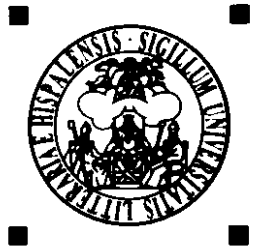

UNIVERSIDAD DE SEVILLA

DEPARTAMENTO DE ECONOMÍA APLICADA I

ANÁLISIS DISCRIMINANTE utilizando MÁQUINAS NÚCLEOS DE VECTORES SOPORTE. FUNCIÓN NÚCLEO SIMILITUD.

Trabajo presentado por **D. Luis González Abril** para la obtención del grado de doctor, dirigido por **Dr. José María Alba Riesco**.

Sevilla, Marzo de 2002

A la memoria de mi mejor amigo.
Mi padre.

Agradecimientos

Quisiera expresar mi más sincero agradecimiento a mi director de Tesis, Dr. D. José María Alba, por todo el esfuerzo y tiempo dedicado a la elaboración de este trabajo.

Así como a todos mis compañeros del Departamento de Economía Aplicada I.

Asimismo, mi agradecimiento a los compañeros del colectivo ARCA por su ayuda desinteresada, en particular al Dr. Cecilio Angulo.

Finalmente, un agradecimiento muy especial a mi mujer, Marilú y a mis hijos Luis y Daniel.

A todos, muchas gracias.

ÍNDICE DE CONTENIDO

2

Problema de clasificación

III Estudio de similitudes entre sucesos a partir de una función núcleo. Análisis práctico de dos problemas de multiclasificación

7
Similitud entre sucesos. Aplicación a unas líneas de investigación

8
Análisis de dos problemas de multiclasificación utilizando SVMs

9
Conclusiones

IV Apéndices

A
Espacios de Hilbert con núcleo reproductor

B

Colección de programas elaborados

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

CAPÍTULO 0

PRÓLOGO

Haciendo un poco de historia, el desarrollo de la Teoría del Aprendizaje Estadístico tuvo sus orígenes en los años 60 del pasado siglo XX, simultáneamente a la incorporación generalizada de los primeros ordenadores a las universidades y a los centros de investigación, lo que permitió guiar a los investigadores en los análisis multivariantes asociados con problemas de la vida real. Desde los primeros resultados de este análisis, quedó claro que los métodos clásicos existentes, para los problemas de estimar funciones cuando se trabajaba en espacios de dimensión pequeña, no reflejaban las singularidades que se daban en los casos de dimensión mayor, ya que había singularidades que no eran captadas por los métodos clásicos. R. Bellman llamó a estas "la maldición de la dimensionalidad".

En sus comienzos supuso una gran ayuda que, en 1958, F. Rosenblatt diseñase una máquina de aprendizaje (programa de ordenador) llamado "Perceptron" para resolver problemas de clasificación, y demostrase que dicha máquina podía generalizarse para su uso con otros problemas. Después del Perceptron, surgieron muchos tipos diferentes de máquinas de aprendizaje, las cuales no se generalizaban peor que el Perceptron, lo que planteó dos cuestiones importantes:

¿Existe alguna cosa en común en estas máquinas?

¿Existe un principio general de inferencia inductiva que ellas implementen?

Rápidamente fue formulado para tales máquinas un principio de inducción general: **el principio de minimización del riesgo empírico** (ERM). Con objeto de alcanzar buenas generalizaciones sobre los datos futuros, el principio ERM sugiere una regla de decisión que minimiza el número de errores ensayados (riesgo empírico). El problema se trasladó al de construir una teoría adecuada para este principio.

Antes de una década, (Vapnik y Chervonenkis, 1968, 1971), la teoría de ERM para los problemas de clasificación fue desarrollada[(1)]. Esta teoría incluye tanto (a) la teoría cualitativa de las generalizaciones, que describe las condiciones necesarias y suficientes del principio ERM (válida para cualquier conjunto de funciones indicadoras en las cuales la máquina minimiza el riesgo empírico), como (b) la teoría cuantitativa general, que describe las cotas sobre las probabilidad de errores en los test (futuros) para la función que minimiza el riesgo empírico. Esta teoría da pie a un nuevo principio; **el principio de minimización del riesgo estructural** (SRM).

La piedra angular de esta teoría es una colección de nuevos conceptos, entre los que destaca el concepto de capacidad de un conjunto de sucesos. De particular importancia es el papel que desempeña en esta teoría la llamada **dimensión VC** del conjunto de sucesos, que caracteriza su variabilidad. Asimismo, se encontró que tanto la condición necesaria, como suficiente, de consistencia y razón de convergencia del principio ERM, dependen de la capacidad del conjunto de funciones implementados por la máquina de aprendizaje. En particular, fue demostrado que, para la consistencia del principio ERM independientemente de la distribución, es necesario y suficiente que el conjunto de funciones implementadas por la máquina de aprendizaje tenga una dimensión VC finita. Se encontró también que las cotas sobre la razón de convergencia uniforme de cualquier distribución depende de la

[(1)] Le Cam, en [Cam00] destaca este resultado como uno de los más importantes en el desarrollo de la estadística desde 1950.

dimensión VC, del número de errores ensayados (del riesgo empírico) y del número de observaciones.

Que la capacidad de generalización de la máquina de aprendizaje depende de la capacidad del conjunto de funciones implementadas, la cual difiere del número de parámetros libres, es uno de los logros más importantes alcanzados por esta nueva teoría.

Posteriormente, ya en los años 80, cuando la teoría de este nuevo enfoque había sido esencialmente desarrollada, se observó que una versión generalizada de uno de los problemas fundamentales de la estadística (el problema de Glivenko-Cantelli) conduce al mismo análisis que fue desarrollado por la teoría del aprendizaje y la generalización de los problemas de clasificación. A mediados de los 80, estos resultados fueron nuevamente reescritos en términos estadísticos tradicionales.

Estos resultados contribuyen a un importante descubrimiento metodológico: El problema de clasificación, uno de los modelos más simples de inferencia inductiva, y sus resultados, pueden ser generalizados a otros modelos más complejos usando, con pequeñas variaciones, las mismas técnicas matemáticas estándar.

De esta forma, en los capítulos 1 y 2 de este trabajo se aborda, en líneas generales, estos resultados. En el capítulo 1, partiendo del problema general de aprendizaje a partir de ejemplos, llegamos al planteamiento del principio SRM, donde se explican todas las ideas subyacentes a partir de una colección de ejemplos.

Así pues, la resolución de estos problemas pasa por diseñar máquinas de aprendizaje para los problemas de clasificación que, posteriormente, son generalizadas para hacer frente a problemas de regresión, problemas de aproximación, etc. En definitiva, se puede decir que la **Teoría del Aprendizaje Estadístico** busca formas de estimar dependencias funcionales a partir de una colección de datos. Este problema es muy general y abarca importantes temas estadísticos, como por ejemplo los problemas de clasificación, problemas de regresión y problemas de estimación de densidades.

Las máquinas de vectores soporte (SVMs) es una técnica desarrollada dentro de esta teoría, con objeto de dar solución al problema fundamental que surge en distintos campos, donde se estudia, la relación entre sesgo y varianza [GB92], el control de capacidad [GVB$^+$92], sobreajuste en los datos [MP92], etc. Este problema consiste en buscar, para una tarea de aprendizaje dada, con una cantidad finita de datos, una adecuada función que permita llevar a cabo una buena **generalización**[(2)] que sea resultado de una adecuada relación entre la precisión alcanzada con un particular conjunto de ensayo[(3)], y **la capacidad de la máquina** (capacidad para aprender con cualquier conjunto de ensayo).

En el capítulo 2 se aborda el problema de clasificación, desde un punto de vista teórico, introduciendo el concepto de capacidad de una máquina de aprendizaje y se obtienen cotas sobre la capacidad de generalización.

Es generalmente aceptado que, no se debe exagerar la capacidad de la máquina, porque en tales casos la máquina memoriza los datos de entrada y salida en lugar de aprender la relación estructural existente entre las variables, de forma que responde perfectamente cuando se introducen los datos utilizados para su entrenamiento y, sin embargo, es incapaz de aproximar una respuesta coherente cuando se le suministra nuevos datos. Así, una máquina con demasiada capacidad es como un especialista que, cuando se le presenta un nuevo árbol para él desconocido, concluye que no se trata de un árbol porque es diferente de todos los que él conoce; de igual manera, una máquina con poca capacidad es como el generalista que declara que, si es verde, entonces es un árbol. Claramente ninguno generaliza bien.

Hay que destacar que el estudio y la formalización de estos conceptos ha resultado ser uno de los desarrollos actuales más importantes dentro de la teoría del aprendizaje estadístico [Vap82].

[(2)] Donde se entiende por generalización, la capacidad de una determinada función de explicar el comportamiento de los datos dentro de un dominio más amplio.

[(3)] Dentro de un conjunto de datos disponibles, el conjunto de ensayo es el formado por aquellos que se utilizan para elegir una función conveniente.

En el capítulo 3 se aborda el estudio de los problemas de clasificación dicotómicos a partir de las máquinas de vectores soporte (SV). Se construye una máquina lineal SV para los casos donde los vectores sean separables y, posteriormente, se generaliza para el caso de vectores no separables. Se introduce un conjunto clave en todos los desarrollos, el conjunto de vectores soporte. También se desarrolla las SVMs no lineales introduciendo el concepto de función núcleo.

En el capítulo 4 se proporciona una interpretación probabilística a las salidas de las SVMs dicotómicas, de tal forma que en el capítulo 5 se propone una nueva máquina de vectores soporte para los problemas de clasificación con más de dos etiquetas que incorpora dicha interpretación probabilística de las salidas, lo que permite llevar a cabo un estudio más detallado y preciso del problema abordado.

Así, en el Capítulo 8, se aplica la metodología expuesta al estudio de los problemas de multiclasificación usando dos conjuntos de datos extraídos de textos de uso frecuente en las enseñanzas de Estadística Multivariante. Además de mostrar la riqueza de información adicional proporcionada por las máquinas de vectores soporte aquí diseñadas, frente a la escueta información de los métodos clásicos del análisis discriminante, se pone de manifiesto la mejor capacidad clasificadora de estas máquinas, así como las ventajas adicionales que esta metodología proporciona al separar los casos de difícil etiquetado, sobre cuyo pronunciamiento, casi con seguridad, se cometerían errores.

Las funciones núcleos, introducidas en el capítulo 3, son estudiadas en el capítulo 6, enlazando esta temática con las redes de regularización y con los espacios de Hilbert con núcleo reproductor[(4)]. De este análisis y también, dentro de esta teoría del aprendizaje, se desarrolla un nuevo enfoque que permite ver las funciones núcleos como cuantificadoras de similitudes, lo que aporta una nueva dirección para la construcción y utilidad de este tipo de funciones.

Como un resultado de las funciones núcleos se desarrolla en el capítulo 7, una función núcleo especial que permite estudiar similitudes entre sucesos. A continua-

[(4)] Estos se desarrollan en el apéndice A.

ción, dentro de este capítulo, se realiza una aplicación práctica que permite estudiar la evolución de las distintas líneas de investigación sobre aprendizaje a partir de datos recogidos en la red internet.

En la elaboración del trabajo ha sido necesario la realización de una serie de programas informáticos que aparecen recogidos en el apéndice B.

Finalizamos el trabajo con el capítulo 9 en el cual se recoge, de manera somera, las principales conclusiones del mismo.

PARTE I

Teoría del Aprendizaje Estadístico

CAPÍTULO 1

INTRODUCCIÓN

Entia præter necessitatem non sunt multiplicanda.
(Los entes no deben multiplicarse más de lo necesario).
–Guillermo de Occam–

En este capítulo inicial se presenta un primer contacto con el enfoque estadístico que se plantea dentro del marco de la Teoría del Aprendizaje Estadístico y es por tanto un punto de referencia necesario para comprender los desarrollos y resultados posteriores. Para su elaboración se ha seguido el esquema planteado en [Gon00], el cual sigue las ideas de V. N. Vapnik recogidas principalmente en [Vap82, Vap98].

El problema fundamental de esta teoría conduce al planteamiento de un problema de optimización: minimizar un funcional riesgo dentro de un conjunto de funciones con una determinada capacidad. Esta primera versión del problema no será resoluble y conduce de forma natural a la definición de un funcional riesgo empírico, el cual sustituye al riesgo en el problema inicial y se tiene una nueva versión del problema fundamental como un problema de optimización que sí tiene solución.

A continuación, se plantea el principio inductivo de minimización del riesgo estructural que constituye la base de los diferentes algoritmos que permiten resolver el problema de optimización y que condicionará la capacidad de la solución para poder generalizar a partir de un conjunto de datos.

Cuando se plantea un determinado algoritmo, con objeto de resolver un problema, necesariamente se han de cubrir dos etapas: (1) existencia de soluciones y (2) convergencia del algoritmo. La primera etapa se satisface, desde el punto de vista de la programación matemática, obligando que los funcionales utilizados en el problema sean convexos. En la segunda etapa, se estudia la **convergencia uniforme** de la metodología donde se introduce un concepto fundamental en esta teoría: **la capacidad de un conjunto de funciones**, es decir la habilidad del conjunto de funciones para aprender a partir de un conjunto de datos. En este primer capítulo se señala la importancia y aplicabilidad de este concepto sin entrar en detalles de su cálculo e interpretación, que se aclara en posteriores capítulos. Además, dentro de la convergencia uniforme del proceso, necesariamente se ha de estudiar como ha de ser el conjunto de funciones de manera que se tenga garantizada una razón de convergencia significativamente grande[(1)].

1.1 Planteamiento del problema

Siguiendo a Vapnik, [Vap98], se entiende por Teoría del Aprendizaje Estadístico la teoría que explora caminos para estimar dependencias funcionales a partir de un conjunto finito de datos. Dentro de esta teoría el problema general se plantea como sigue:

Sean $\mathcal{X}$ e $\mathcal{Y}$ subconjuntos de los espacios vectoriales $\mathbb{R}^d$ y $\mathbb{R}$, respectivamente. Sea

$$X = \{x_1, x_2, \cdots, x_n\} \subset \mathcal{X} \tag{1.1}$$

[(1)]En la práctica, no basta con saber si un determinado algoritmo converge a la solución. Se ha de exigir que esta convergencia sea suficientemente rápida.

un conjunto de n vectores de $\mathcal{X}$ a los que se llamarán indistintamente **vectores inputs** o vectores fuentes, e

$$Y = \{y_1, y_2, \cdots, y_n\} \subset \mathcal{Y} \tag{1.2}$$

un conjunto de n números reales[(2)] a los que se llamaran valores objetivos, **valores outputs**, etiquetas o patrones según sea su procedencia[(3)].

Estos tipos de problemas son conocidos como **problemas de aprendizaje supervisado**, puesto que el conocimiento de los outputs permite cuantificar[(4)] (supervisar) la bondad de los resultados de un determinado problema. Frente a este tipo de problemas se encuentran los problemas de aprendizaje no supervisado donde los outputs son desconocidos o no existen y el objetivo del aprendizaje es obtener algún conocimiento sobre el proceso generador de datos.

Como se ha indicado, el objetivo fundamental de la teoría del aprendizaje estadístico es **aprender** de los datos y para ello se busca la existencia de alguna dependencia funcional entre los vectores inputs $\{x_i,\ i = 1, \cdots, n\}$ y los valores outputs $\{y_i,\ i = 1, \cdots, n\}$. El esquema de trabajo que se sigue para llevar a cabo este fin se encuentra recogido en el libro "Statistical Learning Theory" de V. N. Vapnik ([Vap98]), donde se plantea el siguiente modelo para buscar la dependencia funcional entre los datos, el cual se denomina **modelo de aprendizaje a partir de ejemplos**. Por ello, siguiendo [Ang01] se entenderá por aprendizaje a partir de ejemplos, al proceso de estimar una dependencia desconocida de un sistema entrada-salida utilizando un número limitado de observaciones.

El modelo de aprendizaje a partir de ejemplos presenta tres elementos (ver figura 1.1) claramente diferenciados:

1. El generador de datos (ejemplos), **G**.

[(2)] La generalización al espacio $\mathbb{R}^{d'}$ se sigue de forma natural. Un estudio de este tipo puede encontrarse en [Vap98].

[(3)] Por ejemplo, si se plantea un problema de clasificación donde sobre el conjunto $\mathcal{Y}$ existe una escala nominal, se denominarán etiquetas o patrones.

[(4)] A través de la selección de lo que se conoce como conjunto de test.

2. El operador objetivo (a veces llamado supervisor), **S**.

3. La máquina de aprendizaje, **LM** -del inglés *learning machine*-.

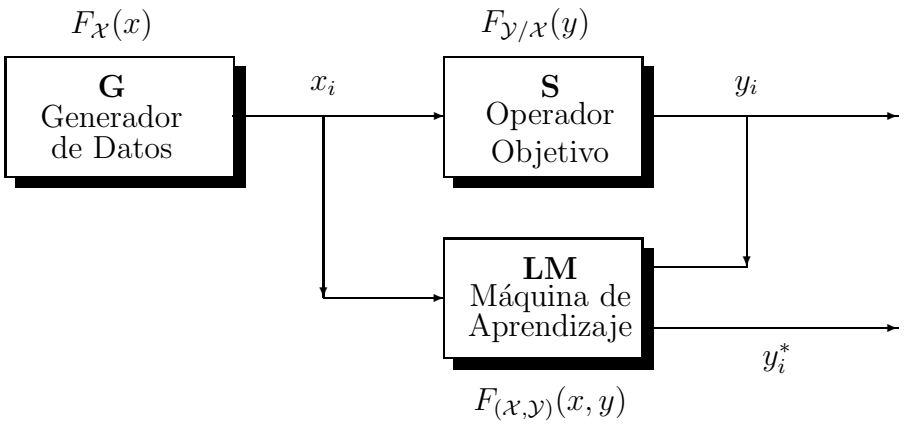

Figura 1.1: Esquema de configuración de una máquina de aprendizaje a partir de ejemplos.

En este modelo, **G** genera los vectores $x_i \in X$, independientes e idénticamente distribuidos (i.i.d.), de acuerdo con una función de distribución de probabilidad $F_{\mathcal{X}}(x)$ desconocida, pero fija. Nótese que una de las consecuencias de ese planteamiento es que las entradas del problema no son controlables, en ningún momento, por el investigador.

Los vectores x_i son los inputs (las entradas) del operador objetivo **S**. Este operador, que transforma estos vectores en valores y_i, es desconocido pero se sabe de su existencia y que no cambiará a lo largo de todo el proceso (El valor y_i es una respuesta del entorno frente a una entrada x_i en el sistema).

De esta forma, la máquina de aprendizaje observa n pares

$$\{(x_1, y_1), \cdots, (x_n, y_n)\}$$

que conforman el denominado **conjunto de ensayo** o **conjunto de entrenamien-**

to[5], los cuales coinciden con los inputs y outputs del supervisor. Se supone que el supervisor **S** obtiene el valor y_i a partir del vector x_i, de acuerdo a una función de distribución condicional $F_{\mathcal{Y}/\mathcal{X}=x_i}(y)$. Así, la máquina de aprendizaje observa el conjunto de ensayo, el cual es obtenido independiente e idénticamente distribuido de acuerdo a una función de distribución conjunta:

$$F_{(\mathcal{X},\mathcal{Y})}(x, y) = F_{\mathcal{X}}(x) \cdot F_{\mathcal{Y}/\mathcal{X}=x}(y).$$

Usando este conjunto de ensayo, la máquina de aprendizaje **construye** una aproximación al operador desconocido. Para construir esta aproximación, la máquina intenta imitar[6] al operador supervisor, es decir, trata de construir un operador el cual proporcione para un generador dado **G**, la mejor aproximación (en algún sentido) a las salidas (outputs) del supervisor.

Formalmente, construir un operador significa que la máquina de aprendizaje implementa un conjunto de funciones, de tal forma que durante el proceso de aprendizaje, elige de este conjunto una función apropiada siguiendo una determinada regla de decisión. Por tanto, **el problema fundamental de aprendizaje** se define como aquel que elige una función dentro de un conjunto de funciones a partir de una determinada regla de decisión.

Dependencia funcional

Dos conjuntos $\mathcal{X}$ e $\mathcal{Y}$ están relacionados por una **dependencia determinística** si para cada $x \in \mathcal{X}$ existe un único $y \in \mathcal{Y}$. Este tipo de relación funcional se dice que

[5]La palabra entrenamiento es la más utilizada en estos problemas, sobre todo por los informáticos, puesto que, como a los atletas, se entrena a una máquina con el fin de dotarla de más capacidad para enfrentarse a los problemas.

[6]Otra formulación distinta se sigue cuando en lugar de intentar imitar el operador objetivo, se pretende identificar dicho operador, es decir, tratar de construir un operador el cual esté cerca (en algún sentido) del operador supervisor. Esta aproximación lleva a plantear problemas sobre ecuaciones integrales, donde lo mejor que se puede conseguir es obtener una sucesión de funciones que converja en probabilidad a la solución cuando el número de observaciones tiende a infinito.

es una función si se trabaja sobre conjuntos de números reales. Sin embargo, en este trabajo se plantea un tipo de dependencia más general; se estudia una relación de **dependencia estocástica**, donde a cada vector $x \in \mathcal{X}$ le corresponde un valor $y \in \mathcal{Y}$ el cual se obtiene como el resultado de un experimento aleatorio. Matizando más, para cada $x \in \mathcal{X}$, sea una distribución $F_{\mathcal{Y}/\mathcal{X}=x}(y)$ definida[7] sobre $\mathcal{Y}$, de acuerdo a la cual se selecciona un valor y como el resultado de una realización de la variable aleatoria $\mathcal{Y}/\{\mathcal{X} = x\}$.

Claramente la dependencia estocástica engloba la dependencia determinística ya que basta con que la distribución de probabilidad $F_{\mathcal{Y}/x}(y)$ sea degenerada, es decir, el espacio muestral tenga un único elemento.

De lo anterior se tiene que cada valor $y_i,\ i = 1, \cdots, n$ es obtenido a partir del vector x_i según una distribución de probabilidad $F_{\mathcal{Y}/x_i}(y),\ i = 1, \cdots, n$ que, al igual que $F_{\mathcal{X}}(x)$, se supondrá que existe y permanece invariante.

Problema de optimización

Sea el conjunto de ensayo

$$Z = \{(x_1, y_1), (x_2, y_2), \cdots, (x_n, y_n)\} \subset \mathcal{X} \times \mathcal{Y} \subset \mathbb{R}^d \times \mathbb{R}$$

el cual determina una muestra aleatoria simple (vectores independientes e idénticamente distribuidos) de acuerdo a una distribución de probabilidad conjunta

$$F_{\mathcal{X}\times\mathcal{Y}}(x, y) = F_{\mathcal{X}}(x) \cdot F_{\mathcal{Y}/\mathcal{X}=x}(y)$$

sobre un determinado espacio probabilístico $(\Omega,\ \mathcal{A}, P)$.

La estimación de la dependencia estocástica basada en un conjunto de datos, significa estimar la función de distribución condicional $F_{\mathcal{Y}/x}(y)$, lo cual en general lleva a un problema complicado, ya que conduce a un problema mal definido[8]. Sin

[7] Abreviadamente y siempre que no lleve a error se utilizará $F_{\mathcal{Y}/x}(y)$.

[8] Siguiendo a Tikhonov ([TA77]), un problema esta bien definido si: i) existe solución; ii) es

embargo, el conocimiento de la función $F_{\mathcal{Y}/x}(y)$ no siempre es necesario; a menudo se esta interesado sólo en alguna de sus características. Por ejemplo se puede buscar la función de esperanza matemática condicional:

$$E[Y/X = x] \stackrel{def}{=} \int y \, dF_{\mathcal{Y}/x}(y). \tag{1.3}$$

Por ello, el objetivo del problema es la construcción de una función $f(x, y)$, dentro de una determinada clase de funciones[9] $\mathcal{F}$ elegida a priori, la cual debe cumplir un determinado criterio de la mejor manera posible. Formalmente, el problema se plantea como un problema de optimización en un determinado espacio de funciones de la siguiente forma:

> *Dado un subespacio vectorial $\mathcal{Z} \subseteq \mathbb{R}^{d+1}$ donde se tiene definida una medida de probabilidad $F_{\mathcal{Z}}(z)$, un conjunto $\mathcal{F} = \{f(z),\, z \in \mathcal{Z}\}$ de funciones reales y un funcional*[10] *$R : \mathcal{F} \to \mathbb{R}$, buscar una función $f^* \in \mathcal{F}$ que minimice $R[f]$, es decir,*
>
> $$R[f^*] = \min_{f \in \mathcal{F}} R[f]. \tag{1.4}$$

Nota 1.1.1 *En muchas situaciones, la existencia del mínimo no está garantizada e incluso puede que no exista; sin embargo, no será un problema tratado explícitamente en este trabajo ya que en estos casos se toma el ínfimo (el cual necesariamente no tiene por qué ser un elemento del conjunto) y ya que dado cualquier valor $\varepsilon > 0$ es posible encontrar un elemento de $f^* \in \mathcal{F}$ que se encuentre a una distancia menor que ε (en un espacio métrico) del ínfimo, se trabaja con esta aproximación al operador objetivo. Por ello, a lo largo del trabajo, se utilizará indistintamente mínimos*

única; y iii) es estable (pequeñas modificaciones en los datos proporciona pequeñas modificaciones en la solución). Puesto que en todos los desarrollos se considerarán funciones convexas sobre dominios compactos, la existencia y unicidad se tienen garantizadas y por ello cuando se indique que el problema esta mal definido se entenderá que el problema no es estable.

[9] En este contexto una determinada clase de funciones es sinónimo de una determinada máquina de aprendizaje.

[10] Una aplicación Q se dice que es un **funcional** si $Q : \mathcal{M} \to \mathcal{N}$ donde $\mathcal{M}$ es un conjunto de funciones y $\mathcal{N}$ es un conjunto de números reales, es decir, para cada $f \in \mathcal{M}$ se tiene $Q[f] \in \mathbb{R}$.

e ínfimos, pero con la ventaja, que presenta trabajar con el mínimo, de poder expresar la solución al problema como un elemento del conjunto, lo que simplificará la notación. ▲

Nota 1.1.2 *El funcional R debe ser tal que cuantifique un determinado criterio, establecido por el investigador, que permita seleccionar una función adecuada dentro del conjunto $\mathcal{F}$. Nótese como los datos entran a formar parte del problema a través del conjunto $\mathcal{Z}$.* ▲

1.2 El funcional riesgo

Diferentes elecciones del funcional R llevaría a considerar distintos problemas, cada uno de los cuales se resolvería utilizando las herramientas adecuadas e incluso, en algunos casos, como en el presente, elaborar toda una teoría que sirva de soporte para su resolución.

Sería muy conveniente elegir el funcional R de tal manera que se pudiese plantear con él, el mayor número de problemas, es decir, que sea lo más general posible. Por ello, y dentro del contexto de la teoría del aprendizaje estadístico se considera el funcional R definido a partir de la esperanza matemática de una determinada función de variables aleatorias respecto a la medida de probabilidad $F_{(\mathcal{X},\mathcal{Y})}(x, y)$. Como se tendrá ocasión de comprobar, esta elección del funcional permite abarcar una gran variedad de situaciones diferentes.

Definición 1.2.1 (de riesgo y pérdida) *Dado $\mathcal{F} = \{f(z),\ z \in \mathcal{Z}\}$ un conjunto de funciones y una medida de probabilidad $F_{\mathcal{Z}}(z)$ se define el* **funcional riesgo** *R como:*

$$R : \mathcal{F} \to I\!R, \quad \textit{donde} \quad R[f] = \int_{\mathcal{Z}} c(z, f(z))\, dF_{\mathcal{Z}}(z). \tag{1.5}$$

A la función $c(\cdot, \cdot)$ se le denomina **función de pérdida** *(o función de coste) y tomará valores no negativos.*

Nota 1.2.2 *A la vista de la figura 1.1 se llega a la conclusión que los valores y_i e y_i^* no coinciden necesariamente. Cuando esto sea así, la máquina de aprendizaje habrá cometido un error que se debe cuantificar de alguna forma. Esta es, precisamente, la misión que dentro de la definición de funcional riesgo tiene la función de coste $c(\cdot,\cdot)$, es decir, sirve para cuantificar la pérdida que para un investigador supone haber cometido un error en la estimación/predicción de las salidas. Por tanto, la función de pérdida es una medida de discrepancia entre la salida del sistema y la proporcionada por la máquina de aprendizaje.*

De lo anterior se sigue que **la función de pérdida depende del investigador** *y por supuesto del problema que pretenda resolver*[(11)].

Por otro lado, nótese que la función de pérdida esta definida de $\mathbb{R}^{d+1} \times \mathbb{R}$ en[(12)] *$\mathbb{R}^+$, es decir,*

$$c(\cdot,\cdot) : \mathbb{R}^{d+1} \times \mathbb{R} \rightarrow \mathbb{R}^+$$

y por tanto $c(z,x) \geq 0,\ \forall z \in \mathbb{R}^{n+1}$, $\forall x \in \mathbb{R}$ y de aquí que

$$R[f] \geq 0, \qquad \forall f \in \mathcal{F}$$

es decir, el riesgo es no negativo, para cualquiera que sea el conjunto de funciones $\mathcal{F}$ que se implemente. ▲

Ejemplo 1.1 *Dentro de los problemas de clasificación y regresión, la función de pérdida debe considerar la "proximidad" entre los valores $f(x_i)$ e y_i con $i = 1, \cdots, n$ para cada función $f(\cdot)$ elegida en el conjunto $\mathcal{F}$. En este caso, el conjunto de funciones $\mathcal{F}$ está definido en $\mathcal{X}$ en lugar de $\mathcal{Z}$, es decir*[(13)], *$\mathcal{F} = \{f(x),\ x \in \mathcal{X}\}$.*

Por ello, se supone (sin pérdida de generalidad) para los problemas de clasificación (y también para los problemas de regresión) que:

[(11)] Se puede encontrar en el capítulo 4 de [Gon00] un estudio detallado de estas funciones dentro de los problemas de regresión.

[(12)] Se considera $\mathbb{R}^+ = [0, +\infty)$.

[(13)] El dominio de las funciones será $\mathcal{X}$ o $\mathcal{Z}$ y no se indicará si queda claro del contexto.

1. *La función de pérdida puede expresarse en la forma*

$$c(z, f^*(z)) \stackrel{def}{=} c(x, y, f(x)) \geq 0$$

lo que significa que la diferencia entre $f(x)$ e y independientemente de si es positiva o negativa, produce una pérdida positiva o nula (la función de pérdida es no negativa);

2. *Si $f(x_i) = y_i$, entonces $c(x_i, y_i, f(x_i)) = 0$, es decir, no se producirá pérdida si la salida que proporciona la máquina, $f(x_i)$, coincide con la salida del supervisor, y_i.*

Con objeto de tener el funcional R bien definido, se supone que la función $c(\cdot, \cdot)$ satisface las condiciones necesarias para que $\int_{\mathcal{Z}} c(z, f(z))\, dF_{\mathcal{Z}}(z) < \infty$, es decir, la función $c(\cdot, \cdot) \in L_1(\mu)$ donde μ es la medida de probabilidad inducida por $F_{\mathcal{Z}}(z)$. ▲

Nota 1.2.3 *De la definición del riesgo se sigue que $R[f]$ es igual al valor esperado de la variable aleatoria $c(Z, f(Z))$ respecto de la medida de probabilidad $F_{\mathcal{Z}}(z)$, es decir, $R[f] = E_{F_{\mathcal{Z}}(z)}[c(Z, f(Z))]$.*

Esta definición es la más natural desde el punto de vista estadístico ya que en Teoría de la Decisión, el riesgo se entiende como el valor esperado de las pérdidas en un determinado problema.

Nótese que la Teoría de la Decisión es más ambiciosa que el enfoque de la Teoría del Aprendizaje, estudiado en este trabajo, ya que como se indica en la figura 1.1, en ésta, se estudia exclusivamente la dependencia funcional entre los datos.

Una introducción simple al riesgo estadístico en Teoría de la Decisión puede encontrarse en [AP95]. Siguiendo la notación de [AP95], se tiene que el conjunto de acciones coincide con el conjunto de funciones $\mathcal{F}$; el conjunto de estados de la naturaleza es el conjunto de outputs ($\mathcal{Y}$); la información viene suministrada por los vectores inputs ($\mathcal{X}$) y la regla de decisión coincide con la regla que resulta de minimizar el riesgo Bayes. Como se indica a lo largo de este trabajo, lo interesante

de este enfoque resulta al elegir criterios para seleccionar un adecuado conjunto de acciones ($\mathcal{F}$). ▲

Utilizando la definición (1.2.1) del riesgo, el problema (1.4) queda planteado como sigue:

> *Dado un subespacio vectorial $\mathcal{Z} \subseteq \mathbb{R}^{d+1}$ donde se tiene definida una medida de probabilidad $F_{\mathcal{Z}}(z)$, un conjunto de funciones $\mathcal{F} = \{f(z),\ z \in \mathcal{Z}\}$ y una función de pérdida, $c(\cdot,\cdot)$, buscar la función f^* tal que $R[f^*]$ alcance el valor*
>
> $$\min_{f \in \mathcal{F}} \int_{\mathcal{Z}} c(z, f(z))\, dF_{\mathcal{Z}}(z). \tag{1.6}$$

Es importante destacar que si en la configuración del problema, la clase de funciones $\mathcal{F}$ y el funcional R estuviesen dado de forma paramétrica, es decir se conoce la forma estructural del riesgo salvo unos determinados parámetros entonces la función f^* que minimiza el funcional se buscaría utilizando las herramientas propias del cálculo variacional, en otras palabras, se plantea un problema de optimización en términos de los parámetros, como se ilustra en el ejemplo siguiente:

Ejemplo 1.2 *Se supone que los vectores inputs siguen un modelo U[0,1] y los valores outputs un modelo U[x, x+1] entonces*[14]

$$dF(x, y) = I_{[0,1]\times[x,x+1]}(x, y)\, dx\, dy$$

en $\mathbb{R}^2$. Sea el conjunto

$$\mathcal{F} = \{f : [0, 1] \to \mathbb{R} \ \textit{tal que}\ f(x) = a + b\, x,\ a, b \in \mathbb{R}\}$$

(funciones lineales) y como función de pérdida se elige

$$c(x, y, f(x)) = (y - f(x))^2$$

[14] La función $g(w) = I_A(w)$ es la función indicadora del conjunto A.

(pérdida cuadrática). El problema que se plantea es determinar los valores a y b de $\mathbb{R}$ *que alcancen el mínimo:*

$$\min_{a,b\in\mathbb{R}} \int_0^1 \int_x^{x+1} (y - a - b\,x)^2\, dy\, dx$$

y la solución que se tiene es $f(x) = \frac{1}{2} + x$ *(*$a = \frac{1}{2}$, $b = 1$*).*

Nótese que se ha supuesto que la función de distribución sigue un modelo uniforme y se ha buscado la solución en $\mathcal{F}$*, y no se ha utilizado en su resolución ningún conjunto de datos, es decir, en estos problemas la solución no depende de los datos, depende del conjunto donde ellos se generen.* ▲

Sin embargo, el problema que se plantea en esta metodología es más complicado ya que a pesar de que el conjunto de funciones $\mathcal{F}$ y la función de pérdida $c(\cdot,\cdot)$ son conocidas, el funcional R no está bien definido, ya que la función de distribución conjunta $F_{\mathcal{X}\times\mathcal{Y}}(x,y)$ se conoce sólo a través de la muestra aleatoria simple (m.a.s.)

$$(x_1, y_1), \cdots, (x_n, y_n). \tag{1.7}$$

Es decir, en este problema la solución depende explícitamente de los datos muestrales y por ello, en lugar de hacer una búsqueda, se ha de **construir la solución** a partir de este conjunto de datos.

En la literatura de las ecuaciones integrales, a este tipo de problemas se les denominan problemas ill-posed (lo que se traduce por: problemas mal planteados, mal definidos o mal situados) y su resolución presenta diferencias substanciales con respecto a la solución que se obtiene en el cálculo variacional.

Cuando el funcional (1.5) se desea minimizar a partir del conjunto de datos (1.7), el problema principal consiste es formular un criterio constructivo para elegir una función del conjunto $\mathcal{F}$ puesto que el funcional (1.5) por si mismo no sirve como criterio de selección, ya que la función $F_{\mathcal{Z}}(z)$ incluida en él es desconocida. Por todo ello, el problema clave que trata de resolver la teoría del aprendizaje estadístico se plantea como sigue:

Construir, dentro de un conjunto de funciones $\mathcal{F}$, una función f^* la cual haga mínimo el riesgo (1.5) cuando la función de distribución es desconocida pero las observaciones (1.7) obtenidas, siguiendo esta distribución, están dadas.

Con objeto de ser más operativo en los siguientes desarrollos habrá veces en las que la familia de funciones $\mathcal{F}$ se represente en la forma

$$\mathcal{F} \stackrel{def}{=} \{f(z,\alpha);\ \alpha \in \Lambda\} \tag{1.8}$$

donde Λ es un conjunto de identificadores que permite para cada α tener diferentes funciones. Se puede pensar que se está restringiendo la clase de funciones a una clase de funciones paramétrica, sin embargo, el conjunto Λ no tiene que ser necesariamente un espacio paramétrico ya que sus elementos pueden ser más abstractos como por ejemplos condiciones[15] o, como en la siguiente nota, un σ-álgebra.

Nota 1.2.4 *Sea un espacio probabilístico $(\Omega, \mathcal{A}, P)$. Si se considera que el conjunto de identificadores Λ coincide con el σ-álgebra $\mathcal{A}$ entonces, si para cada $A \in \mathcal{A}$ la función $f(x, A)$ es una variable aleatoria, el conjunto de funciones $\mathcal{F}$ determina un proceso estocástico.* ▲

El denotar el conjunto de funciones $\mathcal{F}$ de esta forma, permite expresar el problema de minimización como la búsqueda de un determinado $\alpha_0 \in \Lambda$ que identifica f^*, es decir,

$$R[f^*] = \min_{f \in \mathcal{F}} R[f] = \min_{\alpha \in \Lambda} R[\alpha] = R[\alpha_0].$$

1.3 El problema de la clasificación

El problema de clasificación puede formularse como sigue: **un supervisor S clasifica las situaciones observadas G según una de las ℓ-clases diferentes en**

[15] Sirva de ejemplo, dentro del conjunto de las funciones reales de variable real, la función identificada por: $\alpha = \{$función que vale 1 en los números racionales y 0 en los irracionales$\}$.

que se pueden diferenciar. Se quiere construir una máquina, que después de observar las clasificaciones llevadas a cabo por el supervisor, proporcione una clasificación aproximada a la dada por éste. Con posterioridad habrá que cuantificar, en términos probabilísticos, la confianza que nos merece la solución aportada por la máquina.

Utilizando un lenguaje más formal este problema puede formularse como sigue: En un cierto entorno se observan vectores aleatorios x independientes caracterizados por una función de distribución $F(x)$. El supervisor clasifica cada situación de acuerdo a una de las ℓ-clases posibles. Se supone que el supervisor lleva a cabo esta clasificación usando una función de distribución condicional $F(y|x)$, donde $y \in \{0, 1, \cdots, \ell - 1\}$ (por $y = p$, se entiende que el supervisor asigna la clase p a la situación x). No se conoce $F(x)$, ni $F(y|x)$ pero se sabe que existen y por tanto también existe la conjunta $F(x, y) = F(x) \cdot F(y|x)$.

En este caso el conjunto de funciones $\{f(x, \alpha);\ \alpha \in \Lambda\}$, el cual toma solo ℓ valores está dado y representan un conjunto de reglas. Se considera la función de pérdida (más simple)

$$c(y, p) = \begin{cases} 0 & si \quad y = p \\ 1 & si \quad y \neq p \end{cases} \tag{1.9}$$

cuya interpretación es la siguiente: asigna una pérdida nula si la máquina lleva a cabo una clasificación correcta y una pérdida unitaria cuando la máquina proporcione una salida equivocada (da el mismo peso cualquiera que sea la entrada mal clasificada). Obsérvese que la función de pérdida describe un conjunto de funciones indicadoras, es decir, funciones que solo toman dos valores posibles, cero y uno.

El problema de clasificación consiste entonces en minimizar el funcional

$$R(\alpha) = \int c(y, f(x, \alpha))\, dF(x, y) \tag{1.10}$$

sobre el conjunto de funciones $\{f(x, \alpha);\ \alpha \in \Lambda\}$ donde la función de distribución $F(x, y)$ es desconocida pero la muestra aleatoria de pares independientes:

$$(x_1, y_1), \cdots, (x_n, y_n) \tag{1.11}$$

está dada.

Es posible dividir los problemas de clasificación en dos grupos según las salidas que lleve a cabo el supervisor. Si el conjunto de salidas del supervisor y de la máquina de aprendizaje es un conjunto cuyos elementos no presentan ningún tipo de ordenación (escala categórica o nominal) se tendrá un **problema de reconocimiento de patrones**. Como ejemplos de conjuntos de salidas se tienen: sexo de una persona, si se concede o no un préstamo, profesión, estado civil, ...

Por otro lado, si el conjunto de salidas del supervisor y de la máquina de aprendizaje es un conjunto cuyos elementos están ordenados según una escala ordinal y se tiene en cuenta dicha escala en la resolución del problema, se tendrá un **problema de regresión ordinal**. Como ejemplos de conjuntos de salidas se tienen: niveles de estudios, satisfacción por un producto, categoría laboral, ...

1.4 El problema de la regresión

Los problemas de clasificación donde sobre el conjunto de outputs se puede establecer una escala ordinal, pueden ser estudiados como un problema de regresión. Veamos, por tanto, una aproximación de como se plantean estos problemas de regresión dentro del marco de la teoría del aprendizaje[(16)].

Sea el conjunto

$$\{(x_1, y_1), \cdots, (x_n, y_n)\}$$

de pares obtenidos de forma aleatoria e independiente según una distribución conjunta $F(x, y)$. Se busca determinar la función de esperanza matemática condicional, denominada **función de regresión**, siguiente:

$$r(x) \stackrel{def}{=} E[Y/X = x] = \int_{\mathcal{Y}} y \, dF_{Y/x}(y). \tag{1.12}$$

La regresión es una función definida de $\mathbb{R}^d$ en $\mathbb{R}$ tal que aplicada a un vector x, proporciona el valor medio de la variable aleatoria $Y/\{X = x\}$, es decir, aunque las

[(16)] Se puede encontrar un estudio detallado en los capítulos 4 y 5 de [Gon00].

hipótesis iniciales indican que la variable Y se obtiene, en el modelo de aprendizaje por ejemplos, a partir de un experimento aleatorio, la función de regresión asigna a cada vector x un único valor y.

Ejemplo 1.3 *(función de regresión con distribución condicional conocida) Sea X una variable aleatoria continua estrictamente positiva y supongamos como v.a. condicional $Y/\{X = x\} \in Exp(x)$ (modelo exponencial de parámetro x). En este caso, la función de regresión, denotada por $r(x)$, es:*

$$r(x) = \int_{\mathbb{R}^+} y\, d[1 - e^{-xy}] = \int_{\mathbb{R}^+} y\, xe^{-xy} dy = \frac{1}{x}$$

ya que $F_{Y/x}(y) = 1 - e^{-xy}$. ▲

Lema 1.4.1 *El problema de estimar la función de regresión coincide con un problema de minimización del tipo (1.6) si*

$$\int_{\mathbb{R}^2} y^2\, dF(y,x) < \infty, \qquad \int_{\mathbb{R}^2} r^2(x)\, dF(y,x) < \infty.$$

Demostración. Como se tendrá ocasión de comprobar, para la demostración del lema, basta con la hipótesis $\int_{\mathbb{R}^2} (y - r(x))^2\, dF(y,x) < \infty$. Sin embargo, al exigir que $\int r^2(x)\, dF(y,x) < \infty$, se garantiza que la función de regresión pertenezca al conjunto de las funciones de cuadrado integrable $L_2(F)$.
Si $\mathcal{F} \subset L_2(F)$ donde

$$L_2(F) \stackrel{def}{=} \left\{ f : \mathcal{X} \to \mathbb{R} \,/\, \int_{\mathbb{R}} f^2(x)\, dF(x) < \infty \right\},$$

y se elige la función de pérdida

$$c(x, y, f(x,\alpha)) = (y - f(x,\alpha))^2,$$

entonces se tiene que el mínimo del funcional

$$R(\alpha) = \int_{\mathbb{R}^2} (y - f(x,\alpha))^2\, dF(y,x) \tag{1.13}$$

(supuesto que exista[17]) se alcanza en la función de regresión si $r(x) \in \mathcal{F}$. Si, por el contrario, $r(x) \notin \mathcal{F}$, el mínimo es alcanzado por una función $f(x, \alpha^*)$, la cual dentro del conjunto $\mathcal{F}$ es la función que se encuentra más cerca de $r(x)$ en la métrica de $L_2(F)$:

$$\rho(f_1, f_2) = \sqrt{\int_{\mathbb{R}} (f_1(x) - f_2(x))^2 \, dF(x)}.$$

Para la demostración se utiliza una técnica clásica cuando la solución es conocida. Si se denota

$$\triangle f(x, \alpha) \stackrel{def}{=} f(x, \alpha) - r(x),$$

entonces el funcional (1.13) puede escribirse en la forma

$$R(\alpha) = \int (y - r(x))^2 \, dF(x, y) + \int (\triangle f(x, \alpha))^2 \, dF(x) - 2 \int \triangle f(x, \alpha) \, (y - r(x)) \, dF(x, y) \quad (1.14)$$

sin más que desarrollar un binomio de Newton. Si se supone que las variables aleatorias son absolutamente continuas[18] se tiene que $dF(x, y) = g(x, y) \, dx \, dy$, con $g(x, y)$ la función de densidad, y de aquí que en la expresión (1.14), el tercer sumando sea cero, ya que de acuerdo a (1.12) se tiene:

$$\begin{aligned} &\int \int \triangle f(x, \alpha) \, (y - r(x)) \, dF(x, y) \\ &= \iint \triangle f(x, \alpha) \, (y - r(x)) \, g(x, y) \, dx \, dy \\ &= \int \triangle f(x, \alpha) \left(\int (y - r(x)) \, g_{\mathcal{Y}/x}(y) \, dy \right) g_{\mathcal{X}}(x) \, dx \\ &= \int \triangle f(x, \alpha) \left(\int (y - r(x)) \, dF_{\mathcal{Y}/x}(y) \right) dF_{\mathcal{X}}(x) = 0. \end{aligned} \quad (1.15)$$

Así se tiene que

$$R(\alpha) = \int (y - r(x))^2 \, dF(x, y) + \int (f(x, \alpha) - r(x))^2 \, dF(x). \quad (1.16)$$

Ya que el primer sumando no depende de α y es no negativo, la función $f(x, \alpha_0)$, que minimiza el riesgo $R(\alpha)$, es la regresión si $r(x) \in \mathcal{F}$ (ya que anularía el segundo

[17] Si no existe el objetivo es el ínfimo como se indica en la nota 1.1.1.

[18] En otro caso la demostración sería más engorrosa pero se tendría el mismo resultado.

sumando), o es la función más cercana a la regresión en la métrica $L_2(F)$ en el conjunto $\mathcal{F}$, si $r(x) \notin \mathcal{F}$ (ya que minimizaría el segundo sumando). ■

Por tanto, si la función de regresión $r(x)$ pertenece al conjunto dado de funciones $\mathcal{F}$ entonces es la función solución al problema $\inf_{\alpha\in\Lambda} R(\alpha)$. Si se tiene en cuenta que a la hora de plantear un algoritmo que busque la solución al problema $\inf_{\alpha\in\Lambda} R(\alpha)$, no necesariamente se llega a alcanzar, sino que se llega a una función $f(x, \alpha^*)$ tal que el riesgo $R(\alpha^*)$ está ε-cerca del ínfimo

$$R(\alpha^*) - \inf_{\alpha\in\Lambda} R(\alpha) < \varepsilon$$

es necesario saber como de cerca están las dos funciones $r(x)$ y $f(x, \alpha^*)$ en la métrica definida en $L_2(F)$. En estas condiciones se tiene que la función $f(x, \alpha^*)$ está $\sqrt{\varepsilon}$-cerca de la regresión en la métrica $L_2(F)$ ya que:

$$\begin{aligned} \rho^2(f(x,\alpha^*), r(x)) &= \int (f(x,\alpha^*) - r(x))^2 \, dF(x) \\ &= R(\alpha^*) - \int (y - r(x))^2 \, dF(x) \quad \text{por (1.16)} \\ &= R(\alpha^*) - \inf_{\alpha\in\Lambda} R(\alpha) < \varepsilon \end{aligned}$$

lo que implica que

$$\rho(f(x, \alpha^*), r(x)) < \sqrt{\varepsilon}.$$

Al igual que se indica en los problemas de clasificación, una vez resuelto el problema de optimización habrá que cuantificar, en términos probabilísticos, la confianza que nos aporta la solución encontrada.

1.5 El funcional riesgo empírico

Comenzamos la sección con un ejemplo que nos aclara la necesidad de un nuevo concepto:

Ejemplo 1.4 *Una entidad bancaria esta interesada en elaborar un programa informático que le permita, introduciendo un conjunto de valores sobre un cliente, obtener una etiqueta de éste en términos de: "1-Cliente Fiable" y "0-Cliente No Fiable". El responsable del diseño del trabajo selecciona un conjunto de variables relevantes con el problema y recoge de la base de datos de la entidad los siguientes datos:*

$$X = \{(1,2,4,3,4,5),(0,2,3,1,3,4),(2,4,3,4,5,1),\cdots\}$$

$$Y = \{1,0,1,\cdots\}.$$

Tal y como se plantea este problema, en principio, no se tiene ninguna función de pérdida, ni un conjunto de funciones $\mathcal{F}$, ni una medida de probabilidad. ▲

La pregunta que se plantea es:

¿Cómo trabajar con la función de riesgo?

Considerar una determinada función de pérdida puede resultar razonable dentro del contexto del problema, así como la elección de un determinado conjunto de funciones suficientemente sencillas, pero ¿cómo introducir la medida de probabilidad para poder obtener el funcional riesgo, R? La solución más razonable nos conduce al denominado funcional riesgo empírico. La forma de obtenerla es como sigue:

En primer lugar se pretende minimizar el riesgo a partir de un conjunto de datos, y para ello se tiene que, como el funcional riesgo es la esperanza matemática de una variable aleatoria respecto a una medida de probabilidad

$$R[f] = E[c(Z, f(Z))] = \int_{\mathcal{Z}} c(z, f(z))\, dF(z)$$

entonces parece lógico elegir como una estimación de la media poblacional, la media muestral definida a partir de los datos

$$z_1, z_2, \cdots, z_n \tag{1.17}$$

ya que, entre otras buenas propiedades, es un estimador insesgado de la media poblacional. De aquí, la siguiente definición:

Definición 1.5.1 (de riesgo empírico) *Dado un funcional riesgo definido por* $R[f] = \int_{\mathcal{Z}} c(z, f(z))\, dF(z)$, *un conjunto de funciones* $\mathcal{F}$ *y una muestra de tamaño* n, $\{z_1, z_2, \cdots, z_n\}$. *Al funcional* $R_{emp} : \mathcal{F} \to \mathbb{R}$ *definido como*

$$R_{emp}[f] = \frac{1}{n} \sum_{i=1}^{n} c(z_i, f(z_i)), \qquad f \in \mathcal{F} \tag{1.18}$$

se le denomina funcional riesgo empírico o, simplemente, **riesgo empírico**.

Nota 1.5.2 *Es posible dar una representación equivalente del riesgo empírico a partir del conjunto de funciones* $\mathcal{F}$ *expresado en la forma* $\{f(z, \alpha);\, \alpha \in \Lambda\}$ *como sigue: si se denota por* $c(z_i, \alpha) = c(z_i, f(z_i, \alpha))$ *entonces*

$$R_{emp}(\alpha) = \frac{1}{n} \sum_{i=1}^{n} c(z_i, \alpha), \qquad \alpha \in \Lambda.$$

Esta representación es utilizada con bastante asiduidad en los siguientes desarrollos.

▲

Nota 1.5.3 *Nótese, como algo muy importante, que en la definición del riesgo empírico la medida de probabilidad* $F(z)$ *aparece dada implícitamente, a partir de la muestra* $\{z_1, \cdots, z_n\}$ *ya que ésta se obtiene de acuerdo con ella.* ▲

La ventaja de considerar el riesgo empírico estriba en que si se tiene una función de pérdida y un conjunto de funciones, el funcional está dado de forma explícita para cada función; y para cada selección muestral proporciona un valor numérico y es por ello, por lo que será el objeto de la minimización.

La metodología de trabajo que se sigue a partir de este punto es la siguiente: si el valor mínimo del riesgo se alcanza en la función $f(z, \alpha_0)$ y el mínimo del riesgo empírico en la función $f(z, \alpha_n)$, con α_0, $\alpha_n \in \Lambda$, para una muestra (1.17) de tamaño n, entonces se considera que la función $f(z, \alpha_n)$ es una aproximación a la función $f(z, \alpha_0)$ en un determinado espacio métrico (véase la figura 1.4 en la página 43).

Al principio que resuelve el problema de minimización del riesgo utilizando este esquema, se le llama "principio (inductivo) de minimización del riesgo empírico", abreviadamente **principio ERM** (del inglés –*Empirical Risk Minimization*–). Este principio es el utilizado en los desarrollos clásicos, por ejemplo cuando se plantea a partir de un conjunto de datos la regresión lineal mínimo cuadrática.

Pero si se encuentra una función $f^* \in \mathcal{F}$ la cual minimiza el riesgo empírico $R_{emp}[f]$, ¿se puede asegurar que el riesgo $R[f^*]$ está cerca del $\min_{f\in\mathcal{F}} R[f]$? La respuesta es que en general esto no es cierto, como se puede ver en el siguiente ejemplo:

Ejemplo 1.5 *Sean $\mathcal{X}, \mathcal{Y} \subset \mathbb{R}$ y dos muestras $Y = \{y_1, y_2, \cdots, y_n\} \subset \mathcal{Y}$ y $X = \{x_1, x_2, \cdots, x_n\} \subset \mathcal{X}$ con $x_i \neq x_j$ si $i \neq j$. Se considera la función*[19] *$f_{min} : \mathcal{X} \to \mathcal{Y}$ tal que*

$$f_{min}(x) = \begin{cases} y_i & si \quad x = x_i \in X \\ 0 & si \quad x \notin X \end{cases}$$

y como función de pérdida

$$c(x, y, f(x)) = (y - f(x))^2.$$

De la definición de la función f_{min} se tiene que $R_{emp}[f_{min}] = 0$ y se sigue que la función $f_{min}(x)$ es un mínimo del funcional riesgo empírico ya que

$$R_{emp}[f] = \frac{1}{n}\sum_{i=1}^{n}(y_i - f(x_i))^2 \geq 0, \qquad \forall f \in \mathcal{F}.$$

Por otro lado

$$R[f_{min}] = \int_{\mathbb{R}^2}(y - f_{min}(x))^2 dF(x, y) = \int_{\mathbb{R}}\left(\int_{\mathbb{R}} y^2\, dF_{Y/x}(y)\right) dF_X(x)$$

ya que la función f_{min} es cero salvo en un conjunto de medida nula.

Si por ejemplo X= "Precio máximo (en euros) de la gasolina fijado por el gobierno" con $X \sim U[1, 1.1]$ e Y= "Precio de la gasolina (en euros) fijado por las

[19] La condición impuesta sobre los x_i garantiza que ciertamente es una función.

estaciones de servicios" con $Y \sim U[0.8\,x, x]$. *Se tiene que:*

$$R[f_{min}] = \int_1^{1.1} \int_{0.8\,x}^{x} y^2\,\frac{50}{x}\,dy\,dx = \cdots = \frac{50}{9}\left(1 - \frac{8^3}{10^3}\right)\cdot\left(1.1^3 - 1^3\right) \simeq 0.897338.$$

Así, el riesgo siempre vale lo mismo para todas las funciones que minimizan el riesgo empírico. Además, cuando el tamaño muestral crece, no hay ninguna consistencia que permita ir minimizando el funcional riesgo, en otras palabras, no se consigue minimizar el funcional riesgo conforme se aumenta el tamaño de la muestra, de donde se sigue claramente que la función f_{min} *no es adecuada como aproximación a la solución del problema del riesgo ya que no proporciona una respuesta adecuada ante un nuevo input* x *distinto de los inputs utilizados en su construcción (en estos casos se dice que la solución presenta una pobre (nula) capacidad de generalización).*

Evidentemente del ejemplo se tiene que si el precio máximo de la gasolina fijado por el gobierno es un valor que no pertenece al conjunto X *(*$x \neq x_i$, $i = 1, 2, \cdots, n$*), las estaciones de servicios no fijarán el precio de la gasolina en 0 unidades.* ▲

Un estudio minucioso de este ejemplo llevaría a estudiar los fundamentos de los procesos de inducción. Unos comentarios muy acertados sobre este punto se pueden encontrar en el capítulo 3 de [Vap98], donde se ve el problema desde el punto de vista del problema de demarcación de Kant y la teoría de no falsabilidad de Popper.

Un análisis menos filosófico lleva a concluir que la clase de funciones $\mathcal{F}$ no puede ser arbitraria, necesariamente se debe imponer algunas condiciones de regularidad a sus elementos. Por ello, el conjunto $\mathcal{F}$ donde se busca la solución no puede ser arbitrariamente grande pero debe ser suficientemente amplio y flexible, donde se entiende por amplitud y flexibilidad la definiciones dadas en [Ang01], que damos a continuación:

Definición 1.5.4 (de amplitud) *La amplitud de un conjunto de funciones* $\mathcal{F}$ *se define como la capacidad*[20] *para aproximar cualquier función continua,* $f \in \mathcal{C}(\mathcal{X}, \mathcal{Y})$,

[20] Como venimos indicando, un conjunto de funciones es sinónimo de máquina de aprendizaje.

con una precisión especificada cualquiera. Se dirá que $\mathcal{F}$ es denso en $\mathcal{C}(\mathcal{X}, \mathcal{Y})$ si cumple la propiedad de aproximación universal (máxima amplitud).

Definición 1.5.5 (de flexibilidad) *La flexibilidad de un espacio de aproximación se define como la capacidad del espacio para estimar dependencias arbitrarias a partir de un conjunto de datos finito.*

Por otro lado, se demuestra que el problema de minimizar $R_{emp}[f]$ generalmente es un problema mal definido, excepto para una clase de modelos muy restringida [TA77, Vap82], es decir, el funcional $R_{emp}[f]$ puede no ser continuo, y llevar a un comportamiento conocido como sobreajuste en la literatura de redes neuronales [Bis95]. Todo ello lleva a definir el concepto de **consistencia** del principio de minimización del riesgo empírico para un determinado conjunto de funciones[21].

Hay que indicar que, con objeto de conseguir problemas bien definidos, hemos de asumir un conocimiento a priori sobre la forma del modelo, la cual nos permitirá elegir como es el conjunto de funciones $\mathcal{F}$. Este conocimiento, normalmente, será el supuesto de suavidad en la solución, o conocimiento de la función de densidad conjunta, o maximización del margen entre clases, ...

Es conocido que el principio de minimización del riesgo empírico lleva a un proceso de interpolación[22] dentro el conjunto de ensayo que se manifiesta en un fenómeno conocido como[23] "overfitting" (sobreajuste o sobreentrenamiento). Como se indica en [CST00] muchos algoritmos clásicos de máquinas de aprendizajes son capaces de aproximar cualquier función y para conjuntos de ensayos dificultosos proporcionarán una solución que se comporta como un aprendiz novato (del inglés –*rote learner*–),

Por tanto, se puede entender, en general, por capacidad la potencia de la máquina. En este caso, la potencia para elegir una función $f^* \in \mathcal{F}$ que se encuentra tan cerca de f como se quiera.

En el capítulo 2 se da una definición general del concepto de capacidad.

[21] En [Gon00] se puede encontrar un resumen de este tema.

[22] Como ocurre en el ejemplo 1.5.

[23] Un fenómeno no deseable y aún más si se tiene en cuenta que frecuentemente los datos del problema contienen ruido.

donde se entiende por aprendiz novato, aquel que trabaja correctamente con los datos de ensayo pero realiza predicciones sin sentido cuando se enfrentan a nuevos datos. Por ejemplo, los árboles de decisión pueden ser tan grandes que cada hoja represente un ejemplo del conjunto de ensayo. En estos casos la solución llega a ser demasiado compleja y se produce el efecto de sobreajuste. Para evitar en lo posible esta situación se ha de introducir algún tipo de información adicional (por ejemplo una regla de parada, poda del tamaño del árbol, ..) para obtener una solución más adecuada, siguiendo de esta forma el conocido principio que Occam propuso hace más de 700 años: "No se debe multiplicar más de lo necesario", lo que traducido a estos problemas significa que si se tiene una función que explica razonablemente bien la naturaleza del problema, no es necesario buscar una función mucho más complicada si lo que se espera ganar es relativamente poco.

A continuación se presentan ejemplos donde se observa como sería la implementación del principio ERM en dos problemas clásicos de estadística.

Ejemplo 1.6 *Si en los problemas de clasificación se considera como función de pérdida*[(24)]

$$c(x, y, f(x)) = \begin{cases} 1 & si \quad y \neq f(x) \\ 0 & si \quad y = f(x), \end{cases}$$

el funcional riesgo empírico coincide con el método clásico de minimizar el número de errores. ▲

Ejemplo 1.7 *Si en un problema de estimación de la regresión se considera como función de pérdida*

$$c(x, y, f(x)) = (y - f(x))^2,$$

el funcional riesgo empírico queda:

$$R_{emp}(\alpha) = \frac{1}{n} \sum_{i=1}^{n} (y_i - f(x_i, \alpha))^2, \qquad \alpha \in \Lambda$$

[(24)] En el análisis clásico se denomina función test.

por lo que, de acuerdo con el principio ERM, estimar la regresión es minimizar este funcional. Si se considera un conjunto de funciones con una estructura dependiente de unos parámetros, es decir, $\mathcal{F} = \{f(x, \theta);\ \theta \in \Theta\}$*, con* Θ *un espacio paramétrico, entonces el método de minimización de este funcional se conoce como el* **método de mínimos cuadrados ordinarios** *(MCO).* ▲

De esta sección se sigue que la primera dificultad del problema de minimización del riesgo empírico es la búsqueda de condiciones bajo las cuales un conjunto de funciones $\mathcal{F}$ garantiza que valores pequeños del riesgo empírico proporciona valores pequeños del riesgo; es decir, bajo que condiciones se pueden realizar algunas afirmaciones sobre $R[f]$ a partir de la información que viene proporcionada por $R_{emp}[f]$.

1.6 Convergencia Uniforme

Con objeto de garantizar la existencia, unicidad y estabilidad de la solución en los problemas $\text{mín}_{f\in\mathcal{F}} R[f]$ y $\text{mín}_{f\in\mathcal{F}} R_{emp}[f]$ se toman funcionales riesgo convexos sobre un conjunto de funciones compacto en una adecuada métrica. Bajo estas condiciones y en virtud del

Teorema 1.6.1 (Lema de inversión de operadores) *Sea un conjunto compacto* $\mathcal{M}$*, un conjunto de números reales* $\mathcal{N}$ *y una aplicación* $f : \mathcal{M} \to \mathcal{N}$ *continua. Entonces existe una aplicación "inversa"* $f^{-1} : f(\mathcal{M}) \to \mathcal{M}$ *que también es continua.*

cuya demostración se puede encontrar en [RN55], se sigue que el problema de minimización del riesgo empírico está bien definido, ya que si $R_{emp}[f]$ es un funcional continuo del espacio de funciones $\mathcal{F}$ en $\mathbb{R}$, se puede construir un funcional "inverso", y en particular obtener $\arg\text{mín}_{f\in\mathcal{F}} R_{emp}[f]$, el cual queda por tanto bien definido.

Nota 1.6.2 *La aplicación "inversa"* f^{-1} *del teorema no debe interpretarse como la aplicación inversa de* f *ya que en general no se verifica que* $f^{-1} \circ f = f \circ f^{-1} =$

I (identidad). Esta aplicación, que se puede llamar "pseudorecíproca", es una aplicación que se sigue de utilizar el principio de elección para seleccionar una determinada anti-imagen para cada valor $y \in f(\mathcal{M})$. ▲

Las restricciones que se impondrán sobre la clase de funciones $\mathcal{F}$ (máquina de aprendizaje) tienen también otras ventajas. Se demuestra en [VC71] que si el conjunto $\mathcal{F}$ tiene buenas propiedades, el riesgo empírico $R_{emp}[f]$ converge en probabilidad al riesgo $R[f]$ para cualquier función $f \in \mathcal{F}$ cuando el tamaño de la muestra tiende a infinito[(25)], es decir,

$$P\left\{\sup_{f\in\mathcal{F}} |R[f] - R_{emp}[f]| \geq \varepsilon\right\} \to 0 \qquad \text{cuando } n \to \infty \text{ y } \varepsilon > 0. \tag{1.19}$$

Si el conjunto $\mathcal{F}$ esta formado por una única función f, la convergencia (1.19) esta garantizada por la ley de los grandes números. Esta convergencia se cumple también si el conjunto de funciones $\mathcal{F}$ es finito, como se ve a continuación.

Sea un conjunto finito de funciones $\mathcal{F} = \{f_1, \cdots, f_m\}$; y sean ε y δ dos cantidades positivas. Aplicando la ley de los grandes números a las m funciones de $\mathcal{F}$ se sigue que:

$$\forall\, i = 1, 2, \cdots, m; \quad \exists\, n_i \in \mathbb{N} \,/\, \forall\, n \geq n_i : \quad P\left[|R(f_i) - R_{emp}(f_i)| \geq \varepsilon\right] = P[A_i] < \frac{\delta}{m}$$

donde se considera el suceso $A_i = \{|R(f_i) - R_{emp}(f_i)| \geq \varepsilon\}$, y si se toma como valor $n_0 = \underset{1\leq i\leq m}{\text{máx}}\, n_i$, se sigue que para todo $n \geq n_0$ se cumple

$$P\left[\sup_{f\in\mathcal{F}} |R(f) - R_{emp}(f)| \geq \varepsilon\right] \leq P\left[\bigcup_{i=1}^{m} A_i\right] \leq \sum_{i=1}^{m} P[A_i] < m \cdot \frac{\delta}{m} = \delta$$

lo que significa que $\sup_{f\in\mathcal{F}} |R(f) - R_{emp}(f)| \xrightarrow[n\to\infty]{p} 0$.

Sin embargo, para conjuntos $\mathcal{F}$ con infinitas funciones existen contraejemplos donde la convergencia no es cierta (se pueden encontrar algunos en [Vap98]).

[(25)] Se tiene lo que se denomina convergencia de un proceso empírico de dos colas (ver en [Gon00]).

Por tanto, se debe elegir dentro de la máquina de aprendizaje aquellas clases de funciones $\mathcal{F}$ que cumplen la condición impuesta en (1.19), es decir, conjuntos de funciones donde se cumpla **la generalización de la ley de los grandes números**[(26)]. A menudo a estas clases de funciones se les denominan **clases generalizadas de Glivenko-Cantelli**.

La convergencia dada en (1.19) indica que, ya que el supremo es tomado sobre todas las funciones de $\mathcal{F}$, las declaraciones que se realizan son independientes de la función final que se elija, es decir, se verifican para cualquier función del conjunto. Además, de (1.19) se sigue que la situación que se presenta en el peor de los casos (seleccionando dentro del conjunto de funciones la que peor se comporta) es suficiente para dar una cota uniforme de convergencia.

Nota 1.6.3 *Nótese un detalle muy importante que pasa inadvertido muy frecuentemente: realmente, no es la "complejidad" del conjunto $\mathcal{F}$, sino la "complejidad" del conjunto $\mathcal{F}$ inducido por la función de pérdida (el conjunto $c_{\mathcal{F}} = \{c(z, f(z)), f \in \mathcal{F}\}$), la que determina la complejidad global del problema. Esto se puede ver fácilmente sin más que elegir como función de pérdida la función $c(\cdot, \cdot) = 0$ para todos los posibles argumentos. Claramente se tiene que las cotas sobre el supremo son fáciles de obtener*[(27)] *con independencia de la complejidad de $\mathcal{F}$. En general, en circunstancias normales, la clase de funciones $\mathcal{F}$ y la clase $c_{\mathcal{F}}$ inducida por la función de pérdida están muy relacionadas.* ▲

En general, la condición de convergencia uniforme

$$P\left\{\sup_{f\in\mathcal{F}} |R[f] - R_{emp}[f]| \geq \varepsilon\right\} \to 0 \qquad \text{cuando } n \to \infty \text{ y } \varepsilon > 0$$

impuesta a la clase $\mathcal{F}$ es demasiado fuerte; en la práctica lo que se necesita es una condición algo más débil, como que para algún $\eta < 1$ fijado, se cumpla la acotación

[(26)] Se denomina de esta forma cuando se cumple la ley de los grandes números para todas las funciones de una clase dada.

[(27)] Evidentemente las diferencias entre riesgo y riesgos empíricos siempre es nula.

de la probabilidad

$$P\left\{\sup_{f\in\mathcal{F}}|R[f]-R_{emp}[f]|\geq\varepsilon\right\}\leq\eta \tag{1.20}$$

dentro del conjunto de todas las muestras de tamaño n (suficientemente grande). Evidentemente la primera condición implica la segunda condición, sin más que aplicar la definición de límite, y elegir un valor n_0 adecuado, pero no al revés. Puede ocurrir que para $\eta = 0'05$ sea posible obtener la cota, pero no para $\eta = 0'025$.

Las condiciones dadas anteriormente son estudiadas en [Vap98] como la situación presentada en el peor de los casos y una situación más benigna. Intuitivamente la condición (1.20) declara que el investigador espera que la muestra que dispone no sea "mala", en el sentido que no le lleve a aceptar como solución del problema una función dentro del conjunto $\mathcal{F}$ cuyo riesgo empírico se encuentre alejado en exceso del riesgo funcional.

Además desde un punto de vista práctico, no es necesario únicamente la convergencia, sino que también es muy importante la razón de convergencia, es decir, la "rapidez" con la que $R_{emp}[f]$ converge a $R[f]$, apartado que se estudia en el capítulo referente a los problemas de clasificación.

Por otro lado, si se fija un valor $\eta < 1$ se sigue de (1.20) que

$$|R(f)-R_{emp}(f)|\leq\varepsilon,\qquad\forall\, f\in\mathcal{F} \tag{1.21}$$

con probabilidad al menos $1-\eta$; y de aquí que para todas las funciones de $\mathcal{F}$ se cumpla con probabilidad al menos $1-\eta$ que

$$R(f)\leq R_{emp}(f)+\varepsilon$$

donde ε dependerá de la muestra Z de tamaño n, del valor de η y del conjunto $\mathcal{F}$ (a través de la función de pérdida); luego

$$\varepsilon=\Phi(n,\eta,\mathcal{F}).$$

También se sigue que si se tienen dos conjuntos de funciones tales que $\mathcal{F}'\subset\mathcal{F}$ entonces $\varepsilon' = \Phi'(n,\eta,\mathcal{F}') \leq \Phi(n,\eta,\mathcal{F}) = \varepsilon$ ya que la acotación debe ser cierta

para un número mayor de funciones. Así pues, la dependencia de la función Φ del conjunto de funciones $\mathcal{F}$, dado que es monótona, se puede modelizar a través de un indicador $h(\mathcal{F})$ de $\mathcal{F}$, de modo que se puede escribir $\varepsilon = \Phi(n, \eta, h(\mathcal{F}))$. Por tanto con probabilidad al menos $1 - \eta$ se tiene:

$$R[f] \leq R_{emp}[f] + \Phi(\eta, n, h(\mathcal{F})), \qquad \forall\, f \in \mathcal{F}, \tag{1.22}$$

donde n es el tamaño de la muestra y η representa la confianza depositada en las afirmaciones.

La desigualdad (1.22), indica que el problema es **aproximadamente correcto en probabilidad** (del inglés –*probably approximately correct* (pac)–) es decir, se cuantifica en términos probabilísticos la capacidad de la máquina de aprender correctamente a partir de una muestra.

Por otro lado, el indicador[(28)] $h(\mathcal{F})$ es una cantidad que nos mide la **capacidad** del conjunto de funciones $\mathcal{F}$. Nótese que la función Φ es una función no negativa y creciente respecto de $h(\mathcal{F})$. También se sigue de (1.21) que la función $\Phi(\eta, n, h)$ es una cota, con probabilidad al menos $1-\eta$, de la diferencia absoluta entre los riesgos y se suele denominar **cota de generalización**, o **error de generalización**, o **cota pac**. Por tanto el error de generalización se define como la cota de la suma del error de aproximación que se comete al elegir la máquina de aprendizaje (el espacio de funciones $\mathcal{F}$) más el error en la estimación que dentro de esta máquina provoca la finitud del conjunto de ensayo.

Algunos autores, [Bur96, Cor95], llaman a esta función $\Phi(\eta, n, h)$ intervalo de confianza de las muestras ensayadas de tamaño n dentro de una clase de funciones $\mathcal{F}$ con capacidad h, ya que se puede interpretar de forma totalmente análoga a como se hace con los intervalos de confianza clásicos en los desarrollos teóricos de inferencia estadística.

Teniendo en cuenta la condición (1.19) y la desigualdad (1.22) se puede representar gráficamente la llamada **curva de aprendizaje**, la cual describe la variación

[(28)] Si no lleva a equívoco se denotará por h.

del riesgo y del riesgo empírico, como una función del tamaño de la muestra para una clase de funciones $\mathcal{F}$ con una capacidad dada. La figura 1.2 muestra un ejemplo típico de tales curvas de aprendizaje. Nótese que la curva no se debería representar con trazos continuos ya que no es una función continua puesto que la variable n solo toma valores enteros positivos, sin embargo se representa de manera continua por resultar más cómoda su interpretación. Para poder comprender la figura 1.2 se debe tener en cuenta que, para un n fijado, $\alpha_n = \arg\min_{\alpha\in\Lambda} R_{emp}(\alpha)$, las curvas representan las funciones $R(\alpha_n)$, $R_{emp}(\alpha_n)$ y $R(\alpha_0) = \inf_{\alpha\in\Lambda} R(\alpha)$.

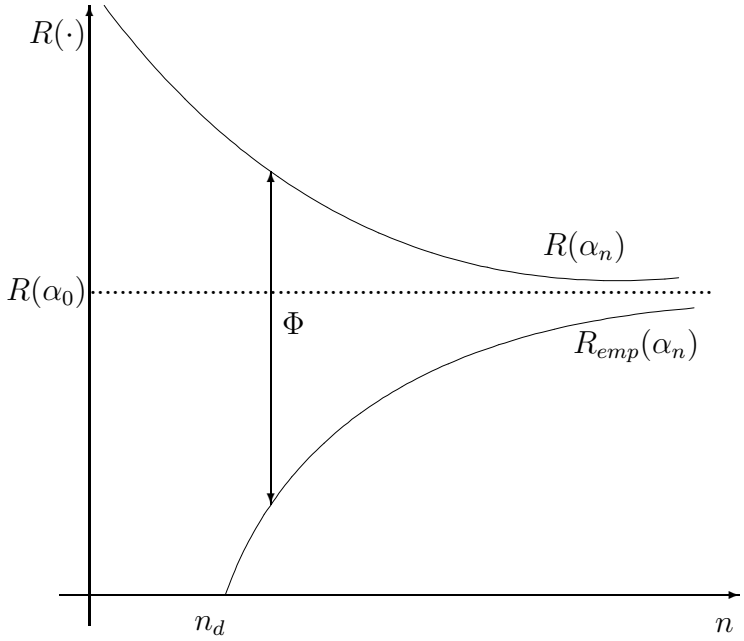

Figura 1.2: Ejemplo de curva de aprendizaje, donde se representa el valor del funcional riesgo y del funcional riesgo empírico en la función $f(z, \alpha_n)$ que minimiza el riesgo empírico. También se indica cual sería el intervalo de confianza Φ.

La figura 1.2 se interpreta como sigue: si el tamaño de la muestra es pequeño, el riesgo empírico es nulo ya que la función elegida dentro de la clase $\mathcal{F}$ ajusta perfectamente todos los datos y normalmente proporcionará una pobre capacidad

de generalización (el intervalo de confianza Φ es muy grande). Cuando la muestra crece, un conjunto con una capacidad finita debe alcanzar el valor crítico n_d, tal que para $n > n_d$ la clase de funciones (máquina de aprendizaje) no puede ajustar todos los elementos de la muestra y por tanto el riesgo empírico ya no se anula[29]. Además, en general, esto lleva a que el mínimo del riesgo empírico crece, $R_{emp}[\alpha_n]$, cuando lo hace el tamaño de la muestra, n. Por otro lado, el riesgo, en general, disminuye cuando datos adicionales son añadidos a la muestra ya que en este caso se dispone de mayor información, es decir, el valor del funcional riesgo en $f(z, \alpha_n)$, $R[\alpha_n]$, decrece cuando el tamaño muestral n crece. Es por ello por lo que se representa el riesgo y el riesgo empírico mediante esos tipos de curvas. Para un máquina de aprendizaje de capacidad finita y en el límite cuando el tamaño de la muestra crece al infinito, las dos curvas proporcionan un valor asintótico común ya que de la condición:

$$P\left\{\sup_{f\in\mathcal{F}} |R[f] - R_{emp}[f]| \geq \varepsilon\right\} \to 0 \qquad \text{cuando } n \to \infty \text{ y } \varepsilon > 0$$

se sigue:

$$\lim_{n\to\infty} \Phi(\eta, n, h) = 0$$

para η y h fijados.

1.7 Principio de minimización del riesgo estructural

En muchas situaciones prácticas la muestra viene dada, lo cual motiva que los razonamientos anteriores no sean aplicables puesto que entonces el tamaño muestral no sería variable sino que sería constante. Esto significa que no se puede considerar un tamaño muestral tan grande como sea necesario para minimizar el riesgo empírico. En estas situaciones se impone seguir otro principio distinto del ERM. El principio considerado por la Teoría del Aprendizaje Estadístico es denominado **principio de**

[29] Pensar en ajustar un conjunto de funciones lineales a un conjunto formado por un dato, dos datos, tres datos, ...

minimización del riesgo estructural, abreviadamente principio **SRM** (del inglés –*Structural Risk Minimization* –).

Para su estudio se parte de la acotación dada en la desigualdad:

$$R[f] \leq R_{emp}[f] + \Phi(\eta, n, h(\mathcal{F})), \qquad \forall\, f \in \mathcal{F}.$$

Esta desigualdad indica que dada una máquina de aprendizaje (clase de funciones) $\mathcal{F}$, una función de pérdida, una cantidad $0 < \eta < 1$ y una muestra de tamaño n, se tiene que el funcional riesgo tiene una probabilidad de al menos $1 - \eta$ de ser inferior al riesgo empírico más la cota de generalización. Estos resultados pueden ser utilizados para acotar el riesgo $R[f]$ para alguna clase $\mathcal{F}$ fija y de aquí que se sea capaz de determinar la bondad de la estimación del riesgo basándose en el riesgo empírico mínimo y en la función Φ.

El principio de minimización del riesgo estructural se basa en minimizar el lado derecho de la desigualdad (1.22), donde se considera la capacidad de la máquina de aprendizaje h como una variable de control (si n es dada y η es el nivel de confianza, es la única que en cierto sentido se puede moldear) ya que con ella es posible controlar el riesgo empírico. Lo adecuado por tanto, dada una muestra Z de tamaño n, es elegir un conjunto de funciones $\mathcal{F}^*$ de una clase $\mathbb{F} = \{\mathcal{F}_i\}_{i \in I}$, donde I es un conjunto de índices, tal que:

$$\min_{f \in \mathcal{F}^*} \{R_{emp}[f, Z] + \Phi(\mathcal{F}^*, Z, \eta)\} = \min_{\mathcal{F} \in \mathbb{F}} \left\{ \min_{f \in \mathcal{F}} R_{emp}[f, Z] + \Phi(\mathcal{F}, Z, \eta) \right\}. \qquad (1.23)$$

Nótese que se ha variado la notación, pero una rápida mirada nos indica que en realidad todo es igual. La idea de cambiarla viene motivada por dar una mayor claridad[30] y donde $\Phi(\mathcal{F}, Z, \eta)$ es, al igual que en (1.22), una cota de la desviación entre el riesgo y el riesgo empírico, la cual se cumple con probabilidad al menos $1 - \eta$.

Del estudio de la consistencia (ver [Gon00]), se llega a la conclusión que una elección adecuada es definir $\mathbb{F}$ con una estructura determinada por una clase de

[30] Si se escribe $\Phi(\eta, n, h)$ se puede pensar que h varía de manera continua cuando en realidad la medida de capacidad es un número natural pues procede de un recuento.

conjuntos compactos anidados de funciones:

$$\mathcal{F}_1 \subset \mathcal{F}_2 \subset \cdots \subset \mathcal{F}_i \subset \mathcal{F}_{i+1} \subset \cdots . \tag{1.24}$$

Ejemplo 1.8 *Una construcción muy habitual de conjuntos anidados de funciones es la que se tiene en los desarrollos en series, por ejemplo los desarrollos de Taylor, Fourier, ... Un esquema general es el siguiente:*

Sea un espacio de funciones generado por un conjunto infinito de funciones $\{\phi_1, \phi_2, \cdots\}$. *Se toma* $\mathcal{F}_i$ *como la clase de funciones generada por las* i *primeras funciones* $\{\phi_1, \phi_2, \cdots, \phi_i\}$. *Evidentemente se tiene que si* $i < j$ *se cumple* $\mathcal{F}_i \subset \mathcal{F}_j$. ▲

Eligiendo una clase $\mathbb{F}$ de conjuntos compactos anidados de funciones se tiene claramente que:

i) El riesgo empírico es monótonamente decreciente dentro de la clase

$$\min_{f \in \mathcal{F}_j} R_{emp}[f, Z] \leq \min_{f \in \mathcal{F}_i} R_{emp}[f, Z], \qquad i < j; \tag{1.25}$$

ii) La función $\Phi(\mathcal{F}_i, Z, \eta) = \Phi(\mathcal{F}_i)$ va creciendo ya que las clases de funciones son cada más vez ricas (mayor capacidad).

Por tanto, debe existir algún[31] $n_0 \in \mathbb{N}$ (ver figura 1.3) donde se alcance el mínimo de la cota global sobre el riesgo. En este conjunto $\mathcal{F}_{n_0}$ se obtiene la función f_{n_0} que alcanza el mínimo del riesgo empírico, y su riesgo será elegido como la aproximación al mínimo del funcional riesgo.

El resultado de este procedimiento de búsqueda de la solución se puede ver gráficamente a partir de la figura 1.4. Si se denota la solución del problema por $f_{opt} = \arg\min R[f]$, el investigador considera una estructura $\mathbb{F}$ de clase de conjuntos de funciones anidadas donde la aproximación a la función f_{opt} es $f(x, \alpha_0) =$

[31] No confundir con el tamaño de la muestra Z, n, ya que aunque dependa de ella, en este caso hace referencia al conjunto de funciones $\mathcal{F}_{n_0}$.

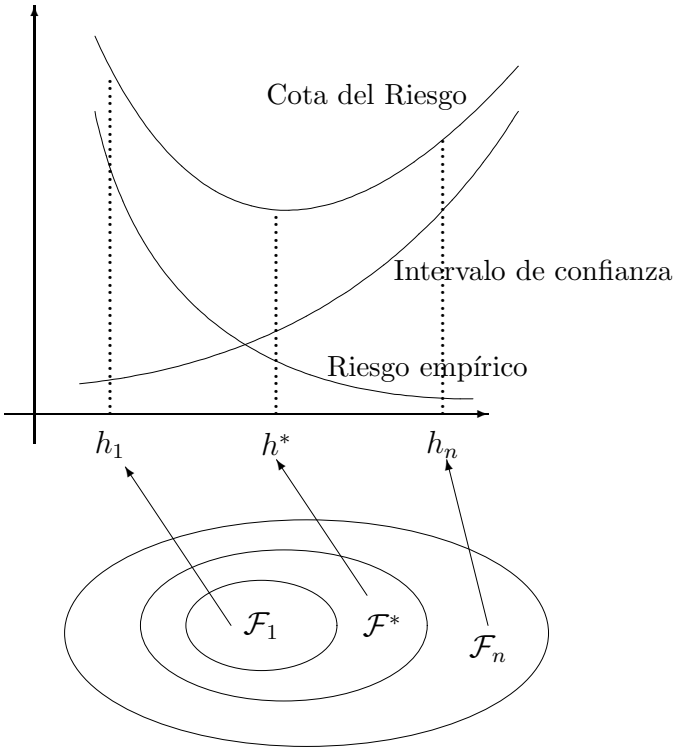

Figura 1.3: Ilustración del principio de minimización del riesgo estructural eligiendo una estructura determinada por una clase de conjuntos compactos anidados.

$\left\{\arg\min_{f\in\mathcal{F}} R[f],\ \mathcal{F}\in\mathbb{F}\right\}$. De la aplicación del principio SRM, selecciona una clase de funciones $\mathcal{F}$ dentro de la estructura $\mathbb{F}$, donde plantea el problema de minimización del riesgo empírico a partir de una muestra de tamaño n y obtiene como solución $f(x,\alpha_n) = \arg\min_{f\in\mathcal{F}} R_{emp}[f]$. Además, considera esta función como una aproximación a la verdadera solución del problema f_{opt}.

Una explicación muy adecuada del objetivo de las máquinas de aprendizaje a partir del gráfico 1.4 dada en [Gun98] es la siguiente: El fin en la modelización de una máquina de aprendizaje es elegir un modelo del espacio de hipótesis, que esté cerca (con respecto a alguna medida) de la función subyacente en el espacio objetivo (target). Los errores que aparecen surgen de dos formas: i) **Error de**

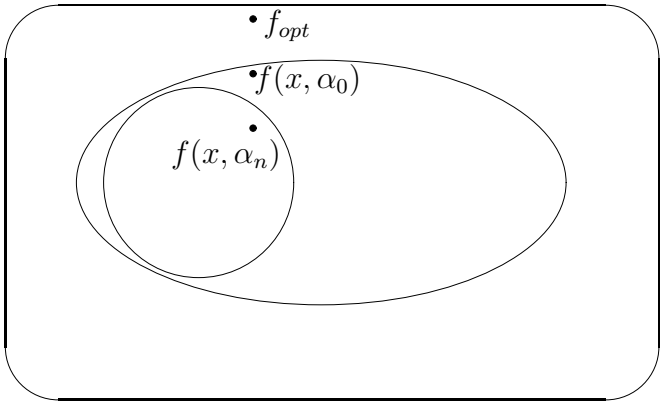

Figura 1.4: Gráfico que muestra una situación de las diferentes soluciones a los problemas de optimización que intervienen en el planteamiento del principio de minimización del riesgo estructural.

aproximación – es consecuencia de que el espacio de hipótesis sea más pequeño que el espacio objetivo, y de aquí que la función subyacente cae fuera del espacio de hipótesis. Una pobre elección del espacio donde se busca el modelo traerá como consecuencia un gran error de aproximación, y se dice que es un modelo mismatch (empareja mal). ii) **Error de estimación** – es el error debido al procedimiento de aprendizaje el cual es el resultado de la selección de técnicas para modelos no óptimos del espacio de hipótesis. A esos dos errores juntos se les denomina error de generalización y, por tanto, se tiene que:

Error de Generalización = Error de Aproximación + Error de Estimación.

Como resultado de todo lo anterior se sigue que el principal inconveniente que plantea la estructura $\mathbb{F}$ es que tiene que ser elegida a priori lo que constituye un problema. Sobre la elección del conjunto de funciones $\mathcal{F}$ se realiza un breve, pero muy útil, esbozo de las aproximaciones en este contexto en [Smo98]. Claramente, lo ideal sería que la verdadera función (desconocida) que resuelve el problema de aprendizaje se encuentre dentro del conjunto de funciones $\mathcal{F}$ seleccionado pero esto no es cierto en general a menos que se le exija a la solución algún tipo de propiedad

con lo cual se entra en el campo del conocimiento a priori.

Por otro lado, es posible plantear este problema como un problema de inferencia bayesiana, donde si se denota por $f_{SRM} \in \mathcal{F}$ la función obtenida, aplicando el principio SRM, por Z el conjunto de ensayo y aplicamos la regla de Bayes se tiene:

$$P(f_{SRM}|Z) = \frac{P(Z|f_{SRM}) \cdot P(f_{SRM})}{P(Z)}.$$

Pero el inconveniente que presenta este enfoque, tal y como se indica en [Vap82] es que puede no tener sentido. Por ejemplo, aplicar este tipo de inferencia a un problema donde el conjunto de funciones $\mathcal{F}$ es un conjunto polinómico si la verdadera función no es polinómica significa que la probabilidad a priori de que cualquier función de $\mathcal{F}$ sea la función solución es nula.

Se concluye esta sección indicando que los análisis en otras ramas de la teoría del aprendizaje han llevado a resultados similares en la relación entre reducir el riesgo empírico y limitar la complejidad del modelo, por ejemplo en la teoría de regularización [TA77], en los estudios de longitud de descripción mínima (MDL) [Kol65], en control de capacidad [GVB+92], en el dilema sesgo-varianza [GB92] (y [Hay94] dentro del contexto de redes neuronales) y en los problemas de sobreajuste [MP92].

1.8 Los funcionales riesgos regularizados

Como se indicó en la sección 1.6, el conjunto de funciones $\mathcal{F}$ no puede ser muy general, necesariamente se debe restringir a un conjunto más concreto. Esto normalmente se lleva a cabo imponiendo un término penalizador convexo sobre alguna de las cantidades involucradas en la función f.

Así, dada una clase de funciones $\mathcal{F}$ se define un funcional

$$Q : \mathcal{F} \rightarrow \mathbb{R}.$$

Dentro de la clase $\mathcal{F}$ se consideran aquellas funciones $f \in \mathcal{F}$ que cumplen la condición

$Q[f] < \Lambda$ para algún[32] $\Lambda > 0$. De esta forma, en una adecuada métrica, resulta que $\mathcal{F}_\Lambda = \{f \in \mathcal{F},\ Q[f] < \Lambda\}$ es un conjunto compacto de funciones con lo que se consigue una de las condiciones para que el problema de minimización del riesgo empírico este bien definido. Por otro lado si se elige una sucesión de números reales, $\{\Lambda_1 \leq \Lambda_2 \leq \cdots\}$ se tiene una estructura de conjuntos compactos anidados de funciones $\mathcal{F}_{\Lambda_1} \subseteq \mathcal{F}_{\Lambda_2} \subseteq \cdots$.

También sobre el funcional $Q[f]$ se realizan dos supuestos:

1. Tiene que ser un funcional convexo, con objeto de asegurar la existencia de soluciones en los posteriores problemas de optimización que se proponen así como la convergencia de los algoritmos que se implementan.

2. La clase de funciones inducida mediante este funcional debe cumplir la convergencia uniforme.

A partir del funcional $Q[f]$ se tienen las siguientes definiciones:

Definición 1.8.1 (Riesgo regularizado) *Dado $\mathcal{F} = \{f(x, \alpha);\ \alpha \in \Lambda\}$ un conjunto de funciones, al funcional*

$$R_{reg} : \mathcal{F} \rightarrow I\!R$$

definido como

$$R_{reg}[f] = R[f] + \lambda \cdot Q[f] \tag{1.26}$$

donde $\lambda \in I\!R^+$, se le denomina funcional riesgo regularizado o simplemente **riesgo regularizado**.

Definición 1.8.2 (Riesgo empírico regularizado) *Dado $\mathcal{F} = \{f(x, \alpha);\ \alpha \in \Lambda\}$ un conjunto de funciones, al funcional*

$$R_{reg,emp} : \mathcal{F} \rightarrow I\!R$$

[32] En estos desarrollos Λ es un número, no es conjunto de índices como en la definición de la clase de funciones $\mathcal{F}$ dada en (1.8), pero se denota de la misma forma ya que es la condición que nos permite distinguir los elementos de una determinada clase de funciones

definido como

$$R_{reg,emp}[f] = R_{emp}[f] + \lambda \cdot Q[f] \tag{1.27}$$

donde $\lambda \in \mathbb{R}^+$*, se le denomina funcional riesgo empírico regularizado o simplemente* **riesgo empírico regularizado**.

Nota 1.8.3 *También es posible dar una definición alternativa del riesgo regularizado en la forma:*

$$R_{reg}[f] = Q[f] + C \cdot R[f] \tag{1.28}$$

$$R_{reg,emp}[f] = Q[f] + C \cdot R_{emp}[f] \tag{1.29}$$

para cualquier función $f \in \mathcal{F}$ *sin más que tomar* $C = \frac{1}{\lambda}$ *si* $\lambda \neq 0$*, lo que será habitual, ya que si es cero no tiene sentido el haber introducido un nuevo tipo de riesgo.* ▲

Nota 1.8.4 *El adjetivo regularizado indica que el funcional* $Q[f]$ *actúa como filtro para que la función* f*, dentro del esquema de trabajo, presente un determinado grado de regularidad (o suavidad). De esta manera se evita trabajar con funciones "extrañas" como la vista en el ejemplo 1.5.* ▲

Nota 1.8.5 *La interpretación del funcional riesgo regularizado desde el punto de vista dado por G. Wahba en [Wah90] es útil también en este contexto, sobre todo desde un punto de vista práctico:*

(a) el riesgo empírico se interpreta como un ajuste de la función a los datos,

(b) el funcional $Q[f]$ *se interpreta como un factor que cuantifica la regularidad de la función* f*; y*

(c) con el riesgo regularizado se pretende, en cierta manera, intercambiar regularidad por ajuste.

La relación entre ajuste-regularidad se controla vía el factor λ. ▲

Como se apunta en [Ang01], el principal defecto de la técnica de regularización es que están diseñadas para identificar el sistema construyendo un operador próximo a él. Sin embargo, el objetivo que se persigue con el aprendizaje a partir de ejemplos es imitar al sistema. Por tanto se busca resolver, como ya se apuntó anteriormente, un problema más complicado cuyos resultados se siguen asintóticamente.

De acuerdo con todo lo desarrollado hasta ahora, el problema de minimizar el riesgo empírico para un conjunto de funciones $\mathcal{F}$ se formula como sigue:

$$\begin{aligned} &\min_{f\in\mathcal{F}} \quad R_{emp}[f] \\ &s.a \quad Q[f] < \Lambda \end{aligned} \tag{1.30}$$

En otras palabras, se minimiza $R_{emp}[f]$ mientras se mantiene la complejidad del modelo fijada, imponiendo para ello una cota superior sobre la medida de complejidad (término de regularización) $Q[f]$ del conjunto de funciones. Esto es lo que se hace cuando se sigue el principio de minimización del riesgo estructural ya que el anidamiento de las clases de funciones nos define una ordenación la cual debe venir impuesta de alguna forma por un funcional adecuado.

En muchas situaciones se puede intentar resolver otro problema diferente (con $\lambda > 0$ fijado), el planteado en términos del funcional riesgo empírico regularizado:

$$\min_{f\in\mathcal{F}} R_{reg,emp}[f] = \min_{f\in\mathcal{F}} \left(R_{emp}[f] + \lambda \cdot Q[f]\right) \tag{1.31}$$

La ventaja de esta nueva formulación es que proporciona problemas de optimización que pueden ser resueltos más fácilmente por métodos numéricos que el problema (1.30).

Finalmente se puede plantear un tercer problema:

$$\begin{aligned} &\min_{f\in\mathcal{F}} \quad Q[f] \\ &s.a \quad R_{emp}[f] < \Lambda' \end{aligned} \tag{1.32}$$

donde se han intercambiado los papeles de los funcionales.

Los tres problemas, cuando tanto $Q[f]$ como $R_{emp}[f]$ son funcionales convexos pueden ser resueltos de forma eficiente y son equivalentes como se demuestra en [Fle87].

La figura 1.5 puede servir para tener una interpretación visual de las tres diferentes elecciones de algoritmos de aprendizajes. La forma de la curva resulta del siguiente razonamiento: a partir del funcional $Q[f]$ se puede construir una estructura de subconjuntos anidados en $\mathcal{F}$ y es evidente que el mínimo de $R_{emp}[f]$ decrece monótonamente cuando $Q[f]$ crece, ya que en este caso el conjunto de funciones es cada vez más grande. De esta forma, se podría parametrizar la solución f_Λ, obtenida cuando minimizamos el riesgo empírico, $R_{emp}[f]$, sobre la clase de funciones $\mathcal{F}_\Lambda = \{f \in \mathcal{F} \,|\, Q[f] \leq \Lambda\}$, por Λ, es decir

$$\min_{f \in \mathcal{F}_\Lambda} R_{emp}[f] = \varphi(Q[f_\Lambda]).$$

En la figura 1.5, la curva, con trazos continuo, representa las soluciones de los problemas de minimización. En ella, se puede especificar la clase de funciones del modelo, $\mathcal{F}_\Lambda = \{f \in \mathcal{F} \text{ tal que } Q[f] \leq Q[f_1]\}$ y obtener como solución f_1 con riesgo empírico $R_{emp}(f_1)$, de esta forma se tendría un problema del tipo (1.30). Si se fija el máximo riesgo empírico que se está dispuesto a asumir se obtiene como solución la función f_3 que pertenece a un conjunto de funciones $\mathcal{F}_\Lambda = \{f \in \mathcal{F} \text{ tal que } Q[f] \leq Q[f_3]\}$, de esta forma se tendría un problema del tipo (1.32). Por último si se determina la relación entre $R_{emp}[f]$ y $Q[f]$ de antemano a través del factor λ se obtiene como solución f_2. En el último caso, λ (coincide con la tangente del ángulo β) es la pendiente negativa de $R_{emp}[f]$ con respecto a $Q[f]$, en la solución óptima: allí donde la derivada del $R_{reg,emp}[f]$ respecto de $Q[f]$ es nula, puesto que

$$\frac{\triangle R_{reg,emp}[f_\Lambda]}{\triangle Q[f_\Lambda]} = \frac{\triangle R_{emp}[f_\Lambda]}{\triangle Q[f_\Lambda]} + \lambda \cdot \frac{\triangle Q[f_\Lambda]}{\triangle Q[f_\Lambda]} = \frac{\triangle R_{emp}[f_\Lambda]}{\triangle Q[f_\Lambda]} + \lambda = 0. \tag{1.33}$$

En otras palabras, se busca una función f' tal que disminuya el riesgo empírico, $R_{emp}[f]$ en alguna cantidad δ y aumente $Q[f]$ en $\dfrac{\delta}{\lambda}$. Esta condición sobre la tangente es única si todos los funcionales son estrictamente convexos, con lo cual se garantiza que la solución óptima al problema es única para un λ fijo y se puede vislumbrar un

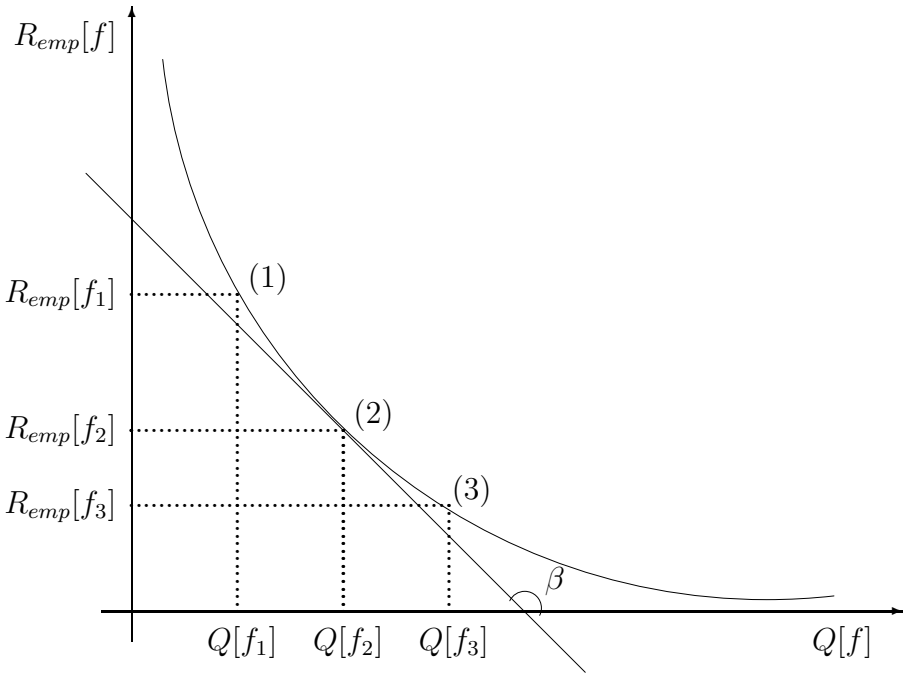

Figura 1.5: Tres algoritmos de minimización de riesgos representados en un mismo gráfico. Los números entre paréntesis indican las soluciones a los problemas planteados.

algoritmo para alcanzar tal solución, sin más que ver como varía el riesgo empírico frente a diferentes conjuntos de funciones. Cuando no se tenga la derivabilidad de $R_{emp}[f]$ sobre un conjunto finito, uno puede aún utilizar las propiedades que presentan las funciones convexas (ver [Rud79]) de $R_{emp}[f]$ para obtener afirmaciones similares.

Finalmente se pueden buscar otras formas de combinar $R_{emp}[f]$ y $Q[f]$ en un funcional global para su posterior minimización, en lugar de una simple combinación lineal como la expresada en el riesgo empírico regularizado. Hay dos razones que hacen que no se deba considerar esta posibilidad. Primero, combinaciones no lineales, en la mayoría de las situaciones, provocan que la minimización del funcional sea más complicada desde el punto de vista teórico y algorítmico. En segundo

lugar, dentro de los cálculos iniciales del factor de relación λ (=constante de regularización) se debe ir variando este parámetro, con objeto de encontrar una clase de funciones particulares (asociada con λ) que proporcione mejores capacidades de generalización. De aquí que una nueva forma (y posiblemente más sutil) de minimizar el funcional riesgo regularizado no tendría la misma influencia sobre la clase de funciones elegida al final.

1.9 Resumen del capítulo

En este capítulo se han puesto las bases que permiten elaborar una nueva metodología a determinados problemas presentes en la Teoría del Aprendizaje Estadístico. Los conceptos de riesgo, riesgo empírico, riesgo estructural y riesgo regularizado permitirán abordar distintos problemas, entre otros los problemas de clasificación y de regresión, de una manera diferente a como se estudian tradicionalmente.

En este nuevo enfoque teórico la característica más importante es que frente al problema de minimizar un riesgo dentro de un conjunto de funciones se plantea el método de minimización del riesgo estructural (SRM), frente al enfoque clásico que minimiza un riesgo empírico (ERM).

CAPÍTULO 2

PROBLEMA DE CLASIFICACIÓN

El conocer la cantidad de incertidumbre asociada a los datos es la llave para tomar la decisión apropiada. Ello nos permite sopesar las consecuencias de diferentes opciones y escoger una que sea la menos perjudicial.

–C. R. Rao–

En este capítulo damos una visión más detallada de algunos de los conceptos introducidos en el capítulo inicial, para a continuación introducir y desarrollar algunos nuevos. Todo ello se lleva a cabo dentro de los problemas de clasificación, sobre los cuales se realizará un estudio teórico (un estudio exhaustivo del tema puede encontrarse en [DGL96]) con objeto de obtener una expresión de las cotas sobre el riesgo.

En este capítulo se dan expresiones sobre cotas del riesgo en términos de una medida de la capacidad del conjunto de funciones $\mathcal{F}$ (la dimensión de Vapnik-Chervonenkis o dimensión VC). Se realiza un estudio de su significado, el cual no solo será válido para los problemas de clasificación, sino que tiene importancia en otra gran variedad de problemas estadísticos (regresión, estimación de densidades,

análisis de componentes principales, ...). Para este fin se necesita introducir los conceptos de entropía, ϵ-recubrimientos, función de crecimiento, familias de conjuntos con estructuras admisibles, ...

Una vez concluido el estudio sobre la dimensión VC se pasa a dar una interpretación más intuitiva del principio de minimización del riesgo estructural y de las cotas obtenidas, todo ello relacionado con los problemas de clasificación.

2.1 Problema de clasificación

El problema de clasificación, tanto si es un problema de reconocimiento de patrones como si es un problema de regresión ordinal, puede formularse de la siguiente forma:

> Un supervisor (ver sección 1.1, página 10) clasifica las situaciones producidas en un determinado entorno según una de las ℓ-clases diferentes en que se pueden diferenciar. El problema consiste en imitar la acción del supervisor, de tal forma que se elija un determinado modo de actuar (función) que se "acerque" lo más posible a las salidas proporcionadas por el supervisor.

Utilizando un lenguaje más formal el problema queda como sigue:

> En un cierto "entorno" se observa un conjunto X de vectores aleatorios independientes e idénticamente distribuidos, caracterizados por una función de distribución $F_{\mathcal{X}}(x)$. Un supervisor clasifica cada vector x de acuerdo a una de las posibles ℓ-clases distintas. Para ello se supone que el supervisor lleva a cabo esta clasificación usando una función de distribución condicional $F_{\mathcal{Y}/\mathcal{X}}(y)$, donde[(1)] $y \in \{0, 1, \cdots, \ell - 1\}$ y el objetivo

[(1)] Etiquetamos cada clase con un valor entero. Aunque en los problemas de reconocimiento de patrones no se debe tener en cuenta el orden que implícitamente aparece cuando usamos números, en los problemas de regresión ordinal sí es adecuado tener en cuenta un determinado orden entre estos números.

es buscar una función que aproxime, en cierto sentido, a esta función de distribución condicional.

En este problema nunca se conocen las funciones de distribución $F_{\mathcal{X}}(x)$ y $F_{\mathcal{Y}/\mathcal{X}}(y)$, pero supondremos que dichas funciones existen y no varían a lo largo del proceso de aprendizaje que se lleve a cabo. Por tanto, se supone la existencia de la distribución conjunta $F(x, y) = F_{\mathcal{Y}/\mathcal{X}}(y) \cdot F_{\mathcal{X}}(x)$. Todo lo anterior nos lleva a encontrarnos con un problema cuya formulación cuadra perfectamente con lo expuesto en el capítulo 1.

Entre los diferentes problemas reales que plantean un problema de clasificación citamos los dos siguientes:

Ejemplo 2.1 *Uno de los problemas de reconocimiento de patrones que más interés ha suscitado dentro de la teoría del aprendizaje es el estudio de la base de datos de la U. S. Postal (servicios postales de los Estados Unidos de América) sobre reconocimiento de caracteres. Se puede encontrar un estudio exhaustivo de este problema en varios de los libros y artículos referenciados en la bibliografía, pero destacamos el trabajo que se realiza en la tesis doctoral de Corinna Cortes [Cor95].* ▲

Ejemplo 2.2 *Dentro del marco de la Economía Internacional y Nacional, existen empresas (empresas de "rating") cuya actividad principal es la de realizar clasificaciones de todo tipo. Una de las clasificaciones más interesante es la que se realiza sobre la solvencia de las empresas de un determinado país*[(2)].

Cuando una empresa solicita un crédito a una entidad bancaria, ésta suele acudir a una empresa de rating, la cual a partir de la documentación que dispone de la empresa la ubica en una de las ℓ clases por ella considerada o las consideradas estándar en el correspondiente entorno. Por supuesto, esta información le supone a

[(2)]Se puede encontrar un estudio de este tipo de problema en [Ang01]

la entidad bancaria, no solo un determinado desembolso de dinero sino también un periodo de tiempo, más o menos largo, de espera en la toma de decisión.

En este caso, el problema que se plantea, por parte de la entidad bancaria, es la búsqueda de un adecuado programa informático que le permita, a partir de un conjunto de datos de la empresa, determinar en que clase de una adecuada clasificación se encuentra, con objeto, por ejemplo, de aceptar o denegar la solicitud de un crédito. ▲

En estos problemas, el conjunto de funciones admisibles[3] $\mathcal{F} = \{f(x,\alpha);\ \alpha \in \Lambda\}$, lo determinan funciones que solamente pueden tomar ℓ valores diferentes. Dentro de este contexto es posible definir distintos tipos de funciones de pérdidas, pero es común para llevar a cabo un estudio teórico considerar la función de pérdida[4]:

$$c(z,\alpha) = c(x,y,f(x,\alpha)) = \begin{cases} 0 & si \quad y = f(x,\alpha) \\ 1 & si \quad y \neq f(x,\alpha) \end{cases} \tag{2.1}$$

que cumple claramente las condiciones que se indican en la nota 1.2.2. Esta función se denomina función test en los análisis estadísticos clásicos. Como ya se ha indicado se podrían haber elegido otras funciones de pérdidas, pero esta función en concreto permite abordar el problema de clasificación, de tal forma que posibilita su generalización no solo a otras funciones sino también a otros problemas diferentes[5].

El problema de clasificación queda, utilizando la notación introducida en el capítulo 1, como sigue:

Minimizar el funcional

$$R(\alpha) = \int c(x,y,f(x,\alpha))\, dF(x,y) \tag{2.2}$$

[3] Como ya hemos comentado el conjunto de funciones es sinónimo de máquina de aprendizaje, pero en un contexto estadístico también se le denomina conjunto de hipótesis.

[4] Que por otro lado es la más simple que se puede considerar.

[5] Como los problemas de regresión (ver [Vap98] y [Gon00]).

sobre el conjunto de funciones $\mathcal{F} = \{f(x,\alpha);\, \alpha \in \Lambda\}$*, cuyo recorrido es* $\{0, 1, \cdots, \ell - 1\}$ *y donde la función de distribución* $F(x,y)$ *es desconocida pero se dispone de información sobre ella a través de una muestra aleatoria independiente,*

$$(x_1, y_1), \cdots, (x_n, y_n) \tag{2.3}$$

donde $y_i \in \{0, 1, \cdots, \ell - 1\}$.

A partir de aquí, y salvo que expresamente se indique lo contrario, se considera el caso (más simple) de dos posibles clasificaciones (caso de dicotomías[6]), es decir, $y \in \{0, 1\}$ (o $y \in \{-1, 1\}$ según sea más adecuado). Se realiza inicialmente este supuesto ya que la generalización al caso de ℓ clasificaciones (multiclasificación) se sigue en función del método elegido para realizar la clasificación, como posteriormente se verá.

Nota 2.1.1 *Si se considera un problema de clasificación dicotómico con etiquetas denotadas por* $\{-1, 1\}$ *entonces la función de pérdida (2.1) se puede expresar de forma compacta, como se puede comprobar fácilmente, de la forma que sigue:*

$$c(x, y, f(x,\alpha)) = \frac{1}{2}\left(1 - y \cdot f(x,\alpha)\right) \tag{2.4}$$

o

$$c(x, y, f(x,\alpha)) = \frac{1}{2}\left|y - f(x,\alpha)\right|.$$

▲

2.1.1 Funcionales riesgos y funciones indicadoras

El objeto de este apartado es proporcionar una nueva interpretación del significado de riesgo y riesgo empírico cuando se trabaja con un problema de clasificación. Para

[6] Si-No, Aceptar-Rechazar, Encendido-Apagado, Comprar-No comprar, Invertir-No invertir, Masculino-Femenino,... .

ello, sea la clase de funciones $\mathcal{F}$ formada por todas las funciones reales que solo pueden tomar los valores $\{0, 1\}$. Entonces, si se considera el espacio input $\mathcal{X} = \mathbb{R}^d$ se tiene que para cada $f \in \mathcal{F}$ existe un conjunto $A \subset \mathbb{R}^d$ tal que

$$f(x) = \begin{cases} 1 & \text{si } x \in A \\ 0 & \text{si } x \notin A \end{cases}$$

luego $f(x) = I_A(x)$, es decir, toda función de $\mathcal{F}$ es una función indicadora[7] de un conjunto $A \subset \mathbb{R}^d$.

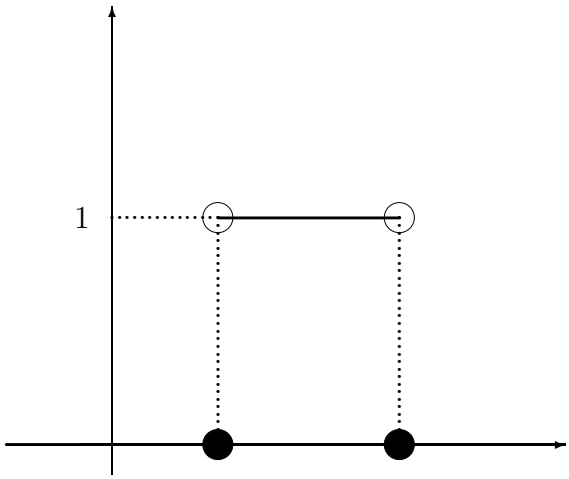

Figura 2.1: Ejemplo de una función indicadora de un intervalo abierto.

Como se ha indicado, en este tipo de problema los valores outputs son $y = 0$ ó $y = 1$, por tanto se tiene que, para α dado, si $f(x, \alpha) = y$ entonces $c(x, y, f(x, \alpha)) = 0$ y si $f(x, \alpha) \neq y$ entonces $c(x, y, f(x, \alpha)) = 1$. De donde resulta que el conjunto de funciones $c_{\mathcal{F}}$ también está formado por un conjunto de funciones indicadoras con lo que:

[7] La función indicadora de un conjunto es una función que toma el valor 1 en todos los elementos del conjunto y cero en el resto.

$$\begin{aligned} R(\alpha) &= \int c(x,y,f(x,\alpha))\, dF(x,y) \\ &= \int I_{\left\{(x,y)\in \mathbb{R}^{d+1} \,/\, c(x,y,f(x,\alpha))=1\right\}}(x,y)\, dF(x,y) \\ &= P\left\{(x,y) \in \mathbb{R}^{d+1} \,/\, c(x,y,f(x,\alpha)) > 0\right\} \\ &= P\left\{(x,y) \in \mathbb{R}^{d+1} \,/\, f(x,\alpha)) \neq y\right\}, \end{aligned} \tag{2.5}$$

y si se denota $A_\alpha = \left\{(x,y) \in \mathbb{R}^{d+1} \,/\, f(x,\alpha)) \neq y\right\}$, se tiene que el funcional riesgo coincide con una probabilidad, $R(\alpha) = P\{A_\alpha\}$, respecto a una medida de probabilidad conjunta[8]. Si se fija α se tiene que $A_\alpha \in \mathbb{R}^{d+1}$ es el conjunto de vectores donde la función $f(x,\alpha)$ realiza una clasificación errónea. Por tanto, las igualdades de (2.5) nos indican que el funcional (2.2) determina la probabilidad de llevar a cabo una clasificación errónea para una determinada función $f(x,\alpha) \in \mathcal{F}$; y de aquí que el objetivo sea buscar aquella función del conjunto $\mathcal{F}$ que proporcione la menor probabilidad de una clasificación errónea, es decir, determinar el índice $\alpha_0 \in \Lambda$ tal que $P(A_{\alpha_0}) = \min_{\alpha\in\Lambda} P(A_\alpha)$.

Nota 2.1.2 *En los contraste de hipótesis, una vez elegido un estadístico se trabaja con dos tipos de errores, error tipo I (α)*[9] *y error tipo II (β) y se habla de potencia de un contraste ($1-\beta$). En el estudio que se está abordando, si se considera la hipótesis nula $H_o : y = f(x,\alpha)$, se esta buscando dentro de un conjunto de funciones*[10] *$\mathcal{F}$ aquella que proporciona un error tipo β, lo menor posible y con ello estamos maximizando la potencia del contraste.* ▲

Por otro lado, para dar una nueva interpretación del funcional riesgo empírico se necesitan introducir algunos conceptos nuevos:

[8] Se supondrá que la medida de probabilidad conjunta es tal que los conjuntos de la forma A_α pertenezcan a la σ-álgebra correspondiente.

[9] Utilizamos α para denotar dos conceptos diferentes, un error y un índice de funciones, pero consideramos no conduce a equivoco y de esta forma mantenemos la notación tradicional.

[10] Supuesto que la función $f(x,\alpha) \in \mathcal{F}$ determina una variable aleatoria como función de la muestra.

Definición 2.1.3 (Probabilidad empírica) *Sea un espacio probabilístico* $(\Omega, \mathcal{A}, P)$ *del cual se obtiene una muestra aleatoria de tamaño* n

$$z_1, \cdots, z_n. \tag{2.6}$$

Para cada suceso fijado A *del* $\sigma-$*álgebra* $\mathcal{A}$*, se considera el valor*

$$v_n(A) \stackrel{def}{=} v(A; z_1, \cdots, z_n) = \frac{n(A)}{n}$$

donde $n(A)$ *es el número de elementos de la muestra* $\{z_1, \cdots, z_n\}$ *que pertenecen a* A*, es decir,*

$$n(A) = \sum_{i=1}^{n} I_A(z_i).$$

Al valor $v_n(A)$ *se le llama* **probabilidad empírica** *o* **frecuencia relativa** *del suceso* A *en una muestra dada de tamaño* n*.*

Nota 2.1.4 *Se tiene que fijado un suceso* A *y un tamaño muestral* n*, la probabilidad empírica* $v_n(A)$ *es una variable aleatoria discreta que toma los siguientes* $n+1$ *valores:* $\{0, 1/n, 2/n, \cdots, 1\}$. ▲

Nota 2.1.5 *Si se toman los conjuntos* $A_x = (-\infty, x]$ *entonces*

$$v_n(A_x) = \frac{1}{n}\sum_{i=1}^{n} n(A_x) = \frac{1}{n}\sum_{i=1}^{n} I_{(-\infty,0]}(x - x_i) = F_n(x),$$

es decir, coincide con la función de distribución empírica. ▲

De la definición de probabilidad empírica y de la definición de los conjuntos A_α, se tiene que:

$$R_{emp}(\alpha) = \frac{1}{n}\sum_{i=1}^{n} c(x_i, y_i, f(x_i, \alpha)) = \frac{n(A_\alpha)}{n} = v_n(A_\alpha),$$

es decir, el riesgo empírico es la frecuencia relativa de ocurrencia del suceso A_α en una muestra. Luego, según el principio de minimización del riesgo empírico (ERM) el problema que se plantea es $\min\limits_{\alpha\in\Lambda} v_n(A_\alpha)$.

Nota 2.1.6 *Supongamos que se tiene un conjunto de vectores* $\{x_1, \cdots, x_n\}$ *de los cuales se conocen sus correspondientes clasificaciones* $\{y_1, \cdots, y_n\}$. *Si inicialmente nos olvidamos de las clasificaciones y sobre los vectores* $\{x_1, \cdots, x_n\}$ *se aplica ("ensaya") una determinada función* $f(x, \alpha)$ *se obtendrá otro nuevo conjunto* $\{y'_1, \cdots, y'_n\}$, *que serán las clasificaciones que resultan de aplicar la función* $f(x, \alpha)$. *Entonces cuando se tenga para algún* $i \in \{1, \cdots, n\}$ *que* $y_i \neq y'_i$ *se dirá que se ha producido un* **error de ensayo** *en el vector* x_i.

Señalemos, como lo concluido hasta el momento coincide con el planteamiento clásico de los problemas de clasificación, donde el objetivo principal es la minimización de la probabilidad de los errores de ensayo. La diferencia radica en los estimadores que resulten, ya que en este caso los estimadores no serán necesariamente paramétricos. ▲

2.2 Entropía. Dimensión de Vapnik-Chervonenkis

Una vez concluida la interpretación del riesgo y el riesgo empírico en términos de probabilidades, se está en condiciones de abordar una de las principales aportaciones de este nuevo esquema, el concepto de medida de la capacidad de un conjunto de funciones.

2.2.1 Entropía de un conjunto de funciones indicadoras

En el apartado 1.6 se ha indicado que las clases de funciones adecuadas para resolver el problema de minimización del riesgo empírico son aquellas que cumplen una ley generalizada de los grandes números, la cual se expresa en la forma:

$$P\left\{\sup_{f\in\mathcal{F}} |R[f] - R_{emp}[f]| \geq \varepsilon\right\} \to 0, \qquad \text{cuando } n \to \infty \text{ y } \varepsilon > 0. \tag{2.7}$$

Teniendo en cuenta la interpretación de los funcionales riesgo y riesgo empírico como una probabilidad y su correspondiente frecuencia relativa, y aplicando el teo-

rema de Bernoulli (una ley débil de los grandes números) al suceso A_α, con α fijado, se tiene que la sucesión de frecuencias converge a la probabilidad cuando el número de observaciones tiende a infinito. Además por la desigualdad Chernoff se tiene:

$$P\{|R(\alpha) - R_{emp}(\alpha)| \geq \varepsilon\} = P\{|P(A_\alpha) - v_n(A_\alpha)| \geq \varepsilon\} \leq 2\exp(-2\varepsilon^2 n) \qquad (2.8)$$

que describe la razón de convergencia.

Si el conjunto de funciones $\mathcal{F}$ contiene un número finito N de elementos, entonces se tendrán N sucesos de la forma A_α, es decir, a partir de la clase de funciones $\mathcal{F}$, se construye una clase finita de sucesos de la forma

$$\mathcal{A}_\mathcal{F} = \{A_1, \cdots, A_N\}$$

y se tiene de la desigualdad de Chernoff que:

$$\begin{aligned} P\left\{\sup_{f\in\mathcal{F}} |R[f] - R_{emp}[f]| \geq \varepsilon\right\} &= P\left\{\max_{1\leq i\leq N} |P(A_i) - v_n(A_i)| \geq \varepsilon\right\} \\ &\leq \sum_{i=1}^{N} P\{|P(A_i) - v_n(A_i)| \geq \varepsilon\} \\ &\leq 2N\exp(-2\varepsilon^2 n) \\ &= 2\exp\left\{\left(\frac{\ln N}{n} - 2\varepsilon^2\right) n\right\} \qquad (2.9) \end{aligned}$$

La expresión (2.9) indica que para obtener la convergencia uniforme para cualquier $\varepsilon > 0$, es necesario que se cumpla:

$$\frac{\ln N}{n} \to 0, \qquad \text{cuando} \quad n \to \infty \qquad (2.10)$$

ya que de esta forma el término $-2\varepsilon^2$ domina en la exponencial. Pero (2.10) siempre es cierto ya que el número $N < \infty$ de funciones queda fijado a priori cuando se elige $\mathcal{F}$. Por tanto se tiene garantizada la condición de convergencia uniforme (2.7) si la clase de funciones es finita.

Nota 2.2.1 *Es conveniente y útil introducir el número de funciones de $\mathcal{F}$, N, dentro de la exponencial en la igualdad (2.9) ya que en los desarrollos de las demostraciones de los resultados siguientes, la condición dada sobre el conjunto de funciones se expresa siguiendo este camino.* ▲

El próximo objetivo consiste en buscar condiciones bajo las cuales una determinada clase de funciones con infinitos elementos cumpla la condición (2.7). Para conseguir esto lo que se hará es, dada una clase de funciones, elegir un conjunto finito de funciones, dentro de ella, que permita garantizar (2.7). Por tanto, se necesita tener algún criterio que permita elegir funciones dentro de un conjunto, y de aquí, las siguientes definiciones.

Definición 2.2.2 *Dos sucesos*[11] *son "distinguibles" entre sí, según una muestra, si existe al menos un elemento de la muestra que pertenece a un suceso pero no pertenece al otro.*

Nota 2.2.3 *Como para cada función $f \in \mathcal{F}$, donde $\mathcal{F}$ es un conjunto de funciones indicadoras se tiene definido un suceso A_f, se puede decir que dos funciones indicadoras $f, g \in \mathcal{F}$ son "distinguibles" si existe algún elemento de la muestra que se encuentra en el suceso A_f pero no en A_g (o al reves), es decir,*

$$f,\, g \in \mathcal{F} \text{ son distinguibles} \quad \Leftrightarrow \quad \{z_1, \cdots, z_n\} \cap A_f \neq \{z_1, \cdots, z_n\} \cap A_g.$$

▲

Ejemplo 2.3 *Sea la muestra $\{0, 1, -1, 2, 0'5, 3\}$ y se considera el conjunto de funciones $\mathcal{F} = \{f : I\!R \to \{-1, 1\} \ / \ f(x) = signo(a\,x + b),\ \text{con } a, b \in I\!R\}$. Sean las funciones $f_1(x) = signo(2x + 3)$ y $f_2(x) = signo(3x + 5)$ se cumple que, a pesar de ser funciones diferentes, a partir de la muestra no podemos diferenciarlas ya que coinciden en todos sus elementos.*

Por otro lado, si se consideran las funciones $f_1(x) = signo(3x - 2)$ y $f_2(x) = signo(3x + 5)$ se tiene que claramente son distinguibles ya que difieren en algún elemento de la muestra ($f_1(0) = -1$ y $f_2(0) = 1$). ▲

[11] O conjuntos.

De la definición de sucesos distinguibles se sigue que, si se dispone de un conjunto de infinitos sucesos (funciones indicadoras), únicamente podemos distinguir un número finito[12] de grupos de sucesos a partir de la muestra $\{z_1, \cdots, z_n\}$, ya que, si se considera el conjunto de vectores n-dimensionales binarios

$$\{ (c(z_1, \alpha), \cdots, c(z_n, \alpha)) ; \quad \alpha \in \Lambda\} \subset \mathbb{R}^n$$

se tiene que para cada α fijo, el vector $(c(z_1, \alpha), \cdots, c(z_n, \alpha))$ determina uno de los vértices del cubo unidad en $\mathbb{R}^n$ (ver figura 2.2); se sigue entonces que el número de vértices no es fijo y depende de la muestra y del conjunto de funciones, por lo que se denota[13]

$$N^{\Lambda}(z_1, \cdots, z_n) = N^{\Lambda}.$$

En la figura 2.2 se puede observar que en el caso tridimensional el número máximo de vectores es $8 = 2^3$. Además se ha de suponer que N^{Λ} es una función medible con respecto a la medida de probabilidad inducida por la muestra.

De la definición de N^{Λ} se sigue que si tiene un valor alto es porque la muestra ha proporcionado una mayor información, donde por información de la muestra se entiende la capacidad de la muestra de permitir distinguir más grupos en el conjunto de funciones (realizar una partición más fina en $\mathcal{F}$). Ya que la muestra es aleatoria, se ha de tener una función que nos permita cuantificar por término medio la información proporcionada por una muestra de tamaño n genérica. Para ello se construye una función, en términos de N^{Λ}, que permite "cuantificar" esta información.

Definición 2.2.4 (Entropía) *A la cantidad*

$$H^{\Lambda}(z_1, \cdots, z_n) = \ln N^{\Lambda}(z_1, \cdots, z_n)$$

[12] Como máximo 2^n si n es el tamaño muestral.

[13] Tradicionalmente se denota utilizando el conjunto de índices en lugar de $N^{\mathcal{F}}$.

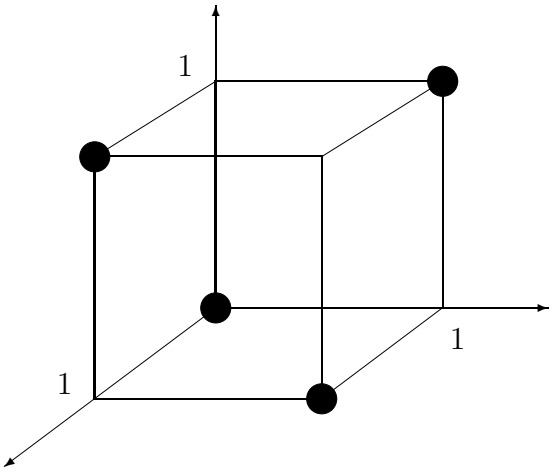

Figura 2.2: Un conjunto formado por cuatro vectores binarios tridimensionales.

se le denomina entropía aleatoria del conjunto $\{f(x,\alpha);\, \alpha \in \Lambda\}$ *de funciones indicadoras sobre la muestra* $\{z_1, \cdots, z_n\}$ *distribuida según la distribución conjunta* $F_{\mathcal{Z}}(\cdot)$.

También se dirá que la cantidad

$$H^{\Lambda}(n) = \int H^{\Lambda}(z_1, \cdots, z_n)\, dF_{\mathcal{Z}}(z_1, \cdots, z_n)$$

es la **entropía** *del conjunto de funciones indicadoras* $\mathcal{F} = \{f(x,\alpha);\, \alpha \in \Lambda\}$ *sobre todas las muestras de tamaño* n *distribuida según la distribución conjunta* $F_{\mathcal{Z}}(\cdot)$.

Nota 2.2.5 *En Mecánica Estadística, la entropía es una medida del desorden de un sistema; en Teoría de la Información expresa una medida del grado de incertidumbre asociada con una fuente de mensaje; en Teoría del Aprendizaje es una medida del número de funciones del conjunto* $\mathcal{F}$ *distinguibles por una muestra de tamaño* n.

La diferencia fundamental radica en que, en este enfoque, la entropía se basa en evaluar logaritmos de números de grupos (particiones) en lugar de trabajar con probabilidades. El que se definan con el mismo nombre se debe a que las propiedades

que presentan son muy similares. Destaquemos que en las demostraciones de los teoremas claves, las ideas que se utilizan se basan en ideas análogas desarrolladas para la entropía de Shannon de Teoría de la Información. Esto no es una coincidencia ya que V.N. Kolmogorov ha tenido un papel clave en todos estos estudios teóricos. ▲

Bajo las apropiadas condiciones de medibilidad del conjunto de funciones $\mathcal{F}$ se tienen los siguientes teoremas[14], cuya demostraciones pueden encontrarse en el capítulo 14, páginas 584-586 de [Vap98].

Teorema 2.2.6 *La convergencia uniforme, dada por*

$$P\left\{\sup_{f\in\mathcal{F}}|R[f]-R_{emp}[f]|\geq\varepsilon\right\}\to 0, \qquad \textit{cuando } n\to\infty \textit{ y } \varepsilon>0,$$

sobre el conjunto de funciones indicadoras $\mathcal{F}=\{f(x,\alpha);\ \alpha\in\Lambda\}$ *se cumple si y solo si*

$$\frac{H^{\Lambda}(n)}{n}\underset{n\to\infty}{\longrightarrow} 0. \tag{2.11}$$

Se puede conseguir una condición más fuerte con las hipótesis de este teorema y es que la convergencia uniforme se tenga casi segura, es decir,

Teorema 2.2.7 *Bajos las condiciones del teorema anterior se garantiza*

$$\sup_{\alpha\in\Lambda}\left|\int c(z,\alpha)\,dF(z)-\frac{1}{n}\sum_{i=1}^{n}c(z_i,\alpha)\right|\underset{n\to\infty}{\longrightarrow} 0 \qquad \textit{(casi seguro).}$$

Nótese que lo que realmente se esta haciendo es sustituir la condición obtenida cuando el número de funciones de $\mathcal{F}$ es finito:

$$\frac{\ln N}{n}\to 0, \qquad \text{cuando} \quad n\to\infty,$$

[14] A lo largo del presente capítulo aparecen enunciados distintos teoremas que no se demuestran pero que se indican donde se encuentran sus demostraciones. Si se actúa de esta forma es por hacer el texto más legíble ya que las demostraciones son en general de carácter técnico, largas y en ningún modo intuitivas.

por el número medio de funciones distinguibles, $H^{\Lambda}(n)$, correspondiente a cada tamaño muestral[15].

Para completar el modelo conceptual de la teoría del aprendizaje se necesita responder a dos preguntas adicionales:

1. ¿Cuáles son las condiciones bajo las cuales se garantiza la existencia de una rápida (con cota exponencial) razón de convergencia uniforme para una medida de probabilidad dada?

2. ¿Cuáles son las condiciones bajo las cuales se garantiza la existencia de una rápida razón de convergencia uniforme para cualquier medida de probabilidad?

En cualquier desarrollo práctico de una teoría basada en muestras es habitual pedir una rápida razón de convergencia con objeto de "garantizar" que los resultados deducidos de la muestra se acerquen a los poblacionales a partir de un "adecuado" tamaño muestral[16]. Para responder a la primera pregunta hemos de describir condiciones para la existencia de dos constantes positivas $a,\ b \in \mathbb{R}$ tal que para un tamaño muestral n suficientemente grande, $n > n(\varepsilon, \mathcal{F}, F_{\mathcal{Z}})$ se cumpla:

$$P\left\{\sup_{\alpha\in\Lambda}\left|\int c(z,\alpha)\,dF(z) - \frac{1}{n}\sum_{i=1}^{n} c(z_i,\alpha)\right| > \varepsilon\right\} < b\,\exp\left\{-a\,\varepsilon^2 n\right\} \tag{2.12}$$

o escrito en forma de riesgos:

$$P\left\{\sup_{f\in\mathcal{F}}|R[f] - R_{emp}[f]| > \varepsilon\right\} < b\,\exp\left\{-a\,\varepsilon^2 n\right\}. \tag{2.13}$$

Para responder a la segunda cuestión, sería igual con $n > n(\varepsilon, \mathcal{F})$ (no depende

[15] La interpretación rigurosa no es ésta ya que la entropía no es el número medio de las funciones distinguibles, sino el número medio del logaritmo neperiano de las funciones distinguibles, pero su interpretación es más intuitiva de esta forma.

[16] En [Gon98] se puede encontrar un sencillo ejemplo de la razón de convergencia de un determinado algoritmo muy simple.

de la distribución seguida por la muestra, $F_{\mathcal{Z}}$) y[17]

$$\sup_{F\in\mathcal{P}} P\left\{\sup_{\alpha\in\Lambda}\left|\int c(z,\alpha)\,dF(z) - \frac{1}{n}\sum_{i=1}^{n} c(z_i,\alpha)\right| > \varepsilon\right\} < b\,\exp\left\{-a\,\varepsilon^2 n\right\} \tag{2.14}$$

o escrito en forma de riesgos:

$$\sup_{F\in\mathcal{P}} P\left\{\sup_{f\in\mathcal{F}}|R[f] - R_{emp}[f]| > \varepsilon\right\} < b\,\exp\left\{-a\,\varepsilon^2 n\right\}. \tag{2.15}$$

Se dan respuestas a estas dos preguntas en el siguiente apartado, basándonos en un concepto clave sobre la capacidad de un conjunto de funciones implementada por la máquina de aprendizaje.

2.2.2 Tres importantes resultados en los problemas de clasificación

El logro más importante de la parte conceptual en teoría del aprendizaje es el hecho de introducir el concepto de **capacidad**, el cual *describe* completamente *el comportamiento cualitativo del proceso de aprendizaje (la consistencia)* y como se verá, la robusta característica de este concepto define también *las singularidades cuantitativas*: las cotas no asintóticas sobre la razón de convergencia, tanto para el caso de procesos de aprendizajes dependientes de la distribución como para aquellos independientes de la distribución.

Así, en la sección anterior se ha considerado, la variable aleatoria

$$N^{\Lambda}(z_1,\cdots,z_n)$$

la cual se ha definido como el número de grupos (separaciones) diferentes que se puede determinar a partir de los datos $\{z_1,\cdots,z_n\}$ y el conjunto $\{f(x,\alpha);\,\alpha\in\Lambda\}$; y a partir de esta variable aleatoria se considera la entropía

$$H^{\Lambda}(n) = E[\ln N^{\Lambda}(z_1,\cdots,z_n)].$$

A continuación se definen dos nuevos conceptos relacionados con estos:

[17] Donde $\mathcal{P}$ denota el conjunto de todas las distribuciones posibles.

Definición 2.2.8 (Entropía suave) *Se define la* **entropía suave** *– del inglés entropy annealed– del conjunto de funciones indicadoras* $\{f(x,\alpha);\, \alpha \in \Lambda\}$ *sobre todas las muestras de tamaño* n *según una distribución* $F_{\mathcal{Z}}$*, como*

$$H^{\Lambda}_{ann}(n) = \ln E[N^{\Lambda}(z_1, \cdots, z_n)].$$

Definición 2.2.9 (Función de crecimiento) *Se define la* **función de crecimiento** *del conjunto* $\{f(x,\alpha);\, \alpha \in \Lambda\}$ *de funciones indicadoras sobre todas las muestras de tamaño* n *como*

$$G^{\Lambda}(n) = \ln \sup_{z_1,\cdots,z_n} N^{\Lambda}(z_1, \cdots, z_n).$$

Nota 2.2.10 *De las anteriores definiciones se sigue que tanto la entropía como la entropía suave dependen de la medida de probabilidad que está subyacente en los datos, ya que se necesita para el cálculo de la esperanza matemática. Sin embargo, la función de crecimiento depende del conjunto* $\mathcal{Z}$ *pero es independiente de la distribución que da origen a los datos.* ▲

Lema 2.2.11 *Entre los conceptos anteriores se tiene la siguiente relación de orden:*

$$H^{\Lambda}(n) \leq H^{\Lambda}_{ann}(n) \leq G^{\Lambda}(n), \qquad \forall\, n \in I\!N.$$

Demostración. La primera desigualdad se sigue a partir de la función $\varphi(x) = \ln x$ que es una función cóncava y que la esperanza matemática $E[X]$ es un operador lineal, entonces aplicando la desigualdad de Jensen se tiene:

$$E[\varphi(X)] \leq \varphi(E[X])$$

y sustituyendo φ por ln, y X por $N^{\Lambda}(z_1, \cdots, z_n)$ se tiene

$$E[\ln N^{\Lambda}(z_1, \cdots, z_n)] \leq \ln E[N^{\Lambda}(z_1, \cdots, z_n)]$$

luego

$$H^{\Lambda}(n) \leq H^{\Lambda}_{ann}(n), \qquad \forall\, n \in \mathbb{N}.$$

La segunda desigualdad se sigue de la siguiente forma:

$$N^{\Lambda}(z_1, \cdots, z_n) \leq \sup_{z_1, \cdots, z_n} N^{\Lambda}(z_1, \cdots, z_n)$$

y tomando valor esperado

$$E[N^{\Lambda}(z_1, \cdots, z_n)] \leq \sup_{z_1, \cdots, z_n} N^{\Lambda}(z_1, \cdots, z_n)$$

y aplicando logaritmos

$$\ln E[N^{\Lambda}(z_1, \cdots, z_n)] \leq \ln \sup_{z_1, \cdots, z_n} N^{\Lambda}(z_1, \cdots, z_n)$$

por lo que

$$H^{\Lambda}_{ann}(n) \leq G^{\Lambda}(n), \qquad \forall\, n \in \mathbb{N}.$$

■

Sobre la base de estas tres funciones se construyen los principales logros de la teoría del aprendizaje.

1. La condición
$$\lim_{n \to \infty} \frac{H^{\Lambda}(n)}{n} = 0$$
es suficiente para la consistencia del principio de minimización del riesgo empírico (depende de la distribución conjunta $F_{\mathcal{Z}}$).

2. La condición
$$\lim_{n \to \infty} \frac{H^{\Lambda}_{ann}(n)}{n} = 0$$
es suficiente para una rápida razón de convergencia definida por la condición
$$P\left\{ \sup_{\alpha \in \Lambda} \left| \int c(z, \alpha)\, dF(z) - \frac{1}{n} \sum_{i=1}^{n} c(z_i, \alpha) \right| > \varepsilon \right\} < b\, \exp\left\{ -a\, \varepsilon^2 n \right\}$$
(depende de la distribución conjunta $F_{\mathcal{Z}}$).

3. La condición
$$\lim_{n\to\infty} \frac{G^{\Lambda}(n)}{n} = 0$$
es necesaria y suficiente para la consistencia del principio ERM para cualquier medida de probabilidad (no depende de la distribución conjunta $F_{\mathcal{Z}}$). Además, es también la condición necesaria y suficiente bajo la cual, la máquina de aprendizaje que implementa el principio ERM, tiene una razón de convergencia asintótica grande, independientemente de la medida de probabilidad.

2.2.3 Cotas sobre la capacidad de generalización en los problemas de clasificación

Las cotas que se exponen a continuación juegan un papel muy importante en la capacidad de generalización, donde por **capacidad de generalización** se entiende la capacidad que tiene una máquina de aprendizaje para poder generalizar los resultados obtenidos para una muestra concreta a un conjunto más amplio. Este es el motivo por lo que suelen llamar cotas sobre la capacidad de generalización.

En el capítulo 4 de [Vap98], se demuestra que dada una muestra aleatoria independiente
$$(x_1, y_1), \cdots, (x_n, y_n)$$
donde $y_i \in \{0, 1\}$ y para un valor $(0 < \eta < 1)$ fijado se tiene con probabilidad al menos $(1 - \eta)$, la siguiente cota sobre el riesgo:
$$R(\alpha_n) \leq R_{emp}(\alpha_n) + \frac{\varepsilon(n)}{2}\left(1 + \sqrt{1 + \frac{4\, R_{emp}(\alpha_n)}{\varepsilon(n)}}\right) \tag{2.16}$$
y con probabilidad al menos $1 - 2\eta$ se tiene:
$$\Delta(\alpha_n) = R(\alpha_n) - R(\alpha_0) \leq \frac{\varepsilon(n)}{2}\left(1 + \sqrt{1 + \frac{4\, R_{emp}(\alpha_n)}{\varepsilon(n)}}\right) + \sqrt{\frac{-\ln \eta}{2\, n}} \tag{2.17}$$
donde se denota por α_n el identificador de la función que minimiza el riesgo empírico en $\mathcal{F}$ a partir de una muestra aleatoria independiente de tamaño n, es decir,
$$\alpha_n = \arg\min_{\alpha\in\Lambda} R_{emp}(\alpha)$$

y por α_0 el identificador de la función que minimiza el riesgo en $\mathcal{F}$, es decir,

$$\alpha_0 = \arg \min_{\alpha \in \Lambda} R(\alpha).$$

Por $\varepsilon(n)$ se denota una función que depende del tamaño muestral que hace cierta las desigualdades (2.16) y (2.17); y cuya forma funcional vendrá dada a partir de los conceptos introducidos en las secciones anteriores como sigue:

1. Si se considera
$$\varepsilon(n) = 4\frac{H^{\Lambda}_{ann}(2n) - \ln(\eta/4)}{n}$$
se tiene la menor cota posible dependiente de la distribución que es válida para una máquina de aprendizaje (conjunto de funciones) y un problema específico (una medida concreta de probabilidad).

2. Si se considera
$$\varepsilon(n) = 4\frac{G^{\Lambda}(2n) - \ln(\eta/4)}{n}$$
se tiene una cota independiente de la distribución válida para una máquina de aprendizaje y cualquier problema (cualquier medida de probabilidad).

Sin embargo, estas dos cotas son más conceptuales que prácticas ya que no se dispone de una metodología adecuada para el cálculo de la entropía suave y la función de crecimiento bajo condiciones generales. Por ello, es necesario buscar alguna expresión de $\varepsilon(n)$ que permita una cota más fácil de implementar y que se cumpla en condiciones muy generales.

2.2.4 Dimensión VC

Se comienza estudiando la función de crecimiento de un conjunto de funciones indicadoras a partir del siguiente teorema cuya demostración aparece en [Vap98, pág. 150-155].

Teorema 2.2.12 *La función de crecimiento $G^{\Lambda}(n)$ de un conjunto de funciones indicadoras $\{f(x,\alpha);\ \alpha \in \Lambda\}$ toma el valor*

$$G^{\Lambda}(n) = n\ \ln 2, \qquad \forall n \in I\!N$$

o esta acotada por la desigualdad

$$G^{\Lambda}(n) = \begin{cases} = n\ \ln 2 & si\ n \leq h \\ \leq \ln\left(\sum_{i=0}^{h} C_n^i\right) \leq \ln\left(\frac{e \cdot n}{h}\right)^h = h\left(1 + \ln\frac{n}{h}\right) & si\ n > h \end{cases}$$

donde[18] *h es el mayor entero para el cual se cumple $G^{\Lambda}(h) = h\ \ln 2$ y $e = 2'7182...$ (número de Euler).*

El teorema afirma que el conjunto de funciones indicadoras puede agruparse en dos categorías diferentes (ver figura 2.3):

1. Conjunto de funciones indicadoras con función de crecimiento lineal.
2. Conjunto de funciones indicadoras con función de crecimiento acotada logarítmicamente con coeficiente h.

Definición 2.2.13 (dimensión VC) *La capacidad de un conjunto de funciones indicadoras con función de crecimiento acotada logarítmicamente puede ser caracterizada por el coeficiente h. El coeficiente h se denomina* **dimensión de Vapnik-Chervonenkis** *(dimensión VC) de un conjunto de funciones indicadoras. Cuando la función de crecimiento es lineal, la dimensión VC se define como infinito.*

El cálculo directo de la dimensión VC a partir de la definición es complicado; por ello se da una definición equivalente que proporciona un criterio para construir la dimensión VC de cualquier conjunto de funciones:

[18] C_n^m denota el número de combinatorio n sobre m, es decir, $C_n^m = \begin{pmatrix} n \\ m \end{pmatrix} = \frac{n!}{m!\,(n-m)!}$.

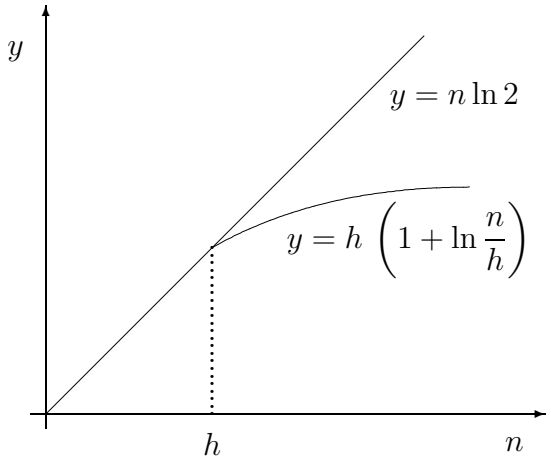

Figura 2.3: Representación gráfica de una función de crecimiento

Definición 2.2.14 *Dado un conjunto* $\{f(x, \alpha);\ \alpha \in \Lambda\}$ *de funciones indicadoras su dimensión VC es igual al mayor número* h *de vectores* $\{z_1, \cdots, z_n\}$ *que pueden ser separados en dos clases diferentes en todas las* 2^h *maneras posibles usando el conjunto de funciones, es decir, la dimensión VC es el mayor número de vectores que pueden ser separados completamente por el conjunto de funciones.*

Nota 2.2.15 *La equivalencia entre las dos definiciones se obtiene directamente a partir de la demostración del teorema 2.2.12.*

Como se sigue de la definición, la dimensión VC de un conjunto de funciones es un número entero, $h \in \{1, 2, 3, 4, 5, \cdots\}$. ▲

Por tanto, para calcular del conjunto de funciones $\{f(x, \alpha);\ \alpha \in \Lambda\}$ su dimensión VC, es suficiente indicar el número máximo de vectores $\{z_1, \cdots, z_n\}$ que pueden separarse completamente por este conjunto de funciones.

Según el teorema 2.2.12 si la dimensión VC de un conjunto de funciones indicadoras es finita, la desigualdad

$$\max_{z_1,\cdots,z_n} N^{\Lambda}(z_1,\cdots,z_n) \leq \sum_{i=0}^{h} C_n^i \tag{2.18}$$

se cumple, donde h es el máximo número tal que

$$\max_{z_1,\cdots,z_h} N^{\Lambda}(z_1,\cdots,z_h) = 2^h.$$

Lema 2.2.16 *La cota obtenida de la función de crecimiento es la menor posible.*

Demostración. Para llevar a cabo la demostración se da un ejemplo donde se alcanza la cota. Sea Z un subconjunto de $\mathbb{R}$,

$$S = \{A \subset Z / \, A \text{ tiene menos de h+1 elementos distintos}\}$$

y sea un conjunto de funciones $\mathcal{F}$ donde las funciones tienen la siguiente forma[19]:

$$f(z,\alpha(A)) = \begin{cases} 1 & \text{si} \quad z \in A \\ -1 & \text{si} \quad z \in Z \setminus A \end{cases}$$

y se tiene que

$$\max_{z_1,\cdots,z_n} N^{\Lambda}(z_1,\cdots,z_n) = \begin{cases} 2^n & \text{si } n \leq h \\ \displaystyle\sum_{i=0}^{h} C_n^i & \text{si } n > h. \end{cases}$$

Aclaremos con un ejemplo considerando el caso $h = 2$ con $Z = \mathbb{R}$. En este caso S es la familia de todos los conjuntos de $\mathbb{R}$ que tiene 0, 1 ó 2 elementos. Si se tiene una muestra formada por 2 elementos $\{z_1, z_2\}$ las funciones indicadoras nos debe discriminar las cuatro posibilidades $\{(+,+),(+,-),(-,+),(-,-)\}$ y esto se consigue con los conjuntos de S siguientes: $A_{(+,+)} = \{z_1, z_2\}$, $A_{(+,-)} = \{z_1, a\}$, $A_{(-,+)} = \{a, z_2\}$ y $A_{(-,-)} = \{a\}$, donde a es un elemento de $\mathbb{R}$ distinto de z_1 y z_2. Sin embargo, si tenemos una muestra de tamaño 3, habría 8 posibilidades distintas

[19] Nótese la forma de los identificadores.

y es posible encontrar funciones indicadoras para todas las posibilidades salvo para $(+,+,+)$. Luego en este caso se tendría que

$$\max_{z_1,z_2,z_3} N^{\Lambda}(z_1,z_2,z_3) = 7 = 1+3+3 = \sum_{i=0}^{2} \begin{pmatrix} 3 \\ i \end{pmatrix} = \sum_{i=0}^{2} C_3^i$$

■

Ejemplo 2.4 *Sea el conjunto* $\{f(x,\alpha);\, \alpha \in \Lambda\}$ *donde* $x = (x_1, x_2) \in I\!R^2$ *y* $f(x,\alpha) = signo(a\,x_1 + b\,x_2 + c)$, *con* $\alpha = (a,b,c) \in I\!R^3$ *y donde la función signo*[20] *se define como:*

$$signo(x) = \begin{cases} 1 & si\ x > 0 \\ -1 & si\ x < 0. \end{cases}$$

Dados tres puntos no alineados siempre es posible encontrar una recta que realice una correcta clasificación, como podemos ver en la figura 2.4. Sin embargo, dado cuatro puntos cualesquiera en $I\!R^2$ *es imposible realizar todas las posibles clasificaciones (ver figura 2.5). De ello se deduce que la dimensión VC del conjunto* $\{f(x,\alpha);\, \alpha \in \Lambda\}$ *es 3, que coincide con el número de parámetros del conjunto de funciones.* ▲

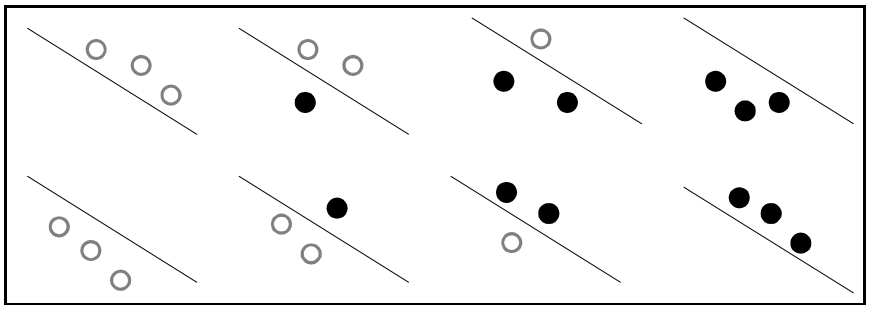

Figura 2.4: Tres puntos no alineados en $\mathbb{R}^2$, pueden ser separados siempre por una recta orientada. El punto hueco representa el valor $y = 1$ y el punto relleno el valor $y = -1$.

[20] Se considera las etiquetas $\{-1, 1\}$.

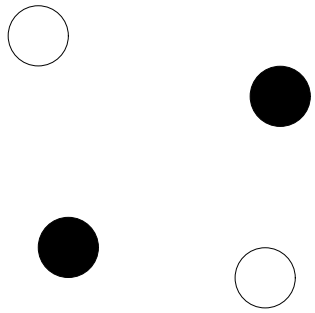

Figura 2.5: Ejemplo de la imposibilidad encontrar una recta que separe cuatro puntos no alineados adecuadamente.

Una idea que es fundamental para los capítulos posteriores es la siguiente: si se considera el problema anterior con $X = \{(1,1),(-1,1),(1,-1),(-1,-1)\}$, los cuatro puntos con etiquetas $Y = \{1,-1,-1,1\}$, se puede elegir un espacio de dimensión superior $\mathbb{R}^3$ y una función adecuada, $f(x_1,x_2) = (x_1,x_2,x_1 \cdot x_2)$, de tal forma que se encuentra un plano $\pi : z = 0$ que divide adecuadamente los cuatro puntos como se puede ver en la figura 2.6.

El ejemplo 2.4 es generalizado en [Bur96] de la siguiente forma:

Teorema 2.2.17 *Sea un conjunto cualquiera de n puntos en $\mathbb{R}^d$ y se elige un punto cualesquiera como origen. Entonces los n puntos pueden ser divididos por hiperplanos orientados*[21] *si y solo si los restantes $n-1$ vectores construidos con los restantes $n-1$ puntos y el origen son linealmente independientes.*

[21] El término hiperplano orientado enfatiza el objeto matemático considerado por el par $\{H,v\}$, donde H es un conjunto de puntos los cuales están en un hiperplano con vector director unitario v. Así $\{H,v\}$ y $\{H,-v\}$ determinan los dos hiperplanos orientados, es decir, el conjunto de puntos de los hiperplanos es el mismo pero las dos regiones en que dividen el espacio cada uno de los hiperplanos tienen signos alternos. Intuitivamente es como si sobre una de las caras del hiperplano se situase un observador el cual considera positivos todos los puntos que se encuentra en la misma región que él, y los puntos que se encuentran en la otra región toman valores negativos (ver figura 2.7).

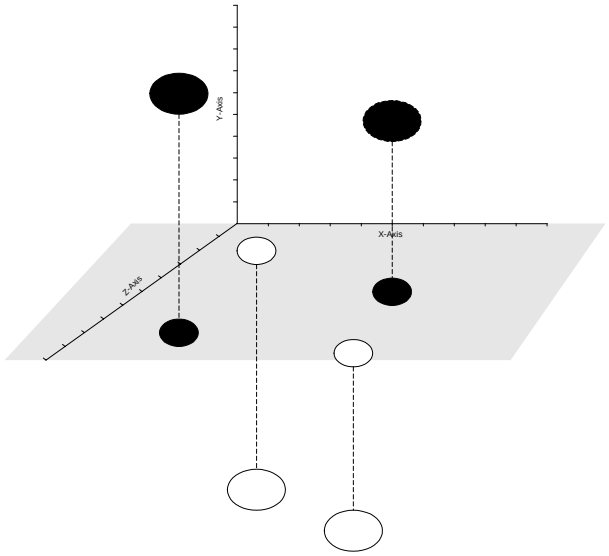

Figura 2.6: Separación de un conjunto de cuatro puntos, no separables en el plano $\mathbb{R}^2$, en el espacio $\mathbb{R}^3$.

Y como consecuencia se tiene:

Corolario 2.2.18 *La dimensión VC del conjunto de hiperplanos orientados en $\mathbb{R}^d$ es $d+1$.*

Demostración. Es evidente, ya que la dimensión de $\mathbb{R}^d$ es d y si se eligen d vectores libres del plano que sean independientes y se centran en el origen, permiten determinar d extremos más el origen que configuran los $d+1$ puntos máximos que pueden ser separados por hiperplanos. ■

La dimensión VC da una noción muy concreta de la capacidad de una clase determinada de funciones. Intuitivamente se puede pensar que una máquina de aprendizaje con muchos parámetros tendrá una alta dimensión VC, mientras que una máquina de aprendizaje con pocos parámetros tendrá dimensión VC pequeña. Sin embargo en [Cor95, Vap95] se da un contraejemplo de esto, debido a E. Levin y

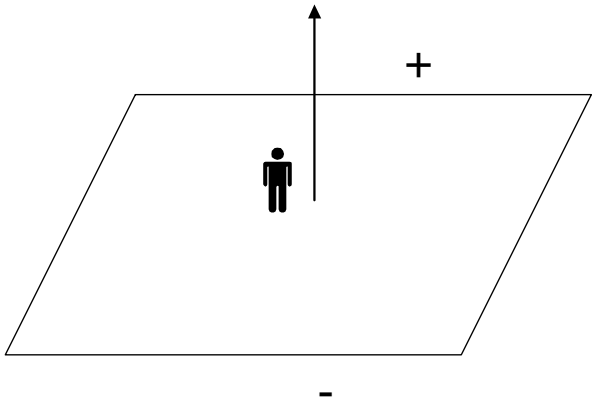

Figura 2.7: Interpretación gráfica de lo que significa un hiperplano orientado en $\mathbb{R}^3$.

J.S. Denker: una máquina con un único parámetro, pero con dimensión VC infinita, es decir, para cualquier tamaño n se encuentra una muestra tal, que las funciones de la clase separa adecuadamente los elementos. El contraejemplo es el siguiente:

Ejemplo 2.5 *Sea la clase de funciones, definida por*

$$f(x,\alpha) = signo(\text{sen}(\alpha\, x)), \qquad x,\, \alpha \in I\!R.$$

Eligiendo cualquier valor natural n, se considera

$$x_i = 10^{-i}, \quad i = 1, \cdots, n$$

y cualquier etiqueta

$$y_1, \cdots, y_n, \qquad y_i \in \{-1, 1\}, \quad i = 1, \cdots, n.$$

Entonces la función $f(x,\alpha)$ proporciona estas etiquetas, si se elige

$$\alpha = \pi \left(1 + \sum_{i=1}^{n} \frac{(1 - y_i)\, 10^i}{2} \right).$$

Por tanto la dimensión VC de esta máquina es infinita ya que conseguimos separar adecuadamente una muestra de cualquier tamaño.

Resulta interesante notar que a pesar de ser un conjunto de funciones de dimensión VC infinita, es posible construir cuatro puntos que no pueden ser divididos adecuadamente, y esto se consigue simplemente tomando los puntos $\{1, 2, 3, 4\}$ *con etiquetas* $\{1, 1, -1, 1\}$. *Veámoslo:*

El valor de $\alpha \in I\!R$ *se puede escribir en la forma*

$$\alpha = 2n\pi + \delta$$

luego para $x_1 = 1$ *se tiene que* $signo(\text{sen}(\alpha x_1)) = signo(\text{sen}(\delta))$ *y para que sea* 1, *se debe cumplir que* $0 < \delta < \pi$. *Para* $x_2 = 2$ *se tiene que* $signo(\text{sen}(\alpha x_2)) = signo(\text{sen}(2\delta)$ *y para que sea* 1 *se debe cumplir que* $0 < \delta < \frac{\pi}{2}$. *Para* $x_3 = 3$ *se tiene que* $signo(\text{sen}(\alpha x_3)) = signo(\text{sen}(3\delta)$ *y para que sea* -1 *se debe cumplir que* $\delta > \frac{\pi}{3}$. *Para* $x_4 = 4$ *se tiene que* $signo(\text{sen}(\alpha x_4)) = signo(\text{sen}(4\delta)$ *y para que sea* 1 *se debe cumplir que* $0 < \delta < \frac{\pi}{4}$, *lo cual lleva a contradicción ya que* $\delta > \frac{\pi}{3}$ *y* $0 < \delta < \frac{\pi}{4}$ *es imposible. Por tanto, se han encontrado 4 puntos que no pueden ser separados adecuadamente por ninguna función de la forma*

$$f(x, \alpha) = signo(\text{sen}(\alpha\, x)), \qquad x,\ \alpha \in I\!R.$$

▲

Para la determinación de la dimensión VC se necesita un determinado tamaño muestral y una clase de funciones, lo que significa que no depende de la medida de probabilidad subyacente a la muestra, es por ello que la dimensión VC juega un papel fundamental en la obtención de una cota libre de la distribución (no depende de la medida de probabilidad desconocida) que puede construirse para evaluar el riesgo a partir de un conjunto de datos y resolver el problema generalizado de Glivenko-Cantelli.

2.2.5 Comentarios sobre el principio de minimización del riesgo estructural

Una vez indicadas algunas cotas sobre la capacidad de generalización en la sección 2.2.3 y la posterior interpretación de las cantidades involucradas en ellas, se tiene una nueva cota de la capacidad de generalización obtenida a partir de la acotación de la función de crecimiento por una función de la dimensión VC. Sea

$$\varepsilon(n) = 4\frac{h(\ln(2n/h)+1) - \ln(\eta/4)}{n}$$

entonces si se denota $\tau = n/h$, se tiene que $\varepsilon(n)$ se expresa como:

$$\varepsilon(n) = 4\left(\frac{\ln(2\tau)+1}{\tau} - \frac{\ln(\eta/4)}{n}\right).$$

Esta última expresión muestra que la capacidad de generalización de la máquina de aprendizaje depende de la razón entre el número de observaciones y la dimensión VC del conjunto de funciones[(22)], ya que para un η razonable el segundo término en la expresión es significativamente más pequeño que el primero. Por tanto, un tema de estudio en este enfoque es la determinación de la dimensión VC para determinados conjuntos de funciones $\{f(x,\alpha);\ \alpha \in \Lambda\}$.

Nota 2.2.19 *Algunos autores indican que esta cota es demasiado ancha en la mayoría de los problemas y señalan que sería más adecuado, si se dispone de información a priori sobre la distribución, atacar el problema directamente y pasar a calcular la entropía y obtener a partir de ella una cota más estrecha (ajustada). Para ver como se aborda este problema en algunos casos se puede ver el estudio que Smola realiza en su tesis doctoral [Smo98].* ▲

La expresión general de la cota de generalización queda entonces como sigue: dado un conjunto de funciones $\mathcal{F} = \{f(x,\alpha);\ \alpha \in \Lambda\}$, con dimensión VC finita, h,

[(22)] Por ello se dice que una muestra es pequeña si la razón n/h es pequeña.

un conjunto de datos dados por (2.3) y un valor $0 < \eta < 1$ fijado. Se tiene con probabilidad al menos $(1 - \eta)$:

$$R(\alpha_n) \leq R_{emp}(\alpha_n) + 2\frac{h\left(\ln\frac{2n}{h} + 1\right) - \ln\frac{\eta}{4}}{n}\left(1 + \sqrt{1 + \frac{n\,R_{emp}(\alpha_n)}{4\left(h\left(\ln\frac{2n}{h} + 1\right) - \ln\frac{\eta}{4}\right)}}\right) \tag{2.19}$$

De los anteriores comentarios se sigue que esta cota no depende de la distribución conjunta $F(x, y)$, luego es válida para cualquier máquina de aprendizaje, independientemente del problema de clasificación que se pretenda resolver. Aplicando el principio de minimización del riesgo estructural se busca la menor de las cotas posibles cuando tanto n como η están fijados. La cota entonces depende de la dimensión VC y del mínimo del riesgo empírico para la muestra dada.

Si se tiene una muestra, podemos plantear un problema equivalente, como se ha visto en el capítulo anterior, buscando la minimización de un funcional riesgo regularizado, donde se obtenga la relación óptima entre dimensión VC y riesgo empírico. Una vez resuelto esto, el cálculo de la cota es inmediato si conocemos la función $f(x, \alpha_n)$. Por tanto, dentro de los problemas de clasificación, se plantea la búsqueda de la función $f(x, \alpha_n)$, la cual es objeto de estudio en la segunda parte del trabajo, que comienza en el capítulo 3.

2.3 Resumen del capítulo

Este capítulo es útil para sentar las bases de lo que serán las máquinas de aprendizaje para el problema de clasificación que utilizan el principio de minimización del riesgo estructural, que se denominan **Máquinas de Vectores Soporte** (del inglés –*Support Vector Machine*–) o **Máquinas de Soporte Vectorial**.

En esta sección hemos estudiado la teoría sobre las cotas de la teoría aprendizaje estadístico asociadas a los problemas de clasificación. En esta teoría aparecen cotas sobre la capacidad de generalización que permiten construir en los siguientes

capítulos unas máquinas de aprendizaje que son consistentes, es decir, se pueden construir unos algoritmos, los cuales, basándose en las cotas sobre la capacidad, son convergentes con una velocidad de convergencia alta.

En [CST00] se pueden encontrar otras teorías sobre como llegar a cotas sobre la capacidad de generalización.

Por tanto, y una vez resuelto el problema teórico de existencia de solución de un determinado problema se plantea su resolución práctica en la segunda parte de este trabajo.

PARTE II

Máquinas Núcleos de Vectores Soporte para Clasificación

CAPÍTULO 3

MÁQUINAS DE VECTORES SOPORTE PARA BI-CLASIFICAR

Los descubrimientos matemáticos, grandes o pequeños, nunca nacen por generación espontánea. Siempre presuponen una tierra plantada con el conocimiento preliminar y bien preparado por medio del trabajo tanto consciente como subconsciente.

–J. H. Poincaré–

Este capítulo aborda el estudio de los problemas de clasificación desde el punto de vista práctico. Como ya se indicó en el capítulo 1, el método que consiste en minimizar el riesgo empírico no resulta adecuado y, por ello, en la resolución numérica de estos tipos de problemas se plantea la minimización de un determinado riesgo regularizado, el cual está definido de forma aditiva a partir de dos funcionales, el propio riesgo empírico y un funcional, que hace las veces de suavizador de las posibles soluciones al problema. Así pues, utilizando el principio de minimización del riesgo estructural (SRM) se construye un método de aprendizaje que nos permite

resolver el problema de aprendizaje a partir de ejemplos.

Para ello, se resuelve el problema de minimización del riesgo estructural de forma indirecta para el caso de los problemas de clasificación, es decir, se plantea un problema de optimización y posteriormente se comprueba que ciertamente se trata de un problema de minimización de un riesgo regularizado. Se aborda así el estudio de las máquinas lineales de vectores soporte (la clase de funciones $\mathcal{F}$ está formada por funciones lineales, Support Vector Machine (SVM) lineales) para los problemas de clasificación tanto para el caso de datos separables como de datos no separables (los cuales serán posteriormente definidos).

De esta forma, se entra de lleno en el planteamiento y resolución de determinados problemas de optimización convexa con restricciones, también convexas. En estos tipos de problemas de optimización se verifican las condiciones de Karush-Kuhn-Tucker (KKT), las cuales dentro de este marco de trabajo son cruciales, ya que será la base que permita, dentro del conjunto de ensayo, obtener un subconjunto particularmente importante, que denominaremos "vectores soporte" – del inglés *Support Vector* (SV)–.

Una vez resuelto los problemas de optimización convexa planteados a partir de un conjunto $\mathcal{F}$ de funciones lineales y a la vista de la estructura de la solución, se generalizará, de manera natural, a conjuntos de funciones no lineales (máquinas de vectores soporte no lineales – SVM no lineales). En esta generalización, la idea clave es la de sumergir el espacio de los inputs ($\mathcal{X}$) dentro de un espacio, dotado de un producto escalar, $\mathcal{H}$ de dimensión superior (que se denomina, espacio característico), utilizando, para ello, una aplicación Φ de $\mathcal{X}$ en $\mathcal{H}$ con unas determinadas propiedades. A partir de esta aplicación Φ se definirá una función real definida en $\mathcal{X} \times \mathcal{X}$ que se denominará función **núcleo** – del inglés *kernel*–, y se denotará por k, la cual nos permitirá expresar la solución del caso no lineal como una continuación lógica del caso lineal. Se indica en este capítulo las ideas básicas de estos espacios característicos así como las de los núcleos y se pospone su estudio más detallado para el capítulo sexto.

Finalizamos este capítulo con un breve resumen del análisis discriminante, con objeto de elegir el método de validación de esta técnica estadística como método de validación de las SVMs.

3.1 Máquinas lineales de vectores soporte

Comenzamos estudiando la máquina de vectores soporte más simple, con objeto de asimilar los conceptos fundamentales involucrados en ellas, para poder comprender mejor su significado cuando se aborden casos más complejos.

3.1.1 El caso separable

Consideramos, como en el capítulo 2, problemas de clasificación con solo dos etiquetas posibles que se denotarán por $Y = \{-1, 1\}$ y donde el conjunto de vectores de ensayo

$$\{(x_1, y_1), \cdots, (x_n, y_n)\}, \qquad x_i \in \mathbb{R}^d,\ y_i \in \{-1, 1\}, \quad i = 1, \cdots, n \tag{3.1}$$

tiene la propiedad de ser un conjunto separable, siendo la definición de conjunto separable la siguiente:

Definición 3.1.1 (de conjunto separable) *Un conjunto de vectores*

$$\{(x_1, y_1), \cdots, (x_n, y_n)\}, \qquad x_i \in I\!R^d,\ y_i \in \{-1, 1\}, \quad i = 1, \cdots, n$$

se dice **separable** *si existe algún hiperplano (llamado hiperplano separable) en $I\!R^d$ que separa, en el sentido de dejar en dos regiones del espacio diferentes, los vectores $X = \{x_1, \cdots, x_n\}$ con valor[(1)] de $y_i = 1$ de aquellos con valor de $y_i = -1$.*

Nota 3.1.2 *En [SWSST99] se da una definición más general de conjunto separable: "un conjunto $\{x_1, \cdots, x_n\}$ se dice separable si existe algún $w \in I\!R^d$ tal que $\langle w, x_i \rangle >$*

[(1)] También se suele decir de etiqueta "$y_i = 1$".

0 *para* $i = 1, \cdots, n$*", y se demuestra que la definición 3.1.1 es un caso particular de ésta. Para los objetivos de este trabajo es más adecuado dar la definición 3.1.1, ya que proporciona una visión más adecuada de los problemas que se plantean, puesto que en ella se relacionan los vectores inputs con los valores outputs (en la versión más general los vectores inputs aparecen sin etiquetas).* ▲

Por tanto, sea un conjunto de vectores separable

$$\{(x_1, y_1), \cdots, (x_n, y_n)\}, \qquad x_i \in \mathbb{R}^d,\ y_i \in \{-1, 1\}, \quad i = 1, \cdots, n,$$

entonces, de la propia definición, se tiene garantizada la existencia de un hiperplano que separa los datos[(2)],

$$\pi :\ w \cdot x + d = 0$$

(ver figura 3.1), donde $w \in \mathbb{R}^d$ es un vector normal (vector perpendicular) al hiperplano π, $|d| \,/\, \|w\|$ es la distancia perpendicular del hiperplano π al origen, y $\|w\|$ es la norma euclidea de w.

Sea $\mathbf{A}$ la región delimitada por el hiperplano π tal que $x \cdot w + d > 0$, y $\mathbf{B}$ la región delimitada por π tal que $x \cdot w + d < 0$. Se consideran los valores reales

$$\begin{aligned} d_- &= \quad \text{mín}\,\{x_i \cdot w + d,\ x_i \in \mathbf{A}\} \\ d_+ &= -\,\text{máx}\,\{x_i \cdot w + d,\ x_i \in \mathbf{B}\}. \end{aligned}$$

Si suponemos que dentro del conjunto de ensayo existen vectores inputs con etiquetas[(3)] 1 y -1, es decir, $\{x_i \cdot w + d,\ x_i \in \mathbf{A}\} \neq \emptyset$ y $\{x_i \cdot w + d,\ x_i \in \mathbf{B}\} \neq \emptyset$ entonces de la definición 3.1.1, se tiene garantizada la existencia de los valores d_- y d_+ y se sigue que: $-d_+ < 0 < d_-$, ya que para todo $x_i \in \mathbf{A}$ se tiene que

[(2)]Se utilizará indistintamente la notación $w \cdot x$ y $\langle w, x\rangle$ para denotar el producto escalar de dos vectores. Abreviadamente utilizaremos $w \cdot x$ en lugar de $w \cdot x^t$ donde x^t indica el vector traspuesto si dicha notación no lleva a confusión.

[(3)]Si no se realiza este supuesto se trabajaría con un conjunto de ensayo que solo presenta una etiqueta con lo cual estaríamos ante un problema poco realista. Por otro lado, desde el punto de vista operativo, no se tendría ninguna característica para poder discriminar la segunda etiqueta y se tendría un problema irresoluble.

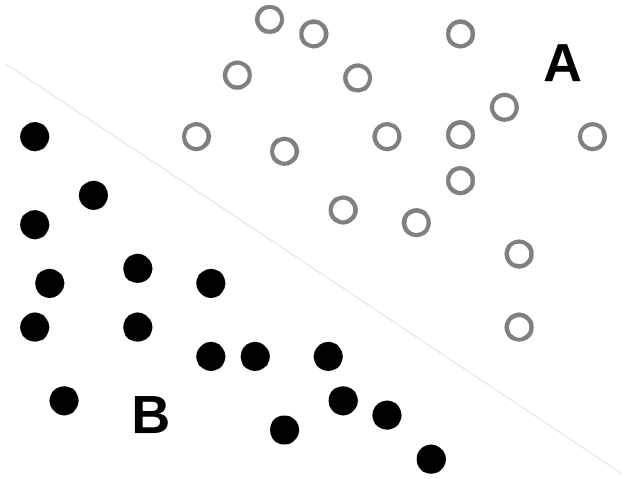

Figura 3.1: Conjunto de puntos separables e hiperplano separable en el plano real. Los puntos huecos representan los datos con etiqueta $y = 1$ y los puntos rellenos los datos con etiqueta $y = -1$.

$x_i \cdot w + d > 0$ y como a lo más hay n vectores diferentes se sigue que el mínimo se alcanza dentro del conjunto y por tanto si éste se denota por $x_{i,\mathbf{A}}$ se tiene que: $d_- = \text{mín}\{x_i \cdot w + d,\ x_i \in \mathbf{A}\} = x_{i,\mathbf{A}} \cdot w + d > 0$.
Por otro lado, para todo $x_i \in \mathbf{B}$ se tiene que: $x_i \cdot w + d < 0$ y como a lo más hay n vectores diferentes se sigue que el máximo se alcanza dentro del conjunto y por tanto si éste se denota por $x_{i,\mathbf{B}}$ se tiene que $d_+ = -\,\text{máx}\{x_i \cdot w + d,\ x_i \in \mathbf{B}\} = -(x_{i,\mathbf{B}} \cdot w + d) > 0$ y de aquí que $-d_+ < 0$.

El valor $d' = d_- + d_+$ nos proporciona el **margen** de un hiperplano separable[(4)].

En la región $\mathbf{A}$ se tiene:

$$x_i \cdot w + d \geq d_- \Rightarrow x_i \cdot w + d - \left(\frac{d_- - d_+}{2}\right) \geq d_- - \left(\frac{d_- - d_+}{2}\right)$$

lo que implica que:

$$x_i \cdot w + d_1 \geq \left(\frac{d_- + d_+}{2}\right)$$

[(4)]El significado de margen será comprensible en los desarrollos siguientes.

donde $d_1 = d - \left(\frac{d_- - d_+}{2}\right)$ y dividiendo ambos miembros de la desigualdad por $\left(\frac{d_- + d_+}{2}\right)$, que claramente es distinto de cero, se tiene:

$$x_i \cdot w' + b' \geq 1 \qquad \text{en la región } \mathbf{A}.$$

Análogamente se obtiene que[5]:

$$x_i \cdot w' + b' \leq 1 \qquad \text{en la región } \mathbf{B}.$$

Por tanto los vectores[6] de ensayo satisfacen las condiciones[7]:

$$x_i \cdot w + b \geq +1 \qquad \text{para } y_i = +1 \tag{3.2}$$

$$x_i \cdot w + b \leq -1 \qquad \text{para } y_i = -1 \tag{3.3}$$

donde

$$w = \frac{2\,w'}{d_- + d_+} \quad \text{y} \quad b = \frac{d - \left(\frac{d_- - d_+}{2}\right)}{\frac{d_- + d_+}{2}} \ .$$

Se simplifican estas dos desigualdades en una única de la forma:

$$y_i \left(x_i \cdot w + b\right) - 1 \geq 0, \qquad i = 1, \cdots, n. \tag{3.4}$$

Sean, entonces los puntos para los cuales se cumple la igualdad en (3.2). Estos puntos pertenecen al hiperplano $\pi_1 : \ x \cdot w + b = 1$, con vector normal w y distancia perpendicular hasta el origen igual a $|1 - b| \,/\, \|w\|$. Análogamente, los puntos que cumplen la igualdad de (3.3) pertenecen al hiperplano $\pi_2 : \ x \cdot w + b = -1$ con vector normal w y distancia perpendicular hasta el origen igual a $|-1 - b| \,/\, \|w\|$. De todo ello se tiene que los hiperplanos π, π_1 y π_2 son paralelos y que: $d_+ = d_- = 1/\,\|w\|$;

[5] Es curioso que esta demostración no la hemos encontrado en ninguno de los trabajos que hemos leído lo que nos parece extraño ya que es muy simple y tiene mucha importancia en los siguientes desarrollos.

[6] Ya que a lo largo de los siguientes temas se utilizará mucho la representación gráfica, se llamaran indistintamente puntos o vectores.

[7] Con objeto de no introducir nuevas constantes renombramos w' y b' como w y b.

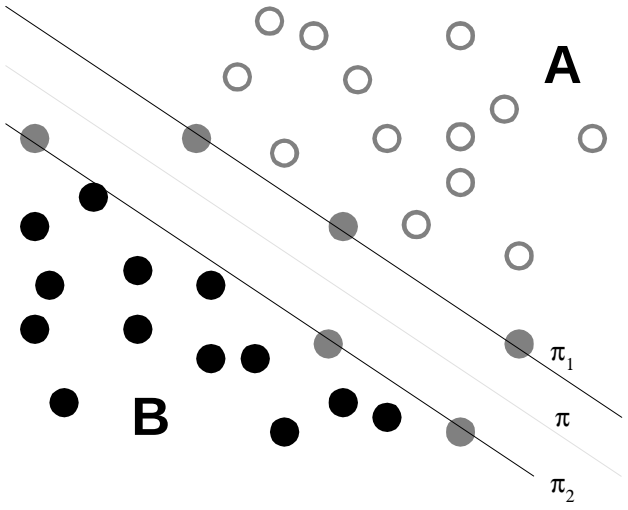

Figura 3.2: Hiperplanos paralelos π, π_1 y π_2 y un conjunto de vectores de ensayo separable en $\mathbb{R}^2$. Los posibles vectores soporte se indican con relleno en gris. El margen es la distancia entre el hiperplano π y el π_1 (o el π_2),

y el margen de separación entre ellos es $d' = 2/\|w\|$. Nótese que los hiperplanos π_1 y π_2 además de ser paralelos, ya que tienen el mismo vector normal, cumplen que no existe ningún vector del conjunto de ensayo que se encuentre entre ellos dos (ver figura 3.2).

De entre todas las posibles elecciones de los hiperplanos π_1 y π_2, parece natural elegir aquella que proporcione una mayor separación entre ellos, ya que de esta forma permitiría discriminar de forma más clara cada una de las regiones donde caen los puntos con distintas etiquetas. De esta forma, se plantea el problema de encontrar un par de hiperplanos π_1 y π_2, los cuales den el máximo margen (minimizando $\|w\|$ o su cuadrado), sujetos a las restricciones (3.4), es decir,

$$\begin{aligned} &\min_{w\in\mathbb{R}^d} \frac{1}{2} \|w\|^2 \\ &s.a. \quad y_i \left(x_i \cdot w + b\right) - 1 \geq 0, \quad \forall i = 1, 2, \cdots, n. \end{aligned} \tag{3.5}$$

La solución para el caso de dimensión dos se puede interpretar gráficamente a partir de la figura 3.2. Obsérvese que algunos de los vectores de ensayo para los cuales se cumple la igualdad de (3.4) se encuentran sobre el hiperplano π_1 o sobre el hiperplano π_2. Algunos de estos vectores son llamados **vectores soporte** (los posibles vectores soporte se indican con relleno en gris en la figura 3.2), y se puede ver como son los únicos que intervienen en la definición de los hiperplanos separadores ya que si sobre la figura 3.2 se mueven los restantes vectores, sin desplazar ninguno entre los hiperplanos π_1 y π_2, los hiperplanos separadores no cambian, es decir, la solución del problema 3.5 permanece invariante.

Hay que hacer notar que si se añade o elimina cualquier número de puntos que cumplan la desigualdad estricta (3.4), la solución no se ve afectada, sin embargo, basta con añadir un punto que se encuentre entre los hiperplanos π_1 y π_2, para que la solución cambie totalmente. Esta propiedad es la característica principal que permite elaborar una algoritmo de búsqueda que obtenga rápidamente la solución al problema (3.5), ya que, si se considera una solución lineal en los inputs, los pesos de los inputs que verifican la desigualdad estricta serían nulos, puesto que no afectan a la posible solución, y los que se encuentran sobre los hiperplanos separadores son cero o distintos de cero (según que su inclusión se utilice en la expresión de la solución del hiperplano separador[(8)]).

Para la resolución del problema de optimización con restricciones (3.5) se utiliza la técnica de los multiplicadores de Lagrange. Esto se hace por dos razones. La primera es que las restricciones (3.4) que resultan de este nuevo problema quedan como restricciones sobre los multiplicadores de Lagrange, las cuales resultan más fáciles de manejar. La segunda razón es que, en esta reformulación del problema, la solución aparece expresada en términos de los vectores inputs como productos escalares entre ellos. Esta propiedad es crucial y permite generalizar el problema para el caso de máquinas de aprendizajes no lineales.

[(8)] Se comenta con más detalle en la nota 3.1.3.

Sean los multiplicadores de Lagrange[9] positivos $\alpha_i,\ i = 1, \cdots n$, uno para cada una de las n restricciones de (3.4) con la siguiente función objetivo[10]:

$$\begin{aligned} L_P(w, b, \alpha_i) &= \frac{1}{2}\|w\|^2 - \sum_{i=1}^{n} \alpha_i \ (y_i\,(x_i \cdot w + b) - 1) \\ &= \frac{1}{2}\|w\|^2 - \sum_{i=1}^{n} \alpha_i\, y_i\,(x_i \cdot w + b) + \sum_{i=1}^{n} \alpha_i. \end{aligned} \tag{3.6}$$

De esta forma, el problema que se debe resolver es minimizar la función $L_P(w, b, \alpha_i)$ respecto a w y b, y simultáneamente requerir que las derivadas parciales de L_P con respecto a los multiplicadores de Lagrange α_i sean todas nulas, todo ello sujeto al conjunto de restricciones $C_1 = \{\alpha_i \geq 0,\ i = 1, \cdots, n\}$; a este problema se le denomina **problema primal**. Así, el problema inicial queda como un problema de programación cuadrática donde la función objetivo es convexa, y los puntos que satisfacen las restricciones forman un conjunto convexo. Esto significa que se puede resolver el siguiente **problema dual** asociado al problema primal: maximizar la función $L_P(w, b, \alpha_i)$ respecto a las variables duales α_i sujeta a las restricciones impuestas para que los gradientes de L_P con respecto a w y b sean nulos, y sujeta también al conjunto de restricciones $C_2 = \{\alpha_i \geq 0,\ i = 1, \cdots, n\}$. Esta particular formulación del problema dual se denomina **problema dual de Wolfe** [Fle87], y verifica que el máximo de $L_P(w, b, \alpha_i)$ respecto de la variables duales, sujeto a las restricciones C_2, coincide con los mismos valores para w, b y α_i, que el mínimo de $L_P(w, b, \alpha_i)$ respecto a w y b sujeto a las restricciones de C_1, es decir, la solución al problema planteado es un punto silla de la función $L_P(w, b, \alpha_i)$. Sobre este tema puede encontrarse un desarrollo muy adecuado en [CST00].

Sea $w = (w_1, \cdots, w_d) \in \mathbb{R}^d$ un vector y $b \in \mathbb{R}$ ambos desconocidos, y $x_i = (x_{i1}, \cdots, x_{id}) \in \mathbb{R}^d$, $y_i \in \mathbb{R}$, $\forall\, i = 1, \cdots, n$ (donde por x_{ij} denota la componente j-ésima del vector x_i) los vectores de entrenamiento que son conocidos. De esta

[9] También llamados variables duales del problema.

[10] Nótese que se ha utilizado para denotar los multiplicadores de Lagrange el mismo símbolo que se utiliza, en la primera parte de este trabajo, para identificar una determinada función de una clase $\mathcal{F}$, ello es debido a que precisamente estos multiplicadores permiten identificar la solución dentro del conjunto de funciones $\mathcal{F}$ que se disponga.

forma, la función objetivo se escribe:

$$L_P(w,b,\alpha) = \frac{1}{2}\sum_{j=1}^{d} w_j^2 - \sum_{i=1}^{n} \alpha_i\, y_i \left(\sum_{j=1}^{d} x_{ij}\, w_j + b\right) + \sum_{i=1}^{n} \alpha_i$$

donde se trabaja con la norma euclidea. Las derivadas parciales respecto de w_j para $j = 1, \cdots, d$ quedan:

$$\frac{\partial}{\partial w_j} L_P(w,b,\alpha_i) = w_j - \sum_{i=1}^{n} \alpha_i\, y_i\, x_{ij}$$

e igualando a cero y resolviendo, queda:

$$w_j - \sum_{i=1}^{n} \alpha_i\, y_i\, x_{ij} = 0 \Rightarrow w_j = \sum_{i=1}^{n} \alpha_i\, y_i\, x_{ij} \Rightarrow$$

$$w = \sum_{i=1}^{n} \alpha_i\, y_i\, x_i. \tag{3.7}$$

Así una vez conocido los multiplicadores de Lagrange somos capaces de calcular el vector de pesos w. Por otro lado, la derivada parcial de $L_P(w,b,\alpha_i)$ respecto de b resulta:

$$\frac{\partial}{\partial b} L_P(w,b,\alpha_i) = \sum_{i=1}^{n} \alpha_i\, y_i$$

e igualada a cero proporciona la ecuación:

$$\sum_{i=1}^{n} \alpha_i\, y_i = 0. \tag{3.8}$$

Ya que estas restricciones son las mismas para el problema dual, se sustituye en (3.6) y se tiene:

$$\begin{aligned} L_P(w,b,\alpha_i) &= \frac{1}{2}\sum_{i,j=1}^{n} \alpha_i\,\alpha_j\, y_i\, y_j\, x_i\, x_j - \sum_{i=1}^{n} \alpha_i\, y_i\, x_i \sum_{j=1}^{n} \alpha_j\, y_j\, x_j - \sum_{i=1}^{n} \alpha_i\, y_i + \sum_{i=1}^{n} \alpha_i \\ &= \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{n} \alpha_i\,\alpha_j\, y_i\, y_j\, x_i\, x_j \end{aligned}$$

luego la función objetivo dual queda:

$$L_D(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{n} \alpha_i\,\alpha_j\, y_i\, y_j\, x_i\, x_j. \tag{3.9}$$

Hay pues, dos funciones lagrangianas diferentes: L_P y L_D que provienen de la misma función objetivo pero con diferentes restricciones; la solución es buscar el mínimo de $L_P(w, b, \alpha_i)$ respecto de w y b, o el máximo de $L_D(w, b, \alpha_i)$ respecto a α_i, $i = 1, \cdots, n$. Hemos de hacer notar que si se formula el problema con $b = 0$, lo que significa que todos los hiperplanos pasan por el origen, la restricción (3.8) no aparece. Esto es una restricción débil para espacios de dimensión superior pero se puede solventar añadiendo al espacio de dimensión superior una dimensión más[(11)].

Los vectores soporte[(12)] (para el caso separable, caso lineal) por consiguiente maximizan la función $L_D(w, b, \alpha_i)$ con respecto a los α_i, sujeto a la restricción (3.8) y C_2; y la solución viene dada por (3.7). En la solución, algunos de los vectores x_i para los cuales $\alpha_i > 0$ son llamados "vectores soporte" (coincide con la condición que se apuntó anteriormente), y se encuentran en uno de los hiperplanos π_1 o π_2. Para este tipo de máquina de aprendizaje, los vectores soporte son los elementos críticos del conjunto de ensayo, ya que ellos son los que proporcionan la aproximación del problema, puesto que si todos los restantes elementos del conjunto de ensayo son eliminados (o son cambiados por otros que no se encuentren entre los dos hiperplanos) y se repite el problema de optimización, se encuentran los mismos hiperplanos separadores (la misma solución del problema).

Nota 3.1.3 *Es importante notar, en este momento, que a pesar de ser la solución del problema de optimización única, la representación de w en términos de los vectores de ensayo no lo es. Basta con observar en la figura 3.2 que para determinar, por ejemplo, la recta π_1 podemos elegir dos cualesquiera de los tres puntos que están sobre ella, obteniendo claramente 6 formas (variaciones de 3 elementos tomados de 2 en 2) diferentes de expresar w.* ▲

[(11)] En [Bur96] se indica simplemente que en un espacio de dimensión muy alta una dimensión de más afecta poco en la solución final, pero nos parece que lo que se indica en este trabajo es más preciso y adecuado.

[(12)] Nótese que son necesariamente del conjunto de ensayo.

3.1.2 Las condiciones de Karush-Kuhn-Tucker

Como se indicaba en la sección anterior, los únicos vectores que intervienen en la expresión de la solución son los "vectores soporte", los cuales llevan asociado un multiplicador de Lagrange $\alpha_i > 0$ y, además, como se encuentran en uno de los dos hiperplanos separadores, cumplen la igualdad: $y_i \cdot (x_i \cdot w + b) - 1 = 0$, luego se tiene que:

$$\alpha_i \cdot (y_i \cdot (x_i \cdot w + b) - 1) = 0. \tag{3.10}$$

Por otro lado, no todos los vectores que se encuentran en los hiperplanos π_1 o π_2 son vectores soporte, pero es evidente que satisfacen la igualdad (3.10). Los restantes vectores de ensayo cumplen la desigualdad estricta (3.4) y como no son vectores soporte cumplen que sus multiplicadores de Lagrange son nulos ($\alpha_i = 0$) y entonces verifican:

$$\alpha_i = 0, \quad y_i \cdot (x_i \cdot w + b) - 1 > 0$$

y de aquí

$$\alpha_i \cdot (y_i \cdot (x_i \cdot w + b) - 1) = 0.$$

Por tanto se tiene que todos los vectores de ensayo satisfacen un nuevo conjunto de restricciones:

$$\alpha_i \cdot (y_i \cdot (x_i \cdot w + b) - 1) = 0, \qquad \forall\, i = 1, \cdots, n. \tag{3.11}$$

Notemos que si se consideran las restricciones de los problemas primal y dual, se tiene que las restricciones (3.11) indican que el producto de las restricciones del problema primal ($y_i \cdot (x_i \cdot w + b) - 1 \geq 0$) y las restricciones del problema dual ($\alpha_i \geq 0$) se anulan en todos los vectores de ensayo. Ello hace que nos preguntemos si se cumplen estas restricciones adicionales en todos los problemas de las máquinas de soporte vectorial (SVM). La respuesta es afirmativa; estas condiciones adicionales junto con las impuestas en un problema de optimización con restricciones se denominan **condiciones de Karush-Kuhn-Tucker** (KKT) y juegan un papel central tanto en la teoría como en la práctica de estos tipos de problemas.

Teniendo en cuenta los resultados previos, se sigue que las condiciones de Karush-Kuhn-Tucker para el problema primal, definido a partir de la función objetivo dada

en (3.6) son las siguientes :

$$\frac{\partial}{\partial w_j} L_P = w_j - \sum_{i=1}^{n} \alpha_i \, y_i \, x_{ij} = 0 \qquad j = 1, \cdots, d \tag{3.12}$$

$$\frac{\partial}{\partial b} L_P = -\sum_{i=1}^{n} \alpha_i \, y_i = 0 \tag{3.13}$$

$$y_i \cdot (x_i \cdot w + b) - 1 \geq 0 \qquad \forall i = 1, \cdots, n \tag{3.14}$$

$$\alpha_i \geq 0 \qquad \forall i = 1, \cdots, n \tag{3.15}$$

$$\alpha_i \cdot (y_i \cdot (x_i \cdot w + b) - 1) = 0 \qquad \forall i = 1, \cdots, n. \tag{3.16}$$

En [McC83] se demuestra que las condiciones KKT se cumplen para la solución de cualquier problema de optimización, sea este convexo o no, con cualquier clase de restricciones, siempre que se cumpla que la intersección de los conjuntos de direcciones factibles con el conjunto de direcciones descendentes coinciden con la intersección del conjunto de direcciones factibles por restricciones lineales con el conjunto de direcciones descendentes. Estos supuestos de regularidad se cumplen para todas las máquinas de vectores soporte, ya que las restricciones impuestas en estos problemas son siempre lineales. Aún más, en el caso de los problemas de SVMs, se tiene que las condiciones KKT son necesarias y suficientes para que w, b y α sea solución, véase [Fle87] y [CST00]. Así resolver el problema SVM es equivalente a encontrar una solución a las condiciones KKT, lo que en los desarrollos numéricos supone una simplificación del problema original.

Nótese que en los desarrollos anteriores no se tiene una forma explícita de determinar el valor b de la función solución. Una de las primeras utilidades de la condición KKT "complementaria" (la igualdad (3.16)) es precisamente la de poder determinar este valor. Para ello, basta elegir un $\alpha_i > 0$, es decir, un vector soporte y sustituir en la igualdad $y_i \cdot (w \cdot x_i + b) = 1$ y despejar $b = y_i - w \cdot x_i$. Aunque se ha determinado el valor b a partir de un vector soporte, es más adecuado realizar los cálculos con todos los vectores soporte y elegir como valor de b un valor promedio de los resultados obtenidos, con objeto de uniformizar los errores intrínsecos asociados a todo método de cálculo numérico (el valor de b es único). Por ejemplo, si se elige como promedio la media aritmética, denotamos por s_i los vectores soporte y por

N_{SV} el número de vectores soporte se tiene que:

$$b = \frac{1}{N_{SV}} \sum_{i=1}^{N_{SV}} (y_i - w \cdot s_i).$$

3.1.3 Prueba

Los desarrollos del apartado anterior permiten transformar el problema (3.5), en un problema de optimización donde las restricciones son más manejables que las dadas en (3.2) y (3.3). Para la resolución práctica de estos problemas, a partir de un conjunto de entrenamiento, se requiere normalmente el uso de métodos numéricos, debido a la gran cantidad de operaciones involucradas en estos problemas, por lo que una simplificación de este tipo permite agilizar el proceso de resolución. Una vez resuelto un problema SVM y obtenida la función solución, el siguiente paso es ver como se lleva a cabo el proceso de generalizar los resultados a otros vectores inputs distintos de los de entrenamiento.

Si se observa la figura 1.1 de la página 12, en la fase de aprendizaje ya se ha completado el primer paso. Este paso ha consistido en observar como se obtienen las etiquetas Y a partir de los vectores inputs X y se ha encontrado una función que imita la secuencia de outputs, proporcionada por el supervisor, de manera óptima.

El segundo paso consiste en poner en funcionamiento la máquina de aprendizaje y dar, ante un nuevo input, un valor que esté "próximo" al valor que proporciona el supervisor. En concreto, en los problemas de clasificación, este paso consiste en determinar en que lado del hiperplano frontera (el hiperplano que se encuentra justamente en la "mitad" entre los hiperplanos π_1 y π_2) se encuentra el nuevo input x y asignar la correspondiente clase (-1 ó 1), es decir, se asigna la clase

$$y = signo(w \cdot x + b)$$

al input x.

Ejemplo 3.1 *Sean los vectores inputs dados por*[13]*:*

$$\begin{aligned} X = \; & \{(180,80),(173,66),(170,80),(176,70),(160,65), \\ & (160,61),(162,62),(168,64),(164,63),(175,65)\} \end{aligned}$$

e

$$Y = \{1,1,1,1,1,-1,-1,-1,-1,-1\}\,.$$

Para la resolución de este problema se ha utilizado el programa MatLab, versión 5.3.0 junto con el paquete SVM implementado por Steven Gunn, puesto a disposición pública en su página Web[14]*.*

La solución gráfica, a este problema, se da en la figura 3.3.

Por otro lado, la resolución analítica proporciona los siguientes multiplicadores de Lagrange:

$$\{0, 1'2346, 0, 0, 0, 0, 0, 0'3210, 0, 0'9136\}$$

con lo cual se han obtenido tres vectores soporte que son el segundo, el octavo y el décimo vector. La tabla 3.1 muestra los resultados asociados a este ejemplo.

De estos valores se sigue que

$$\begin{aligned} w &= 1'2346\,(1)\begin{pmatrix}173\\66\end{pmatrix} + 0'3210\,(-1)\begin{pmatrix}168\\64\end{pmatrix} + 0'9136\,(-1)\begin{pmatrix}175\\65\end{pmatrix} \\ &= \begin{pmatrix}-0'2222\\1'5556\end{pmatrix} \end{aligned}$$

y de la condición

$$\alpha_i \cdot (y_i \cdot (x_i \cdot w + b) - 1) = 0, \qquad \forall\, i = 1, \cdots, n$$

[13] Estos datos han sido tomados de [SWG+96] y los vectores inputs son de dimensión dos, donde la primera componente representa la altura en centímetros y la segunda el peso en kilogramos de una persona. Las salidas (etiquetas) corresponde con $y = 1$ si es hombre e $y = -1$ si es mujer.

[14] Existen distintas implementaciones de libre disposición de las SVMs biclasificadoras. Otra posibilidad es elegir el paquete, también en MatLab del profesor Cecilio Angulo.

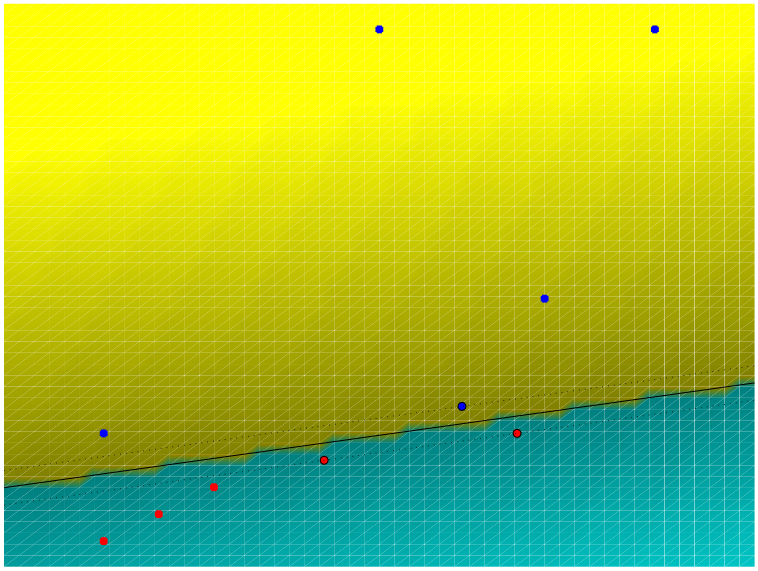

Figura 3.3: Solución gráfica al problema planteado en el ejemplo 3.1. Los puntos en azul representan los vectores inputs con etiqueta 1 (hombres) y los puntos en rojo los de etiqueta -1 (mujeres). Los puntos marcados representan los vectores soporte.

se sigue que ya que $\alpha_i \neq 0$ *para* $i = 2,\ 8\ y\ 10$*:*

$$y_i \cdot (x_i \cdot w + b) - 1 = 0, \qquad para\ i = 2, 8, 10$$

de donde resulta que para $i = 2$ *se tiene:*

$$1\left((-0'2222, 1'5556)\begin{pmatrix}173\\66\end{pmatrix} + b\right) - 1 = 0$$

y se tiene que el término independiente b *es igual a* $-63'229$*. Para los valores* $i = 8$ *e* $i = 10$ *se tiene que* $b = -63'2288$ *y* $b = -63'229$*, respectivamente y por tanto en promedio*[15] *es* $b = -63'22893$*. Como resultado de lo anterior se tiene que el hiperplano separador es :*

$$\pi(x_1, x_2) : -0'2222\, x_1 + 1'5556\, x_2 - 63'22893 = 0$$

[15]Nótese como el valor de b sale ligeramente diferente entre cada vector soporte, como ya se indicó debido a errores de cálculo.

x_{1i}	x_{2i}	y_i	α_i	$\pi(x_{i1}, x_{i2})$	$signo(\pi)$
180	80	1	0	21.2230	1
173	66	1	1.2346	1.0000	1
170	80	1	0	23.4450	1
176	70	1	0	6.5558	1
160	65	1	0	2.3330	1
160	61	-1	0	-3.8894	-1
162	62	-1	0	-2.7782	-1
168	64	-1	0.3210	-1.0000	-1
164	63	-1	0	-1.6670	-1
175	65	-1	0.9136	-1.0000	-1

Tabla 3.1: Resultados del ejemplo 3.1, sobre el sexo de una persona en función del peso y la altura.

(en este caso x_1 y x_2 denotan la primera y segunda componente de un vector $x \in I\!R^2$).

Nótese, en la tabla 3.1, como la imagen de cada vector soporte según el plano π es la unidad lo que significa que se encuentran en uno de los dos hiperplanos separadores.

Si se tiene un nuevo vector input (altura y peso de una persona), por ejemplo $\{195, 95\}$, se calcula

$$\pi(195, 95) = -0'2222 \cdot 195 + 1'5556 \cdot 95 - 63'229 = 41'224 > 0$$

lo que significa que la etiqueta que nosotros le asignamos es $y = 1$, es decir, se indicaría que esta nueva persona es un hombre. La duda básica que surge en todos estos problemas de aprendizaje es: ¿Qué etiqueta le pondrá el supervisor? ▲

3.1.4 El caso no separable

En los problemas reales, no siempre se encuentran conjuntos de vectores de ensayo separables. Por ejemplo, en la figura 3.4 se observa que hay dos puntos huecos

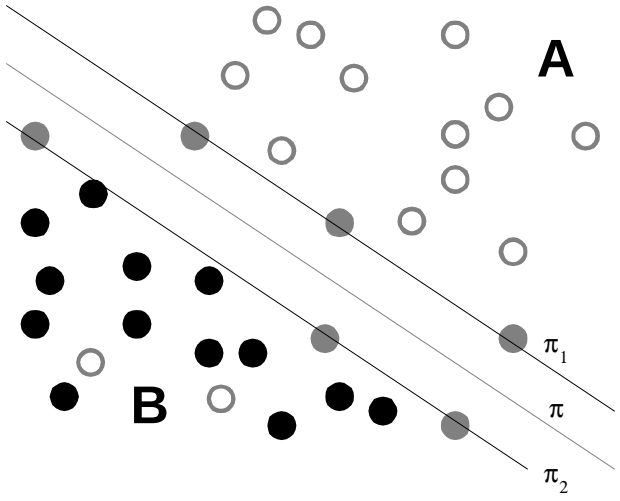

Figura 3.4: Ejemplo de hiperplanos separadores para el caso de datos no separables.

dentro de la región correspondiente a los puntos rellenos que nunca podrán ser separados de ellos por medio de hiperplanos. En estos casos se dirá que el conjunto de ensayo es un **conjunto de datos no separable**. Ante estos casos, el problema de optimización (3.5), no encuentra una solución posible (factible) y ello es evidente sin más que observar como la función objetivo (la función lagrangiana dual) (3.9) crece de forma arbitraría ya que el multiplicador de Lagrange correspondiente a este vector se puede tomar arbitrariamente grande sin que viole las restricciones. Sin embargo, no es difícil extender las ideas generales del caso separable al caso no separable. Veámoslo.

Para ello, se deben relajar, en primer lugar, las restricciones

$$x_i \cdot w + b \geq +1 \qquad \text{para } y_i = +1$$
$$x_i \cdot w + b \leq -1 \qquad \text{para } y_i = -1,$$

pero solo cuando sea necesario, y penalizar con un coste adicional la función objetivo cuando se violen las restricciones. Este coste se puede indicar de forma explícita introduciendo una variable de holgura $\xi_i,\ i = 1, \cdots, n$ en las restricciones, como en [CV95], y planteando los siguientes conjuntos de restricciones:

$$x_i \cdot w + b \geq +1 - \xi_i \qquad \text{para } y_i = +1, \tag{3.17}$$

$$x_i \cdot w + b \leq -1 + \xi_i \qquad \text{para } y_i = -1, \tag{3.18}$$

$$\xi_i \geq 0 \quad \forall i = 1, \cdots, n. \tag{3.19}$$

Se tiene ahora que para que se produzca un error (un vector input que no pueda ser ubicado en su correspondiente región) es necesario que el valor correspondiente ξ_i sea superior a la unidad (ver figura 3.4). Luego como para que se produzca un error se tiene que cumplir que $\xi_i \geq 1$, entonces $\sum_i \xi_i$ es una cota superior del número de errores que se comete dentro del conjunto de ensayo, lo que significa que el coste está acotado por $\sum_i \xi_i$ y de aquí que el coste medio por cada vector de ensayo es menor que $\frac{1}{n}\sum_{i=1}^{n} \xi_i$.

Ya que en el caso no separable, necesariamente se han de cometer errores, parece natural asignar a la función objetivo un coste extra, que en "cierto modo", penalice los errores. Nótese que con ello se hace uso de una función de pérdida, es decir, una función que cuantifica los errores. Por otro lado, como se tendrá ocasión de comprobar posteriormente, esto permite enlazar con los funcionales riesgo regularizado que se estudiaron en la parte primera de este trabajo.

Por todo ello, una opción lógica sería plantear el problema de minimizar

$$\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \xi_i,$$

donde C sea un parámetro elegido por el investigador. También se puede tomar una alternativa más general eligiendo como función objetivo

$$\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \xi_i^r$$

con $r \geq 1$. De esta forma se tiene que un valor C grande, significa que el investigador está asignando un peso a los errores muy alto frente a $\|w\|^2$, y por el contrario si C es pequeño asigna un mayor peso a $\|w\|^2$. Esta interpretación resulta más intuitiva si se interpreta, como posteriormente se verá, que $\|w\|^2$ es un factor de suavizamiento de la solución buscada. Indiquemos que este valor C se corresponde con $1/\lambda$ (valor inverso de la constante de regularización) de la sección 1.8. Por otro lado si r es grande lo que hacemos es dar mucho más peso a los errores cuantos mayores sean éstos, puesto que para que haya errores se ha de cumplir que $\xi_i > 1$ y la función $f(x) = x^r$ es creciente si $x > 1$.

Tal como se esta planteado el problema, se llega a un problema de programación convexa para cualquier valor de r. En especial, para $r = 2$ y $r = 1$ se tiene un problema de programación convexo cuadrático. En el presente trabajo se considera $r = 1$, ya que en este caso se tiene la ventaja de que ningún valor ξ_i, ni ninguno de sus correspondientes multiplicadores de Lagrange, aparecen en el problema dual de Wolfe, lo cual evidentemente simplifica mucho el problema.

El problema se plantea como sigue:

$$\begin{array}{l} \min\limits_{w \in \mathbb{R}^d} \dfrac{1}{2}\|w\|^2 + C\sum\limits_{i=1}^{n} \xi_i \\ s.a. \quad \begin{cases} y_i\,(x_i \cdot w + b) - 1 + \xi_i \geq 0, \ \forall i \\ \xi_i \geq 0, \ \forall i \end{cases} \end{array} \tag{3.20}$$

Utilizando la técnica de los multiplicadores de Lagrange, como en las secciones anteriores, se tiene que la función primal queda:

$$L_P(w, b, \xi) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i \cdot (y_i \cdot (x_i \cdot w + b) - 1 + \xi_i) - \sum_{i=1}^{n} \mu_i\,\xi_i, \tag{3.21}$$

donde los α_i son los multiplicadores de Lagrange correspondiente a las restricciones $y_i \cdot (x_i \cdot w + b) - 1 + \xi_i \geq 0$ y los μ_i son los multiplicadores de Lagrange que se introducen para forzar la positividad de los ξ_i (la condición (3.19)). Para resolver el problema de calcular el mínimo de la función primal se calculan las parciales de

$L_P(w, b, \xi)$ respecto de w_j, b y ξ_j

$$\begin{aligned}\frac{\partial}{\partial w_j} L_P &= w_j - \sum_{j=1}^{n} \alpha_i y_i x_{ij} \\ \frac{\partial}{\partial b} L_P &= \sum_{i=1}^{n} \alpha_i y_i \\ \frac{\partial}{\partial \xi_j} L_P &= C - \alpha_j - \mu_j\end{aligned}$$

igualando a cero y sustituyendo en (3.21) se tiene la función dual:

$$L_D(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \, \alpha_j \, y_i \, y_j \, x_i \, x_j \tag{3.22}$$

la cual hay que maximizar respecto de α_i sujeto a

$$0 \leq \alpha_i \leq C, \, \forall i \tag{3.23}$$

$$\sum_{i=1}^{n} \alpha_i \, y_i = 0. \tag{3.24}$$

y la solución viene dada por

$$w = \sum_{i=1}^{N_{SV}} \alpha_i \, y_i \, s_i \tag{3.25}$$

donde por N_{SV} se denota el número de vectores soporte y por s_i los vectores soporte del conjunto $\{x_1, \cdots, x_n\}$. Claramente se verifica que $N_{SV} \leq n$ y una de las características más interesante que se desea conseguir con esta nueva metodología es que "N_{SV} sea muy inferior a n", con lo que se consigue una representación "escasa" (del inglés –*sparse*–) de la solución en función de los vectores de ensayo.

Nótese, que la única diferencia en la solución con respecto a la dada en el caso separable es que los multiplicadores de Lagrange, α_i, están acotados superiormente por la constante C. Por ello, si frente a un problema concreto se tiene la seguridad de que el conjunto de vectores de ensayo es separable, se toma $C = \infty$ para forzar a la máquina a realizar una clasificación sin error.

Las condiciones de KKT asociadas a este problema son las siguientes:

$$\frac{\partial}{\partial w_j} L_P = w_j - \sum_{i=1}^{n} \alpha_i \, y_i \, x_{ij} = 0, \tag{3.26}$$

$$\frac{\partial}{\partial b} L_P = -\sum_{i=1}^{n} \alpha_i \, y_i = 0, \tag{3.27}$$

$$\frac{\partial}{\partial \xi_i} L_P = C - \alpha_i - \mu_i = 0, \tag{3.28}$$

$$y_i \, (x_i \cdot w + b) - 1 + \xi_i \geq 0, \tag{3.29}$$

$$\xi_i \geq 0, \tag{3.30}$$

$$\alpha_i \geq 0, \tag{3.31}$$

$$\mu_i \geq 0, \tag{3.32}$$

$$\alpha_i (y_i \, (x_i \cdot w + b) - 1 + \xi_i) = 0, \tag{3.33}$$

$$\mu_i \, \xi_i = 0. \tag{3.34}$$

Como se comentó en el caso separable, se pueden usar las condiciones complementarias de KKT, es decir, las igualdades (3.33) y (3.34), para determinar el valor de b. Además, la ecuación (3.28) combinada con la ecuación (3.34) muestra que, si $\xi_i = 0$, entonces $\alpha_i < C$. Así se puede simplificar el cálculo de b tomando los puntos de ensayo tales que, $0 < \alpha_i < C$ y usar la ecuación (3.33) con $\xi_i = 0$ (como ya se indicó anteriormente, es más adecuado promediar este valor entre todos los puntos de ensayo con $\xi_i = 0$).

En definitiva, en el problema de minimización (3.20) se busca determinar un hiperplano separador óptimo en la forma:

$$\pi \equiv f(x; w, b) = \langle w, x \rangle + b = 0$$

donde $w \in \mathbb{R}^d$ y $b \in \mathbb{R}$ son los parámetros que deben ser estimados. Por ello se puede considerar que el conjunto de funciones $\mathcal{F}$ sobre la que se busca la solución al problema (3.20) es $\mathcal{F} = \left\{ f(x; w, b) = \langle w, x \rangle + b, \quad w \in \mathbb{R}^d, \, b \in \mathbb{R} \right\}$, es decir, se tiene que

$$\min_{w \in \mathbb{R}^d} L_P(x; w, b) = \min_{f \in \mathcal{F}} L_P(x, f),$$

y como ya se indicó en la página 103, $L_P(x; f)$ puede ser considerado como un riesgo empírico regularizado $L_P(x; f) \equiv R_{reg,emp}[f]$, por lo que

$$\min_{w \in \mathbb{R}^d} L_P(x; w, b) = \min_{f \in \mathcal{F}} R_{reg,emp}[f]$$

y nos encontramos con un problema de los estudiados en la parte primera de este trabajo.

Por otro lado, ya que una vez determinado el valor óptimo de w, a partir de las condiciones de KKT se obtiene el parámetro b, el objeto principal de estudio se reduce en esencia al vector de parámetros $w \in \mathbb{R}^d$. Este vector de parámetros se obtiene según una combinación lineal de los vectores del conjunto de ensayo por la expresión[16] $w = \sum_{i=1}^{N} \alpha_i \, y_i \, x_i$, de donde se sigue que la función solución al problema de clasificación $f \in \mathcal{F}$ se expresa como

$$f(x) = \sum_{i=1}^{n} \alpha_i \, y_i \, \langle x_i, x \rangle + b. \tag{3.35}$$

3.2 Máquinas no lineales de vectores soporte

Hasta ahora se ha trabajado, tanto en el caso separable como el caso no separable, con conjuntos de funciones lineales en los parámetros[17], las cuales salvando el caso de las funciones constantes, constituye el conjunto de funciones más simple posible. En esta sección se aborda el problema de generalizar los desarrollos anteriores para el caso de conjuntos de funciones no necesariamente lineales en los parámetros.

Algunos autores, entre los que destacan, B.E. Boser, I.M. Guyon y V. N. Vapnik [BGV92], demostraron que utilizando las ideas que M.A. Aizerman, E.M. Braverman y L.I. Rozonoér, expusieron en [ABR64], se puede conseguir plantear los problemas de optimización de la máquina de vectores soporte a partir de conjuntos de funciones

[16] Para aquellos vectores que no sean vectores soporte se tendrá que $\alpha_i = 0$.

[17] La relación que liga los parámetros del modelo $w \in \mathbb{R}^d$ y $b \in \mathbb{R}$ se expresa en términos de sumas y diferencias.

no lineales en los parámetros y por tanto la solución al problema que se plantee es una solución no lineal. Para conseguir este fin, lo primero es observar que los vectores inputs entran a formar parte de la solución (3.35) del problema de clasificación, a través de los productos escalares, $\langle x_i, x\rangle$, $i = 1, \cdots, n$.

Las ideas dadas, en [ABR64], llevan al siguiente desarrollo: Sea una aplicación, a la que se denota por Φ, del conjunto de inputs $\mathcal{X} \subset \mathbb{R}^d$ en un espacio vectorial $\mathcal{H}$ (en la práctica se utilizan espacios de dimensión mucho mayor que d) dotado de un producto escalar:

$$\Phi : \mathcal{X} \subset \mathbb{R}^d \longrightarrow \mathcal{H}. \tag{3.36}$$

Ahora, en lugar de considerar el conjunto de vectores $\{x_1, \cdots, x_n\}$ se consideran los vectores transformados $\{\Phi(x_1), \cdots, \Phi(x_n)\}$ y se tiene que, si se plantea el problema de optimización original a estos vectores, es decir, se cambia de vectores inputs y de espacio input, entonces se tiene que los nuevos vectores entran a formar parte de la solución del problema, solo a través del producto escalar definido en $\mathcal{H}$ como funciones de la forma[18] $\langle \Phi(x_i), \Phi(x)\rangle_{\mathcal{H}}$. Por tanto si se considera una función

$$k : \mathcal{H} \times \mathcal{H} \rightarrow \mathbb{R}$$

a la que se le denomina **función núcleo**, tal que

$$k(x_i, x) = \Phi(x_i) \cdot \Phi(x) = \langle \Phi(x_i), \Phi(x)\rangle,$$

solo es necesario conocer la función núcleo para resolver el algoritmo y no se necesita tener la forma explícita de la aplicación[19] Φ.

Por tanto si se reemplaza $\langle x_i, x\rangle$ por $k(x_i, x)$ en la solución de los problemas de optimización, se habrá conseguido una máquina de vectores soporte planteada en un espacio de dimensión superior (incluso puede que sea interesante considerar espacios de dimensión infinita), y además, lo que resulta muy importante en la práctica, la ejecución de un programa que lleve a cabo esta técnica no lineal, consume la misma cantidad de tiempo (recursos computacionales) que si la técnica fuese lineal.

[18] Si del contexto se tiene el espacio donde se toma el producto escalar, el subíndice no se indicará.

[19] Recuérdese que esta idea ya fue indicada cuando comentamos la figura 2.6.

Así, al resolver el problema, de la máquina de vectores soporte, en un espacio de dimensión superior al del espacio de inputs, donde se trabaja con un conjunto de funciones lineales, la solución que resulta es lineal en este espacio, pero no es necesariamente lineal en el espacio input $\mathcal{X}$, con lo cual se está generalizando el problema a conjuntos de funciones no lineales.

En la sección anterior se trabajaba con funciones lineales de la forma:

$$f(x) = x \cdot w + b = \langle w, x \rangle + b$$

donde el vector solución w venía dado por $w = \sum_{i=1}^{N} \alpha_i \, y_i \, x_i$ y en términos de los vectores soporte $w = \sum_{i=1}^{N_{SV}} \alpha_i \, y_i \, s_i$. Si se lleva a cabo la transformación de los datos, el vector solución w queda $w = \sum_{i=1}^{N} \alpha_i \, y_i \, \Phi(x_i)$ y en términos de los vectores soporte

$$w = \sum_{i=1}^{N_{SV}} \alpha_i \, y_i \, \Phi(s_i) \tag{3.37}$$

donde por $\Phi(s_i)$ denotamos los vectores soporte del conjunto $\{\Phi(x_1), \cdots, \Phi(x_n)\}$.

Nota 3.2.1 *Nótese que los vectores soporte, $\Phi(s_i)$, se encuentran dentro del conjunto $\{\Phi(x_1), \cdots, \Phi(x_n)\}$ pero no son necesariamente los transformados de los vectores soporte s_i que se encuentran dentro del conjunto $\{x_1, \cdots, x_n\}$, entre otras cosas porque con los vectores sin transformar no se realiza ningún algoritmo. A pesar de esta indicación se sigue con esta notación, puesto que es la utilizada tradicionalmente en la literatura clásica.* ▲

Claramente, se ve que la aplicación Φ aparece explícitamente utilizada en la solución del vector $w \in \mathcal{H}$ dada en (3.37), pero cuando sobre un nuevo vector input x se realiza la fase de prueba, no es necesario tener identificada la transformación Φ ya que la solución viene dada por:

$$f(x) = \sum_{i=1}^{N_{SV}} \alpha_i \, y_i \, \langle \Phi(s_i), \Phi(x) \rangle + b \tag{3.38}$$

y escrita en términos de la función núcleo:

$$f(x) = \sum_{i=1}^{N_{SV}} \alpha_i \, y_i \, k(s_i, x) + b. \tag{3.39}$$

Nota 3.2.2 *Al usar* $\Phi : \mathcal{X} \to \mathcal{H}$ *se trabaja en un nuevo espacio* $\mathcal{H}$*, por lo cual el vector solución* w *se encuentra en este espacio. Sobre la aplicación no se tiene la condición de sobreyectividad, lo que significa que puede que no exista un vector* $x_w \in \mathcal{X}$ *tal que* $\Phi(x_w) = w$*. Sin embargo, si existiese dicho vector origen, entonces* $f(x)$ *puede calcularse en un único paso ya que*

$$f(x) = \sum_{i=1}^{N_{SV}} \alpha_i \, y_i \, k(s_i, x) + b = k(x_w, x) + b$$

y de esta forma se evita la suma, y se consigue un algoritmo SVM, N_{SV} *veces más rápido.*

A pesar de no tener necesariamente una aplicación sobreyectiva, la idea anterior puede ser usada para aumentar significativamente[20] *la velocidad de los algoritmos en la fase de prueba [Bur98].* ▲

Ejemplo 3.2 *Veamos un ejemplo de una función núcleo construida a partir del producto escalar definido en el espacio de los inputs* $\mathcal{X} \subset I\!R^2$*.*

Consideramos la función:

$$k : I\!R^2 \times I\!R^2 \to I\!R$$

definida como sigue:

$$k(x_i, x_j) = \langle x_i, x_j \rangle^2 = (x_i \cdot x_j)^2.$$

Si $x_i = (x_{i1}, x_{i2})$ *y* $x_j = (x_{j1}, x_{j2})$ *entonces:*

$$k(x_i, x_j) = (x_i \cdot x_j)^2 = (x_{i1}\, x_{j1} + x_{i2}\, x_{j2})^2 = x_{i1}^2\, x_{j1}^2 + 2\, x_{i1}\, x_{j1}\, x_{i2}\, x_{j2} + x_{i2}^2\, x_{j2}^2$$

[20] Como se indica en la introducción, la utilidad de esta nueva técnica se plantea desde el punto de vista de la resolución de problemas donde intervienen una gran cantidad de variables y por tanto la velocidad del proceso debe ser estudiada y optimizada.

para todo x_i, $x_j \in \mathcal{X}$ *y tomemos* $\Phi : I\!R^2 \to I\!R^3$ *como*

$$\Phi(x_i) = \Phi(x_{i1}, x_{i2}) = \begin{pmatrix} x_{i1}^2 \\ \sqrt{2}\, x_{i1}\, x_{i2} \\ x_{i2}^2 \end{pmatrix}$$

entonces

$$\langle \Phi(x_i) \cdot \Phi(x_j) \rangle_{I\!R^3} = (x_i \cdot x_j)^2, \quad \forall x_i, x_j \in \mathcal{X}$$

y se tiene que la función k *ciertamente es una función núcleo ya que se corresponde con un producto escalar en* $I\!R^3$.

En muchos problemas de clasificación es común elegir como $\mathcal{X} = (-1,1) \times (-1,1) \subset I\!R^2$ *(por ejemplo, cuando se plantea el problema de digitalizar una imagen en escala de grises [Cor95]) y la imagen de* Φ *queda dada según la figura 3.5. Esta*

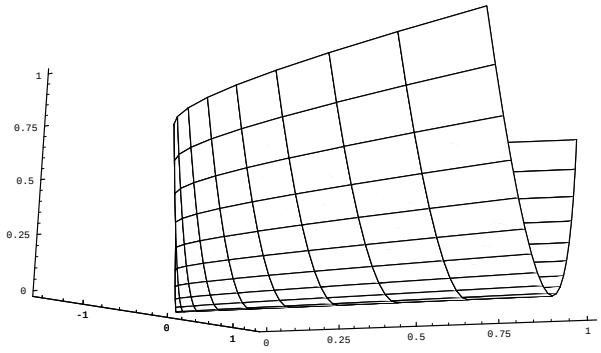

Figura 3.5: Imagen en $\mathbb{R}^3$ del cuadrado $(-1,1) \times (-1,1) \subset \mathbb{R}^2$ bajo la aplicación Φ del ejemplo 3.2.

figura permite ver qué es, en realidad, lo que se hace cuando se aplica una transformación Φ *al espacio de inputs: la imagen de* $\mathcal{X}$ *por* Φ*, es decir,* $\Phi(\mathcal{X})$ *se encuentra en un espacio de dimensión superior* $\mathcal{H}$ *pero en realidad se trata de una superficie cuyo cardinal es el mismo (o menor) que el cardinal de* $\mathcal{X}$*, algo que es elemental ya que cardinal* $\psi(A) \leq$ *cardinal* A *cualquiera que sea la función* ψ.

Por otro lado, no existe una correspondencia uno a uno entre la función núcleo y la aplicación Φ, ya que puede haber diferentes Φ para un núcleo dado k. Por ejemplo con la función núcleo anterior, se podría haber elegido como función Φ las dos siguientes:

$$\Phi(x_i) = \Phi(x_{i1}, x_{i2}) = \frac{1}{\sqrt{2}} \begin{pmatrix} (x_{i1}^2 - x_{i2}^2) \\ 2\, x_{i1}\, x_{i2} \\ (x_{i1}^2 + x_{i2}^2) \end{pmatrix}$$

o

$$\Phi(x_i) = \Phi(x_{i1}, x_{i2}) = \begin{pmatrix} x_{i1}^2 \\ x_{i1}\, x_{i2} \\ x_{i1}\, x_{i2} \\ x_{i2}^2 \end{pmatrix}$$

donde también es posible observar que el espacio de dimensión superior $\mathcal{H}$ no tiene necesariamente que ser único para un núcleo k dado. ▲

Ejemplo 3.3 *Consideramos los datos del ejemplo 3.1 pero utilizando, en este caso, la función núcleo de Gauss o RBF (del inglés –Radial Function Basis– (función de base radial) la cual tiene la forma:*

$$k(x, y) = e^{-\|x-y\|^2/2\sigma^2}, \qquad \sigma \in I\!R^+. \tag{3.40}$$

Se toma $\sigma = 10$, y como sabemos que los datos son separable elegimos como valor de la constante, del problema de optimización, $C = \infty$ (damos un peso infinito a los errores con objeto de que la solución al problema no cometa ninguno).

La solución gráfica aparece en la figura 3.6.

Por otro lado, la resolución analítica proporciona los siguientes multiplicadores de Lagrange:

$$\{0,\ 105'8140,\ 0,\ 0,\ 10'1914,\ 0,\ 0,\ 40'7896,\ 0,\ 75'2159\}$$

con lo cual se han obtenido cuatro vectores soporte. ▲

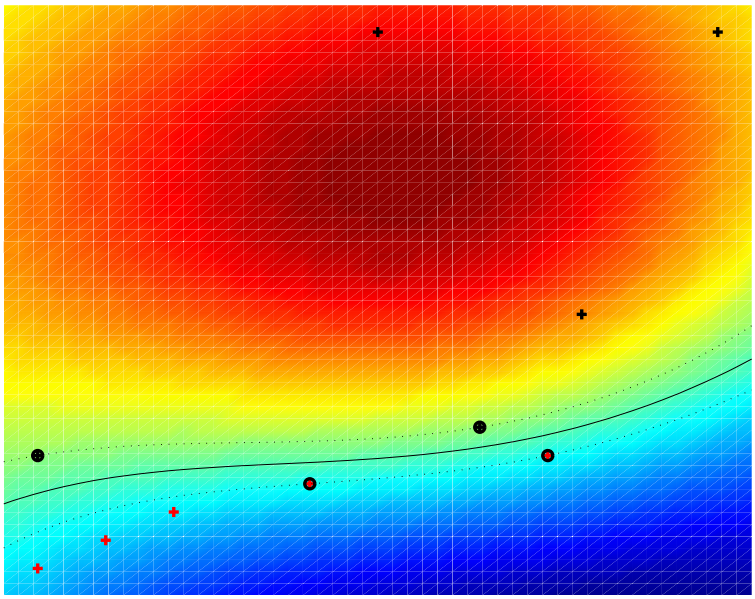

Figura 3.6: Solución gráfica al problema planteado en el ejemplo 3.3. Los puntos "+"en negro representan los vectores inputs con etiqueta 1 (hombres) y los puntos "+"en rojo los de etiqueta -1 (mujeres). Los puntos en círculos representan los vectores soporte.

3.3 SVMs y Análisis discriminante

En Estadística, las técnicas más usuales para resolver los problemas de clasificación son el análisis discriminante y la regresión logística (se estudia en el siguiente capítulo).

El objetivo del análisis discriminante es estimar la relación existente entre una variable dependiente categórica y un conjunto de variables independientes métricas. Así, sea Y una variable dependiente (no necesariamente métrica) que permite la identificación de un conjunto de grupos diferentes, y $X_1, \cdots, X_d$ un conjunto de variables explicativas, entonces a partir de un conjunto de datos se busca obtener una función discriminate, que toma la forma:

$$Z = w_1 X_1 + \cdots + w_n X_d + b.$$

Si denotamos $\mathbf{X} = (X_1, \cdots, X_d)$, la función discriminante se expresa en la forma $Z = w \cdot \mathbf{X} + b$. De esta forma, aunque siguiendo un planteamiento distinto a las máquinas de vectores soporte, el objetivo es el mismo, la búsqueda de un determinado hiperplano en $\mathbb{R}^d$.

Sin embargo, el problema de optimización que se plantea es distinto. En la función discriminante, para cada asignación del vector de variables $\mathbf{X}$, se determina el valor Z, que se denomina **valor teórico**; y el objetivo es buscar aquellas ponderaciones w_i, del valor teórico, para cada variable X_i de tal forma que maximicen las varianzas entre grupos frente a la varianza intra-grupos (dentro del grupo). De esta forma, si la varianza entre los grupos es grande en comparación con la varianza intra-grupos se dirá que la función discriminante separa bien los grupos. Así, después de establecer un conjunto de hipótesis sobre el modelo, se sigue que el análisis discriminante es una técnica estadística que permite contrastar la hipótesis de que las medias de los grupos de un conjunto de variables independientes para dos o más grupos son iguales.

En este planteamiento nosotros consideraremos solo dos grupos como en el caso SVM, aunque si ℓ es el número de grupos diferentes, el análisis discriminante calcula $\ell - 1$ funciones discriminantes.

Frente al análisis seguido en las SVMs donde no se establecía ninguna hipótesis sobre la distribución seguida por los vectores de ensayo, en el análisis discriminante para la obtención de la función discriminante se realizan dos supuestos: i) las variables explicativas se distribuyen según un modelo normal multivariante; y ii) las matrices de varianzas-covarianzas y dispersiones (desconocidas) son iguales para todos los grupos. Por otro lado, es conocido la alta sensibilidad del análisis discriminante[(21)] cuando no se cumple alguno de los supuestos[(22)].

[(21)] Ver [HATB00].

[(22)] En estos casos, se suele utilizar la regresión logística, ya que ésta se ve menos afectada cuando no se cumple los supuestos básicos, concretamente la normalidad de las variables. Además, de proporcionar resultados predictivos y clasificatorios similares y emplear las mismas medidas de validación.

Es conocida la robustez de las SVMs cuando el conjunto de entrenamiento es pequeño (ver por ejemplo [Ang01]), sin embargo, el análisis discriminante es bastante sensible al ratio entre el tamaño muestral y el número de variables explicativas; y por ello, hay que tener en cuenta que los resultados pueden llegar a ser inestables a medida que el tamaño muestral disminuye en relación al número de variables explicativas. También puede afectar a la estimación de la función discriminante y a la clasificación de las observaciones que el tamaño de los grupos varíen ampliamente[(23)].

La ventaja que supone, desde el punto de vista estadístico, el análisis discriminante es la de poder realizar contrastes de hipótesis sobre las principales características del modelo. Sin embargo, los contrastes estadísticos para valorar la significación de las funciones discriminantes no informan sobre lo correctamente que predice la función. Por ejemplo, supongamos dos grupos que son significativamente diferentes por encima del nivel de significación, 0'01 y tamaños muestrales suficientemente grandes. En este caso, las medias de los grupos (centroides) podrían ser virtualmente idénticas y aún se tendría significación estadística y sin embargo, los valores teóricos entre grupos diferentes estarían muy próximos, proporcionando una pobre capacidad predictiva. Otra forma de indicar esta posibilidad es la siguiente: en el análisis discriminante, el porcentaje de aciertos (porcentaje correctamente clasificado) es análogo al R^2 de la regresión mínimo cuadrática. Así, el contraste ji-cuadrado de significación del análisis discriminante, con un tamaño muestral suficientemente grande, podría proporcionar una diferencia estadística significativa entre dos o más grupos y sin embargo clasificar correctamente sólo el 40%.

Por todo ello, para determinar la capacidad predictiva de una función discriminante, el investigador debe acudir a otro tipo de técnica. Este es el fin de las matrices de clasificación, y de igual manera que se emplea en el análisis discriminante, es posible utilizarlas para abordar el problema de medir lo bien que clasifican las SVMs.

[(23)] Nosotros realizaremos un estudio con datos de este tipo en el capítulo 8.

3.3.1 Matrices de clasificación para las SVMs

Como se ha comentado, cuando se plantea la función objetivo, a minimizar, en los problemas de máquinas de vectores soporte se busca una solución que mantenga una relación entre ajuste a los datos y suavidad. Una vez obtenida ésta, tendremos que validarla de alguna forma, puesto que podría ocurrir que la solución sea suficientemente suave y, sin embargo, clasificase correctamente menos de un 10% de los datos.

Para validar las máquinas de vectores soporte (o la función discriminante o la regresión logística u otras técnicas semejantes) para la clasificación por medio de matrices de clasificación, la muestra $\{(x_1, y_1), \cdots, (x_n, y_n)\}$ ha de dividirse aleatoriamente en dos grupos. Uno de los grupos (el conjunto de entrenamiento) se utiliza para calcular la solución SVM; y el otro grupo (el conjunto de validación o conjunto de test) se usa para elaborar la matriz de clasificación y validar los resultados.

Respecto al número de datos de cada conjunto no hay un criterio estándar. Algunos autores consideran una relación $50\% - 50\%$, otros $60\% - 40\%$, ... por supuesto todo ello dependiendo del número de datos de origen. En cuanto a la forma de seleccionar los elementos de cada grupo, se realiza un muestreo estratificado en relación al número de datos de cada clase.

Si el número de datos es pequeño, para realizar una división, se aconseja tomar como conjunto de validación, el mismo que el conjunto de entrenamiento ya que es preferible realizar algún tipo de validación que no realizar ninguna.

Ejemplo 3.4 *Supongamos que tenemos un conjunto de validación formado por* 100 *datos de los cuales sabemos que* 75 *se etiquetan con* $Y = 1$ *y el resto con etiqueta* $Y = -1$. *Una vez aplicada la solución SVM, obtenida a partir del conjunto de entrenamiento, podemos ver la clasificación que resulta de aplicarla al conjunto de validación y obtener la matriz de clasificación que aparece en la tabla 3.2. De la tabla se sigue que se han etiquetado erróneamente* 14 *datos de validación (*5 *datos con etiquetas* $Y = -1$ *se han etiquetado con* $Y = 1$, *y* 9 *datos con etiquetas* $Y = 1$ *se*

Tabla 3.2: Ejemplo de matriz de clasificación. Las iniciales PCC significa: Porcentaje Correctamente Clasificado.

Etiquetas	Predichas			
Reales	-1	1	Total Real	PCC
-1	20	5	25	80%
1	9	66	75	88%
Total Predicha	29	71	100	86%

han etiquetado con $Y = -1$*). Dentro de la clase* $Y = -1$ *se etiqueta correctamente el* 80% *y en la clase* $Y = 1$ *el* 88%*, y teniendo en cuenta el tamaño relativo se tiene un* 86% *de clasificaciones correctas.* ▲

En este ejemplo, la precisión total de la clasificación alcanzada es del 86%. Pero nos podríamos preguntar ¿es aceptable un 60% o debería esperarse un 80% o un 90% de capacidad predictiva? Para tener un valor de referencia se podría tomar el **criterio de máxima aleatoriedad**, el cual se determina tomando el porcentaje de datos representado por el más grande de los grupos. Así por ejemplo, si se tienen dos grupos, uno con un 75% de datos y otro con el 25%, eligiendo como función discriminate aquella que asigna a todos los datos el grupo más grande se tendría un 75% de capacidad predictiva. Evidentemente esta función sería muy suave ya que toma un único valor[(24)] (es una función constante).

La aproximación anterior tiene el inconveniente de no identificar elementos perteneciente a otros grupos, de esta forma si se desea identificar adecuadamente los miembros de dos o más grupos se utiliza el **criterio de aleatoriedad proporcional**. Por ejemplo, si se tienen solo dos grupos y la proporción del primero es p

[(24)] En el ejemplo anterior sería la función $\pi(x) = 1$.

entonces la fórmula para el criterio es[25]:

$$C_{PRO} = p^2 + (1-p)^2.$$

En el ejemplo anterior sería $C_{PRO} = 0'75^2 + 0'25^2 = 0'625$. De esta forma, si la precisión clasificatoria no es más grande que la que cabría esperar aleatoriamente apenas se aportaría nada a la interpretación de la solución SVM.

La justificación para dividir la muestra en dos conjuntos de datos es que aparece un sesgo al alza en la capacidad predictiva de la máquina de vectores soporte[26] si los datos incluidos en la construcción de la matriz de clasificación son los mismos que aquellos utilizados para calcular la solución SVM. Aun más, es conocida la gran capacidad de ajuste de las SVMs. Por ejemplo eligiendo un adecuado parámetro de escala σ en las funciones de base radial, es posible discriminar correctamente todos los datos de entrenamiento. En este punto, y aprovechando la potencia de cálculo de los ordenadores, sería adecuado, con objeto de tener una mayor confianza en la validez de la técnica repetir el proceso de tomar conjuntos de entrenamiento y de validación, varias veces, y promediar los resultados obtenidos.

3.4 Resumen del capítulo

Se estudian los problemas de clasificación donde se ven de forma natural las principales herramientas de trabajo de la metodología de las SVMs, las condiciones de Karush-Kuhn-Tucker, conjuntos separables, hiperplanos separadores, espacios característicos, función núcleo, vectores soporte, ...

También se toma del análisis discriminante un par de criterios que pueden ser útiles en la determinación de la capacidad predictiva de las SVMs. Además de utilizar las matrices de clasificación con objeto de medir lo bien que clasifican las máquinas de vectores soporte.

[25] La demostración para cuatro grupo (la generalización se sigue de éste de forma natural) se da en el capítulo 8.

[26] En realidad, esta afirmación es cierta en todas las técnicas que entrenen con datos.

CAPÍTULO 4

ASPECTOS PROBABILÍSTICOS DE LAS MÁQUINAS DE VECTORES SOPORTE

Es una gran verdad que cuando no está a nuestro alcance determinar lo que es verdadero, debemos aceptar aquello que sea más probable.

–Descartes (1596-1650)–

En este capítulo comenzamos introduciendo brevemente la regresión logística y, a continuación, buscamos siguiendo un camino similar a este tipo de técnica de clasificación dar una interpretación de las máquinas de vectores soporte como un problema de maximizar una determinada función de verosimilitud. Tras estudiar las distintas formulaciones de este problema, llegamos a la conclusión que la más idónea es la dada por Peter Sollich.

Una vez estudiada esta formulación, se está en condiciones de asignar probabilidades a los diferentes sucesos que tienen lugar cuando se lleva a cabo una clasificación utilizando las máquinas de vectores soporte.

4.1 Regresión logística

La regresión logística es un tipo especial de regresión que se utiliza, al igual que el análisis discriminante, para predecir y explicar una variable categórica a partir de un conjunto de variables explicativas. Esta técnica, muy utilizada en Estadística, proporciona resultados predictivos y clasificatorios comparables a los del análisis discriminante y emplea medidas de validación muy similares.

En general, cuando la variable dependiente presenta solo dos etiquetas, se prefiere la regresión logística frente al análisis discriminante por varios motivos. En primer lugar, el análisis discriminante descansa sobre un cumplimiento estricto de los supuestos de normalidad multivariante de las variables explicativas y la igualdad de matrices de varianzas covarianzas entre los diferentes grupos, supuestos que no siempre se verifican. Además, la regresión logística es mucho más robusta cuando estos supuestos no se cumplen y puede permitir la utilización de variables no métricas, dentro del modelo, a través de las variables ficticias.

4.1.1 Interpretación económica de la regresión logística

La interpretación, desde un punto de vista económico, de la regresión logística (o modelo logit) parte del principio de maximización de la utilidad que proporciona cada una de las posible opciones a las que se enfrenta un agente económico. Así pues, sobre la base de este planteamiento teórico cabe establecer que la probabilidad de que el individuo i-ésimo escoja una de los dos alternativas, que denotamos por las etiquetas "0" y "1", a la que se enfrenta, depende de que la utilidad que le proporciona dicha decisión sea superior a la que proporciona su complementaria. La formalización de esta teoría parte del supuesto de que la utilidad derivada de una elección, $U_{i,0}$ o $U_{i,1}$, es función de las variables explicativas de dicha decisión, además de una perturbación aleatoria ε_{ij} que recoge las desviaciones que los agentes tienen respecto a lo que sería el comportamiento del agente medio.

De esta forma el problema de clasificación se puede plantear en los siguientes términos: Sea $U_{i,0}$ la utilidad que proporciona al agente i la elección 0, y $U_{i,1}$ la utilidad que proporciona al agente i la elección 1; y sean $X_{i,0}$ el vector de variables explicativas que caracteriza la elección de la alternativa 0 por parte del agente i, y $X_{i,1}$ el vector de variables explicativas que caracteriza la elección de la alternativa 1 por parte del agente i. Suponiendo la linealidad de las funciones, que intervienen en el modelo, se plantea:

$$\begin{aligned} U_{i,0} &= \overline{U}_{i,0} + \varepsilon_{i0} &= b_0 + w \cdot X_{i,0} + \varepsilon_{i0} \\ U_{i,1} &= \overline{U}_{i,1} + \varepsilon_{i1} &= b_1 + w \cdot X_{i,1} + \varepsilon_{i1} \end{aligned}$$

donde $\overline{U}_{i,0}$ y $\overline{U}_{i,1}$ representan las utilidades medias asociadas a cada elección. Sobre ε_{i0} y ε_{i1} se supone que son variables aleatorias ruido blanco.

A partir de esta construcción, se sigue que el agente i elige la opción que más utilidad le proporcione, es decir,

$$Y_i = \begin{cases} 1 & \text{si } U_{i,1} > U_{i,0} \\ 0 & \text{si } U_{i,0} > U_{i,1}. \end{cases}$$

Por tanto, la probabilidad de que un individuo elija la opción uno será:

$$\begin{aligned} P[Y=1] &= P[U_{i,0} < U_{i,1}] = P[\varepsilon_{i0} - \varepsilon_{i1} < \overline{U}_{i,1} - \overline{U}_{i,0}] \\ &= P[\varepsilon_{i0} - \varepsilon_{i1} < (b_1 - b_0) + w \cdot (X_{i1} - X_{i0})] = F(w \cdot X + b) \end{aligned} \tag{4.1}$$

donde F representa una función de distribución, y así se garantiza que la probabilidad se encuentra entre 0 y 1. De esta forma, la probabilidad de que el individuo elija la opción uno depende de las variables explicativas que explican el proceso de decisión y de la función de distribución que se supone sigue dicha probabilidad.

Si denotamos por $\theta(x) = w \cdot X + b$, se tiene que $P[Y = 1] = F(\theta(x))$; de tal manera que si se suponen distintas funciones de distribución, se tendrán diferentes modelos. Los más habituales son:

- Modelo Logit[(1)]: La función que se utiliza es la logística.

$$P[Y=1] = F(\theta(x)) = \frac{e^{\theta(x)}}{1 + e^{\theta(x)}}$$

[(1)]Berkson especificó este término en 1944 y es una abreviatura de *logistic probability unit.*

- Modelo Probit: La función que se utiliza es la normal tipificada.
- Modelo lineal de probabilidad: La función que se utiliza es la uniforme.

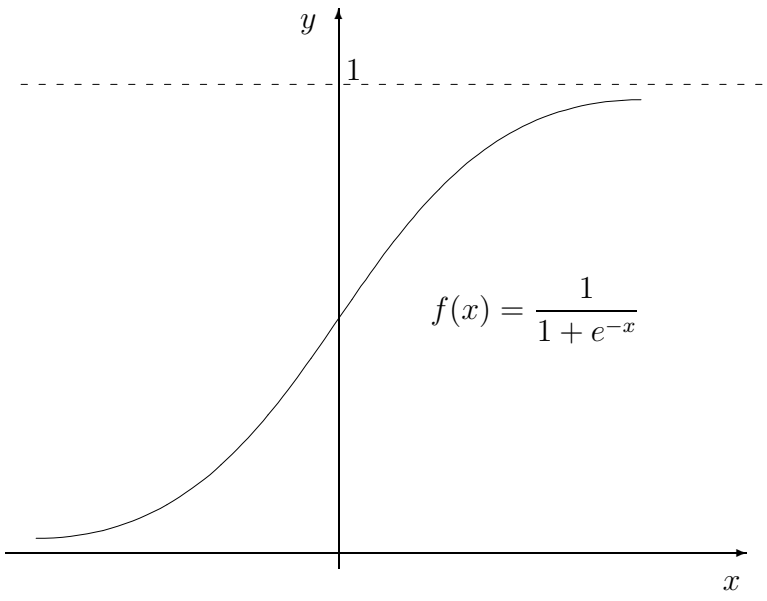

Figura 4.1: Representación gráfica de la función logística.

Como se observa, el proceso de selección entre opciones depende de la probabilidad asociada a cada una de las alternativas posibles que tiene el individuo. A partir de ahora, consideraremos exclusivamente el modelo Logit.

De esta forma, si se tiene un dato $\mathbf{X}_i$ conocido entonces la probabilidad de asignarle la etiqueta $Y_i = 1$ es:

$$P[Y_i = 1/\mathbf{X}_i] = \frac{e^{\theta(\mathbf{X}_i)}}{1 + e^{\theta(\mathbf{X}_i)}} = p_i$$

y

$$P[Y_i = 0/\mathbf{X}_i] = (1 - p_i)$$

Así pues, si después de estimar el modelo se concluye que la probabilidad para un suceso es mayor que $0'5$, entonces se predice la ocurrencia del suceso y se le asigna

la correspondiente etiqueta; en caso contrario, se dirá que el suceso no va a ocurrir y le asignamos la etiqueta complementaria.

Por tanto, el siguiente paso es la estimación de los parámetros b y w incorporados en el modelo. Para ello hay que tener en cuenta que la naturaleza no lineal de la transformación logística hace que no sea aplicable el método de mínimos cuadrados y en su lugar se utiliza el método de máxima verosimilitud para encontrar la estimación más verosimil de los parámetros. Para su estimación se ajusta una curva logística a los datos.

Es conocido que las estimaciones de los parámetros obtenidas por el proceso de máxima verosimilitud son consistentes y asintóticamente eficientes. Igualmente son asintóticamente normales, con lo que los contrastes de hipótesis son asintóticos. De esta forma cuando el tamaño de la muestra tiende a infinito, el contraste de significatividad de los parámetros individuales se puede realizar a través de una distribución normal.

Una vez obtenidos los parámetros por máxima verosimilitud, y al igual que el análisis discriminante, podemos utilizar el método de las matrices de clasificación para evaluar la exactitud predictiva en términos de pertenencia al grupo.

4.2 Interpretación probabilística de las SVMs

Las SVMs estándar no proporcionan probabilidades en el sentido, apuntado en la regresión logística, de estimar la probabilidad de acertar en las predicciones. Es decir, de estimar la distribución condicional $P[Y|X = x]$ con objeto de cuantificar la incertidumbre asociada a nuestra predicción. En este sentido, dentro del contexto de las máquinas de vectores soporte se han elaborado diferentes aproximaciones a este problema.

Un primer camino fue dado en [Wah98] donde se considera una función núcleo clasificadora (un modelo expresado de igual manera que aparece la solución de una

SVM) en la forma:

$$f(x) = \sum_{i=1}^{n} w_i \, k(x_i, x) + w_0$$

y, supone que la probabilidad de asignar la etiqueta $y = 1$ al vector input x en términos de la función logística es como sigue:

$$P[y = 1|x] = \sigma(x) = \frac{1}{1 + \exp(-f(x))} = \frac{\exp(f(x))}{1 + \exp(f(x))}$$

Dada una muestra aleatoria simple $\{(x_1, y_1), \cdots, (x_n, y_n)\}$ considera la función de verosimilitud[(2)]

$$\mathcal{L}(\mathbf{w}) = \prod_{i=1}^{n} \sigma(x_i)^{y_i} (1 - \sigma(x_i))^{1-y_i} \tag{4.2}$$

con $\mathbf{w} = (w_0, w_1, \cdots, w_n)$ y su logaritmo

$$\ell(\mathbf{w}) = \sum_{i=1}^{n} (y_i \, \log \sigma(x_i) + (1 - y_i) \, \log(1 - \sigma(x_i)))$$

De esta forma, teniendo en cuenta que el núcleo k define dentro de un espacio de Hilbert con núcleo reproductor[(3)] – del inglés *Reproducing Kernel Hilbert Spaces* (R.K.H.S.) – una norma $\|\cdot\|$ y siguiendo el esquema de suavizamiento dado para determinadas funciones splines (ver en [Wah90]), plantea la búsqueda del vector de pesos $\mathbf{w}$ minimizando

$$-\ell(\mathbf{w}) + \lambda \|f\|$$

El principal inconveniente de esta aproximación probabilística, como se apunta en [Pla99], es que la solución obtenida no corresponde con una solución escasa –del inglés *sparse*–, como ocurre con el planteamiento de las SVMs, donde por solución escasa se entiende que la solución del problema se expresa en términos de un conjunto reducido de vectores de entrenamiento (los vectores soporte). Hay que indicar que esta característica de solución escasa es la que proporciona una adecuada capacidad de generalización a las SVMs.

[(2)] En este caso es conveniente trabajar con etiquetas $Y = \{0, 1\}$, para poder utilizar la distribución de Bernouilli. Si las etiquetas son $Y = \{-1, 1\}$ se aplica la transformación $t_i = \frac{1 + y_i}{2}$ y se tienen etiquetas $t = \{0, 1\}$.

[(3)] Una introducción a estos espacios se puede encontrar en el apéndice A.

En [Tip00] se sigue el mismo planteamiento anterior buscando maximizar la función de verosimilitud pero sin el término $\|f\|$ y además supone que los pesos siguen una distribución normal multivariante. Esta técnica es conocida como máquinas de vectores relevantes –del inglés *Relevance Vector Machine* (RVM) –, y presenta, con respecto a la anterior, la ventaja de obtener una solución escasa en términos de un conjunto de vectores que se denominan **vectores relevantes**.

Otra aproximación dada en [Pla99] es la siguiente: A partir de un conjunto de entrenamiento se entrena una máquina de vectores soporte obteniendo la solución

$$f(x) = \sum_{i=1}^{n} w_i \, k(x_i, x) + w_0$$

y se consideran las probabilidades a priori $P[y = -1]$, $P[y = 1]$ y las verosimilitudes

$$P[f(x)|y = 1] = \exp(-C\,(1 - f(x))), \qquad P[f(x)|y = -1] = \exp(-C\,(1 + f(x))).$$

Aplicando el teorema de Bayes se tiene que:

$$P[y = 1|\, f(x)] = \frac{1}{1 + \exp\left(-2C\, f(x) + \log\left[\dfrac{P[y = -1]}{P[y = 1]}\right]\right)}$$

obteniendo una representación de la probabilidad a posteriori de forma sigmoidal con pendiente en $f(x) = 0$ igual a $2C \cdot r \cdot (1 + r)^{-2}$ donde r es el cociente de las probabilidades a priori. A partir de este desarrollo, se toma como solución aquella que resulta de ajustar una función sigmoidal de la forma

$$\sigma(x) = \frac{1}{1 + \exp(A \cdot f(x) + b)}$$

a los datos $\{(f(x_1), y_1), \cdots, (f(x_n), y_n)\}$.

Sin embargo, consideramos que ninguno de estos planteamientos se ajusta adecuadamente al problema de dar una interpretación probabilística a las máquinas de vectores soporte, a pesar del reciente trabajo dado en [LWZL01], ya que todos estos planteamientos resuelven un problema de optimización diferente del dado en las SVMs. No obstante, la aproximación dada en [Sol00] donde el objetivo es buscar un determinado conjunto de hipótesis, de tal manera que el problema que resulte de

maximizar una función de verosimilitud proporciona la solución SVM, nos parece más acertada y la pasamos a desarrollar a continuación, ya que la necesitaremos para la elaboración de nuestra máquina de soporte vectorial multiclasificadora.

4.2.1 Probabilidades en las SVMs

Supongamos dado un conjunto de entrenamiento $Z = \{(x_1, y_1), \cdots, (x_n, y_n)\}$ donde $\{x_1, \cdots, x_n\} \subset \mathcal{X} \subset \mathbb{R}^d$, espacio de los inputs, y los outputs $y_i \in \mathcal{Y} = \{-1, 1\}$ correspondientes a dos clases. Como se apuntó anteriormente, en el análisis de las SVMs, el primer paso teórico es transformar los inputs x en otros inputs $\phi(x) \in \mathbb{R}^{d'}$ dentro de algún espacio característico de grandes dimensiones ($d' >> d$) y dotado de un producto escalar.

Dentro de este espacio característico, consideramos los hiperplanos de decisión $\pi \equiv \mathbf{w} \cdot \phi(x) + b = 0$, de tal forma que el problema de optimización (3.20) queda como sigue:

$$\begin{array}{l} \min\limits_{\mathbf{w} \in \mathbb{R}^{d'}} \dfrac{1}{2} \|w\|^2 + C \sum\limits_{i=1}^{n} \xi_i \\ s.a. \quad \begin{cases} y_i \left(\mathbf{w} \cdot \phi(x_i) + b\right) - 1 + \xi_i \geq 0, \ \forall i \\ \xi_i \geq 0, \ \forall i \end{cases} \end{array} \tag{4.3}$$

En el conjunto de entrenamiento, se cumple que en aquellos vectores (x_i, y_i) donde $y_i(\mathbf{w} \cdot \phi(x_i) + b) \geq 1$ se verifica que $\xi_i = 0$, y por tanto al no incurrir en error no penaliza la función objetivo. Por otro lado, los restantes vectores de entrenamiento contribuyen cada uno a penalizar la función objetivo en la cantidad

$$C \cdot \xi_i = C \cdot [1 - y_i(\mathbf{w} \cdot \phi(x_i) + b)]$$

ya que de la igualdad (3.33) se sigue que de ser $\alpha_i \neq 0$ entonces $y_i(\mathbf{w} \cdot \phi(x_i) + b) - 1 + \xi_i = 0$ y de aquí $\xi_i = 1 - y_i(\mathbf{w} \cdot \phi(x_i) + b)$.

De esta forma el problema de optimización de las máquinas de vectores soporte queda de la siguiente forma: encontrar el vector de parámetros $\mathbf{w} \in \mathbb{R}^{d'}$ y el

parámetro $b \in \mathbb{R}$ que minimiza

$$\frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} l(y_i(\mathbf{w}\cdot\phi(x_i)+b)) \tag{4.4}$$

donde $l(z)$ es la denominada función de pérdida bisagra –del inglés *hinge loss*–, también llamada función de pérdida de margen suave –del inglés *soft margin*–,

$$l(z) = (1-z)\cdot H(1-z) \tag{4.5}$$

donde $H(z)$, la función de salto de Heaviside, es definida como sigue: $H(a) = 1$ si $a \geq 0$ y $H(a) = 0$ en otro caso.

Nota 4.2.1 *En muchos artículos es habitual encontrar otra expresión de la función bisagra. Si se define la función* $|x|_+ = \max\{x, 0\}$ *entonces se sigue que la función bisagra* $l(z) = |1-z|_+$ *como es fácil comprobar.* ▲

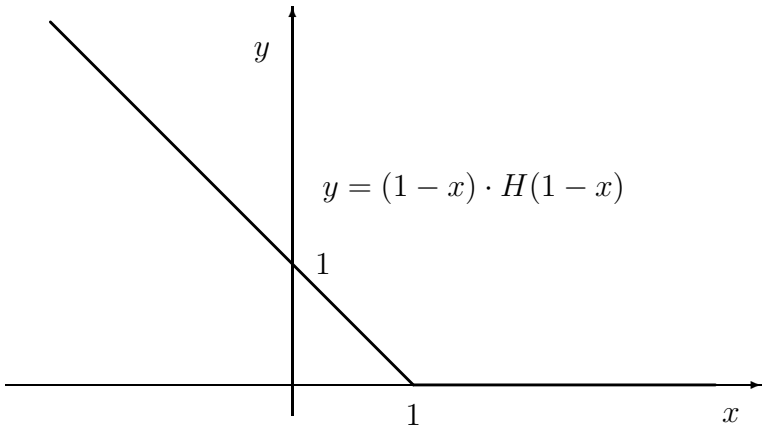

Figura 4.2: Función de pérdida bisagra o función hinge loss.

Para interpretar probabilísticamente las SVMs, debemos notar que en el problema (4.3), la solución

$$\mathbf{w} = \sum_{i=1}^{N_{sv}} \alpha_i\, y_i\, \phi(s_i), \qquad \text{y} \qquad b = \frac{1}{N_{sv}}\sum_{i=1}^{N_{sv}} (y_i - \mathbf{w}\,\phi(s_i) - y_i\,\xi_i)$$

son claramente funciones de la muestra $Z_\phi = \{(\phi(x_1), y_1), \cdots, (\phi(x_n), y_n)\}$. Luego si suponemos que la muestra Z_ϕ se ha obtenido siguiendo una determinada distribución de probabilidad, entonces tanto $\mathbf{w}$ como b son variables aleatorias por ser combinaciones lineales de variables aleatorias.

La idea es expresar (4.4) como menos el logaritmo de una función de verosimilitud[4] para los parámetros $\mathbf{w}$ y b de la SVM, dada a partir de un conjunto de ensayo Z_ϕ. Siguiendo este desarrollo, se tiene que la clasificación llevada a cabo por una máquina de vectores soporte es interpretada como la solución de un problema de maximización de una función de verosimilitudes (problema de maximizar a posteriores –MAP–) y entramos dentro de la inferencia estadística. A diferencia de las aproximaciones anteriores, el objetivo es identificar un problema de máxima verosimilitud de tal forma que su formulación coincida con la formulación del problema SVM.

Veamos como se puede interpretar el primer término de (4.4). Si consideramos que el vector de pesos $\mathbf{w}$ sigue una distribución normal conjunta con las componentes incorreladas y de varianza unidad, es decir $\mathbf{w} \in N_n(0, I)$ y b sigue una distribución normal de media 0 y varianza B^2, es decir $b \in N(0, B^2)$ e independiente de $\mathbf{w}$, entonces la distribución (a priori) conjunta es normal con función de densidad dada por:

$$Q(\mathbf{w}, b) \propto \exp\left(-\frac{1}{2}\|\mathbf{w}\|^2 - \frac{1}{2}b^2 B^{-2}\right)$$

Si sobre $Q(\mathbf{w}, b)$ tomamos logaritmo y cambiamos de signo se tiene

$$-\ln Q(\mathbf{w}, b) = \frac{1}{2}\|\mathbf{w}\|^2 + \frac{1}{2}b^2 B^{-2}$$

que es claramente distinto de $\frac{1}{2}\|\mathbf{w}\|^2$; esto es debido a que en (4.4) se ha supuesto un alisamiento en el sentido de considerar $B \to \infty$. Aunque en las SVMs no se tengan en cuenta el segundo sumando es preferible mantener B finito[5] dentro de

[4] En los problemas que se resuelven por máxima verosimilitud se máxima una función, si se toma la función objetivo cambiada de signo el problema se convierte en un problema de minimización.

[5] En algunos de los núcleos que se utilizan, como por ejemplo $k(x, x') = (\langle x, x' \rangle + 1)^p$, el factor b cae dentro del núcleo, luego su incorporación dentro del marco probabilístico es directa.

una interpretación probabilística (ver nota 4.2.3).

Puesto que es la función $\theta(x) = \mathbf{w} \cdot \phi(x) + b$, la que aparece en el segundo término de (4.4), parece lógico expresar la distribución a priori directamente como una distribución en $\theta(x)$. Para un input x fijado, $\phi(x)$ es un vector y

$$\theta(x) = \sum_{i=1}^{d'} w_i \, \phi_i(x) + b$$

es una variable aleatoria, que por ser combinación lineal de variables normales, sigue una distribución normal. Además se tiene que la covarianza de la función $\theta(\cdot)$ para los inputs x y x' viene dada por:

$$Cov(\theta(x), \theta(x')) = \phi(x) \cdot \phi(x') + B^2$$

ya que:

$$\begin{aligned} Cov(\theta(x), \theta(x')) &= Cov(\mathbf{w} \cdot \phi(x) + b, \mathbf{w} \cdot \phi(x') + b) \\ &= Cov(\mathbf{w}\,\phi(x), \mathbf{w}\,\phi(x')) + Cov(b, b) + 2\,Cov(\mathbf{w}\,\phi(x), b) \\ &= \phi(x) \cdot \phi(x')\,Cov(\mathbf{w}, \mathbf{w}) + Cov(b, b) + 2\,\phi(x)\,Cov(\mathbf{w}, b) \\ &... \quad \text{como varianza}(\mathbf{w}) = 1 \text{ y varianza}(b) = B^2 \text{ son independientes} \\ &= \phi(x) \cdot \phi(x') + B^2. \end{aligned}$$

Luego el primer término de (4.4) es a priori un proceso de Gauss – del inglés *Gauss Process* (GP) – sobre el conjunto de funciones $\{\theta(x)\}_{x \in \mathcal{X}}$, con media cero y función de varianza-covarianza:

$$k(x, x') = \widehat{k}(x, x') + B^2, \qquad \widehat{k}(x, x') = \phi(x) \cdot \phi(x'), \quad \forall x, x' \in \mathcal{X}. \tag{4.6}$$

Como ya indicamos, esta función (exceptuando el término aditivo B^2, el cual surge aquí cuando se incorpora el término independiente dentro de $\theta(x)$) es llamada función núcleo.

Nota 4.2.2 *Nótese que $\{\theta(x)\}_{x \in \mathcal{X}}$ es un proceso de Gauss ya que a priori hemos supuesto que $\mathbf{w} \in N(0, 1)$ y $b \in N(0, B^2)$, lo cual no tiene por qué ser cierto. Es importante indicar que Vapnik insiste mucho en distinguir el aprendizaje estadístico de un proceso gaussiano, aunque no es menos cierto que en la práctica se acaba por utilizar núcleos gaussianos.* ▲

Nota 4.2.3 *Intentemos aclarar el papel de B en estos desarrollos. Para ello escribimos $\theta(x) = \widehat{\theta}(x) + b$, donde $\widehat{\theta}(x)$ es un proceso de Gauss de media cero y b sigue una distribución normal de media cero y desviación típica B. Consideramos el caso donde $\widehat{k}(x,x)$ es independiente de*[6] *x, entonces una muestra proveniente de la distribución a priori tendrá valor de $\widehat{\theta}(x)$ dentro de un rango de orden*[7] *$\left(\widehat{k}(x,x)\right)$, alrededor de cero. Si $B^2 \gg \widehat{k}(x,x)$, entonces en $\theta(x) = \widehat{\theta}(x) + b$, el segundo término domina sobre el primero, lo que implica que probablemente todos los inputs tengan el mismo signo (el signo de b); y esto llega a ser cierto con probabilidad 1 si $B \to \infty$. La distribución a priori asigna entonces probabilidades distintas de cero, solo a dos clasificadores, aquellos, los cuales retorna la misma etiqueta para todos los inputs, es decir, los dos únicos clasificadores posibles son $f(x) = 1$ y $f(x) = -1$ para todo vector input x. Por ello con objeto de evitar esta situación patológica es adecuado mantener B finito en un contexto probabilístico.*

Además, es común, en muchos trabajos considerar $b = 0$, con lo que se consigue que el problema de optimización cuadrática presente una forma más compacta, en el sentido de expresar el conjunto de restricciones de manera más fácil. Esto es posible conseguir añadiendo una nueva componente al vector de input que recoge este término independiente. ▲

Veamos a continuación como podemos conseguir que el segundo término de (4.4) se exprese como un logaritmo de verosimilitudes con signo negativo. Para ello, definimos en primer lugar la[8] "probabilidad" de obtener el output y para x y θ

[6] Como en el caso de los núcleos RBF: $k(x,y) = \exp\left(-\frac{1}{2\sigma^2}\|x-y\|^2\right)$.

[7] Sería el resultado de aplicar el significado de la desviación típica como dispersión de los valores respecto de su media, en este caso $\mu = 0$.

[8] Las comillas se deben interpretar como que realmente no es una probabilidad, como posteriormente se demuestra, pero si es la base para su futura construcción.

dados como sigue:

$$Q(y = \pm 1|x, \theta) = \kappa(C) \exp\left[-C \cdot l(y\,\theta(x))\right] \tag{4.7}$$

donde $\kappa(C) = 1/\left[1 + \exp\left(-2C\right)\right]$ para asegurar que $Q(y = \pm 1|x, \theta)$ no alcance valores mayores que la unidad. Comprobemos que esto es cierto:
Sea $Q(y = \pm 1|x, \theta) = q(y)$, entonces como y solo puede tomar dos valores se debe cumplir que:

$$\sum_y q(y) = q(1) + q(-1) = \kappa(C)\left[\exp(-C \cdot l(\theta(x)) + \exp(-C \cdot l(-\theta(x))\right] \leq 1.$$

Teniendo en cuenta la definición de la función de pérdida $l(z) = (1 - z)\, H(1 - z)$ se tiene:

- Si $\theta(x) \leq -1$, entonces $-\theta(x) \geq 1$, luego

$$\begin{aligned} &\max_{\theta(x)<-1}\left[\exp(-C \cdot l(\theta(x)) + \exp(-C \cdot l(-\theta(x))\right] = \\ &\max_{\theta(x)<-1}\left[\exp(-C \cdot (1 - \theta(x)) + \exp(-C \cdot 0)\right] = \exp(-2C) + 1. \end{aligned}$$

- Si $\theta(x) \geq 1$, entonces de forma análoga al caso anterior se sigue que $\max_{\theta(x)>1}\left[\exp(-C \cdot l(\theta(x)) + \exp(-C \cdot l(-\theta(x))\right] = 1 + \exp(-2C)$
- Si $-1 < \theta(x) < 1$, entonces $\max_{-1<\theta(x)<1}\left[\exp(-C \cdot l(\theta(x)) + \exp(-C \cdot l(-\theta(x))\right] = 2\exp(-C)$

y como $2\exp(-C) \leq 1 + \exp(-2C)$, ya que $C > 0$ y $(1 - \exp(-C))^2 \geq 0$, se sigue que $\max_{\theta(x))} \sum_y q(y) = \kappa(C)(1 + \exp(-2C))$; luego tomando $\kappa(C) = 1/\left[1 + \exp(-2C)\right]$ se tiene que $\sum_{y=\pm 1} Q(y|x, \theta) \leq 1$.

Veamos como se interpreta $Q(y = 1|x, \theta)$ (análogamente para $y = -1$), es decir cual es la "probabilidad"de asignar la etiqueta $y = 1$ en función de los valores que tome $\theta(x)$. Intuitivamente, si $\theta(x) \geq 1$, la probabilidad de asignar la etiqueta $y = 1$ debe crecer conforme $\theta(x)$ lo haga, puesto que se aleja cada vez más del hiperplano separador. Por otro lado si $\theta(x) < 1$, la probabilidad de asignar la etiqueta $y = 1$

debe disminuir conforme $\theta(x)$ lo haga, puesto que nos vamos aproximando a la región correspondiente a la clase $y = -1$. La "probabilidad" $q(1)$ se interpreta de esa forma, ya que si $\theta(x) \geq 1$, entonces $q(1) = \kappa(C)\exp(-C \cdot 0) = ...$ (ya que $l(\theta(x)) = 0$) $.. = 1/\left[1 + \exp(-2C)\right]$ que es máximo. Si $\theta(x) < 1$ entonces $q(1) = \kappa(C)\exp(-C \cdot (1 - \theta(x))$, que disminuye conforme lo hace $\theta(x)$ y en el límite se tiene que $q(1) \to 0$ si $\theta(x) \to -\infty$.

Estos desarrollos quedan recogidos gráficamente en la figura 4.3. Como se puede

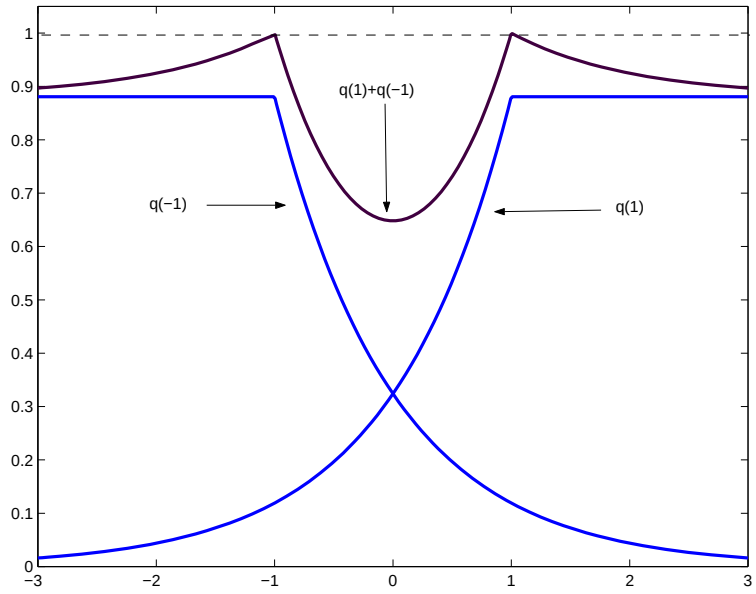

Figura 4.3: Representación gráfica de las funciones $q(1) = Q(y = 1|x, \theta)$, $q(-1) = Q(y = -1|x, \theta)$ y $\nu(\theta(x))$ con $C = 1$.

observar en la figura, la función $q(1)$ tiene una forma casi sigmoidal[(9)], por ello se podría pensar en redefinir la función de coste para obtener una función de este tipo, pero como ya se comentó anteriormente presenta el inconveniente de que se pierde

[(9)] Hay que destacar que a diferencia de las funciones sigmoidales, ésta no es derivable en el punto $\theta(x) = 1$. En el mundo de las redes artificiales, las sigmoides se introdujeron para substituir la función signo en la regla de aprendizaje del perceptrón y así se pudo generar la regla BackPropagation. Es por ello, por lo que el profesor Cecilio Angulo indica que sería más adecuado indicar que presenta forma de exponencial creciente truncada.

la propiedad de escasez en los vectores soporte.

La verosimilitud para el conjunto de entrenamiento Z es:

$$Q(Z|\theta) = \prod_{i=1}^{n} Q(y_i|x_i,\theta)\, Q(x_i)$$

por la independencia de la muestra (m.a.s) y ya que dado un valor concreto de θ, los valores y_i están condicionados a él según (4.7) pero no los de x_i. Por $Q(x)$ denotamos la distribución de probabilidad que siguen los inputs.

Hemos de destacar que $Q(y|x,\theta)$ no se encuentra normalizada para los diferentes valores de x y θ, ya que si denotamos

$$\nu(\theta(x)) = Q(1|x,\theta) + Q(-1|x,\theta)$$

(función de verosimilitud del parámetro $\theta(x)$), ya se demostró anteriormente que $\nu(\theta(x)) < 1$, excepto cuando $|\theta(x)| = 1$ (ver figura 4.3).

Por otro lado, la suma de todas las verosimilitudes $Q(Z|\theta)$ sobre todos los posibles conjuntos de entrenamientos Z con un tamaño dado n es:

$$\sum_{Z} Q(Z|\theta) = \left(\sum_{x} Q(x)\, v(\theta(x)) \right)^{n}. \tag{4.8}$$

Para la demostración hacemos uso de la independencia de la muestra aleatoria simple, y la realizamos para el caso continuo:

$$\begin{aligned}
\textstyle\int_Z Q(Z|\theta)\, dZ &= \int_{(x_1,y_1)} \cdots \int_{(x_n,y_n)} \prod_{i=1}^{n} Q(y_i|x_i,\theta)\, Q(x_i)\, d(x_1,y_1) \cdots d(x_n,y_n) \\
&= \prod_{i=1}^{n} \int_{(x_i,y_i)} Q(y_i|x_i,\theta)\, Q(x_i)\, d(x_i,y_i) \\
&= \left(\int_{(x,y)} Q(y|x,\theta)\, Q(x)\, d(x,y) \right)^{n} \\
&= \left(\int_x Q(x) \int_y Q(y|x,\theta) dy\, dx \right)^{n} = \left(\int_x Q(x)\, \nu(\theta(x))\, dx \right)^{n}
\end{aligned}$$

y en el caso discreto se tiene (4.8). En general (4.8) es menor que la unidad, por la

propia definición de las $Q(y|x,\theta(x))$ y supondremos a partir de ahora que trabajamos en el caso discreto[10].

Puesto que $\nu(\theta(x)) = \sum_y Q(y|x,\theta(x)) \leq 1$, parece evidente que necesariamente hemos llevar a cabo la normalización de las "probabilidades" $Q(y|x,\theta(x))$. La normalización lógica sería reemplazar $Q(y|x,\theta(x))$ por $Q(y|x,\theta(x))/\nu(\theta(x))$ y de esta forma se tendrían las probabilidades normalizadas, pero presenta el inconveniente de no poderse escribir la función a minimizar en las SVMs como la solución a un problema de maximización de verosimilitudes, ya que $-\ln Q(y|x,\theta(x))/\nu(\theta(x))$ no se puede expresar como una cantidad proporcional a la función de pérdida $l(z)$:

$$-\ln Q(y|x,\theta(x))/\nu(\theta(x)) = C\cdot l(y\theta(x))) - \ln\left(\sum_y Q(y|x,\theta(x))\right) \neq Cte\cdot l(y\theta(x)))$$

Para solucionar este problema consideramos el siguiente modelo de probabilidad conjunta:

$$P(Z,\theta) = Q(Z|\theta)\cdot Q(\theta)/\mathcal{N}(Z) \tag{4.9}$$

En este modelo la probabilidad a posteriori $P(\theta|Z) = Q(\theta|Z) \propto Q(Z|\theta)\,Q(\theta)$ es independiente del factor de normalización $\mathcal{N}(Z)$, y de esta forma queda igual que el modelo sin normalizar. Por construcción, el valor que maximiza la función de verosimilitud de θ coincide con la solución dada por la SVM. Así, se elige $\mathcal{N}(Z)$ tal que normalice $P(Z,\theta)$: $\int_Z\int_\Theta P(Z,\theta)\,d\theta\,dZ = 1$, luego:

$$\begin{aligned}\mathcal{N}(Z) &= \int_Z\int_\Theta Q(Z|\theta)Q(\theta)\,d\theta\,dZ = \int_\Theta Q(\theta)\left(\int_Z Q(Z|\theta)\,dZ\right)d\theta \\ &= \int_\Theta Q(\theta)\left(\int_x Q(x)\nu(\theta(x)\,dx\right)^n d\theta\end{aligned}$$

es decir (tomando el espacio input discreto),

$$\mathcal{N}(Z) = \mathcal{N} = \int_\Theta Q(\theta)\,N^n(\theta)\,d\theta, \qquad N(\theta) = \sum_x Q(x)\,\nu(\theta(x)\,dx. \tag{4.10}$$

[10]De esta forma se evita trabajar con determinantes e inversos de operadores que complican de sobremanera la interpretación probabilística, además de proporcionar un marco suficientemente general para una aplicación práctica. También, al menos conceptualmente, un espacio input continuo puede ser siempre discretizado tomando una partición suficientemente fina.

Si se calcula la distribución marginal de θ a partir de (4.9) y (4.10) se tiene:

$$P(\theta) \propto Q(\theta)\, N^n(\theta) \tag{4.11}$$

que depende del tamaño del conjunto de ensayo n, que refleja, lo que se conoce como, la probabilidad de "supervivencia" de θ, ya que $\nu(\theta(x))$ es más pequeña[(11)] dentro del conjunto $|\theta(x)| < 1$ y será tanto menor cuanto mayor sea n.

Si tomamos $Z = \{(x, y)\}$ (una muestra de tamaño 1) se tiene que:

$$\begin{aligned} P(\{(x,y)\}\,|\theta) &= \frac{P(\{(x,y)\},\theta)}{P(\theta)} = \frac{Q(y,x|\theta)\,Q(\theta)/\mathcal{N}(Z)}{Q(\theta)\,N(\theta)/\mathcal{N}(Z)} \\ &= \frac{Q(y,x|\theta)}{N(\theta)} = \frac{Q(y|x,\theta)}{\nu(\theta(x))} \cdot \frac{Q(x)\,\nu(\theta(x))}{N(\theta)} \\ &= P(y|x,\theta)\,P(x|\theta) \end{aligned}$$

La verosimilitud

$$P(Z|\theta) = \prod_{i=1}^{n} P(y_i|x_i,\theta)\,P(x_i|\theta) \tag{4.12}$$

es un producto de verosimilitudes de muestras de tamaño uno. La probabilidad condicional para el output y,

$$P(y|x,\theta) = \frac{Q(y,x|\theta)}{\nu(\theta(x))} \tag{4.13}$$

ya se encuentra normalizada y se puede calcular explícitamente[(12)]:

$$P(y|x,\theta) = \begin{cases} \dfrac{1}{1+e^{-2\,C\,y\,\theta(x)}} & \text{si} \quad |\theta(x)| \le 1 \\ \dfrac{1}{1+e^{-C\,y\,[\theta(x)+signo(\theta(x))]}} & \text{si} \quad |\theta(x)| > 1 \end{cases} \tag{4.14}$$

La representación gráfica de las funciones $P(1|x,\theta)$ y $P(-1|x,\theta)$ se tiene en la figura

[(11)] Para que esto sea verdad hemos de suponer que se toma $C > \ln 2$, ya que si $-1 \le \theta(x) \le 1$ entonces $\nu(\theta(x)) = 2\exp(-C) \le 1$. De esto, se tiene que para valores pequeños de C, $\nu(\theta(x))$ es mayor dentro de $-1 \le \theta(x) \le 1$ y el modelo tiene menos sentido intuitivo, ya que la asignación de etiquetas debe ser tanto más clara cuanto mayor sea $|\theta(x)| > 1$, puesto que la lógica indica que si $\theta(x) > 1$, se asigna la etiqueta $y = 1$, y si $\theta(x) < -1$, se asigna la etiqueta $y = -1$. La zona de mayor incertidumbre debe quedar entre estos dos extremos.

[(12)] El cálculo no es complicado pero si algo engorroso.

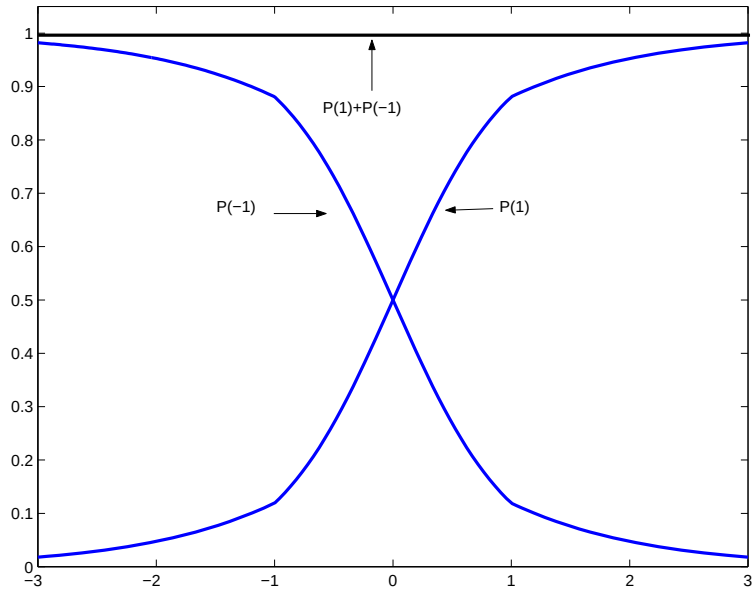

Figura 4.4: Representación gráfica de las funciones $p(1) = P(y = 1|x, \theta)$, $p(-1) = P(y = -1|x, \theta)$ y su suma para $C = 1$.

4.4, donde se observa que las probabilidades tienen forma sigmoidal dependiente de $\theta(x)$, con discontinuidad en la pendiente en los puntos $\theta(x) = \pm 1$.

Si consideramos que $P(y|x, \theta) = f(\theta(x))$ y calculamos su derivada en $-1 < \theta(x) < 1$ se tiene que:

$$\frac{\partial f}{\partial \theta(x)} = \frac{e^{-2\,C\,y\,\theta(x)} \cdot 2\,C\,y}{\left(1 + e^{-2\,C\,y\,\theta(x)}\right)^2} \quad \text{y si } \theta(x) = 0, \quad \left.\frac{\partial f}{\partial \theta(x)}\right|_{\theta(x)=0} = \frac{\pm\, C}{2}$$

(ya que $y = 1$ ó $y = -1$) se sigue, entonces, que el parámetro que penaliza las clasificaciones erróneas C es proporcional a la pendiente a las curvas en el origen y podemos interpretar $1/C$ como el nivel de ruido el cual mide, cómo de estocásticos son los outputs. Esto puede observarse en la figura 4.5 donde se tiene que para $C = 5$ la mayor aleatoriedad de y se presenta cuando $-0'5 < \theta(x) < 0'5$, para $C = 2$ la mayor aleatoriedad de y se presenta cuando $-1'5 < \theta(x) < 1'5$ y sin embargo para $C = 1$, la mayor aleatoriedad de y se presenta en un entorno más amplio. Es interesante esta interpretación del valor C de intercambio entre suavidad y ajuste, ya que nos permite tener un criterio para elegir este parámetro dentro del problema

de optimización SVM.

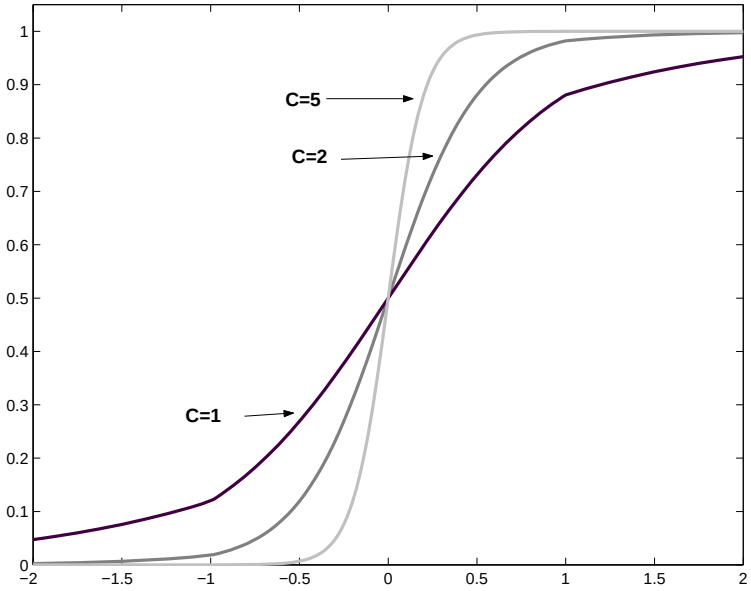

Figura 4.5: Representación gráfica de las funciones $P(y = 1|x, \theta)$ para los valores de C= 1, 2 y 5 ($C > \ln 2$).

De (4.12) se sigue que la función de densidad de los inputs es:

$$P(x|\theta) = \frac{Q(x)\,\nu(\theta(x))}{\sum_x Q(x)\,\nu(\theta(x))} \tag{4.15}$$

que claramente se encuentra normalizada, y dentro de $-1 < \theta(x) < 1$ la probabilidad respecto de $Q(x)$ se reduce, puesto que $\nu(\theta(x))$ tiende a ser más pequeña. De esta manera, el modelo asume implícitamente que las regiones de datos con probabilidad grande en los inputs y outputs bien determinados (valores grandes de $|\theta(x)|$) son separados por la región $|\theta(x)| < 1$, con una probabilidad pequeña en los inputs, ya que la función $\nu(\theta(x))$ es más pequeña en esta región cuanto más próximo se este de $\theta(x) = 0$ (ver 4.3); y unos outputs más inciertos, ya que las probabilidades $P(1/x, \theta)$ y $P(-1/x, \theta)$ toman valores alto, como puede observarse en la figura 4.4.

Las ecuaciones (4.11-4.15) definen un modelo de inferencia cuya solución $\theta^* = \arg\max P(\theta|Z)$ para un conjunto de entrenamiento Z coincide con la solución que

se obtiene en la SVM estándar. La verosimilitud (4.12, 4.13, 4.15) no solo definen una distribución condicional de los outputs para cada función θ, sino también una distribución de los inputs (relativa a alguna distribución arbitraria $Q(x)$). Así, tenemos realmente un modelo conjunto input-output. Además, todas las propiedades relevantes del espacio característico están recogida en el proceso de Gauss $Q(\theta)$, elegido a priori, con función de covarianza igual al núcleo $k(x, x')$.

Tratando de forma conjunta la interpretación probabilística de (4.4) como el logaritmo de una función de verosimilitud se tiene:

$$\ln P(\theta|Z) = \frac{1}{2}\sum_{i,j=1}^{n} \theta(x_i)\, k^{-1}(x_i, x_j)\, \theta(x_j) - C\sum_{i=1}^{n} l(y_i\theta(x_i)) + const \tag{4.16}$$

(donde $k^{-1}(x_i, x_j)$ son los elementos de la inversa de $k(x_i, x_j)$, considerando $\{k(x_i, x_j)\}$ como una matriz de doble entrada), donde $\theta = (w, b)$ y claramente por construcción, se tiene que el máximo de esta verosimilitud $\theta^*(x)$ coincide con la solución SVM. Por tanto es posible escribir θ^* en la forma:

$$\theta^*(x) = \sum_{i=1}^{n} \alpha_i\, y_i\, k(x, x_i) + b \tag{4.17}$$

Si denotamos $\theta^*(x_i) = \theta_i$, se sigue que la función $f(x, y) = y \cdot \theta^*(x)$ divide el espacio de los inputs en tres bloques, dependiendo de si $y_i\,\theta_i > 1$, $= 1$, ó < 1, donde respectivamente se tiene que $\alpha_i = 0$, $\alpha_i \in [0, C]$ ó $\alpha_i = C$. Estos datos de ensayo son llamados fáciles (clasificación correcta), marginales o difíciles, respectivamente. Como ya se sabe a los datos de ensayos tales que $\alpha_i > 0$ se les denominan vectores soporte.

4.3 Comentarios sobre el capítulo

En este capítulo hemos dado a las salidas de las máquinas de vectores soporte biclasificadoras una interpretación probabilística, sin modificar el problema de optimización original.

Además, el proceso de generalización tampoco ha sufrido modificación ya que si la solución $\theta^*(x) > 0$ entonces $P(Y = 1/\theta^*(x)) > P(Y = -1/\theta^*(x))$ y la salida de la máquina es $Y = 1$; y análogamente si $\theta^*(x) < 0$, la salida de la máquina es $Y = -1$.

Sin embargo, como veremos en el siguiente capítulo, cuando se plantee un problema de multiclasificación utilizaremos las probabilidades para llevar a cabo el proceso de interpretación de la solución y resolver posibles empates entre etiquetas.

CAPÍTULO 5

MÁQUINAS DE VECTORES SOPORTE PARA LA MULTICLASIFICACIÓN

Suponer es barato pero una suposición errónea puede ser muy costosa.

–Proverbio chino–

Los errores de un cocinero con salsa se cubren. Los errores de un arquitecto con flores se cubren. Los errores de un médico se cubren .... con tierra.

–Anónimo–

Hasta este momento, se han considerado problemas de clasificación con solo dos clases. Sin embargo, frecuentemente en los problemas reales es necesario discriminar entre más de dos clases ($\ell > 2$). Como comenta el profesor Angulo en [Ang01]:

"Cuando se realiza trabajo teórico de una máquina de aprendizaje, si ésta ha sido especialmente diseñada para casos binarios como las SVMs,

se soluciona la posibilidad de trabajo en entorno multiclase afirmando que su generalización a tales problemas es evidente".

Al igual que él, pensamos que el paso a un problema de multiclasificación no es tan evidente como, a primera vista, puede parecer y se ha de elaborar una metodología precisa que nos permita resolver este problema de la forma más adecuada posible. Por ello, en este capítulo proponemos una nueva máquina de vectores soporte para la multiclasificación que basada en la máquina ℓ-SVCR proporciona una salida probabilística que resuelve el problema de asignación de etiqueta en caso de empate y nos proporciona un grado de confianza de la fiabilidad que un investigador deposita en el modelo.

5.1 Introducción

Sea $\{x_1, \cdots, x_n\} \subset \mathcal{X} \subset \mathbb{R}^n$ un conjunto de vectores inputs e $\mathcal{Y} = \{\theta_1, \theta_2, \cdots, \theta_\ell\}$ el conjunto de todas las posibles etiquetas, con $\ell > 2$ (si ℓ=2 se tiene las SVMs estudiadas en los capítulos anteriores).

En primer lugar, nótese que el espacio de outputs no es, necesariamente, un subconjunto de $\mathbb{R}$ como en estudios anteriores, pero esto es así simplemente por notación, ya que es posible asignar a cada etiqueta un valor entero del conjunto $\{1, 2, \cdots, \ell\}$, pero debemos recordar que sobre estas salidas no podemos realizar ningún tipo de estudio que se base en la ordenación pues las etiquetas utilizan una escala de tipo nominal. Sin embargo, puesto que en la construcción de las distintas máquinas hay que renombrar las salidas es adecuado considerar, en los problemas de multiclasificación, como output directamente las etiquetas θ_k, $k = 1, \cdots, \ell$.

Dentro del conjunto de entrenamiento $Z = \{(x_i, y_i)\}_{i=1}^n$ es conveniente realizar una partición a partir de los conjuntos

$$Z_k = \{(x_i, y_i), \text{ tales que } y_i = \theta_k\},$$

con lo que se tiene: $\bigcup_{k=1}^{\ell} Z_k = Z$ y para cualquier $k \neq h$ se sigue $Z_k \bigcap Z_h = \emptyset$, como es fácil comprobar. Denotamos por n_k el número de vectores de entrenamiento del conjunto Z_k con lo que se tiene: $n = n_1 + n_2 + \cdots + n_\ell$; y por I_k el conjunto de índices i tales que $(x_i, y_i) \in Z_k$ de donde se sigue que $\bigcup_{i \in I_k} \{(x_i, y_i)\} = Z_k$.

La forma, más habitual, de utilización de las máquinas de vectores soporte para resolver problemas de multiclasificación admite dos tipos de arquitectura:

- Máquinas biclasificadoras SV generalizadas: Construyen una función clasificadora global a partir de un conjunto de funciones clasificadoras dicotómicas (biclasificadoras).
- Máquinas multiclasificadoras SV: Construyen una función clasificadora global directamente considerando todas las clases a la vez.

5.2 Máquinas biclasificadoras SV generalizadas

Este tipo de máquinas da solución al problema de la multiclasificación transformando las ℓ particiones del conjunto de entrenamiento en un conjunto de L biparticiones, en las cuales construye la correspondiente función discriminadora (es lo que se denomina **esquema de descomposición**) obteniendo $f_1, \cdots f_L$ clasificadores dicotómicos o biclasificadores. A continuación, mediante un **esquema de reconstrucción**, realiza la fusión de los biclasificadores f_i, $i = 1, \cdots, L$ con objeto de proporcionar como salida final, una de las ℓ clases posibles, $\{\theta_1, \cdots, \theta_\ell\}$.

Dentro del esquema de descomposición, las máquinas más utilizadas son:

- Máquinas 1-v-r SV (iniciales de *one- versus- rest*). Máquinas de vectores soporte, donde cada función clasificadora parcial f_i, enfrenta la clase θ_i contra el resto de las clases.
- Máquinas 1-v-1 SV (iniciales de *one- versus- one*). Máquinas de vectores

soporte, donde cada función clasificadora parcial f_{ij}, enfrenta la clase θ_i contra la clase θ_j, sin considerar las restantes clases.

Una vez construidas las L máquinas biclasificadoras, según estos métodos, se ha de determinar la respuesta global de la máquina frente a un nuevo input x. Para ello, se tiene en cuenta principalmente las L etiquetas proporcionadas (se incluye una nueva etiqueta $\theta_0 = \emptyset$, para aquellos casos en los que la máquina no selecciona una etiqueta concreta, contabilizando de esta forma todos los "No" votos en esta etiqueta artificial) por las funciones discriminadoras f_k, así como sus correspondientes salidas numéricas $f_k(x)$, $k = 1, 2, \cdots, L$.

El método de reconstrucción más habitual, es el **esquema de votación**, donde se tiene en cuenta exclusivamente las etiquetas proporcionadas por las L máquinas biclasificadoras, $\{f_1, \cdots, f_L\}$. De esta forma, el esquema de reconstrucción parte de un conjunto formado por las etiquetas $\{\theta_k^*\}_{k=1}^L$ donde $\theta_k^* = \theta_i$ para algún $i = 0, 1, \cdots, \ell$. A partir de este conjunto realiza un recuento de todas las etiquetas (sin tener en cuenta, cuando aparezcan, la etiqueta $\theta_0 = \emptyset$):

Etiquetas	Votos
θ_1	m_1
$\vdots$	$\vdots$
θ_k	m_k
$\vdots$	$\vdots$
θ_ℓ	m_ℓ
	$L - m_0$

donde m_i, con $i = 1, \cdots, \ell$ es el número de veces que las máquinas f_k, $k = 1, \cdots, L$ asignan sus votos a la etiqueta θ_i; y m_0 es el número de veces que las máquinas f_k, $k = 1, \cdots, L$ no asignan ninguna etiqueta concreta.

Un esquema de reconstrucción posible, es la **votación por unanimidad**. En este esquema se toma como salida global de la máquina aquella etiqueta que haya obtenido todos los votos posibles. A veces, es más adecuado considerar un esque-

ma de **votación por mayoría absoluta** donde se toma como salida global de la máquina aquella etiqueta que haya obtenido más de la mitad de los votos posibles. Otra posibilidad, es considerar un esquema de **votación por mayoría simple** donde se toma como salida global de la máquina aquella etiqueta que haya obtenido más votos (la moda de la distribución de las etiquetas $\{\theta_k^*\}_{k=1}^L$).

En este último esquema de votación se puede presentar empates entre etiquetas. Este problema permite diferentes soluciones, según el tipo de arquitectura de la máquina, pero actualmente no existe una solución aceptada como válida por todos los investigadores.

5.2.1 Máquinas 1-v-r SV

Esta aproximación del problema fue dada por Vapnik en [Vap98]. Este tipo de máquina multiclasificadora construye ℓ bi-clasificadores donde la función discriminante f_k, $k = 1, 2, \cdots, \ell$ discrimina los vectores de entrenamiento de la clase k, Z_k, del resto de vectores de las otras clases, $Z \setminus Z_k$, esto es, si el biclasificador f_k lleva a cabo la discriminación de las clases sin error, entonces $sign(f_k(x_i)) = 1$, si el vector $x_i \in Z_k$ y $sign(f_k(x_i)) = -1$, si el vector $x_i \in Z \setminus Z_k$.

De esta forma, dado un nuevo input x, la salida numérica de la máquina $f_k(x)$ se interpreta de la siguiente forma:

$$\Theta(f_k(x)) = \begin{cases} \theta_k & \text{si } sign(f_k(x)) = 1 \\ \theta_0 & \text{si } sign(f_k(x)) = -1 \end{cases} \tag{5.1}$$

es decir, la función $\Theta(\cdot)$ etiqueta cada vector input x, en función del valor $f_k(x)$ dado por la máquina.

En este esquema, por construcción, ninguna etiqueta puede tener dos votos puesto que aparece explícitamente en una sola máquina. Si existe un único voto entonces la máquina global asignara aquella clase que haya obtenido este voto, puesto que en estas circunstancias todos los métodos de votación coinciden. El problema se planteará cuando haya más de una etiqueta con votos.

Las características de este multiclasificador son las siguientes:

c1.- Se necesitan estimar ℓ funciones biclasificadoras, es decir, se han de resolver ℓ problemas SVM estándar.

c2.- En la construcción de los biclasificadores intervienen todos los elementos del conjunto de ensayo Z, es decir, los biclasificadores disponen de toda la información proporcionada por los datos.

c3.- Es conocido[(1)], y está contrastado, que este procedimiento normalmente proporciona buenos resultados.

Los dos principales inconvenientes que presentan este tipo de máquinas multiclasificadoras son:

i1.- En caso de tener dos o más etiquetas empatadas en número de votos, no se encuentra dentro de la construcción un indicador que nos permita discriminar entre ellas. Podría pensarse en la utilización de las salidas numéricas $f_k(x)$, pero éstas no son adecuadas, por la propia naturaleza de las SVMs, puesto que la solución SVM se construye obligando a que la separación entre las dos clases sea la unidad[(2)].

i2.- No es posible asignar un nivel de confianza a la salida global, a partir de los ℓ biclasificadores.

5.2.2 Máquinas 1-v-1 SV

En esta aproximación del problema de multiclasificación se construyen $L = \dfrac{\ell \cdot (\ell - 1)}{2}$ biclasificadores donde la función discriminante f_{kh}, $1 \leq k < h \leq \ell$ discrimina los vectores de entrenamiento de la clase k, Z_k, de los vectores de entrenamiento de la

[(1)] Ver por ejemplo en [Vap98].

[(2)] Ver [MA99], para un estudio más detallado.

clase h, Z_h, esto es, si el biclasificador f_{kh} lleva a cabo la discriminación de las clases sin error, entonces $sign(f_{kh}(x_i)) = 1$, si el vector $x_i \in Z_k$ y $sign(f_{kh}(x_i)) = -1$, si el vector $x_i \in Z_h$. Los restantes vectores de entrenamiento $Z \setminus \{Z_k \bigcup Z_h\}$ no se consideran en la construcción del problema de optimización.

De esta forma, dado un nuevo input x, la salida numérica de la máquina $f_{kh}(x)$ se interpreta de la siguiente forma:

$$\Theta(f_{kh}(x)) = \begin{cases} \theta_k & \text{si } sign(f_{kh}(x)) = 1 \\ \theta_h & \text{si } sign(f_{kh}(x)) = -1. \end{cases}$$

En esta construcción, el número máximo de votos que puede tener una determinada clase θ_k es $\ell - 1$, ya que es el número de veces que aparece en la construcción de las funciones f_{kh}. De esta forma, si se utiliza el esquema de votación por unanimidad, se considera como salida global de la multiclasificación aquella etiqueta θ_k que haya obtenido $\ell - 1$ votos, que por construcción, si existe, debe ser única[(3)]. Si no es posible aplicar este esquema, porque no existe tal etiqueta, se puede aplicar el esquema de votación por mayoría absoluta, en este caso se toma como salida global, la etiqueta θ_k tal que $m_k > m_h$ donde $h \neq k$ para todo h y $m_k > \frac{\ell-1}{2}$. Si se adopta el esquema de votación por mayoría simple entonces se toma como salida global, la etiqueta θ_k tal que $m_k > m_h$ donde $h \neq k$ para todo h. Si finalmente, se presenta una situación con dos o más etiquetas empatadas en número de votos, habrá necesariamente que acudir a alguna otra característica procedente de la máquina 1-v-1, que nos permita elegir entre ellas.

Las características, más significativas, de este multiclasificador son las siguientes:

c1.- Se necesita estimar $\frac{\ell \cdot (\ell-1)}{2}$ funciones biclasificadoras, es decir, es necesario entrenar $\frac{\ell \cdot (\ell-1)}{2}$ máquinas SV estándar, aunque con un conjunto de entrenamiento más reducido.

[(3)] Ya que es imposible que para un input x se diese que $m_i = m_j = \ell - 1$, puesto que la máquina que las enfrenta f_{ij} solo puede asignar una etiqueta a la entrada x.

c2.- Es posible asignar un nivel de confianza a la salida global, utilizando la interpretación probabilística de las SVMs[(4)].

c3.- Es conocido que este procedimiento es, generalmente, preferido al esquema 1-v-r como así lo demuestran diferentes estudios empíricos[(5)].

Los dos principales inconvenientes que presentan este multiclasificador son:

i1.- Cada uno de los biclasificadores es entrenado con datos extraídos de solo dos clases del conjunto de entrenamiento por lo que la varianza es mayor y no proporciona información sobre el resto de clases. Además, cada máquina f_{kh} entrenada, no utiliza la información disponible en los datos que quedan fuera de las etiquetas θ_k y θ_h, lo que supone una preocupante pérdida de información.

i2.- El número de clasificadores, en comparación con las máquinas 1-v-r es alto, si el número de etiquetas ℓ es grande. Por ejemplo, si $\ell = 6$ se tendría que entrenar 15 máquinas.

Nota 5.2.1 *A pesar de los buenos resultados apuntados por este esquema de multiclasificación, el principal inconveniente que presenta es el de no utilizar toda la información disponible dentro del conjunto de entrenamiento. Por ejemplo, si consideramos la función biclasificadora f_{kh} y tomamos cualquier vector input del conjunto de entrenamiento x_i tal que $i \notin I_k \bigcup I_h$, para que esta función no nos proporcione una salida incorrecta debe verificar que $f_{kh}(x_i) = 0$. De esta forma, en la construcción de esta máquina estamos obligando a que todas las clases distintas de θ_k y θ_h estén dentro del hiperplano $f_{kh}(x) = 0$, y lo que resulta menos creible, sin tener en cuenta las características de estas clases.* ▲

[(4)]Este punto se verá con detalle cuando se estudien las máquinas ℓ-SVCRs. Indicar que no hemos encontrado ningún trabajo que asigne probabilidades en un entorno de multiclasificación al igual que nosotros hacemos en este trabajo.

[(5)]Ver por ejemplo en [Kre99].

5.3 Máquinas multiclasificadoras SV

Dentro de las máquinas de vectores soporte, también, es posible obtener de forma directa un multiclasificador, incorporando todas las etiquetas directamente en la configuración de un único problema de optimización.

Una primera aproximación se da en [Vap98]. Siguiendo la notación dada en la anterior referencia, denotamos los vectores de entrenamiento por

$$x_1^1, \cdots, x_{n_1}^1, \cdots, x_1^\ell, \cdots, x_{n_\ell}^\ell$$

donde el superíndice k en x_i^k denota que el vector pertenece a la clase k. Consideramos el conjunto de funciones lineales

$$f_k(x) = w^k \cdot x + b_k, \qquad k = 1, \cdots, \ell.$$

El objetivo es construir ℓ funciones (obtener ℓ pares (w^k, b_k) de parámetros) tales que para cada vector input x, el clasificador

$$m = \arg \max_{k=1,\cdots,\ell} \left\{ w^k \cdot x + b_k \right\}$$

discrimine adecuadamente todos los vectores de entrenamiento sin error. Esto es, que las desigualdades

$$w^k \cdot x_i + b_k - w^m \cdot x_i - b_m \geq 1$$

sean ciertas para todo $k = 1, \cdots, \cdots, \ell$, $m \neq k$ e $i = 1, \cdots, n$. Si existen soluciones a este problema, se elige entre ellas, el par de parámetros (w^k, b_k), $k = 1, \cdots, \ell$ para el cual, el funcional

$$\sum_{k=1}^{\ell} \left\| w^k \right\|^2 = \sum_{k=1}^{\ell} w^k \cdot w^k$$

sea mínimo[6].

[6] Se sigue de la configuración que para el caso $\ell = 2$ se tiene el problema estándar de SVM para la biclasificación.

Si, por contra, el conjunto de entrenamiento no puede ser discriminado sin provocar error en la clasificación, entonces el objetivo es minimizar el funcional[7]

$$\sum_{k=1}^{\ell} \left\|w^k\right\|^2 + C \sum_{k=1}^{\ell} \sum_{i=1}^{n} \xi_i^k$$

sujeto a las restricciones

$$w^k \cdot x_i + b_k - w^m \cdot x_i - b_m \geq 1 - \xi_i^k,$$

para $i = 1, \cdots, n$ y $1 \leq k,\, m \leq \ell$. Para resolver este problema de optimización se usa las mismas técnicas de optimización que en el caso de las SVMs con dos clases y se obtiene que:

1. La función $f_k(x)$ presenta el siguiente desarrollo en términos de los vectores soporte:

$$f_k(x) = \sum_{m \neq k} \sum_{i=1}^{n_k} \alpha_i(k,m)\, x \cdot x_i^k - \sum_{m \neq k} \sum_{j=1}^{n_m} \alpha_i(m,k)\, x \cdot x_i^m + b_k$$

2. Los coeficientes $\alpha_i(k,m)$, $k = 1, \cdots, m$, $m \neq k$, $i = 1, \cdots, n_k$, $j = 1, \cdots, n_m$ de este desarrollo tienen que maximizar la forma cuadrática:

$$\begin{aligned} W(\alpha) = \sum_{k=1}^{\ell} \sum_{m \neq k} \Bigg[& \sum_{i=1}^{n_k} \alpha_i(k,m) \\ & -\frac{1}{2} \sum_{m^* \neq k} \Bigg(\sum_{i,j=1}^{n_k} \alpha_i(k,m^*)\, \alpha_j(k,m)(x_i^k \cdot x_j^k) \\ & + \sum_{i=1}^{n_m} \sum_{i=1}^{n_m^*} \alpha_i(m,k)\, \alpha_j(m^*,k)(x_i^m \cdot x_j^{m^*}) \\ & -2 \sum_{i=1}^{n_k} \sum_{i=1}^{n_m} \alpha_i(k,m^*)\, \alpha_j(m,k)(x_i^k \cdot x_j^m) \Bigg) \Bigg] \end{aligned}$$

sujetos a las restricciones

$$0 \leq \sum_{m \neq k} \alpha_i(k,m) \leq C,$$

[7] El superíndice k en ξ_i^k denota el error que proporciona la función f_k en el input x_i.

$$\sum_{m\neq k}\sum_{i=1}^{n_k}\alpha_i(k,m)=\sum_{m\neq k}\sum_{j=1}^{n_m}\alpha_j(m,k),$$
$$k=1,\cdots,\ell.$$

Así, se tiene que para $\ell>2$ se han de estimar simultáneamente $n\,(\ell-1)$ parámetros $\alpha_i(k,m)$, con $i=1,\cdots,n_k$, $m\neq k$, $k=1,\cdots,\ell$, donde $n=\sum_{k=1}^{\ell}n_k$.

Como en el caso del biclasificador SVM, para construir la máquina de vectores soporte no lineal basta con sustituir el producto escalar $x_i^r\cdot x_j^s$ por una función núcleo $k(x_i^r,x_j^s)$ en las ecuaciones correspondientes.

Otra aproximación a estos problemas de multiclasificación, aparece en [WW98], donde según los autores, es una aproximación más natural que las dadas por los esquemas de descomposición y reconstrucción, seguidos con los biclasificadores, ya que se considera todas las clases a la vez[(8)]. Esta aproximación sigue un camino muy similar al dado en [Vap98] donde el problema de optimización consiste en minimizar la función

$$\frac{1}{2}\sum_{k=1}^{\ell}\left\|w^k\right\|^2+C\cdot\sum_{i=1}^{n}\sum_{m\neq k}\xi_i^m$$

sujeto a las siguientes restricciones:

$$w^k\cdot x_i+b_k\geq w^m\cdot x_i+b_m+2-\xi_i^m,$$
$$\xi_i^m\geq 0,\quad i=1,\cdots,n,\quad m\in\{1,2,\cdots,\ell\}\setminus k.$$

La función de decisión viene dada por:

$$f(x)=\arg\max_k\left(w^k\cdot x_i+b_k\right),\qquad k=1,\cdots,\ell.$$

La resolución del problema de optimización proporciona la siguiente solución:

$$f(x)=\arg\max_k\left(\sum_{i:\,y_i=k}A_i\,(x_i\cdot x)-\sum_{i:\,y_i\neq k}\alpha_i^n\,(x_i\cdot x)+b_k\right),$$

donde

$$A_i=\sum_{k=1}^{\ell}\alpha_i^k,\quad c_i^k=\begin{cases}1 & \text{si } y_i=k\\ 0 & \text{si } y_i\neq k\end{cases},\quad \sum_{i=1}^{n}\alpha_i^k=\sum_{i=1}^{n}c_i^k\,A_i,\quad k=1,\cdots,\ell$$

[(8)]Desde nuestro punto de vista, puede ser más natural, pero menos operativa.

y

$$0 \leq \alpha_i^k \leq C, \quad \alpha_i^k = 0, \quad i = 1, \cdots, n, \quad m \in \{1, 2, \cdots, \ell\} \setminus k.$$

Claramente, como en todas las máquinas de vectores soporte, si se reemplaza el producto escalar $(x_i \cdot x_j)$ por una función núcleo $k(x_i, x_j)$, se tiene una máquina no lineal.

Como se apunta y comprueba empíricamente en [WW98], esta máquina de multiclasificación, tiene resultados similares, en términos de porcentaje de errores, a las máquinas (1-v-1) y (1-v-r).

Nota 5.3.1 *Para las máquinas de vectores soporte biclasificadoras, la esperanza de la probabilidad de cometer un error sobre un conjunto de test esta acotada por la razón entre el número de vectores de entrenamiento que son vectores soporte y el número de vectores en el conjunto de entrenamiento [Vap95]:*

$$E[P(error)] = \frac{E[\textit{número de vectores de entrenamientos que son vectores soporte }]}{\textit{número de vectores de entrenamiento} - 1}.$$

Esta cota se cumple en el caso de multiclasificación por el esquema de votos (1-v-r) y (1-v-1) y para los métodos de vectores soporte multiclasificación con una única función de decisión. ▲

Las características de estos dos multiclasificadores, que incorporan todas las clases a la vez, son las siguientes:

c1.- Se necesita estimar una única funciones multiclasificadora, pero sobre un problema de optimización que resulta mucho más complejo que en los problemas de biclasificación.

c2.- La salida que resulta de la máquina no necesita ser interpretada en términos de un esquema de votación, y por tanto, el investigador no tiene que establecer un método de votación a priori, ya que va recogido dentro de la configuración de la máquina.

Los dos principales inconvenientes que presentan estos multiclasificadores son:

i1.- Pensamos que el mayor inconveniente que presentan estas configuraciones de "todas las clases a la vez", es él de ser una caja negra, en el sentido de no poder, evaluar a través de una salida intermedia, cómo se ha llegado a la salida última y medir la bondad de la misma.

i2.- No es posible asignar[(9)] un nivel de confianza a la salida global proporcionada por la máquina.

Veamos más detenidamente el inconveniente dado anteriormente en primer lugar. Si se nos presenta un nuevo input x cuyo etiquetado, en caso de realizarse mal, pueda traer desagradables consecuencias, lo correcto sería estudiar en profundidad cómo el multiclasificador ha llegado al etiquetado final, con objeto de corregir o tener en cuenta donde se ha producido el error. Esta posibilidad de trabajo es posible realizarla con las máquinas SV de multiclasificación siguiendo un esquema 1-v-1, o un esquema 1-v-r, sin más que exigir a la máquina que nos presente los resultados intermedios[(10)].

Por todo ello, consideramos más adecuado trabajar con máquinas multiclasificadoras que siguen un esquema de descomposición y reconstrucción, puesto que con éstas, podemos obtener como salidas los resultados de todas y cada una de las máquinas implementadas, y de esta forma disponer de un conjunto de resultados que nos permita tener una mayor capacidad de evaluación de la funcionalidad global.

Por otro lado, como empíricamente se ha demostrado que las máquinas 1-v-1 proporcionan mejores resultados que las máquinas 1-v-r, nosotros optamos por aquellas. Sin embargo, hemos apuntado anteriormente una serie de inconvenientes

[(9)] Siendo más preciso, aún no se ha encontrado una forma de asignar confianza a sus predicciones.

[(10)] En Econometría, cuando se plantea un problema de clasificación, por ejemplo a través de un modelo logit, no solo se presentan los resultados de la clasificación, sino que también aparecen indicadores (en términos de valores de algún estadístico) que permiten dar una mayor fiabilidad en la toma de decisión. Se puede ver diferentes ejemplos muy instructivos de este tema en [CSS01].

que sería adecuado paliar en lo posible, modificando la configuración inicial de estas máquinas, antes de su utilización.

5.4 Máquinas ℓ-SVCR para multiclasificación

En [Ang01] se introduce un nuevo tipo de máquina de vectores soporte para la multiclasificación, que el autor denomina ℓ-SVCR, con objeto de evitar el principal inconveniente que presentan las máquinas 1-v-1, esto es, la no inclusión de todos los vectores de entrenamiento en la configuración del problema de multiclasificación, pero manteniendo todas las ventajas de este tipo de esquema.

Con objeto de dar una mayor claridad a los posteriores desarrollos, supongamos que queremos buscar una función que clasifique los vectores inputs correspondiente a la clase θ_1 de los de la clase θ_2[(11)]. Realizamos una ordenación de los vectores de entrenamiento de tal forma que los n_1 primeros pertenezcan a la clase θ_1, los n_2 siguientes a la clase θ_2 y los restantes ($n_3 = n - n_1 - n_2$) pertenecen al resto de las clases, $\{\theta_3, \cdots, \theta_\ell\}$.

Como en el problema clásico de las SVMs buscamos, inicialmente, un hiperplano $f_{12}(x) = 0$ que separe adecuadamente las clases θ_1 y θ_2, pero ahora imponemos, además, que se tenga en cuenta el resto de las clases, en la construcción del problema de optimización. De esta forma, al hiperplano $f_{12}(x)$ se le exige que deje los vectores inputs de la clase θ_1 en la región $\left\{x \in \mathbb{R}^d, \text{ tal que } f_{12}(x) \geq 1\right\}$, a los vectores inputs de la clase θ_2 en la región $\left\{x \in \mathbb{R}^d, \text{ tal que } f_{12}(x) \leq -1\right\}$ y para los vectores inputs restantes se le asignan una región dependiente de un parámetro $0 \leq \delta < 1$ de tal forma que todos ellos caigan en la región $\left\{x \in \mathbb{R}^d, \text{ tal que } |f_{12}(x)| \leq \delta\right\}$, es decir, a diferencia de las máquinas 1-v-1, con δ se habilita una región de holgura donde incluir todos los restantes vectores de entrenamiento.

Si dicha solución es posible (caso separable) a partir de un hiperplano de la forma

[(11)] La generalización a otras etiquetas es clara.

$f_{12}(x) = \langle w, x\rangle + b$, entonces se podrá resolver el siguiente problema ℓ-SVCR sin pérdidas:

$$\min_{w\in\mathbb{R}^d} \frac{1}{2}\|w\|^2 \tag{5.2}$$

sujeto a

$$y_i\left(x_i \cdot w + b\right) - 1 \geq 0, \qquad \forall i = 1, 2, \cdots, n_1 + n_2, \tag{5.3}$$

$$-\delta \leq \langle w, x\rangle + b \leq \delta \qquad \forall i = n_1 + n_2 + 1, \cdots, n, \tag{5.4}$$

con $0 \leq \delta < 1$, y la clase que proporciona como salida la máquina ante un nuevo vector input x se interpreta de la siguiente forma:

$$\Theta(f_{12}(x)) = \begin{cases} \theta_1 & \text{si } f_{12}(x) > \delta \\ \theta_2 & \text{si } f_{12}(x) < -\delta \\ \theta_0 & \text{si } |f_{12}(x)| < \delta \end{cases} \tag{5.5}$$

Si no existe solución al problema de optimización anterior (caso no separable), entonces se relaja las restricciones (5.3) y (5.4) utilizando variables holguras y se busca, igual que en el caso anterior, un hiperplano de la forma $f_{12}(x) = \langle w, x\rangle + b$, que resuelva el siguiente problema ℓ-SVCR:

$$\min_{w\in\mathbb{R}^d} \frac{1}{2}\|w\|^2 + C_1 \cdot \sum_{i=1}^{n_1+n_2} \xi_i + C_2 \cdot \sum_{i=n_1+n_2+1}^{n} (\varphi_i + \varphi_i^*) \tag{5.6}$$

sujeto a

$$y_i\left(x_i \cdot w + b\right) \geq 1 - \xi_i, \qquad \forall i = 1, 2, \cdots, n_1 + n_2 \tag{5.7}$$

$$-\delta - \varphi_i^* \leq \langle w, x\rangle + b \leq \delta + \varphi_i \qquad \forall i = n_1 + n_2 + 1, \cdots, n \tag{5.8}$$

$$\begin{aligned} \xi_i &\geq 0 \qquad \forall i = 1, 2, \cdots, n_1 + n_2 \\ \varphi_i^*, \varphi_i &\geq 0 \qquad \forall i = n_1 + n_2 + 1, \cdots, n \end{aligned} \tag{5.9}$$

La solución a este problema aparece resuelta en [Ang01] y esta dada en la forma:

$$f_{12}(x) = \sum_{i=1}^{N_{SV}} \alpha_i \langle x_i, x\rangle + b$$

donde los α_i son los multiplicadores del Lagrange asociados al problema (5.6) cumpliendo

$$\sum_{i=1}^{N_{SV}} \alpha_i = 0$$

y b se obtiene a partir de las igualdades proporcionadas en las restricciones por los vectores soporte.

La solución aportada por la máquina ℓ-SVCR puede ser generalizada al caso no lineal utilizando funciones núcleos, obteniéndose de esta forma como solución general a un problema ℓ-SVCR:

$$f_{12}(x) = \sum_{i=1}^{N_{SV}} \alpha_i \, k(x_i, x) + b \tag{5.10}$$

5.4.1 Parámetros en la máquina ℓ-SVCR

Dentro del problema de optimización de la máquina ℓ-SVCR aparecen los siguientes parámetros:

- k Función núcleo.
- C_1 Ponderación dada a la suma de los errores de las dos clases que se discriminan.
- C_2 Ponderación dada a la suma de los errores de las restantes clases.
- δ Factor de insensibilidad.

Respecto a la función núcleo se dedica, dentro de este trabajo, todo un tema donde se estudiará con detalle. Sin embargo, es conveniente destacar dentro de este apartado, la gran importancia que tiene, ya que debe estar definida en un espacio característico con una alta dimensión, con objeto de que la máquina presente un buen funcionamiento, puesto que hemos de obligar a que todos los vectores inputs con etiqueta θ_k, con $k = 3, \cdots, \ell$ estén dentro de una región relativamente "pequeña". En [Ang01, pag. 95-102] se lleva a cabo un estudio empírico sobre este tema.

Para los parámetros C_1 y C_2 (que relacionan el intercambio entre ajuste y suavidad de la solución), al igual que en el caso general, no hay ninguna regla adecuada para asignarles unos valores concretos, salvo el método de validación cruzada, con el coste en términos de datos que esto supone[(12)]. En la siguiente sección, veremos un criterio que puede sernos útil para establecer, al menos, una relación entre estos dos parámetros de manera subjetiva, sin sacrificar dato alguno.

El parámetro δ, debe estar entre 0 y 1 con objeto de no solapar las regiones de decisión. Como se indica en [Ang01], cuanto menor sea δ, menor es la capacidad de generalización[(13)] de la máquina para los patrones a los cuales se etiqueta por θ_0 y mayor es el número de vectores soporte necesarios en la construcción de la solución. La idea que fundamenta la adopción de este parámetro esta relacionada con la función de ε-insensibilidad, desarrollada por Vapnik, que aparece en el tratamiento de los problemas de regresión resueltos a partir de las máquinas de soporte vectorial[(14)].

Con objeto de ver como queda gráficamente la ejecución de este tipo de clasificador consideramos el siguiente ejemplo.

Ejemplo 5.1 *Sea un conjunto formado por* 250 *vectores bidimensionales de entrenamiento generados a través de un modelo aleatorio basado en la distribución normal y asignamos etiquetas, siguiendo el siguiente cuadro:*

Nº datos	x_{1i}	x_{2i}	*Etiqueta*
50	$N(-5,1)$	$N(0,1)$	-1
50	$N(5,1)$	$N(0,1)$	1
50	$N(5,1)$	$N(5,1)$	-1
50	$N(0,1)$	$N(5,1)$	0
50	$N(0,1)$	$N(-5,1)$	0

[(12)] Debido a que el objetivo final de estos estudios por nuestra parte, es la implementación a datos de tipo económicos, destacamos este punto ya que en general, el número de datos suele ser limitado y costoso.

[(13)] Se sigue de la aclaración dada anteriormente con las funciones núcleos.

[(14)] Ver en [Gon00].

donde x_{1i} *representa la primera componente y* x_{2i} *la segunda componente de los vectores de entrenamiento* x_i. $N(\mu, 1)$ *denota que los datos han sido obtenidos a partir de un modelo normal de media* μ *y varianza* 1.

Aplicando una máquina 3*-SVCR a estos datos con función núcleo gaussiana con* $\sigma = 2$, *con constantes de ajuste* $C_1 = C_2 = 10$ *y factor de insensibilidad* $\delta = 0'5$, *se obtiene la figura 5.1. En esta gráfica podemos observar como se ha cometido un*

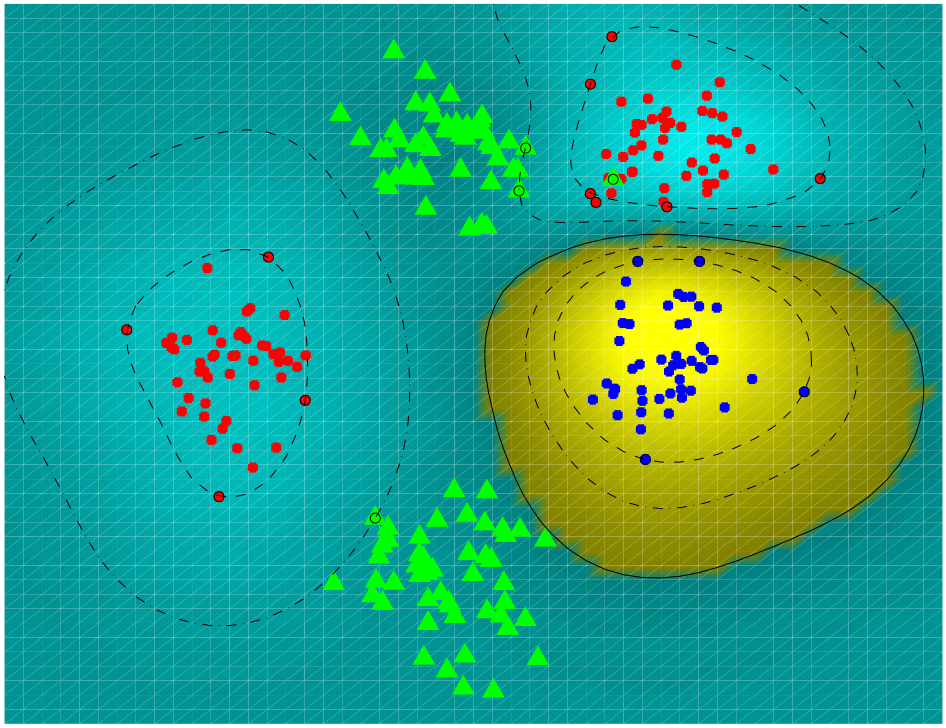

Figura 5.1: Máquina 3-SVCR con núcleo gaussiano de parámetro $\sigma = 2$, con constantes de ajuste $C_1 = C_2 = 10$ y insensibilidad $\delta = 0'5$. Los puntos rojos representan los vectores inputs de etiqueta -1, los azules representan los vectores de etiqueta 1 y los triángulos verdes representan los vectores de etiqueta 0. Las curvas en trazos discontinuos representan las funciones $f(x) = \pm 1$ y $f(x) = \pm\delta$.

único error (existe un vector de etiqueta 0 *en la esquina superior derecha dentro de una región con etiqueta* -1*) y se tiene solo,* 18 *vectores soporte lo que supone sólo*

el 7'2% del total de vectores de entrenamiento. ▲

5.4.2 Probabilidades en las máquinas ℓ-SVCR

En el capítulo dedicado a la interpretación probabilística de las SVMs se introdujo un problema de maximizar una función de verosimilitud que tenía la misma forma que el problema de minimización que se plantea en las máquinas de soporte vectorial para la clasificación dicotómica. En este apartado vamos a generalizar esta aproximación para las máquinas ℓ-SVCR.

Como ya sabemos, el problema general de este tipo de máquina es el siguiente:

$$\min_{w\in\mathbb{R}^d} \frac{1}{2}\|w\|^2 + C_1 \cdot \sum_{i=1}^{n_1+n_2} \xi_i + C_2 \cdot \sum_{i=n_1+n_2+1}^{n} (\varphi_i + \varphi_i^*) \tag{5.11}$$

sujeto a

$$y_i\,(x_i \cdot w + b) \geq 1 - \xi_i, \qquad \forall i = 1, 2, \cdots, n_1 + n_2 \tag{5.12}$$

$$-\delta - \varphi_i^* \leq \langle w, x\rangle + b \leq \delta + \varphi_i \qquad \forall i = n_1 + n_2 + 1, \cdots, n \tag{5.13}$$

$$\begin{aligned} \xi_i &\geq 0 \qquad \forall i = 1, 2, \cdots, n_1 + n_2 \\ \varphi_i^*, \varphi_i &\geq 0 \qquad \forall i = n_1 + n_2 + 1, \cdots, n \end{aligned} \tag{5.14}$$

Sea $\theta(x) = w \cdot x + b$ una posible solución de la máquina, dependiendo de los parámetros w y b, con $w \in \mathbb{R}^d$ y $b \in \mathbb{R}$. Se sigue que:

- Si el vector x_i tiene etiqueta θ_1, entonces dentro de la clasificación llevada a cabo por la máquina ℓ-SVCR la salida correcta sería $\theta(x_i) \geq 1$ ya que en este caso la etiqueta asignada en el planteamiento del problema es $y_i = 1$, que se corresponde con θ_1. En caso contrario, se sigue de (5.12) que la pérdida que ocasiona dentro de la función objetivo es $\xi_i = 1 - \theta(x_i) \geq 0$.

- Si el vector x_i tiene etiqueta θ_2, entonces dentro de la clasificación llevada a cabo por la máquina ℓ-SVCR la salida correcta sería $\theta(x_i) \leq -1$ ya que en

este caso la etiqueta asignada en el planteamiento del problema es $y_i = -1$, que se corresponde con θ_2. En caso contrario, la pérdida que ocasiona dentro de la función objetivo es $\xi_i = 1 + \theta(x_i)$.

- Si el vector x_i tiene etiqueta θ_k con $k \neq 1, 2$, entonces dentro de la clasificación llevada a cabo por la máquina ℓ-SVCR, la salida correcta sería $|\theta(x_i)| \leq \delta$ ya que en este caso la etiqueta asignada en el planteamiento del problema es $y_i = 0$ que se corresponde con θ_0. En caso contrario, la pérdida que ocasiona es: $\varphi_i^* = -\theta(x_i) - \delta$ si $\theta(x_i) < -\delta$ y $\varphi_i = \theta(x_i) - \delta$ si $\theta(x_i) > \delta$.

Si consideramos la función bisagra, dada en el capítulo anterior, $l(z) = z\,H(z)$, entonces asignamos a las salidas $y = 1$ e $y = -1$ de la máquina ℓ-SVCR las siguientes[15] "probabilidades", en función de un nuevo input x, y de los parámetros w y b:

$$Q[y = 1|\theta(x)] = \kappa(C_1, C_2)\, exp\left[-C_1 \cdot l(\theta(x))\right],$$
$$Q[y = -1|\theta(x)] = \kappa(C_1, C_2)\, exp\left[-C_1 \cdot l(-\theta(x))\right],$$

con $\kappa(C_1, C_2)$ a determinar, igual que se hacia cuando se planteaba la máquina de soporte vectorial con dos etiquetas posibles.

Si consideramos la función de insensibilidad[16] δ:

$$|z|_\delta = \begin{cases} -z - \delta & \text{si} \quad z < -\delta \\ 0 & \text{si} \quad -\delta \leq z \leq \delta \\ z - \delta & \text{si} \quad \delta < z \end{cases}$$

entonces asignamos a la salida $y = 0$ de la máquina ℓ-SVCR, la siguiente "probabilidad", en función de un nuevo input x, y de los parámetros w y b:

$$Q[y = 0|\theta(x)] = \kappa(C_1, C_2)\, exp\left[-C_2 \cdot |\theta(x)|_\delta\right].$$

[15] Igual que en el capítulo anterior, no es realmente una probabilidad pero nos sirve de base para construir una.

[16] Función que fue introducida por primera vez por V. N. Vapnik. Su desarrollo tiene como origen, el de proporcionar una función mezcla entre la función de pérdida robusta de Hubber y una que permitiese una reducción del número de vectores soporte.

Del desarrollo de la interpretación probabilística para el caso de dos etiquetas se sigue que es posible llevar a cabo la generalización para un número finito de etiquetas y en particular cuando, como en este caso, se tiene tres.

Para conseguir que ciertamente sean probabilidades basta considerar que $\kappa(C_1, C_2)$ es igual al recíproco de

$$v(\theta(x)) = \sum_{y \in \{-1,0,1\}} Q[y|\theta(x)]$$

y se tiene que:

$$\begin{aligned} P[y=1|\theta(x)] &= exp\left[-C_1 \cdot l(\theta(x))\right]/v(\theta(x)), \\ P[y=-1|\theta(x)] &= exp\left[-C_1 \cdot l(-\theta(x))\right]/v(\theta(x)), \\ P[y=0|\theta(x)] &= exp\left[-C_2 \cdot |\theta(x)|_\delta\right]/v(\theta(x)). \end{aligned} \tag{5.15}$$

La figura 5.2 nos proporciona una gráfica donde se puede ver como quedan las

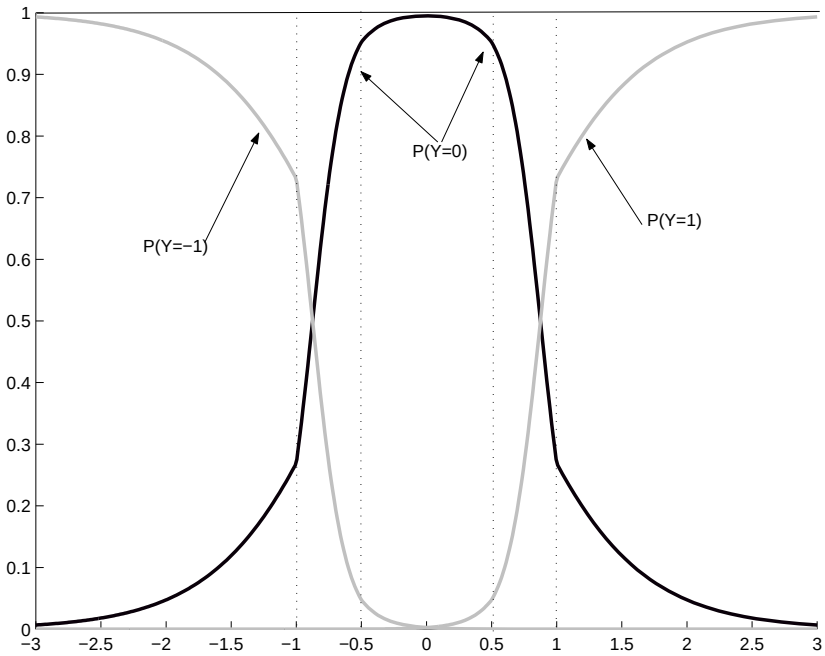

Figura 5.2: Funciones de probabilidad para $\delta = 0'5$, $C_1 = 6$ y $C_2 = 2$.

probabilidades dependiendo del valor que tome la función $\theta(x)$ dentro del intervalo $[-3, 3]$.

Nótese cómo se siguen los resultados que intuitivamente se esperan tenga la máquina, ya que:

- Si $\theta(x) < -1$, la probabilidad de asignar la etiqueta $y = -1$ es mayor que las otras dos probabilidades, y es tanto mayor cuanto menor es $\theta(x)$.
- Si $\theta(x) > 1$, la probabilidad de asignar la etiqueta $y = 1$ es mayor que las otras dos probabilidades, y es tanto mayor cuanto mayor es $\theta(x)$.
- Si $-\delta < \theta(x) < \delta$, la probabilidad de asignar la etiqueta $y = 0$ es mayor que las otras dos probabilidades, y es tanto mayor cuanto más próximo se encuentre de 0.

Por otro lado, esta construcción de las probabilidades de asignación de etiquetas, nos da un criterio subjetivo para elegir los valores de los parámetros C_1 y C_2. Veamos como se podría, a priori, elegir los parámetros C_1 y C_2 a la vista de las gráficas que aparecen recogidas en la figura 5.3.

$C_1 = C_2 = 1$ No sería una elección adecuada ya que se establece una probabilidad para la etiqueta $y = 0$ muy por debajo de las probabilidades asignadas a las etiquetas $y = 1$ e $y = -1$.

$C_1 = 1$, $C_2 = 10$ Se tendría una situación parecida a la anterior, pero además en las cercanías de $|\theta(x)| = 1$ presenta un resultado patológico, que no sería explicable en ningún sentido.

$C_1 = 3$, $C_2 = 7$ Se tendría una situación parecida a dada para $C_1 = C_2 = 1$.

$C_1 = 10$, $C_2 = 1$ Se tendría una situación parecida a dada para $C_1 = C_2 = 1$ pero cambiando los papeles de las etiquetas.

$C_1 = 5$, $C_2 = 5$ Sería una situación adecuada ya que se establece unos niveles de probabilidades similares entre las etiquetas.

$C_1 = 7$, $C_2 = 3$ Sería análoga a la situación anterior, pero dando un crecimiento mayor a las probabilidades cuando nos acercamos a cero.

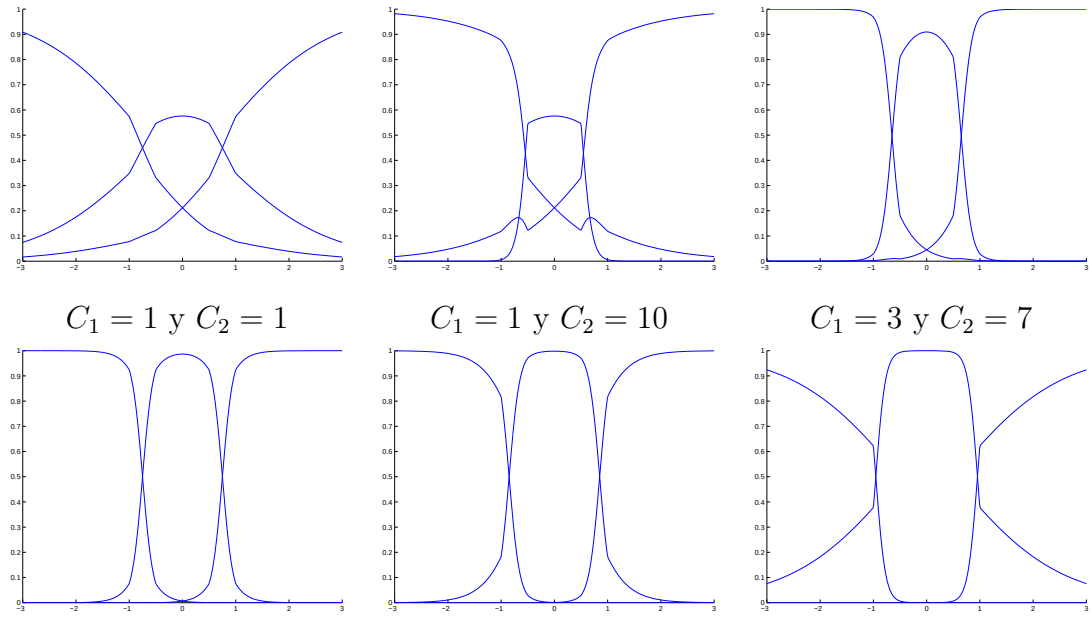

Figura 5.3: Distintos ejemplos de funciones de probabilidad dependientes de C_1 y C_2 para $\delta = 0'5$ y representadas todas ellas en el intervalo $[-3, 3]$.

De estos gráficos, se sigue que el valor asignado al parámetro C_1 debe ser mayor o igual que el asignado a C_2, o en otras palabras, dentro de la función objetivo se ha de dar más peso a los errores cometidos por las clases θ_1 y θ_2 que a los errores en las otras clases, resultado que parece totalmente acorde con lo que la intuición nos indica.

Debemos notar, que este estudio es posible llevarlo a cabo antes de entrenar la máquina puesto que las probabilidades dependen del valor que tome la función $\theta(x)$ y no de cual sea la solución última, proporcionada por la máquina.

5.4.3 Esquema de reconstrucción

Si tenemos en cuenta las probabilidades dadas en (5.15), podemos establecer una función que interprete la solución dada por la máquina ℓ-SVCR diferente a la dada

en (5.5), de la siguiente forma:

$$\Theta(f_{12}(x)) = \begin{cases} \theta_1 & \text{si } P[Y=1] > \text{máx}\{P[Y=0], P[Y=-1]\} \\ \theta_0 & \text{si } P[Y=0] \geq \text{máx}\{P[Y=-1], P[Y=1]\} \\ \theta_2 & \text{si } P[Y=-1] > \text{máx}\{P[Y=0], P[Y=1]\}. \end{cases} \tag{5.16}$$

De esta forma, este nuevo interprete es más reacio a etiquetar θ_1 y θ_2 que la interpretación inicial dada por el profesor Angulo en su tesis doctoral. Pensamos que esta elección mejora la anterior ya que si posteriormente se le da tanta importancia a los votos, con esta interpretación se hace más caro la obtención de ellos.

A continuación, podemos seguir un esquema de votación por mayoría, como en el caso 1-v-1 SVM. El difícil problema de resolver empates entre etiquetas, nosotros lo resolvemos asignando a cada etiqueta un valor promedio de los grados de confianza conseguido con las funciones que lo votan. De esta forma en caso de empate asignamos como salida, aquella etiqueta que tenga una mayor grado de confianza en promedio.

Por ello, en la máquina por nosotros propuesta consideramos más adecuado proporcionar tantas salidas como máquinas ℓ-SVCRs parciales se hayan implementado, donde en cada salida se indique:

- La etiqueta predicha por la máquina.
- Grado de confianza depositado en la salida.

De esta forma, se esta proporcionando al investigador un conjunto de información que le permite tomar una decisión mucho más precisa que si únicamente se le proporciona la salida final de una máquina.

Nota 5.4.1 *Además de las salidas de la máquina, es importante indicar, que cuando se realiza cualquier tipo de análisis discriminante hay que tener en cuenta el tamaño relativo de cada clase, pues si los tamaños son muy diferentes, estos puede acarrear que las interpretaciones de las salidas no sean correctas.* ▲

Aclaremos lo anterior a partir de un ejemplo:

Ejemplo 5.2 *Si tenemos un problema con* 4 *clases distintas en las cuales hemos aplicado el clasificador* 4*-SVCR y se ha obtenido la siguiente salida asociada a un vector input* x*:*

f_{kh}	*1-2*	*1-3*	*1-4*	*2-3*	*2-4*	*3-4*
Etiqueta	θ_1	θ_0	θ_4	θ_0	θ_4	θ_0
Nivel de confianza	*65%*	*80%*	*70%*	*80%*	*80%*	*63%*

En este caso, no se produce ningún empate y la salida global de la máquina para un input x *sería asignar la etiqueta* θ_4 *con un grado de confianza del 75% que es el valor medio de los porcentajes dado por la máquina* f_{14} *(70%) y la máquina* f_{24} *(80%).*

Además, el investigador ha podido observar como el clasificador f_{12} *ha asignado erróneamente la etiqueta* θ_1*, pero también se equivoca el clasificador* f_{34}*, por lo que ha de ser considerado en un estudio a posteriori.*

Si la salida del multiclasificador fuese la siguiente:

f_{kh}	*1-2*	*1-3*	*1-4*	*2-3*	*2-4*	*3-4*
Etiqueta	θ_1	θ_1	θ_4	θ_0	θ_4	θ_0
Nivel de confianza	*65%*	*80%*	*70%*	*80%*	*80%*	*63%*

al igual que antes, la salida global de la máquina para x *sería asignar la etiqueta* θ_4 *con un grado de confianza del 75%, pero se habría producido un empate a dos votos, con la etiqueta* θ_1 *y en el desempate se habría seleccionado* θ_4 *ya que el grado de confianza es mayor. Pero aún más, de las salidas intermedias se tiene que en el emparejamiento de ambas etiquetas, la etiqueta* θ_4 *es la "ganadora", lo cual debe tener un peso alto en la interpretación final dada por el investigador*[(10)]*.* ▲

Con este último ejemplo, hemos intentado dar una visión general de como se trabajaría con este tipo de clasificadores. Por supuesto, habría que valorar todas las posibles combinaciones de etiquetas, por ejemplo ¿qué interpretación haría un investigador si todas los clasificadores etiquetasen θ_0? ¿cómo se interpreta la situación, de un clasificador donde hay dos etiquetas que empatan a votos, presentan el mismo grado de confianza y en el enfrentamiento entre ellas, el clasificado proporciona la etiqueta θ_0? Pensamos que en estos casos la única solución es aquella que resulta del estudio que un experto realice de toda la información proporcionada por la salida de la máquina.

5.4.4 Relación entre los parámetros

Como ya hemos estudiado, en este tipo de máquinas se tienen tres parámetros C_1, C_2 y δ, que deben ser elegidos antes de construir la máquina de vectores soporte para la multiclasificación. Sería muy útil disponer de algún tipo de relación previa que nos facilitara la elección de cada uno de estos parámetros. Esto es posible, en nuestra máquina, si tenemos en cuenta la función interprete dada en (5.16) y se escriben las regiones de definición de forma diferente. Veámoslo:

Por la simetría de las regiones basta con evaluar la frontera en una de ellas ya que la frontera de la otra será su valor opuesto. Consideramos la primera desigualdad:

$$P[Y = 1] > \text{máx}\,\{P[Y = 0], P[Y = -1]\}$$

que por construcción de las probabilidades coincide con

$$P[Y = 1] > P[Y = 0]$$

Por tanto para evaluar la frontera hemos de calcular el valor $\delta^* = \theta^*(x)$ tal que se verifique la ecuación: $P[Y = 1/\theta^*(x)] = P[Y = 0/\theta^*(x)]$. De las definiciones de las probabilidades tenemos garantizada la existencia y unicidad de la solución y,

además ésta debe encontrarse en el intervalo $[\delta, 1]$. Así pues, la ecuación queda:

$$\begin{aligned}
\frac{exp\left[-C_1 \cdot l(\theta^*(x))\right]}{v(\theta^*(x))} &= \frac{exp\left[-C_2 \cdot |\theta^*(x)|_\delta\right]}{v(\theta^*(x))} \\
exp\left[-C_1 \cdot l(\theta^*(x))\right] &= exp\left[-C_2 \cdot |\theta^*(x)|_\delta\right] \\
C_1 \cdot l(\theta^*(x)) &= C_2 \cdot |\theta^*(x)|_\delta \\
C_1 \cdot (1 - \theta^*(x)) &= C_2 \cdot (\theta^*(x) - \delta)
\end{aligned}$$

y la solución es:

$$\theta^*(x) = \delta^* = \frac{C_1 + C_2 \cdot \delta}{C_1 + C_2}$$

Es decir, la función intérprete quedaría explícitamente como sigue:

$$\Theta(f_{12}(x)) = \begin{cases} \theta_1 & \text{si } \theta(x) > \delta^* \\ \theta_0 & \text{si } |\theta(x)| \leq \delta^* \\ \theta_2 & \text{si } \theta(x) < -\delta^* \end{cases} \tag{5.17}$$

que coincide con el interprete dado en (5.5) pero con un valor δ^* en cuya definición aparece de forma explícita los parámetros C_1, C_2 y δ.

Por ejemplo, en el análisis discriminante de Fisher, la frontera se obtiene como una media aritmética ponderada de los valores teóricos de cada uno de los centroides. Pues bien, en nuestro caso, la frontera se obtiene de manera análoga, como una media aritmética ponderada de las fronteras impuestas en las restricciones del problema de optimización (δ y 1), donde las ponderaciones son las constantes C_1 y C_2 (los pesos asignados a cada tipo de error).

A partir de esta expresión de la frontera podemos estudiar su variación con respecto a los parámetros.

- Si C_2 y δ están fijados. Cuanto mayor sea C_1, más importancia se les asignan a los errores que se cometen con las etiquetas θ_1 y θ_2. Nuestra máquina, con objeto de no etiquetar defectuosamente, acerca la frontera hacia 1, es decir, se vuelve más prudente en la asignación del etiquetado aumentando la región de etiquetado $\theta_0 = \emptyset$.

Por el contrario, cuanto menor sea C_1, menos importancia se les asignan a los errores que se cometen con las etiquetas θ_1 y θ_2. Nuestra máquina, se vuelve menos prudente en la asignación del etiquetado disminuyendo la región de etiquetado $\theta_0 = \emptyset$, acercando la frontera hacía δ.

- Si C_1 y δ están fijados. Cuanto mayor sea C_2, más importancia se les asignan a los errores que se cometen con la etiqueta θ_0, y nuestra máquina, se vuelve más prudente en la asignación del etiquetado θ_0 disminuyendo la región de etiquetado $\theta_0 = \emptyset$, acercando la frontera hacía δ, con lo cual potenciamos la asignación del etiquetado θ_1 y θ_2, es decir, tiene el efecto contrario al de C_1.

  Por el contrario, cuanto menor sea C_2, menos importancia se les asignan a los errores que se cometen con la etiqueta θ_0, y nuestra máquina, se vuelve menos prudente en la asignación del etiquetado θ_0 aumentando la región de etiquetado $\theta_0 = \emptyset$, acercando la frontera hacía 1, con lo cual potenciamos la asignación del etiquetado θ_0.

- Si C_1 y C_2 están fijados. La interpretación de $0 \leq \delta < 1$ es exactamente la misma que la dada en la configuración inicial del problema. Cuanto mayor sea, más prudente se vuelve la máquina en el etiquetado de las clases θ_1 y θ_2; y al contrario, cuanto menor sea.

Por otro lado si deseamos estudiar como varía la frontera respecto de la variación conjunta de C_1 y C_2, entonces

$$\begin{aligned} \delta^* &= \frac{C_1 + C_2 \cdot \delta}{C_1 + C_2} \\ &= \frac{C_1 + C_2 + C_2 \cdot (\delta - 1)}{C_1 + C_2} \\ &= 1 + \frac{C_2 \cdot (\delta - 1)}{C_1 + C_2} \\ &= 1 - \frac{1 - \delta}{1 + C_1/C_2} \end{aligned}$$

y de aquí se sigue que:

- Si la razón C_1/C_2 crece, la frontera se acerca hacía 1.
- Si la razón C_1/C_2 decrece, la frontera se acerca hacía δ.
- Como caso particular, si $C_1 = C_2$, la frontera es el punto medio entre δ y 1.

5.5 Comentarios sobre el capítulo

> *En ciencia, la mayor parte del trabajo de investigación consiste en operaciones comparables a abrillantar, tapar un agujero o arreglar una gotera.*
>
> *–C. R. Rao*[17] *–*

En este capítulo hemos analizado las distintas aproximaciones de las máquinas de vectores soporte a los problemas de clasificación con más de dos etiquetas. De entre todas éstas se ha dedicado una especial atención a las máquinas 1-v-1 SV, estudiando sus propiedades más importantes y viendo las limitaciones que presentan con objeto de darle solución.

Con las máquinas ℓ-SVCR se soluciona una de estas limitaciones, a saber, la no inclusión de todos los vectores de entrenamiento en la configuración parcial de los problemas.

Otra limitación importante, el estudio del desempate entre etiquetas, se soluciona utilizando una aproximación probabilística de las SVMs, que permite la comparación entre las salidas numéricas de todas las máquinas parciales. Además de proporcionar un valor que mide la confianza depositada en las distintas máquinas de vectores soporte implementadas.

Una vez diseñada la máquina, el siguiente paso es su construcción y su puesta en funcionamiento. Los programas, escritos en MatLab, que construyen la máquina se encuentran recogidos en el apéndice B. Por otro lado, en el capítulo 8, se resuelve

[17] De su libro "Estadística y Verdad. Aprovechando el azar", [Rao84].

detalladamente dos problemas de clasificación con más de dos etiquetas, utilizando esta nueva máquina de vectores soporte para la multiclasificación.

CAPÍTULO 6

FUNCIONES NÚCLEOS Y SVM

Grandes, eternas e inmutables leyes determinan los caminos que todos recorremos sin rumbo fijo.

–Goethe–

Medir, medir, medir. Medir otra vez y otra vez para encontrar la diferencia y la diferencia de la diferencia.

–Galileo (1565-1642)–

El objetivo de este capítulo es generalizar los problemas de optimización de las máquinas de vectores soporte sobre clases de funciones no necesariamente lineales. Como ya se ha comentado anteriormente, para llevar a cabo esta generalización al caso no lineal, se ha de definir una función real de dos variables con unas características determinadas, a la que se denominará función **núcleo**. De entre todas, destacamos que la principal característica de este tipo de función es que debe venir expresada a través de un producto escalar de una transformación de los vectores

inputs de $\mathcal{X}$ en un espacio característico de dimensión superior, $\mathcal{H}$.

En la sección 6.2 se dará una condición necesaria y suficiente que debe verificar una función de dos variables para ser una función núcleo: la condición de Mercer. Una vez estudiada con detalle esta condición se analiza como se encuentra relacionada con unos espacios de Hilbert concretos, los espacios de Hilbert con núcleo reproductor –del inglés *Reproductor Kernel Hilbert Space*, abreviadamente R.K.H.S.– y se llega a la conclusión que los núcleos de Mercer son núcleos reproductores en el correspondiente espacio R.K.H.S.. Esta relación con los núcleos de Mercer permitirá tener una forma explícita de obtener la transformación de los vectores inputs utilizando la transformada de la función de Green de un determinado operador.

El estudio de la función de Green de un operador llevará a estudiar la relación entre los núcleos de Mercer y los sistemas de regularización, llegando a concluir que los problemas de optimización de vectores soporte son casos particulares de estos sistemas de regularización donde los operadores deben cumplir una determinada condición de consistencia.

Una vez vista la forma que deben tener las funciones núcleos se pasa a dar un breve repaso a los núcleos que más se utilizan en la práctica: núcleos de Gauss (o núcleos de base radial), núcleos basados en B-splines, núcleos polinomiales y núcleos basados en splines.

Se finaliza el capítulo dando un nuevo enfoque a las funciones núcleos como una función que nos posibilita medir similitudes dentro del conjunto de inputs. Además, y lo que resulta más interesante, abre la posibilidad de no considerar exclusivamente como espacio input, un espacio vectorial sino que se puede considerar cualquier conjunto no vacío. Se indica una función núcleo particular que nos permitirá aportar una aproximación muy curiosa a la medida de similitudes dentro de los sucesos de un σ-álgebra.

Al igual que en el capítulo inicial de este trabajo, todos los desarrollos sobre núcleos, son válidos para cualquier problema de máquinas de vectores soporte, tanto si se plantea un problema de clasificación, como si es un problema de regresión, de

estimación de densidades, de análisis de componentes principales, ...

6.1 Introducción

En las diferentes máquinas de vectores soporte, que hemos estudiado, se resolvía un problema de optimización cuadrática sujeto a un conjuntos de restricciones, todas lineales. Así, en el problema de máquina biclasificadora[(1)] estándar:

$$\begin{array}{l} \min\limits_{w\in\mathbb{R}^d} \dfrac{1}{2}\|w\|^2 + C\cdot\sum\limits_{i=1}^{n}\xi_i \\ s.a. \quad \begin{cases} y_i\cdot(x_i\cdot w+b)-1+\xi_i\geq 0,\ \forall i \\ \xi_i\geq 0,\ \forall i \end{cases} \end{array} \tag{6.1}$$

el objetivo es encontrar una función discriminadora (la solución) dentro del siguiente conjunto de funciones lineales (se busca un hiperplano separador en $\mathbb{R}^d$):

$$\mathcal{F} = \left\{ f : \mathcal{X}\subset\mathbb{R}^d \rightarrow \mathbb{R} \,/\, f(x) = \langle w, x\rangle + b,\ \text{donde } w\in\mathbb{R}^d,\, b\in\mathbb{R} \right\} \tag{6.2}$$

donde cada una de las funciones queda determinada a partir de dos parámetros $w \in \mathbb{R}^d$ y $b \in \mathbb{R}$. El parámetro w de la solución se expresa en la forma (ver ecuación (3.25), página 105):

$$w = \sum_{i=1}^{N_{SV}} \alpha_i\, y_i\, s_i = \sum_{i=1}^{n} \beta_i\, x_i$$

donde por N_{SV} se denota el número de vectores soporte y por s_i los vectores soporte del conjunto $\{x_1, \cdots, x_n\}$. Además, se tiene que los $\beta_i (= \alpha_i\, y_i)$ son nulos salvo en los vectores soporte y se cumple la condición:

$$\sum_{i=1}^{n} \beta_i \ = \ 0.$$

Una importante propiedad de la representación del vector solución $w = \sum_{i=1}^{n} \beta_i\, x_i$ es que los vectores inputs intervienen dentro de la solución solo a través de sus

[(1)] Como tendremos ocasión de comprobar, en los desarrollos, la generalización a cualquier otra máquina SV es similar.

productos escalares, $\langle x_i, x_j \rangle$ y nunca por sus atributos individuales, es decir, si se tiene dos vectores x_{i_1} y x_{i_2} distintos pero para todo x_j, $j = 1, \cdots, n$ se cumple $\langle x_{i_1}, x_j \rangle = \langle x_{i_2}, x_j \rangle$ entonces su contribución a la solución final es la misma.

Por otro lado, se tiene que el parámetro b se determina a partir de las condiciones de Karush-Kuhn-Tucker, resolviendo para cada vector soporte una ecuación lineal y tomando como valor b un valor promedio de todas esas soluciones.

De esta forma, la solución lineal al problema de optimización se expresa a través de una función, que utilizando la linealidad del producto escalar, se puede escribir como sigue:

$$f(x) = \langle \sum_{i=1}^{n} \beta_i \, x_i, x \rangle + b = \sum_{i=1}^{n} \beta_i \, \langle x_i, x \rangle + b.$$

Así, en la representación de la función de decisión (función discriminadora), solo se necesita el producto escalar de los vectores de entrenamiento $\{x_1, x_2, \cdots, x_n\}$ con el nuevo vector input $x \in \mathcal{X}$.

Como ya se indicó, lo más importante de esta expresión de la solución es que en ella intervienen los vectores inputs exclusivamente a través del producto escalar, lo cual permite utilizar una técnica introducida por primera vez en[(2)] [ABR64] que consigue generalizar el problema 6.1 a conjuntos de funciones no lineales.

Nota 6.1.1 *Es importante observar que muchos de los posteriores desarrollos de este capítulo no son exclusivos de las máquinas vectores soporte, ya que la condición que se exige es que los datos entren a formar parte de la solución por medio de un producto escalar. Esto significa que cualquier problema que cumpla esta condición es susceptible de ampliar el rango de la clase de funciones en la que se plantea el problema original.* ▲

Sea un conjunto de vectores de entrenamiento:

$$Z = \{(x_1, y_1), \cdots, (x_n, y_n)\} \subset \mathcal{X} \times \mathcal{Y}$$

[(2)] Aizermann y otros en 1964 introducen, por primera vez, una interpretación geométrica de los núcleos como productos escalares en un espacio característico.

siguiendo las ideas expuestas en el capítulo 3, lo práctico sería encontrar un hiperplano separador óptimo en un determinado espacio tal que en dicho espacio el conjunto de entrenamiento fuese separable, con lo que no se tendría ninguna pérdida en la función objetivo.

Por ello, desde el punto de vista del problema de optimización, sería muy conveniente dado un conjunto de vectores de entrenamiento Z, no necesariamente separable, realizar una transformación ϕ de los vectores inputs $X = \{x_1, \cdots, x_n\}$ tal que convierta el conjunto transformado,

$$Z_\phi = \{(\phi(x_1), y_1), \cdots, (\phi(x_n), y_n)\}, \tag{6.3}$$

en un conjunto separable en un espacio adecuado.

Aunque este problema no siempre tendrá solución[(3)], si permitirá ampliar el marco de trabajo al poder considerar otras clases de funciones clasificadoras distintas de las lineales, a la vez que permite retener la capacidad de generalización que proporcionan las funciones lineales en el espacio característico.

Además, de la imposibilidad de poder obtener siempre conjuntos separables, debemos indicar que se ha de tener en cuenta el problema de la generalización. La función solución no puede ser cualesquiera, hay que buscar una relación entre suavidad y ajuste que necesariamente lleva, en los problemas prácticos, a tener que plantear el problema con alguna pérdida.

Formalmente, el problema se plantea como sigue: dado el espacio de los vectores inputs $\mathcal{X}$ se considera una transformación ϕ de este espacio en un espacio vectorial, que se denotará por $\mathcal{H}$ y llamará espacio característico, en la forma:

$$\phi : \mathcal{X} \subset \mathbb{R}^d \rightarrow \mathcal{H} \subset \mathbb{R}^{d'} \tag{6.4}$$

donde normalmente la dimensión de $\mathcal{H}$ (d') es muy superior a la dimensión del espacio $\mathcal{X}$ ($d << d'$). A partir de esta transformación ϕ, en lugar de la clase de

[(3)] Por ejemplo, si se presenta un conjunto de entrenamiento donde haya dos vectores de la forma $(x, 1)$ y $(x, -1)$. Cualquier elección de ϕ necesariamente provocará alguna pérdida.

funciones dada en (6.2), se considera la clase

$$\mathcal{F}_\phi = \left\{ f : \mathcal{X} \subset \mathbb{R}^d \to \mathbb{R} \,/\, f(x) = \langle w, \phi(x) \rangle_{\mathcal{H}} + b, \text{ donde } w \in \mathbb{R}^{d'},\, b \in \mathbb{R} \right\} \quad (6.5)$$

donde $\langle \cdot, \cdot \rangle_{\mathcal{H}}$ denota un producto escalar[4] definido en $\mathcal{H}$.

Planteando el problema de optimización de la SVM a los vectores transformados (6.3), se obtiene un problema de optimización que proporciona, según la clase de funciones dada en (6.5), una solución lineal en el espacio característico, pero no necesariamente lineal en el espacio de los inputs.

Para una mayor simplicidad en los desarrollos y con intención de hacerlos más operativos, se utiliza la siguiente notación para el producto escalar en $\mathcal{H}$:

$$\langle \phi(x), \phi(x') \rangle = k(x, x')$$

y, si se reemplaza en todos los desarrollos del problema (6.1), $\langle x, x' \rangle$ por $k(x, x')$, se obtiene el siguiente problema de optimización convexa no lineal:

$$\begin{array}{l} \min\limits_{w \in \mathbb{R}^{d'}} \dfrac{1}{2} \|w\|^2 + C \cdot \sum\limits_{i=1}^{n} \xi_i \\ s.a. \quad \left\{ \begin{array}{rcl} y_i \cdot (\langle w, \phi(x_i) \rangle - b) - 1 + \xi_i & \geq & 0 \quad \forall\, i \\ \xi_i & \geq & 0 \quad \forall\, i \end{array} \right. \end{array} \quad (6.6)$$

cuya solución viene dada por[5]

$$w = \sum_{i=1}^{n} \beta_i \, \phi(x_i)$$

y proporciona la siguiente función en $\mathcal{F}_\phi$:

$$f(x) = \sum_{i=1}^{n} \beta_i \, k(x_i, x) + b. \quad (6.7)$$

sin más que cambiar el producto escalar $\langle x_i, x \rangle$ por $k(x_i, x)$ para $i = 1, \cdots, n$.

De esta forma se tiene una función $k(\cdot, \cdot)$ que juega un papel muy importante no solo en las máquinas de soporte vectorial, sino en toda la teoría del aprendizaje

[4] El subíndice no se indica si en la exposición queda claro el espacio de trabajo.

[5] Sin más que sustituir en el problema original los vectores x_i por sus transformados $\phi(x_i)$.

estadístico, y que es objeto de muchas investigaciones en la actualidad. Así, se da la siguiente definición:

Definición 6.1.2 (de núcleo de Mercer) *Una función núcleo*[6] *es una función real de dos variables, denotada por k, que verifica:*

$$\begin{aligned} k : \mathcal{X} \times \mathcal{X} &\rightarrow \mathbb{R} \\ (x, x') &\rightarrow k(x, x') = \langle \phi(x), \phi(x') \rangle_{\mathcal{H}} \end{aligned}$$

donde ϕ es una aplicación definida como en (6.4).

Nota 6.1.3 *De esta manera, la idea de trabajar con núcleos ofrece una solución para proyectar un conjunto de datos dentro de un espacio característico de dimensión alta que incrementa la capacidad de generalización de las máquinas lineales de aprendizaje. Por tanto en los problemas de clasificación aumenta la habilidad del modelo para discriminar adecuadamente las etiquetas.*

Por ello, y como desarrollaremos posteriormente, los núcleos generalizan la definición estándar de producto escalar en un espacio de inputs. De esta forma, los núcleos proporcionan un lenguaje descriptivo usado por la máquina para ver los datos. ▲

Por todo lo anterior, se sigue que la solución al problema de optimización (6.6) se expresa en términos de la función núcleo $k(\cdot, \cdot)$, sin tener en ningún momento que conocer la transformación ϕ, ya que ésta aparece implícitamente en (6.7) a través del producto escalar definido en $\mathcal{H}$. Por ello, si la función núcleo $k(x, x')$ es fácil de evaluar, parece razonable utilizar k sin tener que utilizar para nada la aplicación ϕ, ya que entre otras razones:

- suele ser de difícil tratar con ella; y

[6] La definición original de función núcleo proviene del estudio de los operadores integrales. Dentro de este trabajo, todas las funciones núcleos consideradas son núcleos de Mercer y las denotaremos simplemente por funciones núcleos. Hacemos esta distinción ya que existen diferentes tipo de funciones núcleos dentro de la literatura matemática, y no todos ellos son los mismos.

- no siempre es posible determinarla de forma explícita.

Un resumen de lo visto hasta ahora se puede observar en la figura 6.1. En esta figura, se observa que la idea inicial, en las máquinas de vectores soporte, es transformar los vectores inputs $X = \{x_1, \cdots, x_n\}$ en unos nuevos vectores inputs $\phi(X) = \{\phi(x_1), \cdots, \phi(x_n)\}$ dentro de un espacio característico $\mathcal{H}$ de dimensión muy superior, siguiendo una transformación no lineal elegida a priori. De esta forma se consigue tener un mayor grado de libertad para poder actuar sobre los datos[7]. A continuación y dentro de este nuevo espacio, se busca el hiperplano separador óptimo (función discriminadora, función de clasificación, función de decisión) como un desarrollo lineal de funciones núcleos donde una de las dos componentes es un vector input del conjunto de entrenamiento.

Claramente, por definirse a partir de un producto escalar, una función núcleo debe verificar necesariamente que:

1. (Simétrica) $k(x, z) = k(z, x)$, para todo $x, z \in \mathcal{X}$.
2. (Desigualdad de Cauchy-Schwarz) $|k(x, z)|^2 \leq k(x, x)^2 \cdot k(z, z)^2$, para todo $x, z \in \mathcal{X}$

El siguiente ejemplo permite ver como se trabaja siguiendo este esquema y entender la idea de trabajar en un espacio de dimensión superior.

Ejemplo 6.1 *Si a partir de un conjunto de entrenamiento $Z \subset I\!R^{d+1}$, se desea construir una función clasificadora, que se corresponda con un polinomio de grado dos en el espacio $\mathcal{X} \subset I\!R^d$, entonces se puede considerar un espacio característico $\mathcal{H}$ que tiene las coordenadas de la forma*[8]

$$\begin{aligned} g^1 = x^1, \cdots, g^d = x^d \quad & (d \text{ coordenadas}) \\ g^{d+1} = (x^1)^2, \cdots, g^{2d} = (x^d)^2 \quad & (d \text{ coordenadas}) \\ g^{2d+1} = x^1x^2, \cdots, g^{d'} = x^d x^{d-1} \quad & \left(C_d^2 = \frac{d(d-1)}{2} \text{ coordenadas}\right). \end{aligned}$$

[7] En nuestro trabajo, para poder discriminar mejor.

[8] En este ejemplo, por x^i denotamos la i-ésima coordenada del vector $x \in \mathbb{R}^n$.

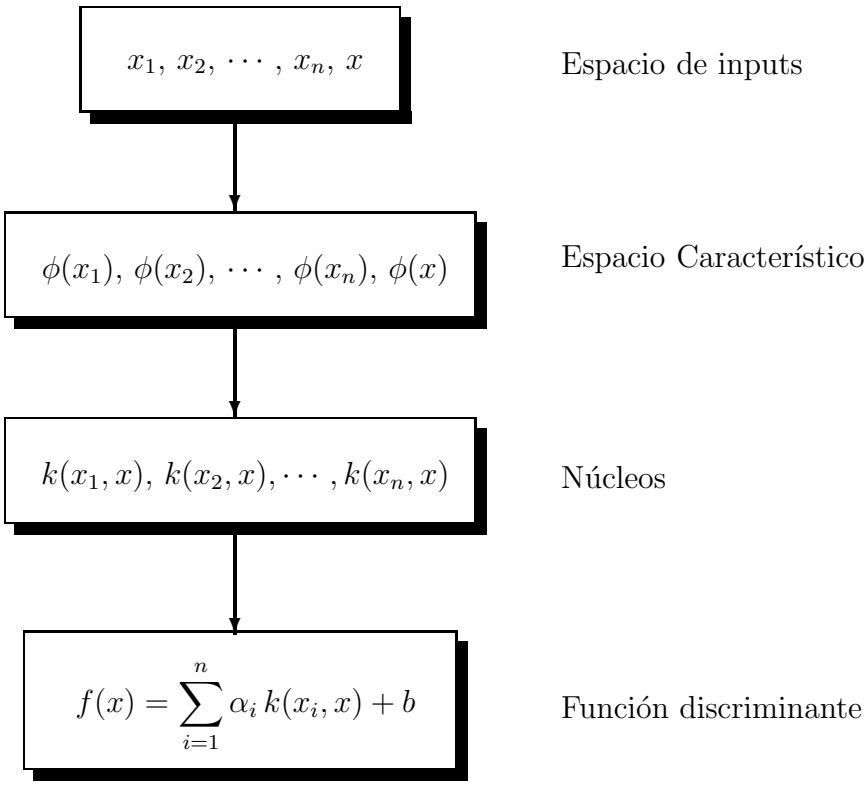

Figura 6.1: Las máquinas de vectores soporte transforman, inicialmente, el espacio input en un espacio característico de dimensión superior y entonces construye la función de clasificación lineal óptima dentro de este nuevo espacio.

Luego

$$d' = d + d + \frac{(d-1)\,d}{2} = \frac{d(d+3)}{2} = \sum_{i=d}^{2+d-1} \left(\begin{array}{c} i \\ d-1 \end{array} \right) = \left(\begin{array}{c} d+2 \\ d \end{array} \right) - 1.$$

Por tanto, si se tiene por ejemplo un conjunto de vectores inputs de dimensión 5 ($d = 5$), entonces el espacio característico tendrá dimensión 20 ($d' = 20$).

Si se desea una función polinomial de grado p donde se parte de un espacio input de dimensión d, entonces el espacio característico tiene dimensión

$$d' = \left(\begin{array}{c} d+p \\ d \end{array} \right) - 1.$$

Si consideramos un conjunto de 12 variables explicativas entonces $d = 12$ *y se consideran polinomios de grado 1 hasta 10 se obtiene la tabla 6.1 que nos indica la dimensión del espacio característico. Nótese el crecimiento de la dimensión del espacio característico cuando aumenta el grado del polinomio.*

Grado	1	2	3	4	5	6	7	8	9	10
d'	12	90	454	1819	6187	18563	50387	125969	293929	646645

Tabla 6.1: Dimensión d' del espacio característico cuando el grado del polinomio clasificador va desde 1 a 10; y el número de variables explicativas originales (d) es 12.

Nótese, que si se dispone, por ejemplo, de 200 variables explicativas y se llega a la conclusión que el grado del polinomio clasificador adecuado es 4, entonces la dimensión del espacio característico es $d' = 70_1\,058\,750$. *Si se aumenta el grado del polinomio a 5 entonces* $d' = 2\,872_1\,408\,790$ *sin embargo, lo realmente sorprendente de este enfoque es que la dificultad numérica y computacional del problema apenas si crece.* ▲

Como ya se ha indicado, este nuevo enfoque esta dirigido principalmente a problemas donde la dimensión del espacio de inputs es grande, es por ello por lo que se debe destacar que, a pesar de ser el espacio característico de una dimensión muy alta, la ejecución de las SVMs no depende de esta dimensión. Por todo ello la situación planteada, en el ejemplo anterior, no es desesperante ya que una de las características más interesante de utilizar las funciones núcleos está en que se puede tratar implícitamente con espacios $\mathcal{H}$ de dimensión arbitrariamente grande sin tener que calcular la transformación ϕ explícitamente, y por tanto no tener que manejar vectores de dimensión tan alta.

6.2 Propiedades de los núcleos

Tal y como se han introducido, las funciones núcleos $k(\cdot,\cdot)$, son funciones definidas de $\mathcal{X}\times\mathcal{X} \rightarrow \mathbb{R}$, pero, claramente, no todas las funciones de este tipo son funciones núcleos ya que éstas deben proceder de un producto escalar definido en un espacio característico $\mathcal{H}$, en principio desconocido.

En esta sección se estudian las propiedades que deben verificar estas funciones núcleos y también se dará un criterio para contrastar cuando, ciertamente, lo son y cuando no.

6.2.1 Normas

Aunque se trabaje con espacios de inputs pseudo-normados todas las declaraciones que se hagan en este capítulo son válidas para un espacio pseudo-métrico. En esta sección se introducen las pseudo-normas que se van a considerar en los desarrollos siguientes.

Definición 6.2.1 *Sea $x = (x^1, \cdots, x^d) \in I\!R^d$. La norma ℓ_p^d se define como:*

$$\begin{aligned} \|x\|_{\ell_p^d} &= \|x\|_p = \left(\sum_{j=1}^{d} \left|x^j\right|^p\right)^{1/p} \quad si \quad 0<p<\infty \\ \|x\|_{\ell_\infty^d} &= \|x\|_\infty = \max_{j=1,\cdots,d} \left|x^j\right| \quad si \quad p=\infty. \end{aligned}$$

Sea $\mathcal{F}$ una clase de funciones definida de $I\!R^d$ en $I\!R$ y $X = \{x_1, \cdots, x_n\} \subset I\!R^d$. Se define la norma ℓ_p^X con respecto a X de $f \in \mathcal{F}$ como:

$$\begin{aligned} \|f\|_{\ell_p^X} &= \|(f(x_1), \cdots, f(x_n))\|_p \quad si \quad 0<p<\infty \\ \|f\|_{\ell_\infty^X} &= \max_{i=1,\cdots,n} |f(x_i)| \quad si \quad p=\infty. \end{aligned}$$

Dado un conjunto $\mathcal{X}$, una medida μ sobre $\mathcal{X}$, algún $1 \leq p \leq \infty$ y una función $f : \mathcal{X} \rightarrow \mathbb{M}$, se define

$$\|f\|_{L_p(\mu,\mathcal{X},\mathbb{M})} = \left(\int_{\mathcal{X}} |f(x)|^p \, d\mu(x)\right)^{1/p} \tag{6.8}$$

si la integral existe, y[9]

$$\|f\|_{L_\infty(\mathcal{X},\mathbb{M})} = ess\sup_{x\in\mathcal{X}} |f(x)| . \tag{6.9}$$

Las normas dadas en (6.8) y (6.9) permiten introducir los siguientes espacios normados para $1 \leq p \leq \infty$:

$$L_p(\mu, \mathcal{X}, \mathbb{M}) = \left\{ f : \mathcal{X} \rightarrow \mathbb{M} \text{ tal que } \|f\|_{L_p(\mu,\mathcal{X},\mathbb{M})} < \infty \right\}.$$

Normalmente, salvo que lleve a confusión, se usará la expresión más corta, $L_p(\mathcal{X})$ para denotar $L_p(\mu, \mathcal{X}, \mathbb{M})$ con μ la medida de probabilidad que determina la distribución uniforme.

6.2.2 Condición de Mercer

Dado un espacio $\mathcal{X}$ dotado de un producto escalar habría que estudiar que propiedades deben cumplir las funciones

$$k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$$

que permiten construir un par $\{\phi, \mathcal{H}\}$, con las propiedades descritas anteriormente, es decir, $k(\cdot, \cdot)$ sea función núcleo.

En [Mer09] se da la condición necesaria y suficiente para que una función núcleo simétrica[10] $k(x, x')$ se corresponda con un producto escalar en algún espacio característico $\mathcal{H}$. En síntesis, una función núcleo se corresponde a un producto escalar en $\mathcal{H}$, si satisface la condición de Mercer, es decir, si genera un operador integral positivo. Esto último se sigue directamente del teorema de Mercer. La versión que de este teorema se indica a continuación, es un caso especial que aparece demostrado en [Kon86], página 145, y que nos parece la más adecuada en este contexto.

[9] *ess* significa esencialmente acotada. Formalmente, dado un espacio de medida $(\mu, \mathcal{X}, \mathbb{M})$ una función f medible sobre $\mathcal{X}$ se dice esencialmente acotada si existe un número $a \geq 0$ tal que se cumple $\mu(\{x \in \mathcal{X} : |f(x)| > a\} \bigcap A) = 0$ para cualquier A perteneciente al σ-algebra.

[10] Claramente, las funciones núcleos son funciones simétricas ya que viene definida a través de un producto escalar real.

Teorema 6.2.2 (Mercer) *Sea $(\mathcal{X}, \mu)$ un espacio medible finito*[11]*. Se supone además que $k \in L_\infty(\mathcal{X} \times \mathcal{X})$ es una función núcleo de dos variable simétrica tal que el operador*[12] *integral*

$$\begin{aligned} T_k : L_2(\mathcal{X}) &\to L_2(\mathcal{X}) \\ T_k(f(x)) &= \int_{\mathcal{X}} k(x,y) \cdot f(y)\, d\mu(y) \end{aligned} \tag{6.10}$$

es positivo[13]*.*

Sean $\psi_j \in L_2(\mathcal{X})$, $j = 1, 2, \cdots$ las autofunciones del operador T_k asociadas con los autovalores $\lambda_j > 0$, $j = 1, 2, \cdots$ (ordenados en orden no decreciente), y normalizadas[14] *entonces:*

1. *$(\lambda_1, \lambda_2, \cdots, \lambda_j, \cdots) \in \ell_1$ (la sucesión formada por todos los autovalores converge uniformemente),*
2. *$\psi_j \in L_\infty(\mathcal{X})$ y $\sup\limits_{j=1,2,\cdots} \|\psi_j\|_{L_\infty} < \infty$,*
3. *Si se denota por N_F el número de autovalores distinto de cero, se tiene que*

$$k(x,y) = \sum_{j=1}^{N_F} \lambda_j\, \psi_j(x)\, \psi_j(y) \tag{6.11}$$

es cierto para casi todo par (x, y). Si $N_F = \infty$ entonces la serie converge absoluta y uniformemente casi por todo.

Con el siguiente teorema[15] se busca aclarar, con más nitidez, lo que realmente se persigue con el teorema de Mercer:

Teorema 6.2.3 *Sea $\mathcal{H}$ un espacio de Hilbert separable*[16] *y T un operador compacto*

[11] Verifica $\mu(\mathcal{X}) < \infty$.

[12] Aplicación entre conjuntos de funciones.

[13] Para todo $f \in L_2(\mathcal{X})$ se tiene $T_k(f) \geq 0$.

[14] Es decir, $\lambda_1 \geq \lambda_2 \geq ...$ y $\|\psi_j\|_{L_2(\mathcal{X})} = 1,\ \forall j = 1, 2, ...$

[15] La demostración de este teorema se encuentra disponible en los apuntes de la asignatura de "Ecuaciones Funcionales I" que se imparte en la Facultad de Matemáticas de la Universidad de Sevilla.

[16] En el sentido topológico.

autoadjunto. Entonces $\mathcal{H}$ *admite una base hilbertiana de vectores propios* $\{\psi_j\}_{j\geq 1}$*de* T*, es decir, existe una sucesión* $\{\psi_j\}_{j\geq 1}$ *de vectores propios de* T *tales que:*

- $\langle \psi_i, \psi_j \rangle = \delta_{ij}, \qquad i, j = 1, 2, \cdots$
- *El espacio vectorial generado por los vectores propios es denso en* $\mathcal{H}$*.*

Si volvemos al teorema de Mercer y se observa la tercera declaración, para un número de autovalores finitos, $N_F < \infty$ se tiene que $k(x, y)$ corresponde con un producto escalar[17] en $\ell_2^{N_F}$, es decir, $k(x, y) = \langle \phi(x), \phi(y) \rangle_{\ell_2^{N_F}}$ con

$$\begin{array}{rcl} \phi : \mathcal{X} & \rightarrow & \ell_2^{N_F} \\ x & \rightarrow & (\sqrt{\lambda_1}\psi_1(x), \cdots, \sqrt{\lambda_{N_F}}\psi_{N_F}(x)) \end{array} \tag{6.12}$$

para todo $x \in \mathcal{X}$.

También siguiendo esta tercera afirmación, si $N_F = \infty$ se tiene que la convergencia uniforme de la serie (6.11) implica que dado $\varepsilon > 0$, se puede encontrar un número natural $n \in \mathbb{N}$ tal que la función $k(\cdot, \cdot)$ puede ser aproximada con una precisión ε como una función definida a través de un producto escalar en $\mathbb{R}^n$, donde la transformación de los inputs se define

$$\phi^n : x \rightarrow (\sqrt{\lambda_1}\psi_1(x), \cdots, \sqrt{\lambda_n}\psi_n(x)).$$

Sin embargo, definir ϕ implícitamente a través de la función $k(\cdot, \cdot)$ crea serios problemas. En su mayor parte, esta aplicación y muchas de sus propiedades son desconocidas. Aún peor, este método no proporciona ninguna regla general sobre que núcleos usar, o que transformación sobre un espacio de dimensión superior proporciona mejores resultados, simplemente el teorema de Mercer nos dice si una determinada transformación puede o no ser utilizada en los problemas SVMs.

Este problema se puede resolver demostrando que los núcleos de Mercer se corresponden con la función de Green del operador P^*P donde P es un operador de regularización y por P^* se denota el operador adjunto de P y, por tanto, se corresponde

[17]Por ℓ_p^N consideramos en este contexto el espacio $\mathbb{R}^N$ con la norma ℓ_p^N.

con el núcleo reproductor de un determinado espacio de Hilbert núcleo reproductor, como se verá en el siguiente apartado.

Por otro lado, dentro el contexto de las SVMs, se utiliza una caracterización más clara de núcleo de Mercer, la cual aparece inicialmente dada en [CH53] y se denomina condición de Mercer. Ésta nos indica lo siguiente: existirá una aplicación Φ y un desarrollo[18]

$$k(x,y) = \sum_{i=1}^{\dim \mathcal{H}} \Phi(x)_i \cdot \Phi(y)_i$$

si y solo si, para cualquier función real $g(x)$ tal que

$$\int g^2(x)\, dx < \infty$$

se cumple:

$$\int k(x,y)\, g(x)\, g(y)\, dx\, dy \geq 0. \tag{6.13}$$

Igual que antes, no es fácil comprobar si la condición de Mercer se cumple o no, ya que la condición (6.13) debe ser satisfecha por todas las funciones $g(x)$ de cuadrado integrable, es decir, funciones $g(x)$ con norma finita en L_2.

Sin embargo, si en un problema la función elegida $k(\cdot,\cdot)$ no cumple la condición de Mercer, será generalmente porque existen conjuntos de entrenamiento para los cuales el Hessiano no este definido, y el correspondiente problema de programación cuadrática no tenga solución (la función objetivo dual puede hacerse arbitrariamente grande). No obstante, para aquellos núcleos que no cumplen la condición de Mercer, para cualquier conjunto de entrenamiento, aún es posible plantear el problema de optimización de la máquina de vectores soporte para un conjunto de entrenamiento dado, con hessiano semidefinido positivo; y en tales casos la convergencia del algoritmo sigue estando plenamente garantizada.

[18] Por $\Phi(x)_i$ se denota la componente i-ésima del vector $\Phi(x)$.

6.2.3 La transformación núcleo reproductor

Normalmente se trabaja con espacios característicos que poseen una estructura determinada, una estructura de espacio de Hilbert tal que exista una función que genera todo el espacio (ver [SMB+99]), es lo que se conoce como espacio de Hilbert con núcleo reproductor[19].

Un espacio de Hilbert con núcleo reproductor $\mathcal{H}$, es un espacio de Hilbert de funciones f sobre algún conjunto $\mathcal{X}$ tal que todos los funcionales evaluadores, es decir, las aplicaciones $T_y : \mathcal{H} \rightarrow \mathbb{R}$ tales que

$$T_y(f) = f(y), \quad \forall f \in \mathcal{H}$$

con $y \in \mathcal{X}$ (fijado), están acotadas para todo $y \in \mathcal{X}$. En estos espacios se tiene, aplicando el teorema de representación de Riesz, que para cada $y \in \mathcal{X}$ existe una única función

$$k(\cdot, y) : \mathcal{X} \rightarrow \mathbb{R}$$

tal que $\forall\, f \in \mathcal{H}$:

$$f(y) = \langle f(\cdot), k(\cdot, y) \rangle_{\mathcal{H}} \tag{6.14}$$

De esta forma se tiene una función

$$k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$$

tal que

$$\langle f(x), k(x, y) \rangle_{\mathcal{H}} = f(y)$$

para todo x, $y \in \mathcal{X}$. A esta función se le denomina **núcleo reproductor** del espacio de Hilbert $\mathcal{H}$.

De la condición (6.14) se sigue que si

$$\langle f(\cdot), k(\cdot, y) \rangle_{\mathcal{H}} = 0, \quad \forall y \in \mathcal{X}$$

[19] Un estudio detallado de estos espacios se encuentra en el apéndice A de este trabajo.

entonces $f(y) = 0$, $\forall y \in \mathcal{X}$, es decir f es idénticamente nula, y de aquí se garantiza que el conjunto de funciones $\{k(\cdot, y);\, y \in \mathcal{X}\}$ genera completamente todo el espacio $\mathcal{H}$. Como una consecuencia de ello, el producto escalar sobre el espacio de Hilbert núcleo reproductor solo es necesario definirlo sobre $\langle f(\cdot), k(\cdot, y)\rangle_{\mathcal{H}}$ con $f \in \mathcal{H}$ ya que puede ser extendido a todo al espacio completo por linealidad y continuidad, como se demuestra en el apéndice A.

Por otro lado, de (6.14) se sigue un caso particular especialmente interesante, sin más que sustituir $f(\cdot)$ por $k(\cdot, x)$, que permite tener una definición más adecuada de la función núcleo reproductor:

$$k(x, y) = \langle k(\cdot, x), k(\cdot, y)\rangle_{\mathcal{H}}, \qquad \forall x,\, y \in \mathcal{X} \tag{6.15}$$

lo que implica que:

- la función k es simétrica por serlo el producto escalar y, lo que resulta más interesante
- se tiene que cualquier núcleo reproductor k se corresponde con un producto escalar en un determinado espacio,

Por tanto, toda función núcleo reproductor de un espacio de Hilbert es un núcleo de Mercer. Veamos que el recíproco también es cierto.

Teorema 6.2.4 *Para cualquier núcleo de Mercer k, existe un espacio de Hilbert $\mathcal{H}$ con núcleo reproductor k.*

Demostración. Sea k una función de dos variables que cumple las condiciones del teorema 6.2.2 (núcleo de Mercer). Sea el conjunto de funciones definidas de $\mathcal{X}$ en $\mathbb{R}$:

$$\mathcal{H} = \left\{ \sum_{i=1}^{\infty} \alpha_i\, k(x, x_i);\ \alpha_i \in \mathbb{R},\, x_i \in \mathcal{X}, \forall i = 1, 2, \cdots \right\}$$

es fácil comprobar que $\mathcal{H}$ es un espacio vectorial.

De las condiciones del teorema de Mercer se sigue que todas las funciones de $\mathcal{H}$ se expresan de la forma:

$$f(x) = \sum_{i=1}^{\infty} \alpha_i \, k(x, x_i) = \sum_{i=1}^{\infty} \alpha_i \sum_{j=1}^{N_F} \lambda_j \, \psi_j(x_i) \, \psi_j(x) \tag{6.16}$$

y a partir de aquí, se considera

$$\langle f(\cdot), k(\cdot, y) \rangle_{\mathcal{H}} = \sum_{i=1}^{\infty} \alpha_i \sum_{j,l=1}^{N_F} \lambda_j \, \psi_j(x_i) \, \langle \psi_j, \psi_l \rangle \, \lambda_l \, \psi_l(y) \tag{6.17}$$

donde por la linealidad de las funciones de $\mathcal{H}$ y del producto escalar en $L_2(\mathcal{X})$ se sigue que ciertamente (6.17) define un producto escalar en $\mathcal{H}$.

Ya que $k(\cdot, \cdot)$ es un núcleo de Mercer, las funciones ψ_i, $(i = 1, \cdots, N_F)$ pueden ser elegidas con respecto al producto escalar definido en $L_2(\mathcal{X})$, tales que

$$\langle \psi_j, \psi_l \rangle = \delta_{jl} / \lambda_j \tag{6.18}$$

(donde por δ_{jl} se denota la delta de Kronecker[20]). Se tiene entonces utilizando (6.17) que

$$\langle f(\cdot), k(\cdot, y) \rangle_{\mathcal{H}} = \sum_{i=1}^{\infty} \alpha_i \sum_{j,l=1}^{N_F} \lambda_j \, \psi_j(x_i) \, \frac{\delta_{jl}}{\lambda_j} \, \lambda_l \, \psi_l(y) = \sum_{i=1}^{\infty} \alpha_i \sum_{j=1}^{N_F} \lambda_j \, \psi_j(x_i) \, \psi_j(y) = f(y)$$

lo que significa que $k(\cdot, \cdot)$ es un núcleo reproductor en $\mathcal{H}$ y de aquí se sigue las tesis del teorema. ■

Nota 6.2.5 *Del teorema anterior se sigue que, si se considera la transformación*

$$\begin{array}{rccc} \widetilde{\phi} : & \mathcal{X} & \rightarrow & \mathcal{H} \\ & x & \rightarrow & k(\cdot, x) \end{array} \tag{6.19}$$

se tiene

$$k(x, y) = \langle \widetilde{\phi}(x), \widetilde{\phi}(y) \rangle_{\mathcal{H}} \tag{6.20}$$

[20] $\delta_{i,j} = \begin{cases} 1 & \text{si } j = i \\ 0 & \text{en otro caso.} \end{cases}$

con la ventaja de trabajar en un espacio de Hilbert $\mathcal{H}$ que presenta una estructura de funciones muy adecuada para la resolución de problemas de clasificación y de aproximación como se pone de manifiesto en [Par62]. ▲

Nota 6.2.6 *Aunque el teorema 6.2.4 proporciona un punto de vista teórico muy interesante, la aplicación $\widetilde{\phi}$ de la nota anterior no es tan útil como a primera vista parece. En la práctica, carecería de sentido esta aplicación que transforma los vectores inputs en funciones, es decir, en objetos de espacios de dimensión infinita. Sin embargo, si se dispone de un conjunto finito de datos, es posible aproximar la transformación $\widetilde{\phi}$ dada en (6.19) por el núcleo evaluado en esos puntos (ver [HT90]).* ▲

Esto permite dar la siguiente definición:

Definición 6.2.7 (Transformación núcleo empírico) *Dado un conjunto de vectores $\{z_1, \cdots, z_m\} \subset I\!R^d$, a la transformación*

$$\begin{aligned} \phi_m : \mathcal{X} \subset I\!R^d &\rightarrow I\!R^m \\ x &\rightarrow k(\cdot, x)|_{\{z_1, \cdots, z_m\}} \\ &= (k(z_1, x), \cdots, k(z_m, x)) \end{aligned} \tag{6.21}$$

se le llama transformación núcleo empírico sobre el conjunto $\{z_1, \cdots, z_m\}$.

Sea $k(\cdot, \cdot)$ un núcleo de Mercer y se considera como conjunto de vectores, los vectores inputs, es decir, $\{z_1, \cdots, z_m\} = \{x_1, \cdots, x_n\} = X$. Entonces, se sigue que el núcleo se evalúa exclusivamente en los vectores inputs, lo cual enlaza con los problemas que se plantean con las SVMs, ya que en éstos la solución aparece exclusivamente mediante desarrollos lineales de estas transformaciones en el conjunto de vectores inputs. Por tanto se puede representar la función $k(\cdot, x)$ de (6.19) como $\phi_n(x)$ sin pérdida de información. Sin embargo, el producto escalar usado en esta representación no es simplemente el producto escalar canónico en $I\!R^n$, ya que normalmente $\{\phi(x_i)\}_{i=1}^n$ no forma un sistema ortogonal. Para conseguir que ϕ_n

proporcione un espacio característico asociado con el núcleo k se necesita dotar a $\mathbb{R}^n$ de un producto escalar $\langle \cdot, \cdot \rangle$ tal como

$$k(x, y) = \langle \phi_n(x), \phi_n(y) \rangle. \tag{6.22}$$

Proposición 6.2.8 *La anterior construcción de la transformación ϕ_n proporciona un espacio de Hilbert núcleo reproductor discreto.*

Demostración. Sea $X = \{x_1, x_2, \cdots, x_n\} \subset \mathbb{R}^d$ un conjunto de vectores inputs y sea un núcleo de Mercer $k(\cdot, \cdot)$, el cual se supone determina una matriz K simétrica definida positiva[21] $\{k_{ij}\}_{i,j=1}^n = \{k(x_i, x_j)\}_{i,j=1}^n$ con inversa[22] $K^{-1} = \{k^{ij}\}_{i,j=1}^n$.

El espacio núcleo reproductor $\mathcal{H}_k$ asociado con el núcleo k consta de todos los vectores n-dimensionales $f = (f_1, f_2, \cdots, f_n)$, donde $f_i = f(x_i)$, con producto escalar dado por

$$\langle f, g \rangle_{\mathcal{H}_k} = \langle f, g \rangle_k = \sum_{s,t=1}^{n} f_s k^{st} g_t = f K^{-1} g.$$

Para probar que se trata de un producto escalar en ese espacio solo es necesario demostrar que se cumple la propiedad reproductora[23]. Utilizamos la notación $k(\cdot, x) = k_{\cdot x}$ a lo largo de lo que resta de demostración, y sea $i = 1, \cdots, n$ entonces se sigue:

$$\langle f, k_{\cdot i} \rangle_k = \sum_{s,t=1}^{n} f_s k^{st} k_{ti} = \sum_{s=1}^{n} f_s \delta_{si} = f_i = f(x_i).$$

Es interesante ver que este producto escalar es posible escribirlo como una

[21] Es lo que se conoce como matriz de Gram, la cual se definirá correctamente en la sección 6.7.

[22] En el peor de los casos se tiene una matriz semidefinida positiva. Aún así el resultado sigue siendo válido con matices como se puede ver en [Smo98]. Al final de la sección se indica como se actuaría cuando la matriz inversa no exista.

[23] Se sigue de los desarrollos del apéndice A, teorema A.3.1.

razón[24] entre dos determinantes.

$$\langle f, g\rangle_k = -\begin{vmatrix} k_{11} & \cdots & k_{1n} & f_1 \\ \vdots & \ddots & \vdots & \vdots \\ k_{n1} & \cdots & k_{nn} & f_n \\ g_1 & \cdots & g_n & 0 \end{vmatrix} \div \begin{vmatrix} k_{11} & \cdots & k_{1n} \\ \vdots & \ddots & \vdots \\ k_{n1} & \cdots & k_{nn} \end{vmatrix}. \tag{6.23}$$

Para demostrar que ciertamente la igualdad dada en (6.23) es un producto escalar basta ver que se verifica la propiedad reproductora, es decir hay que demostrar que $\langle f, k_{\cdot i}\rangle_k = f_i, \quad \forall f \in \mathbb{R}^n,\ i = 1, \cdots, n.$

Se desarrolla en primer lugar el numerador

$$\begin{vmatrix} k_{11} & \cdots & k_{1n} & k_{1i} \\ \vdots & \ddots & \vdots & \vdots \\ k_{n1} & \cdots & k_{nn} & k_{ni} \\ f_1 & \cdots & f_n & 0 \end{vmatrix} = \sum_{j=1}^{n} f_j \begin{vmatrix} k_{11} & \cdots & k_{1,j-1} & k_{1,j+1} & k_{1n} & k_{1i} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ k_{n1} & \cdots & k_{n,j-1} & k_{n,j+1} & k_{nn} & k_{ni} \end{vmatrix} = \cdots$$

donde la igualdad es cierta sin más que desarrollar el determinante por la última fila. Los determinantes que resultan en la segunda expresión para $j \neq i$ son ceros ya que se tendrían determinantes de matrices con dos filas iguales. Si $j = i$ se cambia la última columna y se coloca adecuadamente, con lo cual el determinante que resulta es igual al anterior con el signo cambiado, por tanto

$$\cdots = -f_i \begin{vmatrix} k_{11} & \cdots & k_{1n} \\ \vdots & \ddots & \vdots \\ k_{n1} & \cdots & k_{nn} \end{vmatrix}$$

y sustituyendo en la igualdad (6.23) se obtiene el resultado.

En el caso en que la matriz K determinada por los vectores inputs X sea singular, es posible definir el producto escalar en términos de la pseudoinversa de la matriz K y se mantiene el mismo resultado. ■

[24] La notación $a \div b$ significa $\frac{a}{b}$.

6.3 Relación entre SVM y sistemas de regularización

La relación entre las máquinas de vectores soporte y los espacios de Hilbert con núcleo reproductor queda determinada a partir del teorema 6.2.4 y la nota 6.2.5, pero el problema de elegir un núcleo de Mercer, aún queda sin resolver ya que como se puede seguir de [Gon00], se ha de elegir un operador y a partir de él construir el espacio de Hilbert adecuado (ver figura 6.2).

Los operadores, como se indicará en la sección siguiente, son aplicaciones entre conjuntos de funciones; como casos particulares de operadores se tienen los **operadores de regularización** que[25] son aquellos que aplicados a una determinada función proporciona una nueva función con un determinado grado de regularidad (suavidad).

Ejemplo 6.2 (de un operador de regularización) *Sea $C[a,b]$ el conjunto de todas las funciones reales continuas en el intervalo (a,b). Consideramos el operador $T(f)$ definido en $C[a,b]$, tal que*

$$T(f)(x) = \int_a^x f(t)\,dt.$$

Entonces sin más que aplicar el teorema fundamental del cálculo integral se tiene que $T(f) \in C^1[a,b]$, el conjunto de todas las funciones con derivada primera continua en el intervalo (a,b). ▲

Teniendo en cuenta que se desea tener una colección adecuada de núcleos, que proporcionen soluciones suficientemente suaves a los problemas de optimización, es conveniente estudiar la relación existente entre las SVMs y los operadores de regularización (sistemas de regularización).

[25] Posteriormente se dará una definición rigurosa de operador de regularización, pero es importante indicar cual es su significado intuitivo.

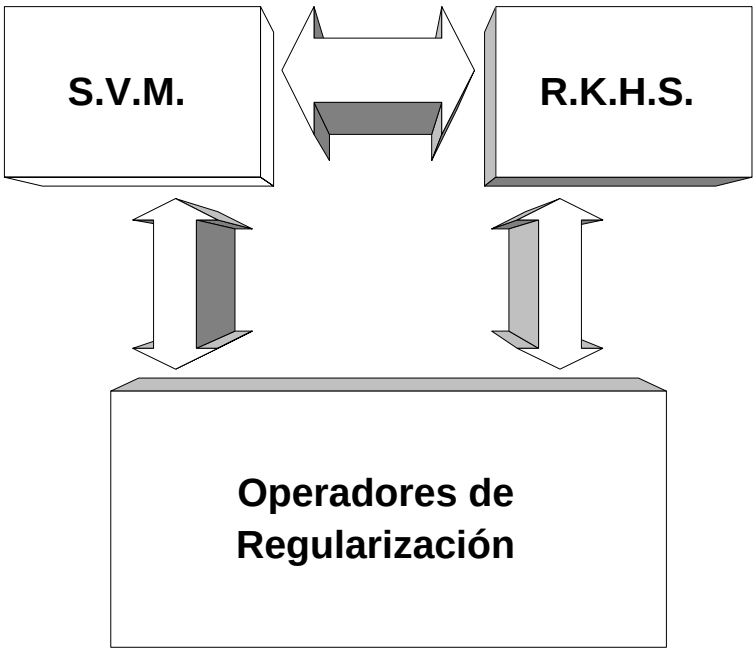

Figura 6.2: Relación Máquinas de Vectores Soporte — Espacios de Hilbert Núcleo Reproductor — Operadores de Regularización.

En los dos apartados siguientes se realiza un breve repaso de algunos conceptos relacionados con operadores y sistemas de regularización que serán necesarios para los desarrollos posteriores.

6.3.1 Operadores

Definición 6.3.1 (Operador) *Una aplicación P se dirá que es un operador si*

$$P : \mathcal{M} \rightarrow \mathcal{N} \tag{6.24}$$

donde $\mathcal{M}$ y $\mathcal{N}$ son conjuntos de funciones.

Si tanto en $\mathcal{M}$ como en $\mathcal{N}$ se tiene definido un producto escalar, entonces se denota por $\langle \cdot, \cdot \rangle_{\mathcal{M}}$ el producto escalar definido en $\mathcal{M}$, y análogamente $\langle \cdot, \cdot \rangle_{\mathcal{N}}$ el producto escalar en $\mathcal{N}$. A partir de los productos escalares en ambos conjuntos se da la siguiente definición.

Definición 6.3.2 (Operador Adjunto) *Dado un operador P definido de $\mathcal{M}$ en $\mathcal{N}$, se define el operador adjunto de P, y se denota por P^*, al operador*

$$P^* : \mathcal{N} \to \mathcal{M} \tag{6.25}$$

tal que para cualesquiera $f \in \mathcal{N}$ y $g \in \mathcal{M}$ cumple:

$$\langle f, Pg\rangle_{\mathcal{N}} = \langle P^* f, g\rangle_{\mathcal{M}} \tag{6.26}$$

Un operador muy interesante que se construye a partir de un operador P y su adjunto P^*, es el operador P^*P que esta definido de $\mathcal{M}$ en $\mathcal{M}$, es decir,

$$\begin{array}{rcl} P^*P : \mathcal{M} & \to & \mathcal{M} \\ f & \to & P^*Pf = g \end{array} \tag{6.27}$$

Nota 6.3.3 *No se debe pensar que el operador P^*P coincide con el operador identidad ya que ello, en general, no es cierto. Para que sea $P^*P = I$ se debe cumplir que $P^* = P^{-1}$, donde P^{-1} es el operador inverso de P, pero puede ocurrir que si bien existe P^* no exista P^{-1}.*

Por otro lado si $\mathcal{M} = \mathcal{N}$ puede ocurrir que $P^* = P$, en cuyo caso se dirá que P es un operador autoadjunto. Este tipo de operadores, aunque no sean tratados en este trabajo, han sido muy estudiados. ▲

Asociado a un operador P aparece la denominada función de Green, la cual fue introducida inicialmente en el contexto de las ecuaciones diferenciales. Para la exposición de este trabajo es suficiente conocer que la **función de Green**, la cual denotaremos por $G(x_i, x) = G_{x_i}(x)$, asociada al operador P^*P cumple

$$(P^*PG_{x_i})(x) = \delta_{x_i}(x) \tag{6.28}$$

donde $\delta_{x_i}(x)$ es la función δ de Dirac, la cual cumple la siguiente propiedad

$$\langle f(\cdot), \delta_{x_i}(\cdot)\rangle_{\mathcal{M}} = f(x_i) \tag{6.29}$$

Ejemplo 6.3 *Un ejemplo de función de Green muy interesante, que aparece desarrollado al final del apéndice A, es el siguiente:*

$$G_m(x,y) = \frac{(x-y)_+^{m-1}}{(m-1)!} \tag{6.30}$$

asociada al problema $D^{m)}f = g$, *con*[26] $f \in W_m^0$; *y donde el operador es* $P = D^{m)}$ *(la derivada m-ésima).*

Esta función es especialmente significativa en el estudio de los problemas de aproximación de funciones, de regresión y de clasificación por splines.

En este punto es necesario destacar que el libro [Wah90] inicialmente pensado para trabajar con funciones splines, han permitido a través de los R.K.H.S. proporcionar más luz en los desarrollos de la Teoría del Aprendizaje Estadístico, así como a las SVMs, como se puede ver en [Wah98]. ▲

6.3.2 Redes (o sistemas) de regularización

En los problemas que se plantean dentro de los sistemas de regularización se minimiza un funcional riesgo empírico $R_{emp}[f]$ más un término de regularización $Q[f]$, donde podemos considerar

$$Q[f] = \frac{1}{2}\|Pf\|^2 \tag{6.31}$$

definido mediante un operador de regularización P en el sentido de Tikhonov y Arsenin ([TA77]) cuya definición precisa es la siguiente:

Definición 6.3.4 (Operador de Regularización) *Un operador* P *definido de* $\mathcal{H}$ *en* $\mathcal{D}$, *donde* $\mathcal{H}$ *es un espacio de Hilbert de funciones y* $\mathcal{D}$ *en un espacio de fun-*

[26] El espacio W_m se denomina espacio de Sobolev, y se define:

$$W_m = \left\{ f : [0,1] \to \mathbb{R} \text{ tal que } f \in C^{m-1}[0,1] \text{ y } f^{m)} \in L_2[0,1] \right\}$$

y $W_m^0 \subset W_m$ tal que $f^{k)}(0) = 0$, para $k = 0, 1, \cdots, m-1$. Estos espacios son muy utilizados en problemas de aproximación mediante splines.

ciones dotado de un producto escalar, se denomina operador de regularización si es semidefinido positivo y la expresión $\langle Pf, Pg\rangle_{\mathcal{D}}$ se encuentra bien definida.

Si se considera el término

$$Q[f] = \frac{1}{2}\langle w, w\rangle = \frac{1}{2}\|w\|^2 \tag{6.32}$$

entonces uno de los objetivos principales de esta sección es demostrar que (6.31) y (6.32) son equivalentes para algún núcleo k correspondiente a un operador de regularización P.

6.3.3 Relación con las SVMs

En los sistemas de regularización, al igual que se planteaba en el capítulo 1 con las máquinas de vectores soporte, se busca minimizar un funcional riesgo regularizado el cual puede venir dado de alguna de las dos formas siguientes:

$$R_{reg,emp}[f] = R_{emp}[f] + \lambda\|Pf\|^2 = \frac{1}{n}\sum_{i=1}^{n} c(x_i, y_i, f(x_i)) + \lambda\|Pf\|^2 \tag{6.33}$$

$$= \|Pf\|^2 + C' R_{emp}[f] = \|Pf\|^2 + C\sum_{i=1}^{n} c(x_i, y_i, f(x_i)) \tag{6.34}$$

donde $C = \frac{C'}{n}$.

Si se considera un desarrollo de la función $f \in \mathcal{H}$ en términos de alguna función simétrica $k(x, x')$, donde $x, x' \in \mathcal{X}$, no necesariamente cumpliendo la condición de Mercer, en la forma

$$f(x) = \sum_{i=1}^{n} \alpha_i\, k(x_i, x) + b \tag{6.35}$$

esto conduciría a un problema de programación cuadrática similar a los obtenidos para las SVMs. Al igual que se seguía en estos desarrollos si se utiliza la notación:

$$D_{ij} = \langle (Pk)(x_i, \cdot), (Pk)(x_j, \cdot)\rangle, \qquad i, j = 1, \cdots, n \tag{6.36}$$

$$K_{ij} = k(x_i, x_j), \qquad i, j = 1, \cdots, n \tag{6.37}$$

se obtiene que el vector solución α se expresa en la forma:

$$\alpha = D^{-1}\, K\, (\beta - \beta^*),$$

con β_i, β_i^* la solución del problema dual de Wolfe asociado (ver [SS98]):

$$\text{Maximizar} \quad \left\{ \begin{array}{l} -\dfrac{1}{2} \displaystyle\sum_{i,j=1}^{n} (\beta_i - \beta_i^*)\, (\beta_j - \beta_j^*)\, (K D^{-1} K)_{ij} \\ + \displaystyle\sum_{i=1}^{n} \left(y_i\, (\beta_i - \beta_i^*) - \varepsilon_i\, \beta_i + \varepsilon_i^*\, \beta_i^* + \dfrac{1}{C}\, (T_i(\xi_i) + T_i^*(\xi_i^*)) \right) \end{array} \right.$$

$$\text{donde} \quad \left\{ \begin{array}{rcl} w & = & \displaystyle\sum_{i=1}^{n} (\beta_i - \beta_i^*)\, x_i \\ T_i^{(*)}(\xi) & = & c_i^{(*)}(\xi) - \xi \cdot \partial_\xi c_i^{(*)}(\xi) \end{array} \right.$$

$$\text{sujeto a} \quad \left\{ \begin{array}{rcl} \displaystyle\sum_{i=1}^{n} (\beta_i - \beta_i^*) & = & 0 \\ \beta & \leq & \dfrac{1}{C}\, \partial_\xi\, c(\xi) \\ \xi & = & \text{ínf} \left\{ \xi \,/\, \dfrac{1}{C}\, \partial_\xi\, c(\xi) \leq \beta \right\} \\ \beta\,, \xi & \geq & 0 \end{array} \right.$$

Comparando este problema de optimización con el planteado en las SVMs, se plantea estudiar bajo que condiciones son equivalentes, lo que nos llevaría a decir que un sistema de regularización coincide con un problema de optimización de una SVM y por tanto se tendrían las condiciones necesarias para que un sistema de regularización proporcione un problema cuya solución se puede expresar mediante una descomposición escasa, es decir, aparecen solo unos pocos de los coeficientes α_i, del desarrollo de la solución f, distintos de cero.

Una condición suficiente (pero no necesaria, dada en [Smo98]) es la que se denomina **condición de consistencia**, e indica que $D = K$, de esta forma se tendrá que $KD^{-1}K = K$, y de (6.36) y (6.37) se sigue:

$$k(x_i, x_j) = \langle (Pk)(x_i, \cdot), (Pk)(x_j, \cdot) \rangle \qquad \text{(condición de consistencia)} \qquad (6.38)$$

El objetivo que se plantea en esta sección, por tanto, es resolver los dos problemas siguientes:

- Dado un operador de regularización P encontrar un núcleo k tal que una SVM usando k no solo fuerce una solución suave en el espacio característico, sino que también se corresponda con una minimización de un funcional riesgo regularizado con P como operador de regularización.
- Dado un núcleo k, asociado a una SVM, encontrar un operador de regularización P tal que la máquina usando este núcleo pueda verse como un sistema de regularización usando P.

Estos dos problemas pueden resolverse de forma conjunta usando las funciones de Green como se describe en [GJP93]. Para ello, se establece una relación entre operadores de regularización y núcleos en el siguiente teorema.

Teorema 6.3.5 *Sea P un operador de regularización y G la función de Green asociada al operador P^*P. Entonces G es un núcleo de Mercer que verifica la condición de consistencia, es decir,*

$$G(x_i, x_j) = \langle (PG)(x_i, \cdot), (PG)(x_j, \cdot) \rangle.$$

Además las SVMs que utilizan como núcleo la función G minimiza el funcional riesgo regularizado con $\|Pf\|^2$ como regularizador.

Demostración. Sustituyendo en (6.29) la función f por G se tiene

$$G(x_i, x_j) = G_{x_i}(x_j) = \langle G_{x_i}(\cdot), \delta_{x_j}(\cdot) \rangle$$

de donde

$$G_{x_i}(x_j) = G(x_i, x_j) = \langle G_{x_i}(\cdot), P^*PG_{x_j}(\cdot) \rangle = \langle (PG_{x_i})(\cdot), (PG_{x_j})(\cdot) \rangle \tag{6.39}$$

De aquí se tiene que la función de Green es simétrica y cumple (6.38). Así el problema de optimización planteado con una SVM es equivalente al planteado bajo un sistema de regularización.

Además G es una función núcleo de Mercer ya que puede escribirse como un producto escalar en un espacio de Hilbert,

$$G(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle \quad \text{con } \phi : x_i \to (PG_{x_i})(\cdot) \tag{6.40}$$

lo cual demuestra el teorema. ■

Este teorema es muy similar al teorema 6.2.4. De hecho se ha demostrado la relación entre núcleo reproductor de un espacio de Hilbert y la función de Green de un determinado operador P. Por tanto, teniendo en cuenta ambos teoremas, se sigue que asociado a un operador P se puede considerar su función de Green que genera un espacio de Hilbert núcleo reproductor en el cual ella es la función núcleo reproductor.

Nota 6.3.6 *Aunque se viene indicando que para la resolución de los problemas de optimización no es necesario conocer explícitamente la transformación del espacio de inputs en el espacio característico, nótese que si se conoce el operador y la función de Green, esta transformación es conocida también ya que su expresión se da en (6.40), es decir,*

$$\begin{aligned} \phi : I\!R^d &\to \mathcal{H} \\ x &\to (PG_x)(\cdot) \end{aligned}$$

En este punto es válido el mismo comentario que en la nota (6.2.6). ▲

Si, como en el teorema 6.2.4, se utiliza un desarrollo en serie en el espacio R.K.H.S. de la función de Green (función núcleo reproductor) veremos que se pueden excluir las autofunciones del desarrollo que tengan una pequeña capacidad y, de esta forma, aproximar los datos con las funciones del desarrollo que tengan mayor capacidad.

Como ya se indicó, la condición (6.38) es una condición suficiente pero no necesaria (ver [Smo98], página 41) y es por ello por lo que se pueden encontrar núcleos que cumplan esta condición y no sean núcleos de Mercer.

6.3.4 Elegir núcleos

Lo expuesto en las últimas secciones proporciona distintas estrategias a la hora de considerar funciones núcleos en los problemas SVM[(27)]. Una de ellas, es elegir un determinado operador y determinar su función de Green y, con ésta, ya se dispone de un núcleo utilizando la expresión dada en (6.40). Otra estrategia pasa por elegir un determinado espacio de Hilbert núcleo reproductor y seleccionar como función núcleo el correspondiente núcleo reproductor. Otra solución es buscar núcleos que cumplan que los problemas SVMs y de los sistemas de regularización sean equivalentes aunque no se cumpla la condición de consistencia (6.38).

En esta sección se consideran los núcleos que más se utilizan en los desarrollos empíricos. En la clasificación de los núcleos es común diferenciarlos según sean o no invariantes frente a traslaciones, y es por ello que en este trabajo se sigue este esquema. Para ello se comienza dando la siguiente definición:

Definición 6.3.7 *Una función núcleo se dice invariante frente a traslaciones si cumple que*

$$k(x, x') = k(x + d, x' + d), \qquad \forall\, x, x', d \in I\!R^d \tag{6.41}$$

Es posible expresar estos tipos de núcleos en función de una única variable ya que se sigue del siguiente desarrollo que $\forall x', x'' \in \mathbb{R}^d$:

$$k(x', x'') = k(x' - x'', x'' - x'') = k(x' - x'', 0) = k(x), \qquad \text{con } x = x' - x'' \in \mathbb{R}^d$$

Por otro lado se tiene que, a la hora de construir núcleos en espacios de dimensión superior, es decir, núcleos definidos de $\mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ con $d > 1$, si no se persigue un tipo de núcleo multidimensional específico para un determinado problema, se

[(27)] Esta construcción no es exclusiva de las SVMs para la clasificación. De hecho, puesto que el problema de regresión es más completo que el de clasificación, los desarrollos de las secciones anteriores han sido desarrollados bajo la perspectivas de SVMs para la regresión y no de la clasificación.

pueden seguir básicamente, dos caminos. En el primero de ellos se tiene:

$$k(x, x') = \prod_{i=1}^{d} k_i(x^i, x'^i), \qquad x, x' \in \mathbb{R}^d \tag{6.42}$$

donde x^i indica la componente i-ésima del vector $x \in \mathbb{R}^d$ y los núcleos $k_i(\cdot, \cdot)$ son núcleos unidimensionales. La demostración, de que ciertamente en un núcleo de Mercer, se encuentra recogida en el apéndice A.

El segundo camino consiste en definir el núcleo multidimensional, cuando el núcleo es invariante frente a traslaciones, como sigue:

$$k(x - x') = k_1(\|x - x'\|) \tag{6.43}$$

donde $k_1(x)$ es un núcleo definido de $\mathbb{R} \to \mathbb{R}$.

6.4 Núcleos invariantes frente a traslaciones

Comenzamos con los núcleos invariantes frente a traslaciones que son los más utilizados en la práctica.

6.4.1 Núcleos de Gauss o Núcleos RBF

Se considera el operador pseudodiferencial[(28)] P cuya norma cuadrática está dada por:

$$\|Pf\|^2 = \int \sum_{n=1}^{\infty} \frac{\sigma^{2n}}{2^n n!} \left(O^n f(x)\right)^2 dx \tag{6.44}$$

con $O^{2n} = \triangle^n$ y $O^{2n+1} = \nabla\triangle^n$, siendo $\triangle$ el operador Laplaciano y ∇ el operador Gradiente. En [GJP93] se demuestra que la función de Green asociada a este

[(28)] Un operador pseudodiferencial difiere de un operador diferencial en la medida en que este contiene un número infinito de operadores diferenciales, lo cual se corresponde con un desarrollo de Taylor de un operador en un dominio de Fourier. Nótese que se ha de requerir que los argumentos se encuentren dentro del radio de convergencia de la serie.

operador tiene la forma:

$$k(x) = \exp\left(-\frac{\|x\|^2}{2\sigma^2}\right) \tag{6.45}$$

y se le denomina núcleo de Gauss o función de base radial, abreviadamente **RBF**. Además de ser invariante ante traslaciones, en [GJP93] se demuestra que este núcleo es invariante frente a rotaciones.

Los experimentos realizados con las SVMs con núcleos de Gauss (ver [SSB+97]) se corresponden con los problemas de minimizar un riesgo empírico con un operador de regularización del tipo (6.44). Indiquemos que utilizar el operador implícito en (6.44) significa que todas las derivadas de la función f son penalizadas con objeto de obtener una función de clasificación muy suave. Esto explica el buen rendimiento de las SVMs en estos casos, pero hay que señalar que esto no significa necesariamente que elegir una función suave en un espacio de dimensión superior se corresponda con una función suave en el espacio de los vectores inputs.

Por otro lado, hay que tener un cierto cuidado con su utilización en los problemas SVMs, ya que se puede encontrar en [Bur96] la demostración de que la dimensión VC de estos tipos de núcleos es infinita, y por tanto, pueden presentar una pobre capacidad de generalización.

En la figura 3.3 del capítulo 3, se puede ver la resolución de un problema de clasificación utilizando uno de estos núcleos. Hay que indicar, que diferentes desarrollos empíricos muestran que los núcleos de Gauss proporcionan muy buen rendimiento bajo supuestos generales de suavizamiento, y por tanto se suelen consideran especialmente cuando no se dispone de ningún conocimiento adicional sobre los datos disponibles. De hecho, son los más utilizados en la práctica, ya que el problema de tener dimensión VC infinita, se evita, en cierta forma, tomando un adecuado valor para el parámetro σ.

Las funciones de base radial, no solo se utilizan en los problemas de máquinas de vectores soporte, de hecho son ampliamente utilizadas en otros campos, como por ejemplo en las redes neuronales, sin mencionar que en el modelo probabilístico normal, la función de densidad tiene la forma de una RBF, de ahí que también se

denomina función gaussiana.

Sin embargo, las técnicas clásicas utilizan las funciones de base radial empleando algún método de determinación de un subconjunto de centros[29], por ejemplo mediante un método de cluster. Una característica atractiva de las SVMs es que esta selección queda implícita dentro de la configuración del problema ya que cada vector soporte contribuye con una función de base radial.

6.4.2 Núcleos de B_p-splines

Dentro de los núcleos invariantes frente a traslaciones aparecen los núcleos generados por B-splines. Estos núcleos dentro de las matemáticas de los ordenadores juegan un papel muy importante y fueron introducidos dentro de este contexto en [VGS97] por primera vez.

Un núcleo de B-splines de grado p, se define como:

$$k(x) = \prod_{i=1}^{d} B_p(x^i) \qquad x = (x^1, \cdots, x^d) \in \mathbb{R}^d \tag{6.46}$$

donde $B_p(x)$ es un B-splin de grado p.

Antes de dar la definición de B-splin de grado p se necesita:

Definición 6.4.1 *Dados los índices $i, m \in \mathbb{N}$ se define la i-ésima diferencia dividida de la función $f : \mathbb{R} \to \mathbb{R}$ en los puntos $t_i, t_{i+1}, \cdots, t_{i+m}$ como el coeficiente líder del polinomio[30] interpolador de orden $m+1$ de la función f en los puntos $t_i, t_{i+1}, \cdots, t_{i+m}$. Se denota por:*

$$[t_i, t_{i+1}, \cdots, t_{i+m}]\, f$$

[29] Puesto que la variable de la función x viene dada a partir de una diferencia $x - x_i$, a x_i se le denomina centro de la función.

[30] Se define el coeficiente líder de un polinomio en x como el coeficiente (no nulo) de la mayor potencia de x.

Definición 6.4.2 (de B-splin) *Sea $t = \{t_1, t_2, \cdots\}$ una sucesión de valores reales no decreciente (la cual puede ser finita o infinita). El i-ésimo B-splin (normalizado) de orden m para la sucesión de nodos t, se denota $B_{i,m,t}$ y se define como:*

$$B_{i,m,t}(x) = (t_{i+m} - t_i)[t_i, \cdots, t_{i+m}](t - x)_+^{m-1} \tag{6.47}$$

donde la diferencia dividida se toma considerando como variable t.

Es común denotar los B-splines simplemente por B_i en lugar de $B_{i,m,t}$ si del contexto se tienen claramente los valores de t y m.

Las propiedades más importantes de los B-splines son las siguientes:

1. El B-splin B_i esta definido en todo $\mathbb{R}$, pero es distinto de cero exclusivamente en el intervalo $[t_i, t_{i+m}]$, es decir

   $$B_i(x) = 0 \qquad \text{para} \quad x \notin [t_i, t_{i+m}]$$

   De esta propiedad también se sigue que de todos los B-splines de orden m para la sucesión de nodos t, solo los m B-splines, $B_{i-m+1}, B_{i-m+2}, \cdots, B_i$ serán no nulos en el intervalo $[t_i, t_{i+1}]$.

2. Se cumple para los B-splines de orden m que:

   $$\sum_{i=1}^{\infty} B_i(x) = \sum_{i=r+1-m}^{s-1} B_i(x) = 1 \qquad \text{para todo} \quad t_r < x < t_s \tag{6.48}$$

   Esto es lo que se conoce como condición de ortonormalización de los B-splines.

3. Los B-splines verifican la siguiente relación de recurrencia:

   $$B_{i,m}(x) = \frac{x - t_i}{t_{i+m-1} - t_i} B_{i,m-1}(x) + \frac{t_{i+m} - x}{t_{i+m} - t_{i+1}} B_{i+1,m-1}(x) \tag{6.49}$$

   con

   $$B_{j,1}(x) = \begin{cases} 1 & t_j \leq x \leq t_{j+1} \\ 0 & \text{en otro caso} \end{cases}$$

4. Los B-splines son estrictamente positivos en todo su dominio, es decir

$$B_i(x) > 0 \qquad \text{para} \quad t_i < x < t_{i+m}$$

Las propiedades anteriores demuestran que la sucesión de B-splines $\{B_i\}$ esta formada por funciones no negativas que suman la unidad, luego $\{B_i\}$ proporciona una partición de la unidad.

6.5 Núcleos no invariantes ante traslaciones

6.5.1 Núcleos de polinomios

Sea $\mathbb{N}_0^d$ el conjunto de los vectores d-dimensionales de números naturales incluido el cero. Se denota por $\mathbf{n} = (n_1, \cdots, n_d) \in \mathbb{N}_0^d$ un índice múltiple y se definen $|\mathbf{n}| = \sum_{i=1}^{d} n_i$ y el coeficiente (denominado coeficiente multinomial):

$$\left(\begin{array}{c} p \\ \mathbf{n} \end{array} \right) = \frac{p!}{(p - |\mathbf{n}|)!\, |\mathbf{n}|!} \tag{6.50}$$

Sea el funcional

$$D_0^{\mathbf{n})} f = \frac{1}{n_1!} \partial_{x_1}^{n_1} \cdots \frac{1}{n_d!} \partial_{x_d}^{n_d} f(x) \Big|_{x=0} \tag{6.51}$$

y $\{e_{\mathbf{n}}\}_{\mathbf{n}=1}^{p}$ una base ortonormal[31] en el espacio de los polinomios con d variables de grado p, es decir, $\langle e_{\mathbf{n}}, e_{\mathbf{n}'} \rangle = \delta_{\mathbf{n}\mathbf{n}'}$. Entonces en [Smo98] se demuestra que el operador[32]

$$P = \sum_{|\mathbf{n}| \leq p} e_{\mathbf{n}} \left(\begin{array}{c} p \\ \mathbf{n} \end{array} \right)^{\frac{1}{2}} D_0^{\mathbf{n})} \tag{6.52}$$

[31] Nótese que el subíndice es un índice múltiple ya que por ejemplo si $|\mathbf{n}| = 2$ y d=2 se tiene $e_{\mathbf{2}} = \{e_{2,0}, e_{1,1}, e_{0,2}\}$

[32] Obsérvese que para cada $\mathbf{n}$, $D_0^{\mathbf{n})}$ extrae exactamente un coeficiente del polinomio múltiple de grado p, él correspondiente al monomio $x_1^{n_1} \cdots x_d^{n_d}$.

es un operador de regularización y satisface la condición (6.38) con función de Green (núcleo reproductor, núcleo de Mercer):

$$k(x, y) = (\langle x, y \rangle + h)^p \tag{6.53}$$

donde x, $y \in \mathbb{R}^d$, $h \in \mathbb{R}$ y $p \in \mathbb{N}$. A este núcleo se le llama **núcleo polinomial de grado p**.

Si $h = 0$ se dirá núcleo polinomial homogéneo, aunque el caso más utilizado es con $h = 1$.

Se puede encontrar en [Bur96] que la dimensión VC de esta familia de funciones, si el espacio input tiene dimensión d, es $\begin{pmatrix} d+p \\ p \end{pmatrix} - 1$.

Estos núcleos junto con los núcleos de base radial, son los más utilizados en la práctica debido a su capacidad de generalización y a que la expresión de la solución, si el grado del polinomio es bajo (uno, dos o tres) permite unas adecuadas interpretaciones de los coeficientes del desarrollo. Nótese que cuando se plantea el problema SVM con conjuntos separables, la máquina lineal no es más que un caso particular de núcleos polinomiales con grado $p = 1$. En el capítulo 3, hemos podido ver un ejemplo de como funcionan.

6.5.2 Núcleos generados por splines

Este tipo de núcleos es una generalización de los núcleos polinomiales, ya que en su versión más simple, los splines no son más que funciones polinómicas a trozos. Por otro lado, como se indicó en los B-splines, los splines juegan un papel muy importante en la teoría de aproximación y, como se sigue de "Splines in Statistic" ([WW83]), desde hace ya más de tres décadas su utilización es cada vez mayor en los planteamientos y resolución de muchos y distintos problemas planteados dentro del marco de la estadística. Por tanto, parece lógico, que su utilización dentro de la Teoría del Aprendizaje Estadístico sea necesaria.

La utilización de este tipo de núcleos dentro de los problemas de las SVMs hace

necesario que se comprendan las raíces y el significado de función splin así como las técnicas de construcción y sus principales características. Un análisis de este tipo puede verse en el apéndice B de [Gon00].

Para la construcción de los núcleos en espacios de dimensión superior se sigue lo apuntado en (6.42) pero construyendo en lugar de d núcleos unidimensionales distintos, un único núcleo de dimensión uno, $k_1 : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$ y a partir de él, se construye un núcleo d-dimensional $k : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ como:

$$k(x, x') = \prod_{i=1}^{d} k_1(x^i, x'^i), \qquad x, x' \in \mathbb{R}^d$$

La construcción de splines se puede realizar con un número finito o infinito de nodos[(33)] (nudos). En ambos casos, la complejidad de la solución aportada por la máquina de vectores soporte depende del número de vectores soporte que se necesite para aproximar la función subyacente a los datos, más que de la dimensión del espacio de trabajo o del número de nodos. Se comienza con los splines con un número finito de nodos.

Splines con un número finito de nodos

Sea el intervalo finito $[0, 1]$, y se elige dentro de él, m puntos t_i, a los cuales se les denomina nodos, cumpliendo

$$0 < t_1 \leq t_2 \cdots \leq t_m \leq 1.$$

Hay que indicar que elegir el intervalo $[0, 1]$ no supone pérdida de generalidad ya que aplicando a este intervalo un cambio de origen y escala adecuado podemos hacer referencia a cualquier intervalo finito $[a, b]$. Esta observación, se debe tener en cuenta en la práctica, ya que normalmente, los programas disponibles antes de utilizar núcleos basados en splines necesitan que estos vengan normalizados.

[(33)] Un nodo es un punto de enlace entre dos polinomios consecutivos que definen una función splin.

La expresión de función splin de variable real de orden $p \in \mathbb{N}$ se puede ver en [Gon00] que se expresa de la siguiente forma:

$$s(x) = \sum_{j=0}^{p} a_j \, x^j + \sum_{j=1}^{m} b_j \, (x - t_j)_+^p \qquad a_j, \, b_j \in \mathbb{R}. \tag{6.54}$$

Cualquier función splin tiene una única representación en la forma (6.54) y ello es cierto debido al siguiente lema:

Lema 6.5.1 *El conjunto de funciones reales de variable real dado por:*

$$\{1, x, \cdots, x^p, (x - t_1)_+^p, \cdots, (x - t_\ell)_+^p\} \tag{6.55}$$

esta formado por funciones linealmente independientes.

Demostración. Se supone que todos los nodos son distintos[(34)] y se considera una combinación lineal de todas las funciones de (6.55) igualada a cero, es decir, se esta considerando una función del tipo (6.54). Igualando a la función idénticamente nula en $[0, 1]$:

$$0 = \sum_{j=0}^{p} a_j \, x^j + \sum_{j=1}^{\ell} b_j \, (x - t_j)_+^p \qquad \forall x \in \mathbb{R}.$$

Para todo $x \in [0, t_1)$, se tiene, por definición de la función de Heaviside, que

$$0 = \sum_{j=1}^{p} a_j \, x^{j-1}$$

y, si un polinomio de grado p coincide con el polinomio nulo es que necesariamente todos sus coeficientes son ceros, lo que implica que $a_j = 0, \; j = 0, \cdots, p$ ya que un polinomio de grado p tiene a los más p ceros.

Si $x \in [t_1, t_2)$ se tiene que:

$$0 = b_1 \, (x - t_1)_+ \qquad \forall x \in \mathbb{R}$$

luego $b_1 = 0$ y si se sigue el mismo razonamiento se llega a que $b_i = 0$, $i = 1, 2, \cdots, \ell$. Por tanto, se tiene demostrada la independencia lineal de las funciones (6.55) ya que

[(34)] Si algunos coinciden la demostración es similar pero algo más engorrosa.

se ha demostrado que cualquier combinación lineal de las funciones se iguala a la función nula solo si todos los coeficientes son nulos. ■

En [Gon00] se encuentra demostrado que el espacio de funciones generado por el conjunto (6.55) es un espacio de Hilbert con núcleo reproductor:

$$k(x,y) = \sum_{j=0}^{p} x^j\, y^j + \sum_{j=1}^{\ell} (y - t_j)_+^p\, (x - t_j)_+^p \tag{6.56}$$

En este caso, se tiene dada explícitamente la transformación de los vectores input en un espacio de dimensión superior como sigue:

$$\begin{array}{rcl} \phi : \mathbb{R} & \rightarrow & \mathbb{R}^{\ell+p+1} \\ x & \rightarrow & (1, x, \cdots, x^p, (x-t_1)_+^{p-1}, \cdots, (x-t_\ell)_+^{p-1}) \end{array} \tag{6.57}$$

También en el apéndice B de [Gon00] se puede encontrar una interpretación de la elección a priori de los nodos asociados a una función splin. Aunque en las SVMs, el número de nodos no juega un papel importante, si es cierto que, si se dispone de información a priori sobre los datos que indican puntos donde la clasificación sufre un cambio estructural, entonces eligiendo estos puntos como nodos se debe obtener una solución ligeramente mejor.

Splines con un número infinito de nodos

Con objeto de simplificar los cálculos es posible elegir funciones splin con un número infinito de nodos definidos en el intervalo $[0, 1]$. Estos tipos de splines se expresan en la forma

$$s(x) = \sum_{j=0}^{p} a_j\, x^j + \int_0^1 a(t)\, (x-t)_+^p\, dt, \qquad \forall x \in \mathbb{R},\ a_j \in \mathbb{R},\ j = 1, \cdots, p \tag{6.58}$$

y $a(t)$ es una función real de variable real[35]. Interpretando los desarrollos dados en [Gon00] se tiene que el conjunto de funciones de la forma (6.58) determina un

[35] Esta expresión de splin con infinitos nodos se sigue de la definición de integral de Riemann en un intervalo finito.

espacio de Hilbert núcleo reproductor con núcleo de Mercer:

$$k(x, x') = \sum_{j=1}^{m} x^j \, x'^j \; + \; \int_0^1 (x - t)_+^p \, (x' - t)_+^p \, dt \tag{6.59}$$

Esta expresión presenta la ventaja, respecto a los splines con un número finito de nodos, que no se tiene que elegir a priori un conjunto finito de valores como nodos.

Es aconsejable tener en cuenta que generalmente los algoritmos que incorporan núcleos de splines están configurados para trabajar en intervalos finitos de la forma $[0, a]$, por ello, si se quiere trabajar en otro intervalo distinto se debe realizar previamente un cambio de origen y de escala.

Aunque este tipo de núcleos pueda parecer de cálculo laborioso, ya que contiene un operador integral, su calculo en la búsqueda de la solución es relativamente fácil puesto que:

$$\begin{aligned} k(x_i, x_j) &= \int_0^1 (x_i - t)_+^p \, (x_j - t)_+^p \, dt + \sum_{k=0}^{p} x_i^p \, x_j^p \\ &= \int_0^{\text{mín}\{x_i, x_j\}} (x_i - t)^p \, (x_j - t)^p \, dt + \sum_{k=0}^{p} x_i^p \, x_j^p \\ &= \int_0^{\text{mín}\{x_i, x_j\}} u^p \, (u + |x_j - x_i|)^p \, du + \sum_{k=0}^{p} x_i^p \, x_j^p \\ &= \sum_{k=0}^{p} \frac{C_p^k}{2p - k + 1} \left(\text{mín}\left\{x_i, x_j\right\}\right)^{2p-k+1} \, |x_j - x_i|^k + \sum_{k=0}^{p} x_i^p \, x_j^p \end{aligned} \tag{6.60}$$

En la práctica los núcleos más utilizados corresponden a los núcleos de grado 1, 2 y 3 en los cuales se tiene:

- splin lineal ($p = 1$):

$$k(x_i, x_j) = 1 + x_i \, x_j + \frac{1}{2} \, |x_i - x_j| \, \text{mín}\left\{x_i, x_j\right\}^2 + \frac{\text{mín}\left\{x_i, x_j\right\}^3}{3}$$

- splin parabólico ($p = 2$):

$$\begin{aligned} k(x_i, x_j) &= 1 + x_i \, x_j + x_i^2 \, x_j^2 + \frac{1}{5} \, \text{mín}\left\{x_i, x_j\right\}^5 + \frac{1}{2} \, \text{mín}\left\{x_i, x_j\right\}^4 \, |x_i - x_j| + \\ &\quad \frac{1}{3} \, \text{mín}\left\{x_i, x_j\right\}^3 \, |x_i - x_j|^2 \end{aligned}$$

- splin cúbico ($p = 3$):

$$k(x_i, x_j) = 1 + x_i\, x_j + x_i^2\, x_j^2 + x_i^3\, x_j^3 + \frac{1}{7} \min\{x_i, x_j\}^7 + \frac{1}{2} \min\{x_i, x_j\}^6 \, |x_i - x_j| + \frac{3}{5} \min\{x_i, x_j\}^5 \, |x_i - x_j|^2 + \frac{1}{4} \min\{x_i, x_j\}^4 \, |x_i - x_j|^3$$

Los núcleos dados en esta sección son algunos de los más utilizados. Sin embargo es posible realizar combinaciones de ellos[(36)] e incluso se tienen otros núcleos como los núcleos sigmoidales, núcleos thin-plate-spline, núcleos de Dirichlet (basados en desarrollos de Fourier), etc.

6.6 Taxonomía Numérica

Siguiendo las reflexiones de [Her01], una de las primeras tareas de los seres humanos, en sus comienzos, fue la de clasificar los objetos y cosas que les rodeaban. ¿Qué se podía comer y que no?, ¿Cuáles son los animales peligrosos y cuáles no?, etc. Inicialmente, esta necesidad de comprender lo que le rodeaba no fue realizada de manera sistemática, de hecho, la primera clasificación sistematizada llevada a cabo por el hombre procede de la Zoología, el "Sistema Natural" de Carolus Linnaeus (1707-1778).

No todas las clasificaciones se llevan a cabo de la misma forma, sin embargo, lo que toda clasificación tiene en común es el objetivo que persiguen: separar o dividir los individuos que se estudian en diferentes clases o grupos, de tal manera que todos los individuos que pertenecen a una misma clase son "parecidos" entre si y "diferentes" de los individuos que pertenecen a otra clase distinta.

La taxonomía es *el estudio teórico de las clasificaciones, incluyendo sus bases, principios, procedimientos y reglas*, donde se entiende por clasificación a la agrupación de individuos en grupos o conjuntos en base a sus relaciones. Su objetivo principal es el de conocer la esencia de lo general de cada grupo y dejar de lado las diferencias particulares de los individuos.

[(36)] Ver por ejemplo en [CST00] distintas formas de construir núcleos.

Como caso particular, la taxonomía numérica estudia *la agrupación por métodos numéricos de individuos (unidades taxonómicas) en grupos (o taxas) sobre las bases de sus estados característicos.* Es decir, lleva a cabo las clasificaciones utilizando criterios cuantitativos.

Los principales fundamentos de la Taxonomía numérica son:

1. Cuanto mayor sea el contenido de la información de cada individuo y cuantas más variables o caracteres lo definan, en general, mejor será la clasificación que se obtendrá.

2. Inicialmente, cada carácter tiene el mismo peso que los demás para la determinación de las clases de individuos.

3. La similitud de conjunto entre dos clases cualesquiera es función de las similitudes individuales de los caracteres que se están comparando.

4. Se pueden reconocer grupos diferentes por la correlaciones de los diversos caracteres en los grupos de individuos.

Sin embargo, de la simplicidad de los fundamentos no se debe entender que todo es sencillo. En muchas ocasiones pueden aparecer problemas derivados de la falta de concreción en la definición de las clases; por un exceso de información que impida determinar características generales; por una mala elección en las características que definan la clasificación; por una clasificación en la que se produzcan solapamientos, etc.

El objetivo de la Taxonomía numérica es el de construir clasificaciones, que sigan una determinada jerarquía, basadas en las semejanzas de los individuos a clasificar. Que las clasificaciones sean jerárquicas tiene su explicación por el hecho de que una vez que se consideren los individuos objeto del estudio y éstos se van agrupando, es necesario establecer un proceso jerárquico de modo que el procedimiento de agrupación pueda dar lugar a grupos más o menos homogéneos, según el grado de precisión que se desee conseguir.

Como ejemplo de la necesidad de establecer una jerarquía podemos considerar él siguiente. Se estudia dentro de las empresas del sector hotelero, el conjunto de empresas que han quebrado en el periodo 1990-1995 en España. De este conjunto de empresas se registra información relacionada con distintos aspectos relevantes de su gestión. Como una primera clasificación se podría considerar una clase formada por todas las empresas, porque todas tienen en común el hecho de ser empresas del sector hotelero. Otra clasificación extrema sería aquella que considera tantas clases como empresas, es decir, cada clase la forma una única empresa, puesto que todas las empresas son distintas. Entre estas dos clasificaciones caben clasificaciones intermedias, como por ejemplo distinguir entre empresas que no han sido capaces de satisfacer las deudas con proveedores y las restantes.

El proceso que se sigue para lograr el objetivo de clasificar los individuos es:

1. Se seleccionan los individuos y se codifican.
2. Se calculan las semejanzas (similitudes) entre dichos individuos.
3. Se construyen los grupos (taxas) en función de dichas similitudes.
4. Se hacen generalizaciones acerca de los grupos obtenidos.

6.6.1 Medidas de similitud

Como ya se apuntó anteriormente, el concepto de similitud es fundamental en estos desarrollos. Así podemos definir, la **similitud entre objetos** como una medida de correspondencia, o parecido, entre objetos que van a ser agrupados.

La similitud puede medirse de varias formas, pero siguiendo a [SS73], se pueden considerar cuatro grandes tipos de medidas de similitud:

1. Distancias: Representan la similitud como la proximidad entre dos objetos. En realidad, las distancias son las medidas inversas de las similitudes, es decir, son disimilitudes. La más utilizada es la distancia euclidea.

2. Coeficientes de asociación: Se basa en la utilización de datos cualitativos, aunque es posible utilizar con datos cuantitativos si se esta dispuesto a perder parte de información. Estas medidas son una forma de medir concordancia o conformidad entre los estados de dos variables o atributos asociados a todos los elementos de la muestra.

3. Coeficientes angulares: Mide la proporcionalidad e independencia entre los vectores que definen los elementos de la muestra. El más conocido es el coeficiente de correlación aplicado a variables continuas. También, son utilizadas las covarianzas asociadas a variables tanto continuas como discretas. Este tipo de similitud será la que nosotros consideraremos en la siguiente sección.

4. Coeficientes de similitud probabilística: Incluyen información estadística y miden la homogeneidad del sistema por particiones o subparticiones del conjunto muestral. La idea de utilizar estos coeficientes se basa en relacionarlos con diferentes clasificaciones utilizando para ello criterios de bondad de ajuste. En general, estos coeficientes se distribuyen como un modelo χ^2. De esta forma, es posible en algunos casos establecer un contraste de hipótesis y estudiarlos por los métodos inferenciales clásicos.

6.7 Núcleos como medida de similitud

La siguiente sección proporciona un camino alternativo de introducir la función núcleo, tanto para los problemas de clasificación como para los problemas de regresión y en general para cualquier otro problema que pueda ser resuelto utilizando el método de los núcleos. Para completar la sección se ha seguido principalmente [Skh00], donde se da un tratamiento muy adecuado y de una manera muy pedagógica.

Sea un conjunto de vectores inputs $X = \{x_1, \cdots, x_n\} \subset \mathcal{X}$. Como en secciones anteriores consideramos el problema de clasificación con dos etiquetas posibles $\mathcal{Y} = \{-1, 1\}$

Hasta ahora, en todos los desarrollos del trabajo se ha supuesto que el conjunto de inputs, $\mathcal{X} \subset \mathbb{R}^d$. Sin embargo, en este caso supondremos que el conjunto de inputs, $\mathcal{X}$, puede ser cualquier conjunto no vacío, sin necesidad de tener una determinada estructura. De esta forma, estamos generalizando el problema de clasificación en otro aspecto más.

Dado un nuevo input $x \in \mathcal{X}$, la máquina SV debe elegir una etiqueta $y \in \mathcal{Y}$, de tal forma que el par (x, y) sea en "algún sentido" similar a los datos del conjunto de entrenamiento. Para ello, debemos necesariamente medir similitudes en $\mathcal{X}$ ya que en $\mathcal{Y}$ es fácil de establecer comparaciones, es decir, hemos de buscar una función que a cada par de inputs x, x' le asocie un número que cuantifique "lo parecido" que son. Si, como veníamos trabajando, estamos en $\mathbb{R}^d$, esta similitud venía dada por el producto escalar $\langle x, x' \rangle$, pero ahora al no tener una estructura de espacio vectorial dotado de un producto escalar en $\mathcal{X}$, no podemos hacer uso de ella.

De esta forma, se sigue que una de las más importante características de la función núcleo es la de medir similitudes. Dada una función

$$k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$$

con el valor $k(x, x')$, para cada par de inputs x, x', se pretende medir similitudes entre ellos. Como ya indicamos en el párrafo anterior, la medida de similitud más estudiada, cuando $\mathcal{X} \subseteq \mathbb{R}^n$, es el producto escalar[37] $\langle x, x' \rangle = \sum_{i=1}^{n} x_i \cdot x'_i$, ya que este nos proporciona la distancia euclidea.

Al considerar como medida de similitud entre dos inputs, la que se sigue de un producto escalar, se esta utilizando una medida de similitud basada en **coeficientes angulares**; y por tanto, mide la proporcionalidad e independencia entre los vectores que definen los correspondientes inputs.

Por tanto, para poder utilizar como medida de similitud en el espacio input $\mathcal{X}$, un producto escalar, necesariamente hemos de incrustar $\mathcal{X}$ dentro de un espacio característico dotado de un producto escalar $\mathcal{H}$, a partir de una aplicación ϕ definida

[37] Ver una interpretación geométrica muy acertada en [Skh00].

de $\mathcal{X}$ en $\mathcal{H}$.

Este espacio $\mathcal{H}$ presenta, según los autores de [Skh00], tres ventajas:

1. Permite definir similitudes en $\mathcal{X}$, a través del producto escalar en $\mathcal{H}$:

   $$k(x, x') = \langle \phi(x), \phi(x') \rangle_{\mathcal{H}}$$

   De esta forma estamos en condiciones de realizar algún tipo de análisis cluster dentro del conjunto $\mathcal{X}$.

2. Permite tratar con patrones geométricos, y así se estudian algoritmos de aprendizaje utilizando el álgebra lineal y la geometría analítica.

3. La libertad en la elección de ϕ, proporciona una gran variedad de algoritmos de aprendizaje.

Para continuar con el desarrollo de la sección necesitamos las siguientes definiciones:

Definición 6.7.1 *Dado un núcleo k y un conjunto $\{x_1, x_2, \cdots x_n\} \subset \mathcal{X}$, la matriz de orden $n \times n$:*

$$\mathbf{K} = \{k(x_i, x_j)\}_{i,j=1}^{n}$$

se denomina **matriz de Gram** *(o matriz núcleo) de k respecto de $x_1, x_2, \cdots, x_n$.*

Definición 6.7.2 *Una matriz de orden $n \times n$, $\mathbf{K} = \{k_{ij}\}_{i,j=1}^{n}$ que verifica:*

$$\sum_{i,j=1}^{n} c_i \, c_j k_{ij} \geq 0$$

para todo $c_i \in \mathbb{R}$ se denomina **matriz positiva**

Definición 6.7.3 *Sea $\mathcal{X}$ un conjunto no vacío. Una función*[38] *$k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ tal que para cualquier $x_i \in \mathcal{X}$, y cualquier $n \in \mathbb{N}$ proporciona una matriz de Gram positiva es denominado* **núcleo definido positivo**.

[38] Es posible dar una definición más general, trabajando en el conjunto de los números complejos en lugar de en $\mathbb{R}$, pero puesto que nosotros consideraremos siempre núcleos simétricos, nos parece más adecuado trabajar directamente en el conjunto de los números reales.

En el teorema A.2.7 del apéndice encontramos una demostración de que todo núcleo definido positivo es un núcleo reproductor en un determinado espacio de Hilbert; y por tanto se trata de un núcleo de Mercer.

Nota 6.7.4 *Es un error, muy común, confundir núcleos positivos con núcleos que solo toman valores positivos, ya que esto es falso como podemos ver en el siguiente ejemplo.*

Consideramos como núcleo k, el producto escalar en $\mathbb{R}^2$ y se toman los vectores $a = (1, 2)'$ y $b = (1, -2)'$; entonces:

$$k(a, b) = \langle a, b \rangle = a'\, b = 1 \cdot 1 + 2 \cdot (-2) = -3 < 0.$$

Pensamos que este error se debe a que la mayoría de la veces el núcleo que se utiliza es el núcleo gaussiano y este si es positivo siempre. ▲

Esta forma de introducir los núcleos, nos permite disponer de un nuevo criterio a la hora de su selección, ya que al venir definido como un producto escalar, nos permite trabajar con una noción más intuitiva de similitud en el espacio de los inputs. Por tanto, frente a un determinado problema real, los expertos pueden proporcionar una guía que nos permita interpretar la idea de similitud e implementarla mediante una función núcleo dentro del problema de clasificación que tengamos.

Como ejemplo de función núcleo definido en un conjunto $\mathcal{X}$ que no es un espacio vectorial dotado de un producto escalar tenemos el **núcleo de similitud** de sucesos

$$k : \mathcal{A} \times \mathcal{A} \to \mathbb{R}$$

definido como sigue:

$$k(A, B) = P\left(A \bigcap B\right) - P(A) \cdot P(B), \qquad \forall\, A,\, B \in \mathcal{A}$$

donde $\mathcal{A}$ es el σ-álgebra del espacio probabilístico $(\Omega, \mathcal{A}, P)$.

El estudio exhaustivo de este tipo de núcleo, en el siguiente capítulo, nos permitirá aportar soluciones a distintos problemas.

En [Skh00] se garantiza que ciertamente la función núcleo similitud es una función núcleo, pero no aparece dada su demostración, la cual es como sigue:

Lema 6.7.5 *La función núcleo similitud es un núcleo de Mercer.*

Demostración. Para comprobar que es núcleo de Mercer veremos que la matriz de Gram asociada a una colección A_1, A_2, $\cdots$, A_n de sucesos cualesquiera del σ-álgebra $\mathcal{A}$ es definida positiva.

La idea clave es expresar la función $k(A, B)$, para todo A, $B \in \mathcal{A}$ como una covarianza entre dos variables aleatorias.

Así, sean A, $B \in \mathcal{A}$ y consideramos las variables aleatorias $X = I_A$ e $Y = I_B$, donde como ya se denotó en el capítulo segundo, la función $I_A(\cdot)$ es la función indicatriz del conjunto A, la cual toma el valor 0 si $x \in \overline{A}$ y el valor 1 si $x \in A$.

Antes de calcular el valor esperado de las variables X, Y y $X \cdot Y$ demostramos que $I_A(x) \cdot I_B(x) = I_{A \bigcap B}(x)$:

- Si $x \in A \bigcap B$ entonces $x \in A$ luego $I_A(x) = 1$; y $x \in B$ luego $I_B(x) = 1$. Por tanto $I_{A \bigcap B}(x) = 1 = 1 \cdot 1 = I_A(x) \cdot I_B(x)$.
- Si $x \notin A \bigcap B$ entonces $x \notin A$ o $x \notin B$ luego $I_A(x) = 0$ ó $I_B(x) = 0$ y por tanto $I_A(x) \cdot I_B(x) = 0$ y como $I_{A \bigcap B}(x) = 0$ se tiene la igualdad entre ambas funciones.

Calculamos:

$$\begin{aligned} E[X] &= E[I_A] = 1 \cdot P(X = 1) + 0 \cdot P(X = 0) = 1 \cdot P(A) + 0 \cdot P(\overline{A}) = P(A), \\ E[Y] &= E[I_B] = P(B), \\ E[X \cdot Y] &= E[I_A \cdot I_B] = E[I_{A \bigcap B}] = P(A \bigcap B). \end{aligned}$$

De esta forma se tiene que:

$$\begin{aligned} Cov(X, Y) &= E[X \cdot Y] - E[X] \cdot E[Y] \\ &= P(A \bigcap B) - P(A) \cdot P(B) \\ &= k(A, B). \end{aligned}$$

Luego para todo A, $B \in \mathcal{A}$ se tiene que $k(A,B) = Cov(I_A, I_B)$.

Sean A_1, A_2, $\cdots$, $A_n \in \mathcal{A}$ y $\mathbf{K} = \{k_{ij}\}_{i,j=1}^n = \{k(A_i,A_j)\}_{i,j=1}^n$ la matriz de Gram. Para cualquier vector $\mathbf{c}' = (c_1, c_2, \cdots, c_n) \in \mathbb{R}^n$ se tiene que :

$$\begin{aligned}
\mathbf{c}'\,\mathbf{K}\,\mathbf{c} &= \sum_{i,j=1}^{n} c_i\, c_j\, k(A_i,A_j) = \sum_{i,j=1}^{n} c_i\, c_j\, Cov(I_{A_i}, I_{A_j}) \\
&= \sum_{i=1}^{n}\sum_{j=1}^{n} Cov(c_i\, I_{A_i}, c_j\, I_{A_j}) = Cov\left(\sum_{i=1}^{n} c_i\, I_{A_i}, \sum_{j=1}^{n} c_j\, I_{A_j}\right) \\
&= Varianza\left(\sum_{i=1}^{n} c_i\, I_{A_i}\right) \geq 0.
\end{aligned}$$

Por tanto la matriz de Gram es definida positiva y se tiene concluido el lema. ■

6.8 Algunos comentarios sobre los núcleos

Las máquinas de vectores soporte implementan el principio de minimización del riesgo estructural y de aquí que puedan generalizar bien. Por ello, una elección cuidadosa de la función núcleo es necesaria para poder obtener cotas sobre las generalizaciones en los problemas de clasificación que sean topológicamente apropiadas.

La introducción de los núcleos aumenta significativamente la potencia de las máquinas de aprendizaje a la vez que retiene la linealidad que asegura que los aprendizajes resulten comprensibles. El incremento de la flexibilidad, sin embargo, incrementa el riesgo de sobreajuste con la elección de hiperplanos separables que aumentan los problemas mal propuestos debido al número de grados de libertad. Así, para muchas clases de núcleos, por ejemplo los núcleos gaussianos, siempre es posible encontrar un parámetro del núcleo para el cual los datos llegan a ser separables, salvo los casos patológicos[(3)]. Sin embargo, forzar la separación de los datos puede conducir fácilmente al sobreajuste, en particular cuando hay ruido en los datos. En estos casos, los outliers podrían ser caracterizados por tener multiplicadores de Lagrange muy grande, y el procedimiento podría ser usado para depurar los datos

ya que ello puede clasificar los datos de entrenamiento de acuerdo a la dificultad que ellos presentan para una clasificación correcta ([CST00]).

Resulta muy sorprendente, que aplicando las SVMs con diferentes núcleos, los resultados empíricos conduzcan a resultados muy similares de precisión en la clasificación y de los conjuntos de vectores soporte, como se indica en [SBV95]. De esta forma, el conjunto de vectores soporte parece caracterizar la tarea dada de manera, que es independiente en cierto grado, del tipo de núcleo usado.

Por todo ello, es necesario tener un control adecuado de la flexibilidad del espacio característico inducido por el núcleo. Para ello se requiere una teoría de generalización, la cual sea capaz de describir con precisión que factores han de ser controlados en las máquinas de aprendizaje con objeto de garantizar unas buenas generalizaciones. De aquí que el estudio de los núcleos sea un campo de investigación muy activo hoy día.

PARTE III

Estudio de similitudes entre sucesos a partir de una función núcleo. Análisis práctico de dos problemas de multiclasificación

CAPÍTULO 7

SIMILITUD ENTRE SUCESOS. APLICACIÓN A DISTINTAS LÍNEAS DE INVESTIGACIÓN SOBRE TEORÍA DEL APRENDIZAJE

Su trabajo[1] *pone de manifiesto un hecho extraño de la investigación matemática, a saber, que es imposible predecir que líneas de investigación conducirán a resultados útiles.*

–Revista Investigación y Ciencia. Monotemático sobre Grandes Matemáticos–

Como un ejemplo de la versatilidad de las funciones núcleos, en este capítulo se propone una medida de similitud entre sucesos de un σ-álgebra $\mathcal{A}$ dentro de un espacio probabilístico $(\Omega, \mathcal{A}, P)$, definida a partir de una función núcleo. Para estudiar la validez de esta nueva medida, se consideran diferentes tipos de funciones de similitud entre sucesos y comprobamos que la función de similitud que proponemos mejora a las anteriores en diversos aspectos.

[1] Se refiere a los trabajos sobre ideales de Ernst Eduard Kummer de mediados del siglo XIX.

En primer lugar, tratamos con una función S_1, que mide la diferencia simétrica entre conjuntos como función de similitud. Estudiamos las propiedades relativas a la similitud entre conjuntos, sus consecuencias e interpretación y los inconvenientes que se tienen con esta función. Si se considera S_1 y su complementaria S_2, se observa que ambas no tienen en cuenta el tamaño de los conjuntos estudiados cuando se evalúa la similitud de un suceso consigo mismo. Esta dificultad, la resolvemos, considerando una nueva función S_3, pero ésta presenta otra dificultad, como es la ser nula cuando los conjuntos son disjuntos, sin darnos la posibilidad de medir la no similitud.

Todas estas dificultades quedan resueltas con la función núcleo similitud, por nosotros propuesta, la cual llega a medir tanto la similitud entre dos sucesos, como la no similitud en caso contrario. Además, nos proporciona, en todo los casos posibles, una interpretación muy intuitiva tanto de su cuantificación como de su signo.

Finalizamos este capítulo con varios ejemplos de utilización de la función de similitud. El primero de ellos, nos permite observar de manera intuitiva los resultados teóricos obtenidos así como seguir los cálculos manualmente. El segundo, es un ejercicio que nos permite ver la utilidad que nos reporta si se conoce la función de distribución conjunta de un vector aleatorio. También, en este ejercicio demostramos la relación entre la función núcleo similitud y las funciones de distribución y de densidad[(2)] de un vector aleatorio. El tercer y último ejemplo, es el que nos parece más importante ya que realizamos un estudio de las similitudes que existen entre distintas líneas de investigación asociadas a la Teoría del Aprendizaje, referente al año 2000, que nos permite dar una primera utilidad práctica a esta herramienta.

Además aportamos una serie de representaciones gráficas que nos permiten interpretar gráficamente los resultados obtenidos analíticamente.

[(2)]Si el vector es absolutamente continuo.

7.1 Introducción

Sea un espacio probabilístico $(\Omega, \mathcal{A}, P)$, donde Ω es un espacio muestral, $\mathcal{A}$ es un σ-álgebra de Ω y P es una medida de probabilidad asociada al σ-álgebra $\mathcal{A}$. A los elementos del σ-álgebra $\mathcal{A}$, y los conjuntos (cuando se interpreten como tales) se denotarán con letra mayúscula A, B, C, ... utilizándose subíndices cuando sea necesario.

Dados dos conjuntos A y B, como los que aparecen en la figura 7.1, se construye una partición[3] del espacio muestral Ω en la forma[4]:

$$\Omega = (A \setminus B) \bigcup (B \setminus A) \bigcup \left(A \bigcap B \right) \bigcup \left(\overline{A \bigcup B} \right)$$

A la vista de la figura 7.1, e intuitivamente se indicaría que los conjuntos A y B son tanto más similares cuanto menor sea el conjunto[5] $(A \setminus B) \bigcup (B \setminus A)$, es decir, cuanto menos elementos tenga el conjunto A que no tenga B y recíprocamente. Por tanto, sería adecuado utilizar estos conjuntos para construir una medida de similitud, pero necesariamente hemos de construir una aplicación que nos permita llevar a cabo una cuantificación de este grado de similitud. Para esto, sería muy conveniente utilizar la función de probabilidad P, ya que ésta no solo nos permite llevar a cabo una cuantificación de la similitud sino que, además, nos permite asignar pesos (probabilidades) a los diferentes elementos que constituyen los sucesos[6] A y B.

[3] Una partición de un conjunto Ω está formada por una colección de subconjuntos de Ω que son:

- Mutuamente excluyentes, es decir, la intersección dos a dos es vacía.
- Exhaustivos, es decir, la unión de todos ellos determinan el conjunto Ω.

[4] $A \setminus B = \{w \in \Omega, \text{ tal que } w \in A \text{ y } w \notin B\}$.

[5] Denominado diferencia simétrica.

[6] Se supone a lo largo de este capítulo que todos los conjuntos que intervienen son elementos del correspondiente σ-álgebra.

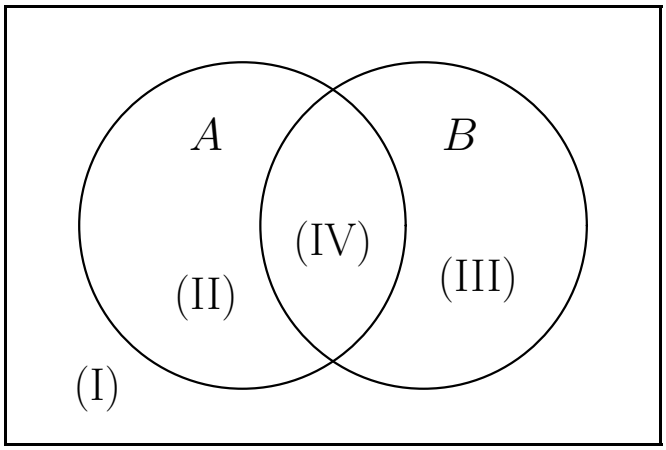

Figura 7.1: Partición del espacio muestral Ω en cuatro conjuntos disjuntos. (I)=$\overline{A \bigcup B}$, (II)=$A \setminus B$, (III)=$B \setminus A$ y (IV)=$A \bigcap B$.

Así pues, utilizando la medida de probabilidad y teniendo en cuenta que son sucesos disjuntos, se tendrá una primera función de similitud:

$$S_1 : \mathcal{A} \times \mathcal{A} \to \mathbb{R}$$

definida como:

$$S_1(A, B) = P(A \setminus B) + P(B \setminus A). \tag{7.1}$$

Las principales características que presenta esta función son:

- $0 \leq S_1(A, B) \leq 1$.
  Esto es evidente ya que la probabilidad de un suceso esta definida entre 0 y 1.

- $S_1(A, B) = S_1(B, A)$ (simetría).
  Evidente a partir de la definición de la función S_1.

- $S_1(A, A) = 0$ para todo $A \in \mathcal{A}$.
  $S_1(A, A) = P(\emptyset) + P(\emptyset) = 0 + 0 = 0$ y puesto que el suceso que más se parece a si mismo es él, se sigue que la máxima similitud se da cuando dos sucesos

A y B coinciden[7]. Este resultado nos indica que en realidad la función S_1 más que medir similitudes cuantifica disimilitudes ya que no sería lógico que la similitud de un mismo suceso tome el menor valor posible. A pesar de ello, seguimos viendo las propiedades que tiene puesto que posteriormente las necesitaremos.

- $S_1(A, \overline{A}) = 1$ para todo $A \in \mathcal{A}$.
  $S_1(A, \overline{A}) = P(A) + P(\overline{A}) = 1$ y puesto que podemos entender que $\overline{A}$ es el suceso más distinto (disimilar o no similar) a A, se sigue que la mínima similitud se da entre un suceso A y su complementario $B = \overline{A}$. En este punto, habría que realizar el mismo comentario que en la propiedad anterior.

- $S_1(A, \overline{B}) = 1 - S_1(A, B)$.
  Teniendo en cuenta que $A \setminus B = A \bigcap \overline{B}$ se sigue que: $S_1(A, \overline{B}) = P\left(A \setminus \overline{B}\right) + P\left(\overline{B} \setminus \overline{A}\right) = P(A \bigcap B) + P(\overline{A} \bigcap \overline{B}) = P(A \bigcap B) + P(\overline{A \bigcup B}) = \cdots$
  de la construcción de la partición dada en la figura 7.1 se cumple
  $\cdots = 1 - P(A \setminus B) - P(B \setminus A) = 1 - S_1(A, B)$.

- $S_1(\overline{A}, \overline{B}) = S_1(A, B)$.
  Evidente sin más que aplicar dos veces el resultado anterior.

Como ya hemos indicado, esta primera cuantificación de la similitud presenta el inconveniente de asignar como valor de máxima similitud el cero y como valor de mínima similitud el uno, lo que no parece lógico desde ningún punto de vista. Así, con objeto de evitar que la máxima similitud tome un valor numérico inferior a la mínima similitud consideramos una segunda función de similitud:

$$S_2 : \mathcal{A} \times \mathcal{A} \rightarrow \mathbb{R}$$

definida como:

$$S_2(A, B) = 1 - S_1(A, B) = P\left(\overline{A \bigcup B}\right) + P\left(A \bigcap B\right) \tag{7.2}$$

[7]Salvo en un conjunto de medida nula. A partir de ahora no se volverá a especificar pero debe estar presente en todos los posteriores razonamientos que, la igualdad entre sucesos se entiende entre clases de sucesos con la misma probabilidad.

que en términos de sucesos, no sería más que considerar el suceso complementario a la diferencia simétrica.

La función S_2, teniendo en cuenta que $P(A \setminus B) = P(A) - P(A \bigcap B)$, se puede expresar en la forma:

$$S_2(A, B) = 1 - P(A) - P(B) + 2\,P(A \bigcap B)$$

expresión que resulta más adecuada en algunos tratamientos algebraicos.

A partir de las propiedades obtenidas para la función S_1 se tiene que:

- $0 \leq S_2(A, B) \leq 1$.
- $S_2(A, A) = 1$.
  De donde se sigue que la máxima similitud se da cuando los sucesos A y B coinciden, y de esta forma se soluciona el problema de incoherencia anteriormente planteado.
- $S_2(A, \overline{A}) = 0$.
  De donde se sigue que la mínima similitud se da cuando $B = \overline{A}$.
- $S_2(A, B) = S_2(B, A)$ (simetría).
- $S_2(A, \overline{B}) = 1 - S_2(A, B)$
  Este resultado nos indica que fijado un suceso $A \in \mathcal{A}$, la máxima similitud que es la unidad, se la reparten entre un suceso B y su complementario $\overline{B}$, es decir, lo que le falta a B para llegar a la unidad lo tiene su complementario.
- $S_2(\overline{A}, \overline{B}) = S_2(A, B)$.

Conviene destacar una propiedad muy importante de la función de similitud $S_2(A, B)$, que es la siguiente:

Lema 7.1.1 *La función $S_2(A, B)$ es una función núcleo definido positivo (núcleo de Mercer).*

Demostración. Para realizar la demostración seguimos la notación dada en el lema 6.7.5.

Sean $A_1, A_2, \cdots, A_n \in \mathcal{A}$ y denotamos por $p_i = P(A_i)$ y $p_{ij} = P(A_i \bigcap A_j)$. Consideramos las variables aleatorias $X_i = I_{A_i}$ y del lema 6.7.5 se sigue que: $Cov(X_i, X_j) = p_{ij} - p_i \cdot p_j$ y a partir de esta igualdad, se sigue que:

$$\begin{aligned} G_{ij} &= S_2(A_i, A_j) = 1 - p_i - p_j + 2\,p_{ij} \\ &= 1 - p_i - p_j + 2\,Cov(X_i, X_j) + 2\,p_i\,p_j \\ &= 1 - p_i - p_j + p_i\,p_j + 2\,Cov(X_i, X_j) + p_i\,p_j \\ &= (1 - p_i) \cdot (1 - p_j) + 2\,Cov(X_i, X_j) + p_i\,p_j. \end{aligned}$$

Si utilizamos notación matricial, $\mathbf{G} = \{G_{ij}\}_{i,j=1}^n$, $\mathbf{K} = \{Cov(X_i, X_j)\}_{i,j=1}^n$ y $\mathbf{p}' = (p_1, p_2, \cdots, p_n)$, entonces:

$$\mathbf{G} = (\mathbf{1} - \mathbf{p}) \cdot (\mathbf{1} - \mathbf{p})' + \mathbf{p} \cdot \mathbf{p}' + 2\,\mathbf{K}.$$

Sea $\mathbf{c}' = (c_1, c_2, \cdots, c_n) \in \mathbb{R}^n$. Entonces

$$\begin{aligned} \mathbf{c}'\mathbf{G}\,\mathbf{c} &= \mathbf{c}'\,(\mathbf{1} - \mathbf{p}) \cdot (\mathbf{1} - \mathbf{p})'\,\mathbf{c} + \mathbf{c}'\,\mathbf{p} \cdot \mathbf{p}'\,\mathbf{c} + 2\,\mathbf{c}'\,\mathbf{K}\,\mathbf{c} \\ &= \|\mathbf{c}'\,(\mathbf{1} - \mathbf{p})\|^2 + \|\mathbf{c}'\,\mathbf{p}\|^2 + 2\,\mathbf{c}'\,\mathbf{K}\,\mathbf{c}, \end{aligned}$$

donde $\|\cdot\|$ es la norma euclídea; y como $\mathbf{c}'\,\mathbf{K}\,\mathbf{c} \geq 0$, se tiene que para cualquier $\mathbf{c} \in \mathbb{R}^n$ se cumple que $\mathbf{c}'\,\mathbf{G}\,\mathbf{c} \geq 0$ y por tanto la función S_2 es un núcleo definido positivo. ■

Sin embargo, podemos enumerar una serie de inconvenientes que presenta esta función de similitud:

1. $S_2(A, \emptyset) = P(\overline{A})$ ¿Cómo se interpretaría este resultado? No tendría sentido decir que si $P(A) = 0'25$ la similitud entre el suceso A y el suceso imposible es igual a $0'75$ y entre el suceso $\overline{A}$ y el suceso imposible es igual a $0'25$ ¿por qué se parece más a uno que a otro?

2. $S_2(A, \Omega) = P(A)$ ¿Cómo se interpreta? Se sigue un razonamiento similar al anterior.

3. Si A y B son independientes, entonces se tiene que:

$$\begin{aligned} S_2(A,B) &= P(\overline{A \bigcup B}) + P(A \bigcap B) = P(\overline{A} \bigcap \overline{B}) + P(A \bigcap B) \\ &= P(\overline{A})\, P(\overline{B}) + P(A)\, P(B), \end{aligned}$$

y, salvo el caso extremo de $A = \emptyset$ y $B = \Omega$, se sigue que $S_2(A,B) \neq 0$, pero ¿cómo se interpreta?

Los anteriores inconvenientes tienen un nexo común y es que la función S_2 no tiene un elemento neutro que nos permita entender cuando dos sucesos pasan de ser similares a ser disimilares (no similares) y al contrario.

7.1.1 Análisis gráfico de la función S_2

Con objeto de tener una representación gráfica que, en cierta forma, aclare las ideas anteriores proponemos el siguiente esquema gráfico, recogido en la figura 7.2 que pasamos a construir.

Fijado un suceso $A \in \mathcal{A}$, consideramos la función:

$$S_{2A} : \mathcal{A} \rightarrow \mathbb{R}$$

definida como:

$$S_{2A}(B) = S_2(A,B) \qquad \forall\, B \in \mathcal{A}.$$

Veamos en que forma su recorrido recoge algunas características de los sucesos A y B. Esto es posible verlo gráficamente utilizando un pequeño truco ya que sin éste, no sería posible su representación gráfica sobre los ejes cartesianos puestos que no tratamos con puntos, sino con sucesos. Sin embargo, puesto que la función S_{2A} recoge las características del suceso B a través su probabilidad podemos considerar la función $S_A(P(\cdot)) = S_2(A,\cdot)$, y como $P(\cdot)$ esta definida en $[0,1]$ obtener una gráfica dentro del cuadrado $[0,1] \times [0,1]$. Veamos como sería esta gráfica.

Si consideramos la definición de $S_2(A,B)$ en la forma:

$$S_2(A,B) = 1 - P(A) - P(B) + 2\, P(A \bigcap B)$$

y denotamos por $p = P(A)$ entonces $S_{2A}(B) = 1 - p - P(B) + 2\,P(A \bigcap B)$. Para cualquier $B \in \mathcal{A}$, consideramos la siguiente descomposición: $B = B_1 \bigcup B_2$ donde $B_1 = A \bigcap B \subseteq A$, $B_2 = \overline{A} \bigcap B \subseteq \overline{A}$ y por construcción son disjuntos. De donde se tiene que:

- Si $P(B) = 0$ entonces $S_{2A}(B) = 1 - p - 0 + 0 = 1 - p$.
- Si $P(B) = x \leq p$, se sigue:

  $$S_{2A}(B) = 1 - p - P(B_1) - P(B_2) + 2\,P(B_1)$$

  y entre todos los sucesos B con $P(B) = x$, el máximo se alcanzará cuando $P(B_2) = 0$ y valdrá $\text{máx}_{P(B)=x}\, S_{2A}(B) = 1-p-x+2x = 1-p-x$ y se alcanza cuando el suceso[(8)] $B \subseteq A$.
- Si $P(B) = x > p$, se sigue:

  $$S_{2A}(B) = 1 - p + P(B_1) - P(B_2)$$

  y entre todos los sucesos B con $P(B) = x$, el máximo se alcanzará cuando $P(B_1)$ es máximo y $P(B_2)$ es mínimo y como se ha de cumplir que $P(B) = x = P(B_1) + P(B_2)$ esto se consigue si $B_1 = A$, y el máximo de $S_{2A}(B)$ valdrá $\text{máx}_{P(B)=x}\, S_{2A}(B) = 1 - p + p - (x - p) = 1 + p - x$ y se alcanza cuando el suceso B cumple: $A \subseteq B$.
- Si $P(B) = 1$ como caso particular del anterior se tiene que $S_{2A}(B) = p$.
- Si $P(B) = x \leq 1 - p$, se sigue:

  $$\begin{aligned} S_{2A}(B) &= 1 - p - P(B_1) - P(B_2) + 2\,P(B_1) \\ &= 1 - p + P(B_1) - P(B_2) \end{aligned}$$

  y entre todos los sucesos B con $P(B) = x$, el mínimo se alcanzará cuando $P(B_1) = 0$ y valdrá $\text{mín}_{P(B)=x}\, S_{2A}(B) = 1-p+0-x = 1-p-x$ y se alcanza cuando el suceso B cumple que $B \subseteq \overline{A}$.

[(8)] Salvo conjunto de medida nula.

- Si $P(B) = x \geq 1 - p$, se sigue:

$$S_{2A}(B) = 1 - p + P(B_1) - P(B_2)$$

y entre todos los sucesos B con $P(B) = x$, el mínimo se alcanzará cuando $P(B_1)$ es mínimo y $P(B_2)$ es máximo y como se ha de cumplir que $P(B) = x = P(B_1) + P(B_2)$ esto se consigue si $B_2 = \overline{A}$, y el mínimo de $S_{2A}(B)$ valdrá $\text{mín}_{P(B)=x} S_{2A}(B) = x - (1 - p)$ y se alcanza cuando el suceso B cumple que $\overline{A} \subseteq B$.

- Si $P(B) = x$ y B es un suceso independiente de A entonces

$$S_{2A}(B) = 1 - p - x + 2\,p \cdot x = (1 - p) + (2\,p - 1)\,x.$$

De esta forma, la representación gráfica del dominio y recorrido de la función de similitud $S_{2A}(P(B))$ es la área del paralelogramo que aparece representado en la figura 7.2.

Como casos extremos del gráfico 7.2, se sigue que si $A = \emptyset$, el dominio y recorrido de $S_{2\emptyset}(\cdot)$ es el segmento que une los puntos $(0, 1)$ y $(1, 0)$; y si $A = \Omega$, el dominio y recorrido de $S_{2\Omega}(\cdot)$ es el segmento que une los puntos $(0, 0)$ y $(1, 1)$.

Nótese como el el área delimitada depende claramente de la probabilidad del suceso A, y es igual a $2\,P(A) \cdot (1 - P(A))$.

7.1.2 Otra medida de similitud

Hemos visto, que para cada $A \in \mathcal{A}$ se da la máxima similitud $S_2(A, A) = 1$, si bien, es cierto que, esto refleja una cuantificación clara de la idea intuitiva de similitud, presenta el inconveniente de no reflejar la "complejidad" del suceso A medida en términos de su probabilidad. Para aclarar este punto, podemos pensar en comparar organismos unicelulares entre sí y en comparar organismos pluricelulares entre si. Claramente son igualmente similares entre ellos, pero no es menos cierto, que entre los organismos pluricelulares es posible encontrar una cantidad mucho mayor de

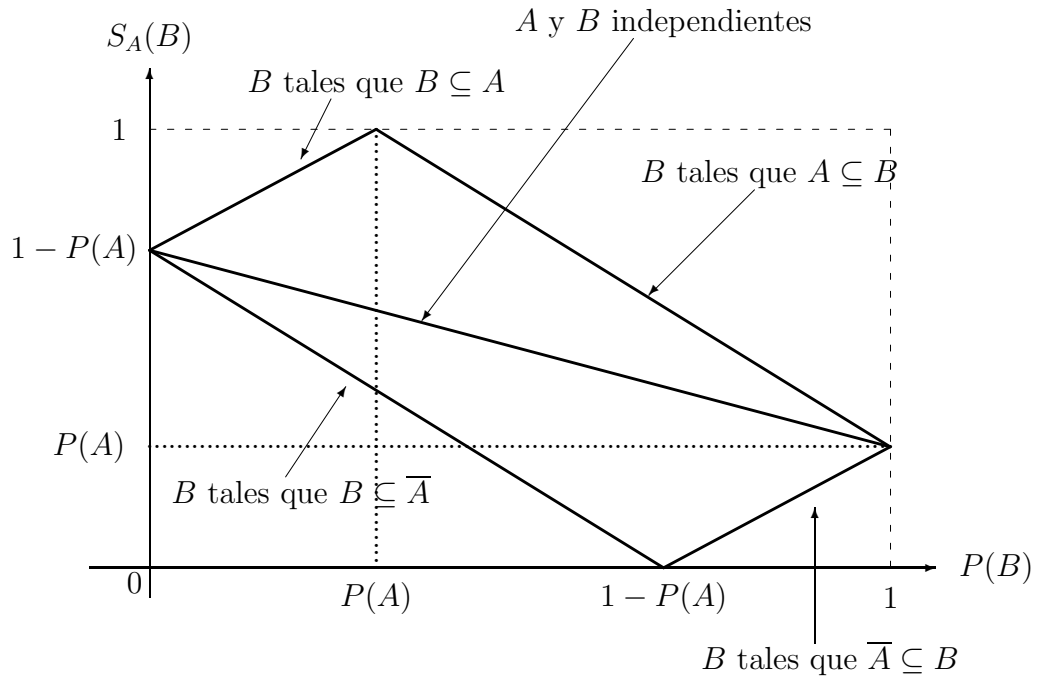

Figura 7.2: La zona delimitada por el paralelogramo representa el dominio y recorrido de la función $S_{2A}(\cdot)$, con A un suceso fijado tal que $0 < P(A) < 1$.

características que nos permitan comparar entre ellos. Por ello, pensamos que sería conveniente buscar una función de similitud que tenga en cuenta este hecho, es decir, que tenga en cuenta el tamaño relativo (en este caso la probabilidad) de los sucesos cuando se comparan con ellos mismos.

Para conseguir tener una función de similitud con esta característica, podemos realizar el siguiente argumento, a partir de un espacio muestral finito $\Omega = \{w_1, \cdots, w_n\}$, con objeto de tener una interpretación más clara e intuitiva. Con la finalidad de tener una medida de similitud entre dos conjuntos A y B, que denotaremos $D(A, B)$, hemos de observar el número de elementos comunes de $A \bigcap B$ y los que no están en ninguno de los dos, $\overline{A \bigcup B}$. Si observamos la figura 7.3 se puede entender que, si las flechas determinan la forma de los conjuntos, cuanto mayor sea

el conjunto de puntos comunes se fuerza a que los conjuntos se parezcan más, y por otro lado cuanto más sean los puntos no comunes a ninguno de ellos, más se parecen los conjuntos. Por tanto si tratamos con un mismo suceso podemos considerar como medida de similitud de un conjunto consigo mismo, la siguiente[(9)]:

$$D(A, A) = \text{Nº elementos de } A \cdot \text{Nº elementos de } \overline{A}.$$

Así pues si:

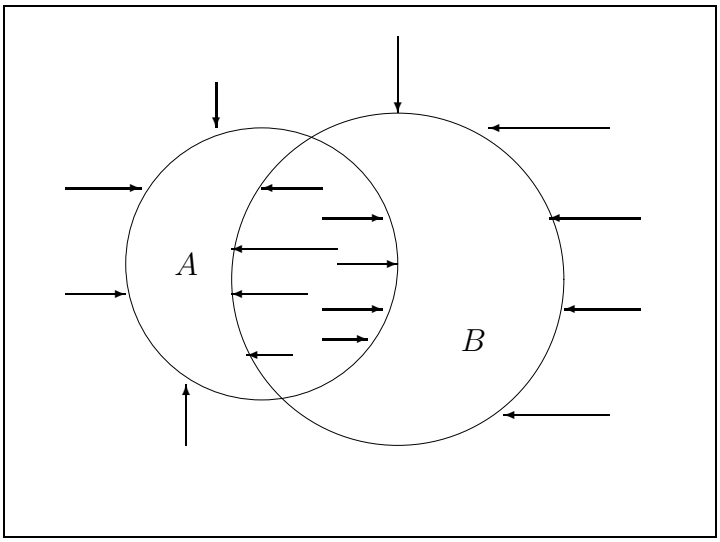

Figura 7.3: Interpretación gráfica de una medida de similitud entre un suceso y otro, la cual tiene en cuenta el tamaño relativo del suceso A.

- $A_0 = \emptyset$ entonces $D(A_0, A_0) = 0 \cdot n$,
- $A_1 = \{w_1\}$ entonces $D(A_1, A_1) = 1 \cdot (n-1)$,
- ....
- $A_k = \{w_1, \cdots, w_k\}$ entonces $D(A_k, A_k) = k \cdot (n-k)$,

[(9)] Claramente si consideramos la suma tenemos la función S_2.

- ....
- $A_{n-1} = \{w_1, \cdots, w_{n-1}\}$ entonces $D(A_{n-1}, A_{n-1}) = (n-1) \cdot 1$,
- $A_n = \Omega = \{w_1, \cdots, w_n\}$ entonces $D(A_n, A_n) = n \cdot 0$.

Si en lugar de considerar elementos consideramos probabilidades no cambian los resultados anteriores y por tanto se podría generalizar la función D en la forma[10]

$$D(A, A) = P(A) \cdot P(\overline{A}).$$

Con esta nueva función resolvemos el problema de no tener en cuenta la "complejidad" de los sucesos ya que en este caso se tiene que si $P(A) = 1/2$, entonces $S_3(A, A) = 1/2 \cdot 1/2 = 1/4$ y si $P(A) = 1/3$, se tiene que $S_3(A, A) = 1/3 \cdot 2/3 = 2/9$.

También, podemos razonar de la siguiente forma: como la función $S_2(A, B)$ viene definida a partir de la probabilidad de $A \bigcap B$ y la probabilidad de $\overline{A \bigcup B}$ como una suma, podemos combinar estas probabilidades utilizando otro tipo de operación, por ejemplo[11]:

$$S_3(A, B) = P\left(\overline{A \bigcup B}\right) \cdot P\left(A \bigcap B\right). \tag{7.3}$$

Por otro lado, esta función además cumple que: $0 \leq P\left(A \bigcap B\right) \leq P(A)$ ya que $A \bigcap B \subseteq A$ y $0 \leq P\left(\overline{A \bigcup B}\right) \leq P(\overline{A})$ puesto que $\overline{A \bigcup B} \subseteq \overline{A}$ luego se sigue que[12] $0 \leq S_3(A, B) \leq S_3(A, A) \leq 1/4$ y se sigue cumpliendo que el suceso más similar con A es el propio A, lo cual no lleva a ningún tipo de inconsistencia de similitudes.

Con objeto de obtener una interpretación visual del comportamiento de esta nueva función de similitud $S_3(A, B)$, fijamos un suceso A y realizamos un tratamiento similar al dado para la función S_{2A} a la función $S_{3A}(B) = S_3(A, B)$. Veamos como sería el dominio y recorrido de esta función.

[10] Nótese que esta función coincide con la varianza de la variable aleatoria $X = I_A$.

[11] En orden de jerarquía de operaciones sería la inmediatamente superior a la operación suma.

[12] La última desigualdad se demostrará posteriormente.

En primer lugar, se tiene que:

$$\begin{aligned} S_3(A,B) &= P\left(\overline{A \bigcup B}\right) \cdot P\left(A \bigcap B\right) \\ &= \left(1 - P(A) - P(B) + P\left(A \bigcap B\right)\right) \cdot P\left(A \bigcap B\right). \end{aligned}$$

Si denotamos por $p = P(A)$, y al igual que anteriormente $B = B_1 \bigcup B_2$ con $B_1 \subseteq A$ y $B_2 \subseteq \overline{A}$ se tiene que:

- Si $P(B) = 0$ entonces $S_{3A}(B) = 0$.
- Si $P(B) = x \leq p$ se sigue que

  $$S_{3A}(B) = (1 - p - P(B_2)) \cdot P(B_1)$$

  y será máximo cuando $P(B_2)$ sea mínimo y $P(B_1)$ sea máximo, lo cual se consigue con $P(B_2) = 0$ y $P(B_1) = x$, y el máximo valdrá $\text{máx}_{P(B)=x}\, S_{3A}(B) = x \cdot (1 - p)$ y se alcanza cuando el suceso $B \subseteq A$.
- Si $P(B) = x \geq p$ se sigue que

  $$S_{3A}(B) = (1 - p - P(B_2)) \cdot P(B_1)$$

  y será máximo cuando $P(B_2)$ sea mínimo y $P(B_1)$ sea máximo, lo cual se consigue con $P(B_1) = p$ y $P(B_2) = x - p$, y el máximo valdrá $\text{máx}_{P(B)=x}\, S_{3A}(B) = p \cdot (1 - p)$ y se alcanza cuando el suceso B tal que $A \subseteq B$.
- Si $P(B) = 1$ entonces $P\left(\overline{A \bigcup B}\right) = 0$ y de aquí $S_{3A}(B) = 0$.

De estos desarrollos se sigue que el dominio y recorrido de la función de similitud S_{3A} es el área encerrada por el triángulo que aparece reflejado en la figura 7.4.

Sin embargo, esta función de similitud presenta un grave inconveniente ya que para todos los sucesos A y B tales que $A \bigcap B = \emptyset$ se sigue $S_3(A,B) = 0$, con lo cual limita el conjunto de sucesos de trabajo a aquellos con intersección no vacía. Por ejemplo, dado tres sucesos A, B, $B' \in \mathcal{A}$ con $A \bigcap B = A \bigcap B' = \emptyset$ lo único que podemos decir es que la similitud de A con B y de A con B' es nula, y no

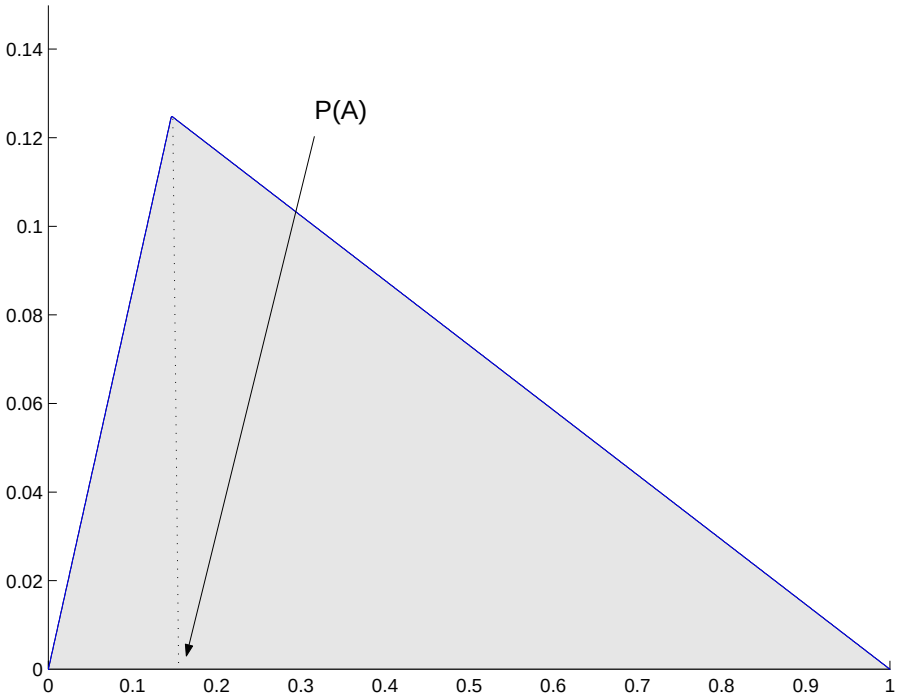

Figura 7.4: Visualización del recorrido de la función $S_{3A}(\cdot)$ para el caso $P(A) = 3/20$.

podemos cuantificar si A se parece más a B o a B', ya que no se tiene en cuenta otras características de los dos sucesos.

Por otro lado, para aquellos sucesos que son independientes, se cumple que:

$$\begin{aligned} S_3(A,B) &= P(A)\cdot P(\overline{A})\cdot P(B)\cdot P(\overline{B}) \\ &= p\cdot(1-p)\cdot x\cdot(1-x) \end{aligned}$$

cuya representación gráfica es una parábola de vértice $(\frac{1}{2}, \frac{1}{4}\,p\,(1-p))$ y pasa por $(0,0)$ y $(1,0)$. Esta gráfica dentro de la figura 7.4 no nos aporta una mejor interpretación de la función S_3 y por ello no se ha representado.

7.2 Función núcleo similitud

Con objeto de evitar los inconvenientes presentados por las anteriores funciones de similitud, proponemos una nueva función de similitud que, como veremos, mejora a las anteriores, además de proporcionar un conjunto de nuevas características que la hacen más versátil y potente.

Si consideramos la figura 7.2, observamos que las similitudes medidas por la función S_2 son todas positivas, lo cual no nos permite señalar cuando existe disimilitud entre sucesos. Sin embargo, si consideramos como eje de simetría la recta que determinan los sucesos independientes, podríamos decir que los que están por encima presentan similitudes positivas y los que están por debajo presentan similitudes negativas. Por ello, teniendo en cuenta que para un suceso A con probabilidad p, y un suceso B con probabilidad x, la recta de independencia, en función de x, es:

$$\text{Independencia:} \qquad y = (1-p) + x\,(2p-1);$$

y la función de similitud $S_2(\cdot,\cdot)$ es:

$$S_2(A,B) = 1 - p - x + 2\,P(A \bigcap B)$$

se tiene que una nueva medida de similitud es:

$$\begin{aligned} k(A,B) &= 1 - p - x + 2\,P(A \bigcap B) - ((1-p) + x\,(2p-1)) \\ &= 1 - p - x + 2\,P(A \bigcap B) - 1 + p - 2\,p\,x + x \\ &= 2\,(P(A \bigcap B) - p\,x) \\ &= 2\,(P(A \bigcap B) - P(A)\,P(B)); \end{aligned}$$

y estamos en condiciones de definir la siguiente función que cuantifica similitudes:

Definición 7.2.1 (función núcleo similitud) *Sea* $(\Omega, \mathcal{A}, P)$ *un espacio probabilístico. Se define la función núcleo similitud:*

$$k : \mathcal{A} \times \mathcal{A} \rightarrow I\!R$$

como sigue:

$$k(A,B) = P\left(A \bigcap B\right) - P(A) \cdot P(B) \tag{7.4}$$

para todo $A, B \in \mathcal{A}$.

De la conclusion del lema 6.7.5 se sigue que $k(A,B)$ es una función que viene dada a partir de un producto escalar en un determinado espacio[13], es decir, existe un espacio característico $\mathcal{F}$ tal que la función $k(A,B) = \langle \phi(A), \phi(B) \rangle_{\mathcal{F}}$ donde ϕ es una aplicación definida desde la σ-álgebra $\mathcal{A}$ en $\mathcal{F}$ (ver figura 7.5). Permitiendo de esta forma, a partir del producto escalar definido en $\mathcal{F}$, establecer similitudes entre los sucesos del σ-álgebra.

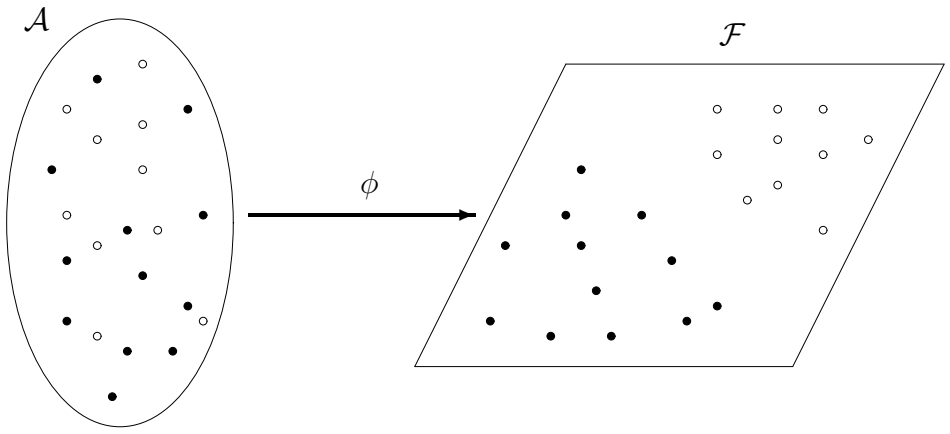

Figura 7.5: Función ϕ que nos permite incrustar el σ-algebra $\mathcal{A}$ dentro de un espacio característico $\mathcal{F}$ dotado de un producto escalar, en él cual podemos establecer una medida de similitud entre los sucesos.

[13] A partir del lema, se podría especificar con más detalle cual es el espacio característico, el producto escalar definido en él, e incluso cual es la función ϕ.

7.2.1 Propiedades de la función núcleo similitud

A continuación, y a lo largo de esta sección, se estudia la interpretación de la función k, en términos de cuantificadora de similitudes, así como las propiedades más importante que presenta.

1.- Si $P(A) = 0$ entonces $k(A, B) = 0$ para todo $B \in \mathcal{A}$.
La demostración es evidente sin más que tener en cuenta que $A \bigcap B \subseteq A$. Inicialmente, no podemos dar una interpretación de esta situación, pero quedará completamente justificada con la propiedad 12.

2.- Si $P(A) = 1$ entonces $k(A, B) = 0$ para todo $B \in \mathcal{A}$.
La demostración es evidente sin más que tener en cuenta que $P(A \bigcap B) = P(B)$. Igual que la propiedad anterior, inicialmente no podemos dar una interpretación, pero ésta podrá ser dada con la propiedad 11.

3.- Sean B_1 y $B_2 \in \mathcal{A}$ tales que $B_1 \bigcap B_2 = \emptyset$ (sucesos disjuntos) entonces para todo $A \in \mathcal{A}$:

$$k\left(A, B_1 \bigcup B_2\right) = k(A, B_1) + k(A, B_2) \tag{7.5}$$

Demostración:

$$\begin{aligned} k(A, B_1 \bigcup B_2) &= P(A \bigcap (B_1 \bigcup B_2)) - P(A)\, P(B_1 \bigcup B_2) \\ &= P(A \bigcap B_1) + P(A \bigcap B_2) - P(A)\,(P(B_1) + P(B_2)) \\ &= k(A, B_1) + k(A, B_2). \end{aligned}$$

Claramente este resultado se puede generalizar para una colección numerable de sucesos disjuntos dos a dos, ya que nos encontramos dentro de un σ-álgebra. Por tanto, sean $B_1, B_2, \cdots \in \mathcal{A}$ tales que $B_i \bigcap B_j = \emptyset$ para todo $i,\ j = 1, 2, \cdots$ con $i \neq j$ entonces[14]:

$$k\left(A, \bigcup_{i=1}^{\infty} B_i\right) = \sum_{i=1}^{\infty} k(A, B_i).$$

$\diamondsuit$

[14]Por la naturaleza de la función k, no plantea problema la convergencia de la serie.

4.- Sea $A \subseteq B$ entonces:

$$k(A, B) = k(A, A) - P(A)\, P(B \setminus A) \tag{7.6}$$

Demostración: De la definición de los sucesos se tiene que $A \bigcap (B \setminus A) = \emptyset$, luego $k(A, B) = k(A, A) + k(A, B \setminus A)$, y $k(A, B \setminus A) = -P(A) \cdot P(B \setminus A)$ y de ambas igualdades se sigue (7.6).

$\diamondsuit$

En esta propiedad lo que estamos viendo es como se comporta la función k cuando uno de los sucesos esta contenido en el otro, $A \subseteq B$. Una de las consecuencias que se sigue de la igualdad (7.6) es que si el suceso $B \setminus A$ es de probabilidad nula entonces $k(A, B) = k(A, A)$ para todo $A \in \mathcal{A}$.
Estos resultados nos permiten realizar la siguiente interpretación cuando $A \subseteq B$:
Habrá menos similitud entre dos sucesos A y B cuanto mayor sea la probabilidad de $B \setminus A$. Si en lugar de un espacio muestral Ω, consideramos un conjunto finito e interpretamos en términos de número de elementos en cada conjunto, esta condición nos indica que los conjuntos A y B con $A \subseteq B$ serán tanto más similares cuantos menos elementos tenga B que no están en A (ver figura 7.6).

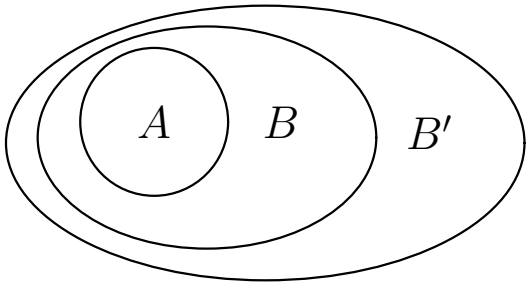

Figura 7.6: Conjuntos anidados. Intuitivamente se declararía que los conjuntos A y B son más similares que los conjuntos A y B'.

5.- Sea $A \subseteq B \subseteq B'$ entonces

$$0 \leq k(A, B') \leq k(A, B) \leq k(A, A), \tag{7.7}$$

Demostración: Si consideramos cualquier suceso $B \in \mathcal{A}$ tal que $A \subseteq B$ entonces

$$k(A,B) = P(A) - P(A) \cdot P(B) = P(A) \cdot (1 - P(B)) \geq 0$$

y junto con (7.6) se sigue que si $A \subseteq B \subseteq B'$ entonces, $P(B \setminus A) \leq P(B' \setminus A)$, con lo que $k(A,B') \leq k(A,B)$ y se tiene el resultado (7.7).

$\diamondsuit$

6.- Para todo suceso $A \in \mathcal{A}$ fijado se verifica:

$$\max_{\{B:\, A \subseteq B\}} k(A,B) = k(A,A).$$

Demostración: Evidente a partir de los resultados anteriores.

$\diamondsuit$

Por otro lado como $0 \leq P(A) \leq 1$,

$$k(A,A) = P(A)P(\overline{A}) = P(A)\,(1 - P(A))$$

pero como la función $f(x) = x(1-x)$ con $0 \leq x \leq 1$ tiene un máximo absoluto en el punto $x_0 = 1/2$; y $f(x_0) = 1/4$, se tiene que

$$\max_{A \in \mathcal{A}} k(A,A) \leq 1/4$$

con lo que el valor máximo de la función k es menor o igual que $1/4$ y se alcanza esta cota si existen sucesos $A \in \mathcal{A}$ tales que $P(A) = 1/2$. Nótese como se llega a la misma interpretación de similitud de un suceso consigo mismo dada en la página 235.

7.- Si A y B son sucesos independientes entonces $k(A,B) = 0$
Demostración: De la independencia se sigue que $P(A \bigcap B) = P(A) \cdot P(B)$ y de aquí que $k(A,B) = 0$.

$\diamondsuit$

En este caso, la interpretación que resulta se expresa diciendo que la similitud entre dos sucesos independientes es nula, es decir no tienen ninguna similitud en común. Puesto que las características de los sucesos vienen en términos de sus probabilidades, la ocurrencia de un suceso B independiente a otro A no nos permite comparación entre ellos ni en términos de similitud ni de no similitud, de ahí que la función k cuantifica la similitud entre ellos como nula.

Estudiamos ahora los casos en los que los sucesos A y B son disjuntos.

8.- Sea un suceso $B \subseteq \overline{A}$ entonces

$$k(A, B) = -k(A, A) + P(A) \cdot P(\overline{A} \setminus B).$$

Demostración: Si A y B son sucesos disjuntos entonces como $A \bigcap B = \emptyset$ se tiene que $P(A \bigcap B) = 0$ luego $k(A, B) = -P(A)\, P(B)$ y como consecuencia de la positividad de las probabilidades: $k(A, B) \leq 0$.
La interpretación de este resultado sería que ya que dos sucesos disjuntos no tienen elementos en común presentan una similitud negativa, que como venimos indicando denominamos disimilitud o no similitud.
Si se considera un suceso $B \subseteq \overline{A}$ entonces a partir de los sucesos A, B y $\overline{A} \setminus B$ se puede expresar $\overline{A} = B \bigcup (\overline{A} \setminus B)$ y de aquí:

$$\begin{aligned} k(A, B) &= -P(A)\, P(B) \\ &= -P(A)\, (P(\overline{A}) - P(\overline{A} \setminus B)) \\ &= -P(A)\, P(\overline{A}) + P(A)\, P(\overline{A} \setminus B) \\ &= -k(A, A) + P(A)\, P(\overline{A} \setminus B). \end{aligned}$$

$\diamondsuit$

9.- Para todo suceso A se tiene $k(A, \overline{A}) = -k(A, A)$.
Demostración: Si se considera $B = \overline{A}$ entonces $\overline{A} \setminus B = \emptyset$ y $P(\overline{A} \setminus B) = 0$ luego se sigue la igualdad.

$\diamondsuit$

10.- Sea $B \subseteq B' \subseteq \overline{A}$ entonces

$$k(A, \overline{A}) \leq k(A, B') \leq k(A, B) \leq 0. \tag{7.8}$$

Demostración: Si $B \subseteq B' \subseteq \overline{A}$ entonces como $(\overline{A} \setminus B') \subset (\overline{A} \setminus B)$, entonces $P(\overline{A} \setminus B') \leq P(\overline{A} \setminus B)$ y se sigue (7.8).

$\diamondsuit$

La interpretación, a la vista de la figura 7.7, en términos de conjunto es la siguiente: Como B' tiene más elementos que no están en A que B, su similitud es mayor en valor absoluto. Otra interpretación sería: A y B son más similares que A y B' ya que entre A y B hay menos posibilidades de discriminación.

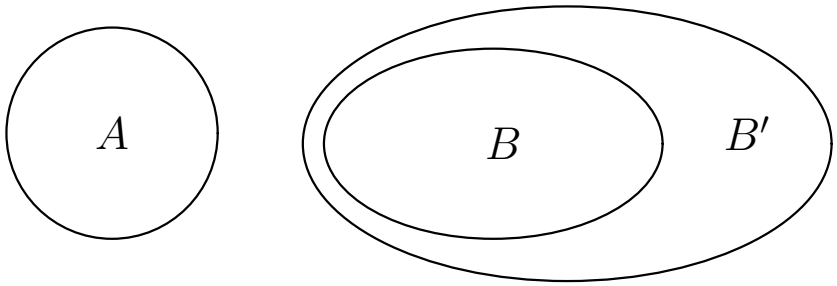

Figura 7.7: A la vista de estos conjuntos intuitivamente se declararía que los sucesos A y B son más similares que los sucesos A y B'.

Por otro lado, como $P(A) \cdot P(\overline{A} \setminus B) \geq 0$ para cualesquiera dos sucesos se tiene que para $A \in \mathcal{A}$ fijado

$$\min_{\{B:\, B \subseteq \overline{A}\}} k(A, B) = k(A, \overline{A}).$$

Por simetría con lo anteriormente apuntado se sigue que

$$\min_{A \in \mathcal{A}} k(A, \overline{A}) \geq -1/4$$

y la cota se alcanza si existe $A \in \mathcal{A}$ tal que $P(A) = 1/2$.

11.- $k(A, \Omega) = 0$ para todo $A \in \mathcal{A}$.

La demostración es evidente, ya que dentro de los $B \supseteq A$ el que menos se parece a A es Ω. Lo cual nos proporciona una interpretación de la similitud entre un suceso y el suceso seguro.

12.- $k(A, \emptyset) = 0$ para todo $A \in \mathcal{A}$.

La demostración es evidente, ya que dentro de los $B \subseteq A$ el que menos se parece a A es $\emptyset$. Lo cual nos proporciona una interpretación de la similitud entre un suceso y el suceso imposible.

13.- Para cualesquiera dos sucesos $A, B \in \mathcal{A}$ se cumple:

$$k(A, \overline{A}) \leq k(A, B) \leq k(A, A). \tag{7.9}$$

Demostración:

Dados $A, B \in \mathcal{A}$ expresamos B en la forma $B = (A \bigcap B) \bigcup (B \setminus A) = B_1 \bigcup B_2$, donde $B_1 = A \bigcap B \subseteq A$ y $B_2 = B \setminus A \subseteq \overline{A}$.

Aplicando los resultados anteriores se sigue que:

$$\begin{aligned} k(A, B) &= k(A, B_1) + k(A, B_2) \leq k(A, B_1) \\ &\leq k(A, A) \end{aligned} \quad \text{ya que } k(A, B_2) \leq 0.$$

$$\begin{aligned} k(A, B) &= k(A, B_1) + k(A, B_2) \geq k(A, B_2) \\ &\geq k(A, \overline{A}) \end{aligned} \quad \text{ya que } k(A, B_1) \geq 0.$$

$\diamondsuit$

Este resultado nos permite realizar la siguiente interpretación: El suceso que más similitud tiene con A es el propio A y el suceso que menos similitud tiene con A es $\overline{A}$, interpretación que resulta la más razonable de todas las posibles.

14.- Para todo $A, B \in \mathcal{A}$:

$$\text{máx}\left\{k(A, \overline{A}), k(B, \overline{B})\right\} \leq k(A, B) \leq \text{mín}\left\{k(A, A), k(B, B)\right\}.$$

La demostración es trivial a partir de las desigualdades (7.9).

15.- $k(A, \overline{B}) = -k(A, B)$, para todo $A,\ B \in \mathcal{A}$.

Demostración:

$0 = k(A, \Omega) = k(A, B \bigcup \overline{B}) = k(A, B) + k(A, \overline{B})$ luego $k(A, \overline{B}) = -k(A, B)$

$\diamondsuit$

Interpretación: Si consideramos que B y $\overline{B}$ son totalmente disimilares, entonces sería lógico que la similitud entre A y B sea opuesta a la de A y $\overline{B}$.

16.- $k(\overline{A}, \overline{B}) = k(A, B)$, para todo $A,\ B \in \mathcal{A}$.

La demostración es consecuencia inmediata de la propiedad anterior, sin más que aplicar ésta dos veces.

17.- Para todo $A,\ B \in \mathcal{A}$ se sigue:

$$|k(A, B)| \leq \text{mín}\left\{k(A, A), k(B, B)\right\}. \tag{7.10}$$

La demostración es consecuencia inmediata de las propiedades anteriores.

Nota 7.2.2 *Nótese, que si en lugar de considerar el núcleo k se toma $k' = 4\,k$ se tendría que*

$$-1 \leq k'(A, B) \leq 1, \qquad \forall A, B \in \mathcal{A}$$

y su interpretación sería muy parecida a la interpretación dada para el coeficiente de correlación lineal:

- *Si $k'(A, B) = 0$, los sucesos son independientes o alguno de los dos sucesos tiene probabilidad nula o alguno de los dos sucesos tiene probabilidad uno.*
- *Si $k'(A, B) = 1$, los sucesos presentan máxima similitud (atracción).*
- *Si $k'(A, B) = -1$, los sucesos presentan máxima disimilitud (repulsión).*

▲

7.2.2 Representación gráfica de la función $k_A(B)$

Fijamos un suceso A, y como se hizo con la funciones de similitudes anteriores, con objeto de obtener una representación gráfica, consideramos la función definida en $[0,1]$: $k_A(P(B)) = k(A,B)$. Veamos como se sigue la representación gráfica de esta función.

Si denotamos por $p = P(A)$ entonces $k_A(B) = P(A \bigcap B) - p \cdot P(B)$. Consideramos la descomposición de un suceso $B = B_1 \bigcup B_2$ donde $B_1 = A \bigcap B \subseteq A$, $B_2 = \overline{A} \bigcap B \subseteq \overline{A}$ y por construcción son disjuntos. De donde se tiene que:

- Si $P(B) = 0$ ya hemos visto que $k_A(B) = 0$.
- Si $P(B) = x \leq p$, se sigue:

$$\begin{aligned} k_A(B) &= k_A(B_1 \bigcup B_2) = k_A(B_1) + k_A(B_2) = P(B_1) - p\,P(B_1) - p\,P(B_2) \\ &= (1-p) \cdot P(B_1) - p\,P(B_2) \end{aligned}$$

  y entre todos los sucesos B con $P(B) = x$, el máximo se alcanzará cuando $P(B_2) = 0$ y valdrá $\text{máx}_{P(B)=x}\, k_A(B) = (1-p) \cdot x$ y se alcanza cuando el suceso[(15)] $B \subseteq A$.

- Si $P(B) = x > p$ se sigue:

$$k_A(B) = (1-p) \cdot P(B_1) - p\,P(B_2)$$

  y entre todos los sucesos B con $P(B) = x$, el máximo se alcanzará cuando $P(B_1)$ es máximo y $P(B_2)$ es mínimo y como se ha de cumplir que $P(B) = x = P(B_1) + P(B_2)$ esto se consigue si $B_1 = A$, y el máximo de $k_A(B)$ valdrá $\text{máx}_{P(B)=x}\, k_A(B) = (1-p) \cdot p - p \cdot (x-p) = p \cdot (1-x)$ y se alcanza cuando el suceso B cumple: $A \subseteq B$.

- Si $P(B) = 1$ como caso particular del anterior se tiene que $k_A(B) = 0$.

[(15)] Salvo conjunto de medida nula

- Si $P(B) = x \leq 1 - p$, se sigue que como $k_A(B) = -k_A(\overline{B})$ entonces $\text{mín}_{\{B/\, P(B)=x\}}\, k_A(B) = -\,\text{máx}_{\left\{\overline{B}/\, P(\overline{B})=1-x\right\}}\, k_A(\overline{B}) = -p\,x$ ya que en este caso $P(\overline{B}) = 1 - x \geq p$ y sustituyendo para este caso, en el desarrollo anterior se tiene que el máximo se da en $p\,(1-(1-x))$ y se tiene cuando $A \subseteq \overline{B}$. De esta forma, el mínimo se tiene cuando el suceso $B \subseteq \overline{A}$.

- Si $P(B) = x > 1 - p$, se sigue que como $k_A(B) = -k_A(\overline{B})$ entonces $\text{mín}_{\{B/\, P(B)=x\}}\, k_A(B) = -\,\text{máx}_{\left\{\overline{B}/\, P(\overline{B})=1-x\right\}}\, k_A(\overline{B}) = -(1-p)\,(1-x)$ ya que en este caso $P(\overline{B}) = 1 - x \leq p$ y sustituyendo para este caso, en el desarrollo anterior se tiene que el máximo se da en $(1-p)\,(1-x)$ y se tiene cuando $\overline{B} \subseteq A$. De esta forma, el mínimo se tiene cuando el suceso $\overline{A} \subseteq B$.

De esta forma, la representación gráfica del dominio y recorrido de la función $k_A(P(B))$ es el área encerrada por el paralelogramo que aparece representado en la figura 7.8. Nótese el gran parecido con la figura 7.2, ya que presentan la misma forma y los límites coinciden con respecto, a cuales son, los sucesos que delimitan el recinto. Además las áreas entre ambas figuras son proporcionales ya que el área delimitada por el paralelogramo de la figura 7.2 es $2\,p\,(1-p)$ y él de la figura 7.8 es $p\,(1-p)$.

Si nos centramos en la representación gráfica de la función núcleo similitud, una pregunta que puede surgir cuando se observa la figura 7.8 es la siguiente: ¿Conociendo las similitudes $k(A,B)$ y $k(A,B')$ podemos realizar alguna afirmación sobre la similitud $k(B,B')$?, es decir, existe alguna forma de transitividad entre las similitudes. La respuesta es que no, y para ello consideramos dos situaciones. Si $A = \emptyset$ y $B = B'$ con $P(B) = 1/2$ entonces $k(A,B) = k(A,B') = 0$ y sin embargo $k(B,B') = 1/4$ (máxima similitud). Por otro lado si tomamos $A = \emptyset$ y $B = \overline{B'}$ con $P(B) = 1/2$ entonces se sigue que $k(A,B) = k(A,B') = 0$ y sin embargo $k(B,B') = -1/4$ (mínima similitud).

Cuando se estudiaron las propiedades de la función núcleo similitud, se vio como se comportaba esta función respecto a distintas formas de los conjuntos A, B y B'. Existen otras posibilidades de combinar los conjuntos, como por ejemplo las

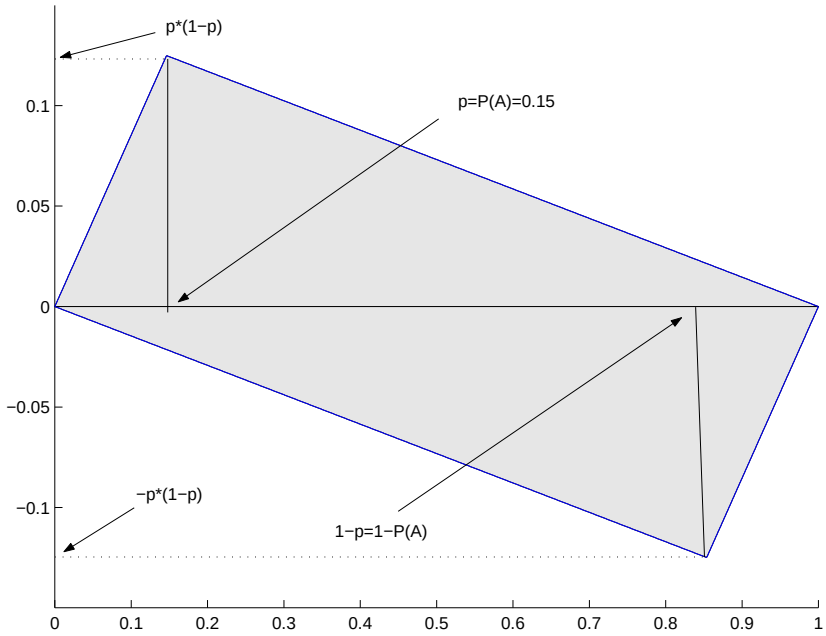

Figura 7.8: La región encerrada entre las dos funciones representan el dominio y recorrido de la función $k_A(P(B))$, cuando se fija un suceso A.

dadas en la figura 7.9. Estos casos son igualmente interesantes, sin embargo, no es posible realizar ninguna afirmación sobre sus similitudes como ya comentamos en el apartado anterior. Esto se ve claro si damos otra expresión de la función núcleo similitud.

Sean los sucesos A y B. Sobre B realizamos la descomposición:

$$B = B_1 \bigcup B_2, \qquad B_1 \bigcap B_2 = \emptyset, \qquad B_1 = A \bigcap B \subseteq A, \qquad B_2 = \overline{A} \bigcap B \subseteq \overline{A}.$$

De esta forma se tiene que la función núcleo similitud queda:

$$\begin{aligned} k(A,B) &= P(A \bigcap B) - P(A)\,P(B) \\ &= P(B_1) - P(A)\,(P(B_1) + P(B_2)) \\ &= P(B_1)(1 - P(A)) - P(A)\,P(B_2) \\ &= P(B_1)\,P(\overline{A}) - P(A)\,P(B_2) \end{aligned}$$

De esta expresión de la función núcleo similitud se tiene claramente que el estudio de

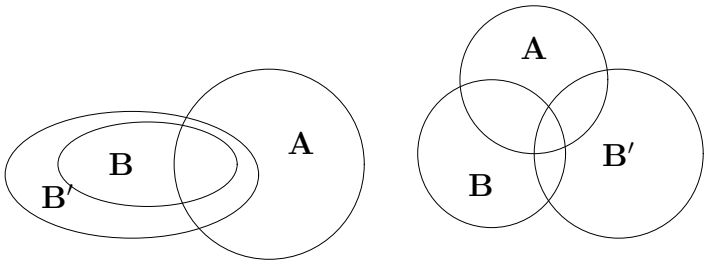

Figura 7.9: Dos ejemplos de posibilidades que se pueden presentar entre los conjuntos A, B y B'

las situaciones presentadas en la figura 7.9, no se pueden abordar de forma general, ya que existen una infinidad de posibilidades de combinar los sucesos A, B y B' en función del tamaño de las intersecciones entre todos ellos.

Por todo ello, la única afirmación que se puede realizar de forma empírica es la siguiente: Si $P(A) \simeq P(B_1) \simeq P(B_2)$ y $k(A, B_1) \simeq k(A, B_2)$ entonces $k(A, B_1) \simeq k(A, B_2) \simeq k(B_1, B_2)$. Por ello, pensamos que sería una buena elección, si se está trabajando con un conjunto de sucesos, elegir como el suceso A de referencia en la función k_A aquel cuya diferencia con todas las restantes probabilidades sea mínima en el sentido de mínimos cuadrados[16], es decir, si tenemos $\{A_i\}_{i=1}^n$ y para cada A_i se considera

$$d(A_i) = \sum_{j=1}^{n} (P(A_i) - P(A_j))^2$$

se ha de elegir como A óptimo aquel que cumpla: $d(A) = \min_i d(A_i)$, con objeto que cuando sea posible se pueda aplicar la afirmación anterior.

[16]Podríamos haber elegido otro criterio, como por ejemplo utilizar valor absoluto en lugar de cuadrados, pero hemos considerado este más adecuado después de llevar a cabo algunos ensayos prácticos.

7.3 Ejemplo de cálculo de similitudes

Con objeto de aclarar los conceptos anteriores hemos seleccionado un ejemplo sencillo en el que se puedan seguir los cálculos implícitos en la función núcleo similitud. Consideramos el experimento aleatorio de lanzamiento de tres monedas y estudiamos las similitudes que se obtienen en los sucesos que tienen en cuenta la distribución de caras y cruces.

Sean los siguientes sucesos:
$A_i = \{\text{Se obtiene al menos } i \text{ caras}\}$, con $i = 0, 1, 2, 3$.
$B_j = \{\text{Se obtiene al menos } j \text{ cruces}\}$, con $j = 0, 1, 2, 3$.

Calculamos las similitudes que presentan los sucesos A_k:

Si los lanzamientos son experimentos independientes y suponemos que la probabilidad de obtener cara es igual a p en las tres monedas entonces la variable aleatoria X definida como "Número de caras", sigue un modelo Binomial de parámetros $(3, p)$, luego

$$P(A_i) = \sum_{j=i}^{3} \left(\begin{array}{c} 3 \\ j \end{array} \right) p^j \, (1-p)^{3-j}$$

y claramente

$$P\left(A_i \bigcap A_{i'}\right) = P(A_i)$$

si $i' \leq i$ ya que entonces $A_i \subseteq A_{i'}$.

Si las tres monedas están perfectamente equilibradas, entonces la probabilidad de obtener cara es $p = 1/2$ y se tiene la siguiente tabla de probabilidades:

i	0	1	2	3
$P(A_i)$	1	7/8	1/2	1/8

A partir de esta tabla, el cálculo de similitudes es fácil ya que, por ejemplo, el

cálculo de $k(A_2, A_2)$ es como sigue:

$$\begin{aligned} k(A_2, A_2) &= P(A_2 \bigcap A_2) - P(A_2)\, P(A_2) \\ &= P(A_2) - P(A_2)^2 \\ &= \left(\frac{1}{2}\right) - \left(\frac{1}{2}\right)^2 = \frac{1}{4} \end{aligned}$$

y el cálculo de $k(A_2, A_3)$ es:

$$\begin{aligned} k(A_2, A_3) &= P(A_2 \bigcap A_3) - P(A_2)\, P(A_3) \\ &= P(A_3) - P(A_2)\, P(A_3) \\ &= \left(\frac{1}{8}\right) - \frac{1}{2}\,\frac{1}{8} = \frac{1}{16}. \end{aligned}$$

Realizando todos los cálculos se obtiene la siguiente tabla de similitudes entre los sucesos A_k:

$64 \cdot k(A_i, A_j)$	A_0	A_1	A_2	A_3
A_0	0	0	0	0
A_1	0	7	4	1
A_2	0	4	16	4
A_3	0	1	4	7

Observamos claramente, que la tabla obtenida es simétrica y además el suceso más similar consigo mismo es A_2, donde se da la máxima similitud ($k(A_2, A_2) = 1/4$) puesto que $P(A_2) = 1/2$. Entre dos sucesos distintos se tiene que los sucesos A_2 y A_3 son tan similares como los sucesos A_1 y A_2, que presentan una mayor similitud que los sucesos A_1 y A_3 lo cual intuitivamente resulta correcto, puesto que las diferencias en el número de caras entre A_1 y A_2 es menor que entre A_1 y A_3. También, de la tabla, se sigue que la similitud entre A_1 y A_2 es igual a la de A_2 y A_3, es decir, la características que los diferencia, que es el número de caras, queda perfectamente recogida por la función núcleo similitud, puesto que si la diferencia entre caras es dos, la similitud es 1/64 y si la diferencia entre las caras es uno, la similitud es mayor e igual 4/64.

Tabla 7.1: Similitudes entre los sucesos A_i =$\{$Se obtienen al menos i-caras en el lanzamiento de tres monedas$\}$ y los sucesos B_j =$\{$Se obtienen al menos j-caras en el lanzamiento de tres monedas$\}$.

$64\,k(A_i, B_j)$	A_0	A_1	A_2	A_3
B_0	0	0	0	0
B_1	0	-1	-4	-7
B_2	0	-4	-16	-4
B_3	0	-7	-4	-1

Calculamos las similitudes conjuntas entre los sucesos A_i y B_j. Comenzamos con los sucesos A_2 y B_3:

$$\begin{aligned} k(A_2, B_3) &= P(A_2 \bigcap B_3) - P(A_2)\,P(B_3) \\ &= P(\{\text{Al menos 2 caras y al menos tres cruces }\}) - P(A_2)\,P(B_3) \\ &= P(\{\emptyset\}) - P(A_2)\,P(B_3) \\ &= 0 - \frac{1}{2}\cdot\frac{1}{8} = -\frac{1}{16} \end{aligned}$$

observando como se obtiene una similitud negativa entre estos dos sucesos. Calculamos la similitud entre A_2 y B_1:

$$\begin{aligned} k(A_2, B_1) &= P(A_2 \bigcap B_1) - P(A_2)\,P(B_1) \\ &= P(\{2 \text{ caras y } 1 \text{ cruz}\}) - P(A_2)P(B_3) \\ &= \frac{3}{8} - \frac{1}{2}\cdot\frac{7}{8} = -\frac{1}{16}. \end{aligned}$$

Realizando todos los cálculos se tiene la tabla 7.1 de similitudes entre los sucesos A_i y B_j.

Observemos también, que los sucesos más disimilares son los sucesos A_2 y B_2 donde se da la máxima disimilitud ($k(A_2, B_2) = -1/4$); y que son tan disimilares los sucesos A_2 y B_1 como los sucesos A_3 y B_2, es decir, la característica que cuantifica la similitud esta estrechamente relacionada con la diferencia entre los índices i y j y esta queda recogida por la función núcleo similitud.

7.4 Función núcleo similitud y función de distribución.

Sea $(\Omega, \mathcal{A}, P)$ un espacio probabilístico y $X_i : \Omega \rightarrow \mathbb{R}$ con $i = 1, \cdots, m$ un conjunto de variables aleatorias asociadas al σ-álgebra $\mathcal{A}$, las cuales si la expresamos en forma de vector, $\mathbf{X} = (X_1, \cdots, X_m)$ nos determina un vector aleatorio m dimensional. Es conocido que en los desarrollos tanto teóricos como prácticos del cálculo de probabilidades, es común buscar variables aleatorias que nos permitan estudiar algunas características de los sucesos de la σ-álgebra $\mathcal{A}$, pudiendo en tales casos hacer uso de todo el aparato matemático que se encuentra disponible en Análisis Real y Complejo de funciones. Así, a partir de esta construcción de variables aleatorias, se está en condiciones de utilizar los conceptos de integrabilidad, diferenciabilidad, paso al límite, etc...

Dado un vector aleatorio $\mathbf{X}$, m dimensional sabemos que su distribución de probabilidades queda completamente identificada a partir de su función de distribución conjunta $F(\mathbf{x})$, la cual se define como:

$$F : \mathbb{R}^m \rightarrow [0, 1]$$

tal que

$$F(x_1, x_2, \cdots, x_m) = P\left(X_1 \leq x_1, X_2 \leq x_2, \cdots, X_m \leq x_m\right)$$

para todo $(x_1, x_2, \cdots, x_m) \in \mathbb{R}^m$. En esta sección, establecemos una relación entre ambas funciones: la función de distribución y la función núcleo similitud.

Sea un vector aleatorio bidimensional (X, Y) asociado a un espacio probabilístico $(\Omega, \mathcal{A}, P)$. Si consideramos los sucesos: $A = (-\infty, x]$ y $B = (-\infty, y]$ para cualesquiera x e $y \in \mathbb{R}$, entonces:

$$\begin{aligned} k(A, B) &= P\left(A \bigcap B\right) - P(A) \cdot P(B) \\ &= P(X \leq x, Y \leq y) - P(X \leq x) \cdot P(Y \leq y) \\ &= F_{(X,Y)}(x, y) - F_X(x) \cdot F_Y(y) \end{aligned}$$

y como tanto A como B quedan determinados a partir de x e y, entonces se tiene que la función núcleo similitud puede expresarse en la forma:

$$k(x,y) = F_{(X,Y)}(x,y) - F_X(x) \cdot F_Y(y).$$

Obteniendo la condición de independencia entre dos variables aleatorias. Por otro lado, si consideramos que el vector aleatorio $\mathbf{X}$ es absolutamente continuo, entonces se sigue que:

$$\frac{\partial k(x,y)}{\partial x\, \partial y} = f_{(X,Y)}(x,y) - f_X(x) \cdot f_Y(y)$$

y se tiene una nueva relación entre la función núcleo similitud y las funciones de densidades conjuntas y marginales.

Veamos a continuación un ejemplo, en el que se calcula de manera explícita una función núcleo similitud en términos de unas determinadas variables aleatorias. Aprovechamos para plantear y resolver un problema, a partir de un ejemplo, que dentro de la Economía de la Empresa puede ser interesante como es comparar la similitud entre situaciones acaecidas en distintas empresas. Nótese que, en este problema, lo realmente complicado, no sería el cálculo de similitudes, sino estudiar un problema fundamental de la Estadística, como es la estimación de las funciones de densidad.

Ejemplo 7.1 *Sea un oligopolio, constituido por tres empresas de las cuales pretendemos estudiar como se comportan sus cuotas de mercado. Para ello consideramos las siguientes variables aleatorias*[17]*:*
X= Cuota de mercado de la empresa A,
Y= Cuota de mercado de la empresa B,
Z= Cuota de mercado de la empresa C,
(todas expresadas en tantos por uno) donde la función de densidad conjunta que liga las variables es:

$$f(x,y,z) = \frac{180}{19}\, z\, (x+2y)\, (5-x-2y)$$

[17] Nótese como se generaliza a un vector m-dimensional y no tenemos que restringirnos a un vector aleatorio bidimensional.

donde $0 \leq x, y, z \leq 1$, $0 \leq x + y + z \leq 1$.

Si la empresa A pretende obtener una cuota de mercado entre $[0'4, 0'5]$*, teniendo en cuenta las estructuras de las empresas (expresada en términos de la función de distribución conjunta) para la empresa B ¿qué resultado sería más similar al resultado de la empresa A, obtener entre* $[0'4, 0'5]$, $[0'3, 0'4]$ *ó* $[0'2, 0'3]$ *de la cuota de mercado?*

Para resolver este problema aplicando la medida de similitud expresada a través de la función $k(x, y)$*, en primer lugar se calculan las funciones de densidades marginales de* X *e* Y*,*

$$f_X(x) = \frac{1}{60}(x-1)^3(4x^2 - 23x - 21)$$
$$f_Y(y) = \frac{1}{120}(y-1)^3(62x^2 - 159x - 23).$$

A continuación calculamos,

$$P[0'4 \leq X \leq 0'5] = \int_{0'4}^{0'5} \frac{1}{60}(x-1)^3(4x^2 - 23x - 21)\,dx = 0'00851304$$

y los restantes cálculos llevan a los siguientes resultados:

$$\begin{aligned} k(0'4 \leq X \leq 0'5, 0'4 \leq Y \leq 0'5) &= 0'0102758 \\ k(0'4 \leq X \leq 0'5, 0'3 \leq Y \leq 0'4) &= 0'0338548 \\ k(0'4 \leq X \leq 0'5, 0'2 \leq Y \leq 0'.3) &= 0'0651678 \end{aligned}$$

Por tanto se concluye que el resultado más similar al de la empresa A, de los tres expuestos es que la cuota de mercado de la empresa B este comprendido entre $[0'2, 0'3]$. ▲

7.5 Aprendizaje en la Red

El objeto de esta sección es utilizar la función núcleo similitud para poder desarrollar una herramienta que permita estudiar la evolución, tanto transversal como longitudinal, que dentro de las páginas Web de la red Internet siguen las diferentes líneas de investigación relativas a un determinado tema.

Tabla 7.2: Relación de las nueve líneas de investigación relacionadas con la Teoría del Aprendizaje por nosotros consideradas.

Items	Temas de Investigación
A_1	Reinforcement Learning
A_2	Learning for Information Retrieval
A_3	Automated Learning
A_4	Automated Discovery
A_5	Hybrid Systems, Neural and Symbolic Processing
A_6	Support Vector Machine, Decision Trees and Decision Trees Graphics
A_7	Machine Learning
A_8	Neural Network
A_9	Data Mining

Tanto las máquinas de vectores soporte como las funciones núcleos forman parte de las líneas de investigación que se enmarcan dentro de lo que se conoce como Teoría del Aprendizaje. Así, en esta sección, estudiamos las similitudes existente dentro de nueve líneas de investigación abiertas relacionadas con la Teoría del Aprendizaje, utilizando para ello la información que nos proporciona a través de la red Internet el buscador Altavista.

Presentamos en la tabla 7.2, las nueve líneas de investigación[(18)] (items) por nosotros consideradas, que denotaremos por A_i, $i = 1, 2, \cdots, 9$. En la mayoría de los buscadores de páginas Web, por ejemplo Altavista, Google, Yahoo, ... existen bases de datos de históricos donde se recogen las direcciones de las páginas por años así como los identificadores de cada una de esas páginas. Dentro de un proyecto de investigación[(19)], se ha elaborado un programa informático que a partir de un conjunto de identificadores, rastrea dentro de estas bases de datos y cuantifica el

[(18)] Nótese como aparece dos temas tratados en este trabajo como son: Support Vector Machine y Machine Learning.

[(19)] En el cual participo.

Tabla 7.3: Número de citas en las que aparecen recogidas algunas de las nueve líneas de investigación relacionadas con la Teoría del Aprendizaje en el año 2000 dentro de las bases de datos del buscador Altavista.

	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9
A_1	1	0	0	0	0	0	0	0	0
A_2	0	75	1	5	2	13	60	26	24
A_3	0	1	679	16	6	95	185	90	268
A_4	0	5	16	1434	24	144	457	210	442
A_5	0	2	6	24	11417	162	490	829	490
A_6	0	13	93	141	161	16984	6271	4474	5372
A_7	0	60	185	457	490	6271	41001	7581	9604
A_8	0	26	90	210	829	4474	7581	85706	6566
A_9	0	24	268	442	490	5372	9604	6566	122970

número de veces que los identificadores aparecen por año. De esta forma, se ha realizado una búsqueda, en las bases de datos del buscador Altavista, del número de páginas donde aparezcan referenciados a la vez dos items concretos en el año 2000; y hemos denotado los números de enlaces (frecuencias absolutas) por n_{ij}, con $i, j = 1, 2, \cdots, 9$.

Después de realizar la búsqueda se han encontrado un total de $N = 250000$ páginas donde al menos aparece un item, obteniendo la tabla 7.3 referente a los n_{ij}. En estos datos, es posible observar como en algunos casos varían ligeramente n_{ij} de n_{ji}, por ejemplo $n_{63} = 93$ y $n_{36} = 95$, lo cual no tiene sentido puesto que por construcción $n_{ij} = n_{ji}$ (simetría). Pensamos que esta diferencia es debida a que las consultas para obtener n_{ij} y n_{ji} se han realizado en distintos instantes de tiempo, lo que puede ocasionar que algunas de las bases de datos estén momentáneamente fuera de servicio o saturadas[(20)]. Con objeto de evitar el problema, de no disponer de

[(20)] Claramente, este problema no está en nuestras manos solucionarlo, lo que si sería adecuado es repetir la misma búsqueda en diferentes instantes de tiempo y tomar como valor de referencia el

Tabla 7.4: Cuantificación de las similitudes encontradas entre las nueve líneas de investigación relacionadas con la Teoría del Aprendizaje en el año 2000 dentro de las bases de datos del buscador Altavista.

	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9
A_1	0.0000	-0.0000	-0.0000	-0.0000	-0.0000	-0.0000	-0.0000	-0.0000	-0.0000
A_2	-0.0000	0.0003	0.0000	0.0000	-0.0000	0.0000	0.0002	0.0000	-0.0001
A_3	-0.0000	0.0000	0.0027	0.0000	-0.0001	0.0002	0.0003	-0.0006	-0.0003
A_4	-0.0000	0.0000	0.0000	0.0057	-0.0002	0.0002	0.0009	-0.0011	-0.0011
A_5	-0.0000	-0.0000	-0.0001	-0.0002	0.0436	-0.0025	-0.0055	-0.0123	-0.0205
A_6	-0.0000	0.0000	0.0002	0.0002	-0.0025	0.0633	0.0139	-0.0054	-0.0119
A_7	-0.0000	0.0002	0.0003	0.0009	-0.0055	0.0139	0.1371	-0.0259	-0.0423
A_8	-0.0000	0.0000	-0.0006	-0.0011	-0.0123	-0.0054	-0.0259	0.2253	-0.1424
A_9	-0.0000	-0.0001	-0.0003	-0.0011	-0.0205	-0.0119	-0.0423	-0.1424	0.2499

una matriz simétrica, hemos tomado la decisión de elegir como valor representativo el mayor entre n_{ij} y n_{ji} puesto que un valor promedio no tiene sentido ya que si una búsqueda encuentra, por ejemplo 1000 enlaces necesariamente se debe cumplir que el número de enlaces debe ser mayor o igual que 1000, en otras palabras, el número real de enlaces ha de ser siempre mayor o igual que n_{ij}, para todo i, j.

Con objeto de enlazar estos datos con la construcción de similitudes hemos tenido en cuenta, que el número de datos disponibles es suficientemente grande, y hemos optado por una interpretación frecuencial de la probabilidad. De esta forma, hemos supuesto que

$$P(A_i \bigcap A_j) = \frac{n_{ij}}{N}, \qquad i, j = 1, 2, \cdots, 9$$

con lo cual estamos en condiciones de aplicar la función núcleo similitud que hemos desarrollado en este capítulo. A partir de la tabla 7.3 hemos realizado los cálculos correspondientes a las similitudes de los items, obteniendo la tabla 7.4.

De esta forma disponemos de una tabla de orden 9×9, la cual si se desea estudiar

valor máximo obtenido.

la evolución concreta de dos items basta con entresacar de ella, los tres valores de referencia (k_{ij}, k_{ii} y k_{jj}). Sin embargo, si se desea tener una visión general del comportamiento de todas las similitudes es muy útil la representación gráfica de la función núcleo similitud dada en la sección 7.2.2 fijando un determinado item A_i.

Consideramos la función $k_A(B)$ con A óptimo en el sentido por nosotros propuesto en la sección 7.2.2, es decir, el item que minimice

$$d(A_i) = \sum_{j=1}^{9} \left(P(A_i) - P(A_j)\right)^2$$

para $i = 1, 2, \cdots, 9$. En este caso se obtiene que $A = A_7$ (Machine Learning), y representamos gráficamente esta función en los items correspondientes, obteniéndose la figura 7.10.

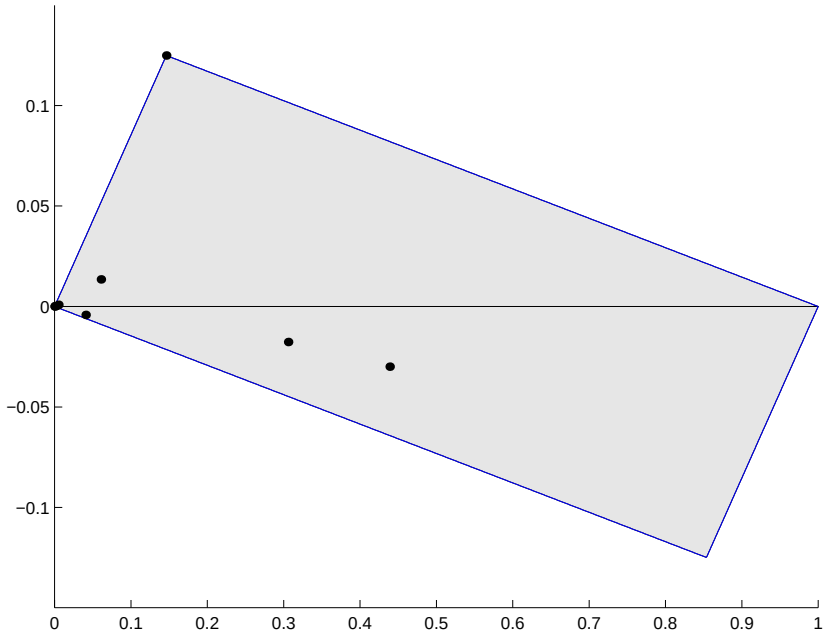

Figura 7.10: Representación gráfica de las similitudes dadas por $k_A(P(A_i))$ con $i = 1, 2, \cdots, 9$ y $A = A_7$ el item que proporciona el mínimo de $d(A_i) = \sum_{j=1}^{9} \left(P(A_i) - P(A_j)\right)^2$ dentro de las líneas de investigación abiertas en la Teoría de Aprendizaje.

A la vista de esta figura[21], se podría indicar que salvo A_6 que presenta una similitud pequeña y positiva, las restantes líneas de investigación presentan una similitud negativa, en especial los items A_5, A_8 y A_9. Todo ello nos permite declarar que no existe una alta similitud entre el suceso A_7 y todos los restantes, lo que significa que la línea de investigación "Machine Learning" presentaba en el año 2000 un conjunto de páginas Web que no tenían muchos nexos con las restantes páginas asociadas a las otras líneas de investigación.

Como ya indicamos anteriormente, con este gráfico solo es posible observar como se comportan los restantes items con respecto al que sirve de referencia. Por ello, si queremos realizar un análisis conjunto hemos de realizar tantas gráficas como items estemos estudiando.

Así, en la figura 7.11 podemos ver la posición de los valores $k_{A_i}(A_j)$ variando A_i desde $i = 1$ hasta 9, con respecto a todos los restantes items. Del estudio de estas gráficas una a una se tiene:

Gr1.- Todas las líneas de investigaciones presentan la máxima disimilitud con la línea "Reinforcement Learning", lo cual resulta evidente desde el primer momento puesto que el número de referencias de este item es casi nulo. En función del número de referencias y su relación con las anteriores se indicaría que es una línea de investigación muy poco desarrollada y apartada totalmente de los desarrollos de las restantes.

Gr2.- Este gráfico expresa las similitudes con la línea de investigación "Learning for Information Retrieval". Lo más destacable es la ausencia de similitud con las redes neuronales (A_8) y su alta similitud con la línea "Machine Learning" (A_7).

Gr3.- Lo más destacable de este gráfico es la alta disimilitud entre "Automated Learning" y las redes neuronales.

Gr4.- Este gráfico es muy similar al anterior.

[21] Para interpretar esta gráfica, y las siguientes, es necesario darse cuenta que los diferentes items se encuentran ordenados en función de sus respectivas probabilidades.

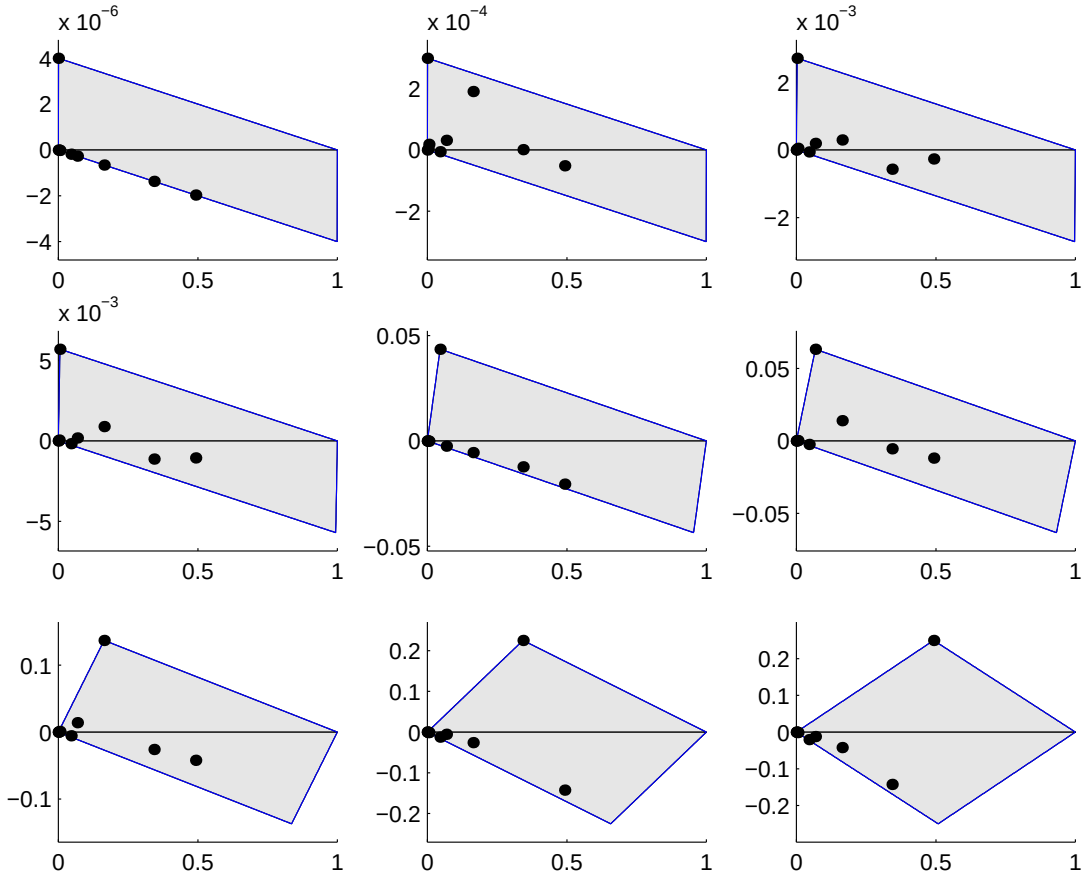

Figura 7.11: Representación gráfica de 9 gráficos donde se observan todas las similitudes dentro de las líneas de investigación abiertas en la Teoría de Aprendizaje. .

Gr5.- Todas las líneas de investigaciones presentan casi la máxima disimilitud con la línea "Hybrid Systems, Neural and Symbolic Processing", por ello, se indicaría que es una línea de investigación apartada totalmente de los desarrollos de las restantes.

Gr6.- La interpretación de este gráfico, referente a la línea de investigación "Support Vector Machine, Decision Trees and Decision Trees Graphics", es muy similar a la dada en el gráfico 3.

Gr7.- Línea "Machine Learning". Lo más resaltable es que presenta un grado inter-

medio de disimilitud con las otras dos grandes líneas de investigación (grandes en el sentido dado por sus probabilidades).

Gr8.- Lo más importante es que la "Neural Network" presenta una muy alta disimilitud con la línea "Data Mining".

Gr9.- Igual comentario que el anterior gráfico.

A partir de estas gráficas podemos concluir que, debido a la naturaleza, todavía joven, de estos estudios sobre Teoría del Aprendizaje, no existen dos líneas de investigación que presenten un alto grado de similitud en el sentido de llegar a tener un conjunto de identificadores comunes.

7.6 Representación gráfica de todas las similitudes en un único gráfico

El principal defecto que presenta la representación gráfica dada en la figura 7.11 es que necesita tantas gráficas como items se estudian. Evidentemente, sería mucho más práctico disponer de una única representación gráfica donde aparezcan recogida toda la información proporcionada por los gráficos anteriores. Para conseguir este fin, proponemos la siguiente solución.

En primer lugar consideramos los sucesos $\{A_1, A_2, \cdots, A_n\}$ ordenados a partir de sus probabilidades, es decir, $0 \leq P(A_1) \leq P(A_2) \leq \cdots \leq P(A_n)$, y se representan estas probabilidades sobre el eje de abscisas. Sobre el eje de ordenadas representamos las similitudes de la siguiente forma: Se toma el primer suceso A_1 y se representa el conjunto de similitudes $\{-k(A_1, A_1), k(A_1, A_2), \cdots, k(A_1, A_n), k(A_1, A_1)\}$ con abscisa $P(A_1)$, a continuación se toma el suceso A_2 y se representa el conjunto de similitudes $\{-k(A_2, A_2), k(A_2, A_3), \cdots, k(A_2, A_n), k(A_2, A_2)\}$ con abscisa $P(A_2)$, y así sucesivamente hasta el suceso A_n. La explicación del por qué se actúa de esta forma viene motivado por la desigualdad (7.10) que nos indica que para cualquier

par de sucesos se sigue:

$$|k(A_i, A_j)| \leq \text{mín}\left\{k(A_i, A_i), k(A_j, A_j)\right\}.$$

Si se tiene que $P(A_n) \leq 1/2$, de la ordenación de las probabilidades y del crecimiento de la función $f(x) = x\,(1-x)$ en $(0, 1/2)$, se deduce que

$$|k(A_i, A_j)| \leq k(A_i, A_i) \qquad \text{si } i < j$$

lo cual valida la construcción realizada. Por otro lado, si existe algún suceso con probabilidad superior a $1/2$ se trabajaría con su complementario y, gracias a las propiedades vista de la función núcleo similitud, de las conclusiones sobre el complementario se tendría la inicial.

La representación gráfica, siguiendo este proceso aplicada a las nueve líneas de investigación relacionadas con la Teoría del Aprendizaje, puede verse en la figura 7.12. En esta gráfica el símbolo $\times$ y el número al lado representa la similitud entre el item que aparece en abscisa y el item que se referencia con el número. Se observa que entre los items con mayor probabilidad (items 6, 7, 8 y 9) no existe mucha similitud, con lo que podemos concluir que estas líneas de investigación siguen caminos distintos con pocas interrelaciones. Por otro lado, nótese que, si se fija el item A_7, las similitudes $k(A_7, A_8)$ y $k(A_7, A_9)$ están próximas, pero sin embargo la similitud entre los items A_8 y A_9 es negativa y alta, lo cual asevera el comentario realizado en la página 248.

También, en la figura 7.12, y debido a que los items 1, 2, 3, 4 y 5 tienen probabilidades pequeñas, no se observan con nitidez, por ello si queremos verlo con más claridad hemos de realizar una ampliación dentro de la figura 7.12 como la que se tiene la figura 7.13. En este gráfico, al igual que ocurría con el anterior, la pequeñez de las probabilidades de los primeros items hace no visible su representación pero se puede ver que la similitud entre ellos es despreciable; y en consecuencia declarar que todas las líneas de investigación estudiadas dentro de la teoría del aprendizaje seguían caminos diferentes con pocos nexos de unión en el año 2000.

Sin embargo, ¿cómo se vería en los dos últimos gráficos que dos items son muy similares o muy disimilares? Para ello, si la similitud entre ellos es positiva, basta

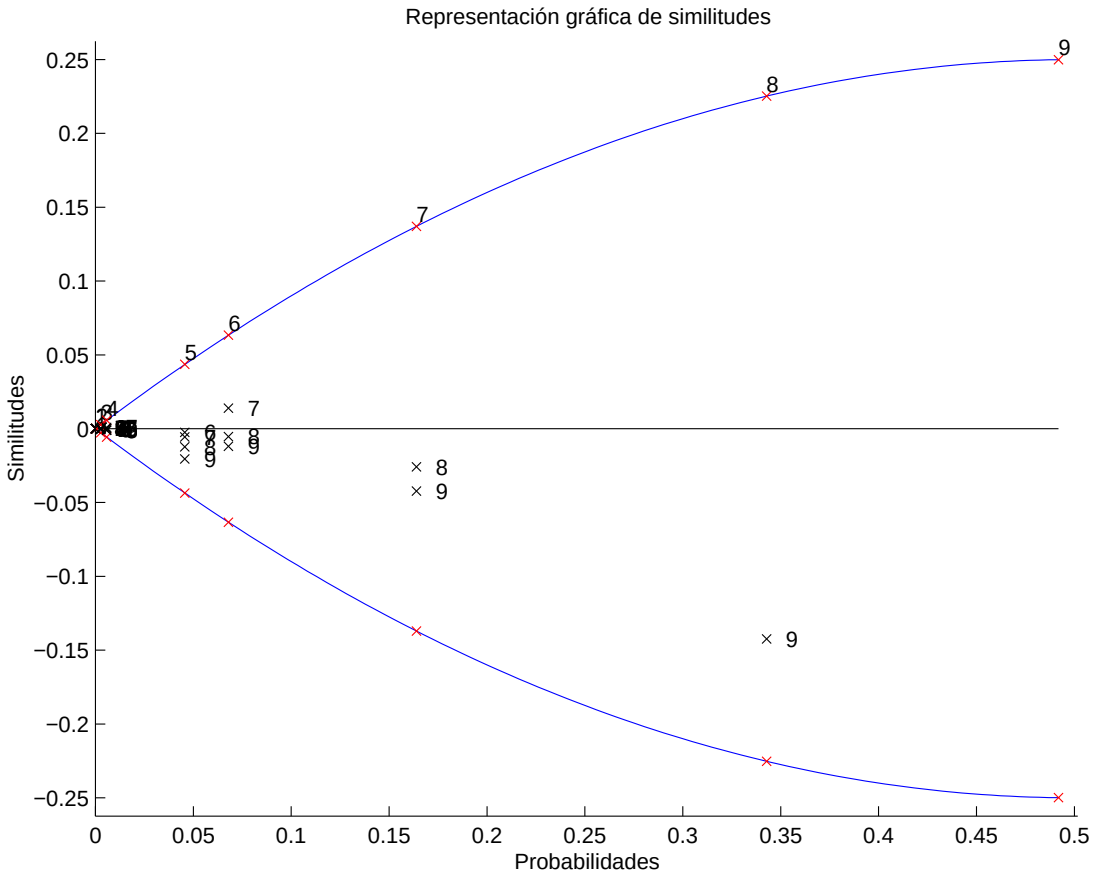

Figura 7.12: Representación gráfica de todas las similitudes, relacionadas con las líneas de investigación en la Teoría del Aprendizaje, en un único gráfico.

con dibujar el triángulo formado por $\Delta_1 = \{k(B,B),\ k(A,B),\ k(A,A)\}$, y si el área encerrada en el triángulo es pequeña entonces se tendría que son sucesos de tamaño parecido y similitud muy alta. De igual manera, si la similitud entre ellos es negativa, basta con dibujar el triángulo formado por $\Delta_2 = \{-k(B,B),\ k(A,B),\ -k(A,A)\}$, y si el área encerrada en el triángulo es pequeña, entonces se tendría que son sucesos de tamaño parecido y disimilitud muy alta. En general, a medida que el área encerrada por el triángulo Δ_1 (si la similitud conjunta es positiva) o el triángulo Δ_2 (si la similitud conjunta es negativa) es mayor, entonces los sucesos son más diferentes en tamaño y presentan menor similitud.

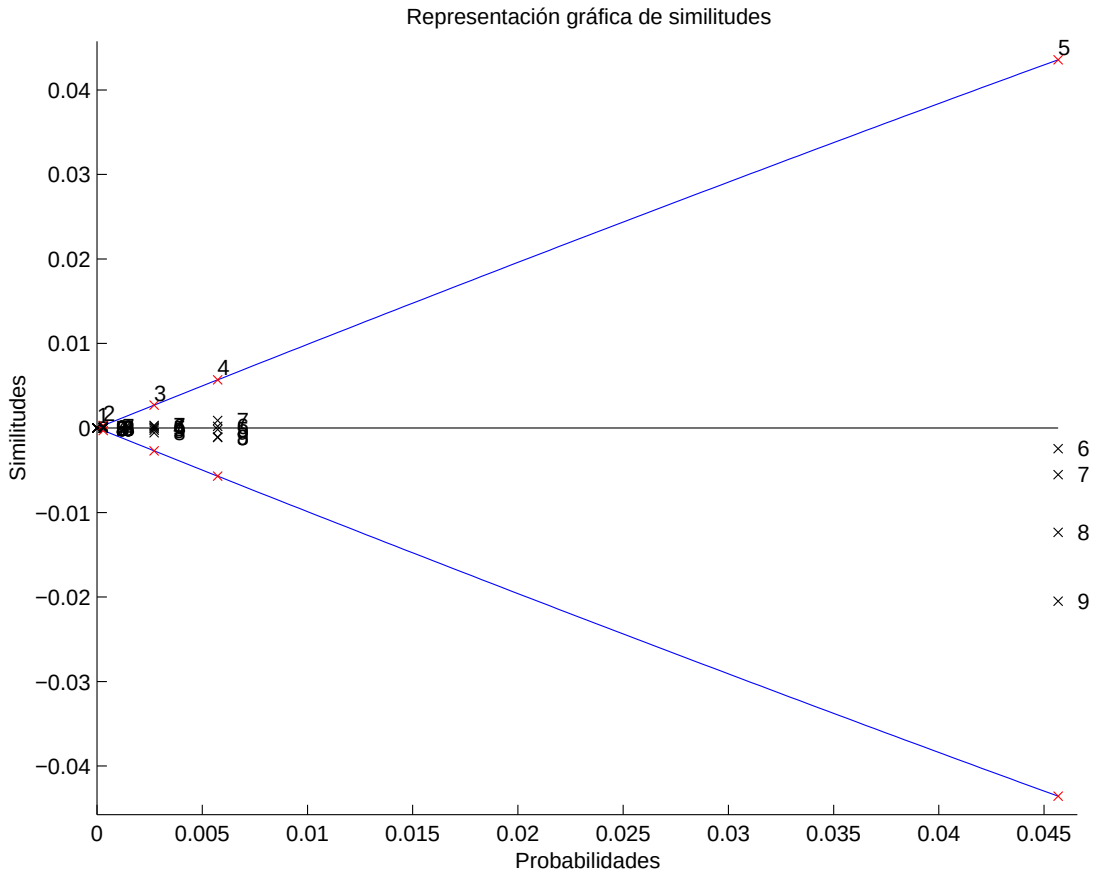

Figura 7.13: Representación gráfica de todas las similitudes, relacionadas con las cinco primeras líneas de investigación en la Teoría del Aprendizaje, en un único gráfico.

7.7 Distancia entre sucesos

En esta sección se define a partir de la función núcleo similitud una distancia entre sucesos de la siguiente manera. Como la función $k(A, B)$ es una función núcleo, es decir, $k(A, B) = \langle \phi(A), \phi(B) \rangle_{\mathcal{F}}$, es posible definir una distancia entre dos sucesos a través de sus transformados $\phi(A)$ y $\phi(B)$, puesto que el objeto de la función $\phi : \mathcal{A} \to \mathcal{F}$ es llevar a cabo una "incrustación" de la clase de sucesos $\mathcal{A}$ dentro del conjunto $\mathcal{F}$ que presenta una estructura de espacio vectorial dotado de un producto escalar. Esta incrustación permite obtener un importante beneficio ya que se puede

tratar con los elementos $A \in \mathcal{A}$ a través de sus transformados desde un punto de vista geométrico y por tanto se puede hacer uso del álgebra lineal y de la geometría analítica.

Si definimos la función:

$$D : \mathcal{A} \times \mathcal{A} \rightarrow \mathbb{R}^+$$

como:

$$D(A, B) = \sqrt{k(A, A) + k(B, B) - 2\,k(A, B)}$$

se tiene que $D(A, B)$ es una función distancia euclidea ya que se obtiene como:

$$D(A, B) = \sqrt{\langle \phi(A) - \phi(B), \phi(A) - \phi(B) \rangle_{\mathcal{F}}}$$

7.8 Comentarios sobre el capítulo

En este capítulo se ha introducido una función, que hemos denominado **función núcleo similitud**, que cuantifica similitudes entre sucesos, a través de sus probabilidades. Se ha realizado un estudio detallado de sus propiedades y se ha dejado un camino abierto para estudiar su relación con la función de distribución y de densidad de un vector aleatorio bidimensional. Todo ello, se ha llevado a cabo de forma analítica y gráfica, construyendo para esta última un conjunto de gráficos que permiten tener una interpretación visual de las similitudes muy intuitiva.

Sin embargo, consideramos que lo más importante ha sido su utilidad práctica en estudiar, a través de una tabla de doble entrada, el comportamiento en un determinado año de diferentes líneas de investigación sobre Teoría del Aprendizaje. También se ha dejado abierta la posibilidad de realizar un estudio longitudinal de estas líneas de investigación, sin más que estudiar como se comportan las similitudes en diferentes años. Evidentemente, este tipo de estudio es posible de realizar sobre cualquier temática.

CAPÍTULO 8

ANÁLISIS DE DOS PROBLEMAS DE MULTICLASIFICACIÓN UTILIZANDO LAS MÁQUINAS DE VECTORES SOPORTE

Saca en primer lugar la viga de tu propio ojo, y entonces podrás ver claramente para sacar la mota del ojo de tu hermano.

–*Berkeley en The Analyst* (1734)–

Este capítulo nos permitirá comprobar el funcionamiento de la máquina de vectores soporte que se ha ido diseñando a lo largo de este trabajo. Para su elaboración hemos necesitado realizar distintos programas que hemos llevado a cabo dentro del entorno de programación del programa Matlab, versión 5.3; y que aparecen desarrollados en la sección B.2 del apéndice B recogido al final de este trabajo.

Hemos decidido seleccionar dos conjuntos de datos, incorporados en sendos libros de textos con un carácter eminentemente didáctico, donde llevamos a cabo un estudio detallado de las propiedades del modelo discriminador por nosotros desarrollado.

La opción de estudiar la versatilidad de la máquina sobre un conjunto real, no nos pareció adecuada debido a que la complejidad propia de la naturaleza de los datos reales pueda oscurecer las características propias de nuestra máquina de vectores soporte para la multiclasificación. Sin embargo, es posible encontrar, una máquina parecida, a la propuesta por nosotros, aplicada a datos reales en la tesis doctoral del profesor Cecilio Angulo de la Universidad Politécnica de Cataluña (U.P.C.), [Ang01].

8.1 Conjunto de datos Hatco

Para la primera implementación de la máquina de vectores soporte para la multiclasificación consideramos el conjunto de datos obtenido del libro "Análisis Multivariante", de Hair, Anderson y Tathan[(1)] [HATB00] que hace referencia a un distribuidor industrial.

De entre todas las variables consideradas en la base de datos HATCO, elegimos las siguientes tres variables explicativas[(2)]:

- X_1 = Velocidad de entrega de un producto.
- X_2 = Nivel de precio.
- X_3 = Flexibilidad de precios (negociación del precio de compra).

Estas tres variables son variables métricas que cuantifican la percepción que los clientes tienen sobre la compañía HATCO (Hair, Anderson, Tathan, Company). Se toma las siguientes variables categóricas dicotómicas que describen las características del comprador[(3)]:

[(1)] Este libro de la editorial Prentice Hall, lleva actualmente 5 ediciones (en español), es uno de los libros clásicos utilizado en la enseñanza de los problemas relacionados con el análisis multivariante, por su calidad y riquezas de contenidos.

[(2)] Son las mismas que utilizan los autores del libro para explicar el comportamiento del análisis discriminante y la regresión logística, aunque sobre un etiquetado dicotómico.

[(3)] Utilizamos la misma numeración de las variables que aparece en el texto original.

- X_9 = Tamaño de la empresa (relativo a otras empresa en el mismo mercado). 1="Empresa grande" y 0="Empresa pequeña".

- X_{11} = Tipo de industria. 1="Industria de una clase A"y 0="Otras clases de industrias".

Con objeto de conseguir un etiquetado más complejo, que el presentado con una variable categórica dicotómica, que nos permita ver la operatividad de nuestra máquina multiclasificadora, construimos una nueva variable categórica (etiquetado) a partir de las dos variables anteriores, que presenta cuatro modalidades, la cual será objeto de estudio por nuestra parte. Esta variable la denotamos por Y y se obtiene como sigue:

$$Y = \begin{cases} 1 & \text{si} \quad X_9 = 0 \quad \text{y} \quad X_{11} = 0 \\ 2 & \text{si} \quad X_9 = 1 \quad \text{y} \quad X_{11} = 0 \\ 3 & \text{si} \quad X_9 = 0 \quad \text{y} \quad X_{11} = 1 \\ 4 & \text{si} \quad X_9 = 1 \quad \text{y} \quad X_{11} = 1 \end{cases}$$

De esta forma, el conjunto de trabajo, que llamaremos **hatco**, consta de 100 datos que presentan el siguiente formato:

$$(\text{Identificador}, X_1, X_2, X_3, Y)\,,$$

como se muestra en la tabla 8.1 para unos cuantos datos de este conjunto.

Una vez seleccionado el conjunto de datos de trabajo comenzamos la elaboración de una función discriminante, utilizando la técnica de las máquinas de vectores soporte, que nos permita llevar a cabo una adecuada clasificación.

En los desarrollos siguientes utilizaremos diferentes tipos de letras para diferenciar las entradas y salidas de los programas de Matlab (tipo romana), de las explicaciones y comentarios realizados sobre el proceso de clasificación (tipo times).

Iniciamos el estudio de los datos con la instrucción (El símbolo >> indica el prompt del programa Matlab)

```
>> clear all
```

Tabla 8.1: Conjunto de datos **hatco** utilizado en el proceso de multiclasificación.

$Identificador$	X_1	X_2	X_3	$Etiqueta$
1	2,4	1,6	8,8	1
2	4,7	1,3	9,9	1
3	3,4	2,0	9,7	1
⋮	⋮	⋮	⋮	⋮
31	4,1	0,6	6,9	2
32	6,0	0,9	9,6	2
33	4,6	2,4	9,5	2
⋮	⋮	⋮	⋮	⋮
61	1,8	3,0	6,3	3
62	1,3	4,2	6,2	3
63	4,0	3,5	6,5	3
⋮	⋮	⋮	⋮	⋮
81	3,4	5,2	5,7	4
82	2,7	1,0	7,1	4
83	1,9	3,3	7,9	4
⋮	⋮	⋮	⋮	⋮
100	0,6	1,6	6,4	4

que nos borra todas las variables de trabajo que hasta ese momento ha almacenado, en la memoria del ordenador, el programa Matlab. A continuación con

```
>> hatco
```

almacenamos (cargamos) en memoria los datos de trabajo. Si deseamos ver como son, basta con volver a ejecutar la misma orden.

Como el etiquetado se encuentra en la última columna de **hatco**, construimos un vector que almacene estas etiquetas, que denominamos **etireal** (abreviatura de

"etiquetado real"), que nos permita ver como es la distribución del etiquetado y posteriormente nos servirá para comprobar el funcionamiento de nuestro discriminador:

```
>> etireal=hatco(:,end)';
```

Este vector debe ser necesariamente un vector fila con objeto de poder ser introducido convenientemente en el programa "clasificacion", uno de los realizados por nosotros. Ejecutamos este programa con el vector **etireal**.

```
>> clasificacion(etireal)

ans =

   100     1     2     3     4     0
     1    30     0     0     0     0
     2     0    30     0     0     0
     3     0     0    20     0     0
     4     0     0     0    20     0
```

Este programa nos proporciona la matriz de clasificación, pero nosotros hemos incorporado más información, ya que esta matriz refleja la salida "0" que realiza nuestra función discriminante para indicar el número de aquellos casos donde no se lleva a cabo un etiquetado, como se desarrolla en el capítulo 5. Además la posición (1,1) de la matriz de salida (primera fila y primera columna) la hemos aprovechado para indicar el número total de vectores estudiados.

La explicación más detallada de esta matriz la posponemos para más adelante cuando realmente podamos indicar el comportamiento del discriminador ya que aún no se ha realizado ninguna clasificación. Por ahora sólo señalamos que se tienen 30 datos con etiqueta "1", 30 datos con etiqueta "2", 20 datos con etiqueta "3" y 20 datos con etiqueta "4" $(30 + 30 + 20 + 20 = 100)$.

Para abordar el problema de clasificación hemos considerado un conjunto de entrenamiento obtenido a partir de un muestreo estratificado de tamaño 70 que guarda la proporción $(3, 3, 2, 2)$ que presenta la distribución de etiquetado de **hatco**. Para realizar esta selección hemos realizado un pequeño programa denominado "walea" que proporciona un vector de identificadores adecuado.

```
>> walea
```

(Para visualizar el conjunto de identificadores basta con volver a ejecutar "walea"). A partir de estos identificadores construimos el conjunto de entrenamiento que denominamos **datos**.

```
>> datos=hatco(walea,:)';
```

Para ver si el número de etiquetas de cada clase es el adecuado ejecutamos las siguientes ordenes (como se hizo anteriormente con el conjunto completo):

```
>> etientre=datos(:,end)';
```

(iniciales de "etiquetas de entrenamiento")

```
>> clasificacion(etientre)

ans =

    70     1     2     3     4     0
     1    21     0     0     0     0
     2     0    21     0     0     0
     3     0     0    14     0     0
     4     0     0     0    14     0
```

Antes de buscar la función discriminadora, veamos cual sería el porcentaje mínimo de aciertos, según la distribución de etiquetas, para este conjunto de datos, es

decir, que proporción de aciertos tendría una función que realizara el etiquetado de forma completamente aleatoria.

Si suponemos que los datos reflejan la verdadera distribución del etiquetado de **hatco** entonces definiendo los sucesos:

$$A_i = \{\text{el dato dado se encuentra en la clase } i\}\,; \qquad i = 1, 2, 3, 4$$

se tiene que:

$$P(A_1) = p_1 = 0'3; \quad P(A_2) = p_2 = 0'3; \quad P(A_3) = p_3 = 0'2; \quad P(A_4) = p_4 = 0'2$$

Si no se dispone de información sobre el etiquetado, salvo que existen cuatro clases con la anterior distribución de probabilidades, una elección aleatoria proporcionaría la siguiente probabilidad de acierto:

$$A = \{\text{Acertar la clase de un determinado dato }\}\,;$$

$$P(A/A_i) = P(A_i);$$

$$P(A) = \sum_{i=1}^{4} P(A/A_i)\, P(A_i) = \sum_{i=1}^{4} p_i \cdot p_i = \sum_{i=1}^{4} p_i^2 = 0'3^2 + 0'3^2 + 0'2^2 + 0'2^2 = 0'26$$

Luego, el porcentaje mínimo de acierto que le debemos exigir a nuestra función discriminante es del 26%, ya que de otra forma no nos proporcionaría ninguna información relevante en el proceso de clasificado.

Antes de ejecutar el programa "discrimina", uno de los dos programas fundamentales en todos los desarrollos, al conjunto de entrenamiento necesitamos aclarar como son las salidas.

La función discriminante f_{ij} (realiza la discriminación entre los datos de etiqueta "i" y "j") tiene la forma:

$$f_{ij}(x) = \sum_{m=1}^{n} \alpha_m^{ij}\, k(x, x_m) \; + \; b^{ij}$$

donde x_m son los vectores de entrenamiento (en este caso, $n = 70$) y $k(\cdot, \cdot)$ es la función núcleo.

La función "discrimina", a partir de un conjunto de datos, proporciona como salida una matriz de orden $\ell \times \ell$ (ℓ = número de clases distintas) y dos hipermatrices de orden $\ell \times \ell \times n$ (n = tamaño del conjunto de entrenamiento) y $\ell \times \ell \times 1$:

- sv(i,j)= número de vectores soporte necesarios en la construcción de la función discriminante f_{ij}.
- galfa(i,j,m)=α_m^{ij} donde $m = 1, 2, \cdots, n$ y $i, j = 1, \cdots, \ell$; que proporciona los multiplicadores de Lagrange de la función f_{ij}.
- gbeta$(i,j,1)$=b^{ij} con $i, j = 1, \cdots, \ell$, que proporciona el sesgo o término independiente (b_{ij}) de la función discriminante f_{ij}.

Los parámetros los hemos elegidos teniendo en cuenta exclusivamente el significado de éstos[(4)]. Así:

- $C_1 = C_2 = 10$. Con esta elección estamos dando el mismo peso a los errores que presentan etiquetado $\{-1, 1\}$ que a los de etiquetado $\{0\}$. Al tomar este valor, también, estamos dando una mayor importancia a la obtención de una función que discrimine adecuadamente frente a la "suavidad" que presente.
- k=Núcleo='rbf'. Elegimos como función núcleo, la función de base radial (función gaussiana). Como ya comentamos en el capítulo 6, cuando no tenemos información adicional sobre los datos de trabajo, la elección de esta función núcleo es la más natural. Hemos de indicar que sobre los datos originales, no se ha realizado ningún tipo de normalización[(5)].

[(4)]Podríamos haber optado por realizar un estudio exhaustivo de este problema en función de los valores dados a los diferentes parámetros de nuestro modelo. Sin embargo, con los dos ejemplos que vamos a estudiar en este capítulo, pretendemos introducir el modelo basado en las SVMs para la multiclasificación, por nosotros propuesto, a los problemas de tipo económico. Por ello, estamos más interesamos en la dinámica de trabajo y en las herramientas implementadas en el modelo que en la realización de una perfecta clasificación.

[(5)]En estos tipos de análisis con máquinas de vectores soporte es habitual realizar un proceso

- $p_1 = 0'5$ (parámetro de la función núcleo).
- $\delta = 0'5$ (parámetro de insensibilidad[6]).

Aplicamos la función "discrimina" al conjunto de entrenamiento:

```
>> [sv,galfa,gbeta]=discrimina(datos);

ans=

Número de etiquetas ....: 4

Número de funciones discriminantes dicotómicas ...: 6
-----------------------------------------------
Calculando la función discriminante ....: 1.2
-----------------------------------------------
Calculando la función discriminante ....: 1.3
-----------------------------------------------
Calculando la función discriminante ....: 1.4
-----------------------------------------------
Calculando la función discriminante ....: 2.3
-----------------------------------------------
Calculando la función discriminante ....: 2.4
-----------------------------------------------
Calculando la función discriminante ....: 3.4
-----------------------------------------------
```

de tipificación con los datos, ya que es conocido que si se tipifican los datos, los resultados de la clasificación son mejores. Por supuesto, esta tipificación habría que realizarla con los vectores de entrenamiento con objeto de no añadir información adicional a la máquina proveniente de los datos test.

[6]Sobre este parámetro se puede ver su importancia cuando se compara los resultados de ambos ejemplos.

```
El programa se ha completado en 3.1 segundos
```

Con objeto de ir viendo la ejecución del programa "discrimina", hemos incorporado un conjunto de salidas que nos indican sobre todo, cual es la función que actualmente esta calculando[7]. Se finaliza el programa mostrando el tiempo computacional total.

La dificultad que supone la realización de la clasificación, en cierta forma, también queda marcada por el número de vectores soporte necesarios para la construcción de las distintas máquinas biclasificadoras f_{ij}, así como la capacidad de generalización. Si queremos obtener todos los vectores soporte necesarios en la ejecución del programa "discrimina" se sigue:

```
>> sv

sv =
     0    39    50    46
     0     0    49    48
     0     0     0    30
```

De esta forma, se tiene que el número de vectores soporte necesario en la construcción de la función discriminante f_{12} es 39 que representa un 55.71% del porcentaje total de datos de entrenamiento. También, se tiene que la función discriminadora que más vectores soporte necesita es f_{13} con 50 (un 71.43%); y la que menos necesita es f_{34} con sólo 30 (un 42.86%).

Veamos como lleva a cabo nuestro modelo, el proceso de clasificación dentro del conjunto de entrenamiento. Este estudio es necesario realizarlo ya que si nuestro discriminador presenta unos pobres resultados con este conjunto, no es de esperar que se comporte bien con el conjunto test, es decir, lo lógico sería que presentase

[7]Si observamos que el cálculo de una determinada función f_{ij} es largo, esto es sintomático de la dificultad de llevar a cabo la discriminación entre los items correspondientes

una pobre capacidad de generalización. Para ello construimos, siguiendo los desarrollos del capítulo 5 (esquema de votación y grado de confianza), a partir de las funciones discriminadoras dicotómicas f_{ij}, una función que interprete los resultados procedentes de todas ellas.

Para este fin, hemos realizado el programa "interprete" que lleva a cabo esta interpretación. Este programa no lo hemos construido como los anteriores (como una función en Matlab) ya que el número de salidas, que nosotros deseamos nos proporcione, es tal que sería muy engorroso la definición de la orden. Igual ocurre con el conjunto de parámetros a tener en cuenta. Por otra parte, de esta forma, tenemos a nuestra disposición todos los resultados intermedios del programa que nos permite estudiar como se lleva a cabo el proceso de multiclasificación en aquellos casos donde nos parezca oportuno.

Por otro lado, sería posible realizar un único programa que lleve a cabo el trabajo de los programas "discrimina" e "interprete", pero ya que el mayor coste computacional es llevado a cabo por el programa "discrimina" donde se resuelve los diferentes problemas de optimización (en este caso seis), no nos ha parecido adecuado. De esta forma se consigue que una vez ejecutado el programa "discrimina" podamos llevar a cabo tantas clasificaciones como queramos a partir del programa "interprete", sin tener que volver a repetir el proceso de optimización.

El programa "interprete" funciona de la siguiente manera: a partir de un conjunto de entrenamiento (**datos**) y un conjunto test (**datostest**) proporciona una matriz de etiquetados con tantas filas como funciones discriminadoras parciales se tengan; y tanta columnas como vectores (datos) se tengan en el conjunto test. También, con la misma dimensión, proporciona una matriz del grado de confianza (una confianza por cada función discriminadora parcial). La primera matriz se denomina **etiqueta**, y la segunda **confianza**. Sin embargo, las salidas más utilizadas, cuando se tengan un nuevo input, serán:

- **etiquetafinal**: Vector fila con las etiquetas proporcionadas a los datos del conjunto test por el programa "interprete".

- **confianzafinal**: Vector fila con los grados de confianzas que la función discriminadora final asigna a cada una de las etiquetas proporcionadas a los datos del conjunto test.

Ejecutamos el programa "interprete":

```
>> interprete
```

Las etiquetas asignadas por nuestro modelo se encuentran en el vector **etiquetafinal**, y a partir de estas construimos la matriz de clasificación, que llamaremos **Aent**, para estas salidas. Se debe tener cuidado y colocar el etiquetado real como la primera entrada del programa "clasificacion":

```
>> Aent=clasificacion(etientre,etiquetafinal)

Aent =

    70     1     2     3     4     0
     1    21     0     0     0     0
     2     0    21     0     0     0
     3     0     0    14     0     0
     4     0     0     0    14     0
```

Observamos que la máquina clasifica correctamente todos los vectores de entrenamiento (volvemos a posponer la interpretación detallada de esta matriz para más adelante).

El siguiente paso es estudiar como se comporta la máquina para el resto de los datos de **hatco**. Para ello volvemos a ejecutar el programa "interprete" ahora sobre todo el conjunto de datos **hatco**, es decir, **datostest=hatco**.

```
>> interprete
```

Veamos la clasificación llevada a cabo, es decir, comparamos las etiquetas reales (**etireal**) y las etiquetas proporcionadas por la máquina (**etiquetafinal**):

```
Atest=clasificacion(etireal,etiquetafinal)
```

```
Atest =

   100     1     2     3     4     0
     1    27     0     0     0     3
     2     0    28     0     0     2
     3     0     0    18     0     2
     4     0     0     0    18     2
```

Ahora que la "matriz de clasificación" tiene un aspecto algo más complicado explicamos su significado.

- Primera fila: El elemento Atest(1,1) muestra el número total de etiquetas comparadas. Los restantes elementos de la fila denotan la etiqueta predicha por la máquina. Como ya indicamos, la etiqueta "0" denota la opción de "no clasificar" ("no etiquetar") de la máquina de vectores soporte.

- Primera columna: Salvo el primer elemento, los restantes denotan el etiquetado real de los datos.

- Segunda fila: Se refiere a los datos con etiquetado real "1"; y los resultados de la máquina. En este caso, de los 30 vectores con etiqueta real "1", la máquina clasifica correctamente 27 vectores y no etiqueta 3.

- Tercera fila: Se refiere a los datos con etiquetado real "2"; y los resultados de la máquina. En este caso, de los 30 vectores con etiqueta real "2", la máquina clasifica correctamente 28 vectores y no etiqueta 2.

- Cuarta fila: Se refiere a los datos con etiquetado real "3"; y los resultados de la máquina. En este caso, de los 20 vectores con etiqueta real "3", la máquina

clasifica correctamente 18 vectores y no etiqueta 2.

- Quinta fila: Se refiere a los datos con etiquetado real "4"; y los resultados de la máquina. En este caso, de los 20 vectores con etiqueta real "4", la máquina clasifica correctamente 18 vectores y no etiqueta 2.

Puesto que de los 100 datos, 70 corresponden a los vectores de entrenamiento, se tiene referente al conjunto test los siguientes resultados: De los 30 vectores, clasifica correctamente 21 (el 70%) y no etiqueta 9 (el 30%). **El porcentaje de clasificados erróneamente es 0**%. Por tanto, nuestra máquina supera con creces el 26% que nos proporciona una máquina que asigna aleatoriamente una etiqueta, es decir, con esta máquina se consigue recoger una gran parte de la información disponible en las variables explicativas para llevar a cabo el proceso de clasificación.

Es importante señalar, que una de las principales objeciones que se realizan sobre las redes neuronales, las cuales siguen un esquema en cierto sentido similar a las SVMs, es la de ser una "caja negra", en el sentido de no proporcionar ningún rastro intermedio que permita su posterior análisis. En la máquina que nosotros proponemos se evita este inconveniente ya que somos capaces de extraer una gran cantidad de información intermedia del proceso de etiquetado.

De esta forma, si deseamos saber cuales son los identificadores de los vectores no etiquetados (o los mal clasificados), se sigue:

```
>>  [w]=find(etireal(:)~=etiquetafinal(:))

ans =
    10
    17
    24
    33
    45
    74
```

```
80
86
100
```

Para ver todos los resultados intermedios que proporciona nuestra máquina, se construye la siguiente matriz:

```
>> todo=[etireal;etiquetafinal;etiqueta;confianza];
```

todo es una matriz de dimensión 10×100 ($(2\ell + 2) \times N$ donde N es el número de vectores estudiados y ℓ es el número de etiquetas diferentes). Las columnas representan los datos y las filas los resultados del "interprete". Si se desea ver exclusivamente los datos no clasificados correctamente entonces:

```
>> todo(:,w);
```

En este ejemplo todas las etiquetas parciales son nulas, por ello no las mostramos[8]. Esta situación indica que ninguna de las seis funciones discriminadoras "se atreve" a dar un etiquetado a estos datos, es decir, para estos datos nuestra máquina "peca de prudente".

8.1.1 Comparativa con el análisis discriminante clásico

Con objeto de contrastar el funcionamiento de nuestra máquina frente a otra técnica de clasificación, hemos evaluado el comportamiento del análisis discriminante de Fisher a estos mismos datos.

Hemos llevado a cabo, utilizando el programa SPSS versión 10.0, un estudio de los datos **hatco** teniendo en cuenta el tamaño y sin tener en cuenta el tamaño de las

[8] Se verá adecuadamente en el siguiente análisis de datos.

diferentes clases[9]. Además, hemos considerado como conjunto de entrenamiento, los 100 datos. Los resultados aparecen en las siguientes tablas:

100	1	2	3	4
1	8	16	4	2
2	6	14	10	0
3	0	2	11	7
4	2	0	6	12

100	1	2	3	4
1	8	16	4	2
2	10	14	6	0
3	4	4	5	7
4	2	0	6	12

En la primera tabla (sin tener en cuenta el tamaño de las clases) el porcentaje de datos correctamente clasificados es del 45%; y en la segunda tabla (teniendo en cuenta el tamaño de las clases), el resultado es peor, un 39% de datos correctamente clasificados.

La justificación de esta pobre capacidad de clasificación, así como la diferencia en el porcentaje de aciertos entre los dos análisis discriminantes, habría que buscarla, entre otras cosas, en el tamaño reducido del conjunto de datos (100 para un problema con 4 clases). Esta es precisamente una de las características más importante de las máquinas de vectores soporte, su robustez cuando se dispone de un conjunto pequeño de datos. Esto es así, ya que en el problema que se plantea con las SVMs, la solución se expresa a partir de los vectores soporte que proporcionan una compresión[10] de los datos, ya que como hemos estudiado la solución no se ve afectada si se modifica cualquier otro vector del conjunto de entrenamiento.

Finalmente, hemos de indicar que nosotros hemos implementado nuestra máquina considerando como conjunto de entrenamiento todos los datos de **hatco**, y nuestro modelo ha clasificado correctamente el 100%.

[9] Si bien en este ejemplo, no es muy significativa esta característica si lo será para el siguiente ejemplo.

[10] En [CST00] se puede encontrar una justificación de esta forma de ver las SVMs.

8.2 Conjunto de datos Empresa

La siguiente base de datos, que llamaremos **empresa**, esta compuesta por 474 vectores, tomados del libro "Técnicas Estadísticas con SPSS" de la editorial Prentice Hall [Pér01], tiene dos variables explicativas y una variable dependiente que presenta las diferentes etiquetas.

Las variables explicativas son:

- X_1 = Experiencia previa del empleado antes de acceder a la empresa.
- X_2 = Nivel de estudio del empleado.

La variable dependiente (etiquetado) es:

- Y = Categoría profesional del empleado.

Esta variable categórica presenta tres modalidades diferentes $Y = \{1, 2, 3\}$ según la categoría actual del empleado. En este caso, a diferencia del anterior, las etiquetas permitirían una ordenación (escala ordinal) entre ellas, la cual nosotros no hemos considerado en este estudio. Pensamos que podría ser objeto de estudio en un futuro, ya que esta situación se aborda considerando una función de pérdida diferente en los problemas de optimización que cuantifica el error cometido entre dos items contiguos con menor peso que cuando están más separados. Estudios de este tipo ya han sido llevado a cabo, por ejemplo en [AC01], sin embargo no se ha estudiado como afectan los parámetros del modelo a la función discriminadora final en términos de ganancia de información[(11)].

Respecto a la elección de estos datos, los hemos elegidos debido a la complejidad que presenta su clasificación ya que como veremos la distribución del etiquetado presenta una etiqueta que domina claramente a los dos restantes.

[(11)] Actualmente existe un grupo de trabajo, bajo la dirección de Grace Wahba, que se encuentra trabajando estos problemas bajo la denominación de clasificación "no standard".

Como en el ejemplo anterior comenzamos limpiando de variables la memoria del programa Matlab.

```
>> clear all
```

y cargamos los datos de trabajo en memoria

```
>> empresa
```

En la tabla 8.2 podemos observar como son los datos de este conjunto. Como en el caso de los datos de **hatco** el etiquetado se debe encontrar en la última columna y el identificador de los datos en la primera.

Tabla 8.2: Conjunto de datos **empresa** utilizado en el proceso de multiclasificación.

Identificador	X_1	X_2	Etiquetado
1	144	15	3
2	36	16	1
3	381	12	1
4	190	8	1
5	138	15	1
6	67	15	1
7	114	15	1
8	0	12	1
⋮	⋮	⋮	⋮
45	307	12,0	2
⋮	⋮	⋮	⋮
474	9,0	12,0	1

Veamos la distribución del etiquetado:

```
>> etireal=empresa(:,end)';
>> clasificacion(etireal)

ans =

   474     1     2     3     0
     1   363     0     0     0
     2     0    27     0     0
     3     0     0    84     0
```

Observamos como el número de vectores con etiqueta "1" es muy superior respecto a las otras dos, representa el 76′58% del total, frente al 5′70% de la etiqueta "2" y del 17′72% de la etiqueta "3".

Así, si asignamos un etiquetado aleatorio, la probabilidad de acierto, manteniendo los mismo supuestos que con los datos de **hatco**, es:

$$A = \{\text{Acertar la clase de un determinado dato }\};$$

$$P(A) = \left(\frac{363}{474}\right)^2 + \left(\frac{27}{474}\right)^2 + \left(\frac{84}{474}\right)^2 = \frac{139554}{224676} = 0'6211,$$

es decir, el 62′11%. Sin embargo, aprovechando la información de la distribución de etiquetas, se podría optar por asignar a cualquier nuevo dato la etiqueta"1", y de esta forma[12], el porcentaje de acierto sería del 76′58%. Por ello, en este caso, debemos exigir que nuestro modelo tenga una proporción de aciertos superior a esta cantidad.

Consideramos como conjunto de entrenamiento para nuestra máquina, él formado por los 200 primeros vectores[13]:

[12] Si los datos de **empresa** reflejan la verdadera proporción de etiquetado.

[13] No hemos considerado un muestreo estratificado ya que, como después mostraremos, la distribución en el etiquetado de estos 200 primeros vectores es muy similar a la del conjunto completo. Una justificación de esta situación es la siguiente:
Un procedimiento de selección de los datos de entrenamiento sería "ordenar aleatoriamente" el

```
>> datos=empresa(1:200,:);
```

Veamos como se distribuye el etiquetado de este conjunto, al cual denominamos **etientre** iniciales de "etiquetas de entrenamiento":

```
>> etientre=datos(:,end)';
>> clasificacion(etientre)

ans =

   200     1     2     3     0
     1   150     0     0     0
     2     0    11     0     0
     3     0     0    39     0
```

La distribución de este etiquetado en porcentajes es:

Etiqueta	1	2	3
Porcentaje	75′0%	6′5%	19′5%

que mantiene aproximadamente las proporciones del conjunto completo.

Para llevar a cabo la clasificación hemos decidido dar más importancia a los errores que presentan los datos cuando se etiquetan como "1" y "-1", en las funciones discriminadoras parciales f_{ij}, que cuando se etiquetan como "0". De esta forma, como se tienen tres etiquetas diferentes, buscamos que la función f_{23} discrimine adecuadamente estos datos; y de esta forma paliar, en lo posible, la influencia del tamaño de la etiqueta "1".

Evidentemente, es posible elegir diferentes parámetros en cada problema de optimización que nos determina las funciones f_{ij}. Sin embargo, si se hiciera de esta

conjunto de todos los datos, y seleccionar, una fracción fija $\alpha \cdot N$ de los primeros así ordenados. Si N es suficientemente grande, esto sería equivalente a un muestreo estratificado.

forma, nos encontraríamos con la dificultad de diseñar el programa "discrimina", ya que presentaría una gran cantidad de parámetros[(14)]. Por otro lado, es conocido que en general, no siempre la unión de los mejores componentes trae como consecuencia la mejor de las máquinas.

Por todo ello, tomamos:

- $C_1 = 5$ y $C_2 = 3$.

Decidimos, también, que nuestra máquina no sea tan conservadora, como con los datos anteriores, y tomamos un parámetro de insensibilidad más pequeño:

- $\delta = 0'25$.

La función núcleo elegida es:

- nucleo = función rbf[(15)]; con $p_1 = 5$,

de esta forma, tomando p_1 tan grande[(16)] buscamos conseguir una función discriminadora "suficientemente" suave[(17)].

Ejecutamos el programa "discrimina" con estos parámetros:

```
>> [sv,galfa,gbeta]=discrimina(datos);
```

[(14)] Cada función necesita 4 parámetros (C_1, C_2, δ, p_1) si se opta por la función rbf (gaussiana). Así en este ejemplo en lugar de tener 4 parámetros se tendría $3 \times 4 = 12$, y en el ejemplo anterior serían $6 \times 4 = 24$. Las posibilidades cambiando la función núcleo ya se escapa de cualquier comentario.

[(15)] Siguiendo el mismo comentario que en el ejemplo anterior.

[(16)] La elección de p_1, si se realiza anteriormente un proceso de tipificación en los datos, no plantea ningún problema ya que se suele tomar $p_1 = 1$. Sin embargo, seguimos inicialmente con la idea de estudiar el comportamiento de la máquina sin explota la transformación de los datos originales.

[(17)] En las funciones rbf cuanto menor sea el valor del parámetro p_1 menos suave es la solución obtenida.

```
Número de etiquetas ....: 3


Número de funciones discriminantes dicotómicas ...: 3

-----------------------------------------------
Calculando la función discriminante ....: 1.2

-----------------------------------------------
Calculando la función discriminante ....: 1.3

-----------------------------------------------
Calculando la función discriminante ....: 2.3

-----------------------------------------------
El programa se ha completado en 161.3 segundos
```

El tiempo de ejecución del programa[18] ha sido de 161.3 segundos, lo cual supone un gasto considerable de tiempo si cada vez que se quisiese llevar a cabo una nueva clasificación se tuviese que consumir. Por ello, como ya comentamos anteriormente, decidimos utilizar dos programas distintos. El primero resuelve los diferentes problemas de optimización, que requieren un considerable gasto computacional, mientras que el segundo, que lleva a cabo el nuevo etiquetado, es casi instantáneo.

Por otro lado, como se indica en los capítulos teóricos, los problemas de optimización que se plantean con las máquinas de vectores soporte siempre tienen solución (y es única) y los diferentes algoritmos de optimización que existen siempre llegan a calcularla (de forma aproximada). Por ello, el tiempo de ejecución del programa es síntoma de la dificultad que presenta el proceso de clasificación. Nótese que el tiempo de ejecución con la datos de **hatco** fue muy pequeño comparado con el consumido con los datos de **empresa** a pesar de no existir una diferencia tan grande con el número de datos considerados y tener que calcular el doble de funciones clasificadoras.

Antes de estudiar como se comporta nuestra máquina con los datos de entrena-

[18] Evidentemente, este tiempo depende del hardware utilizado en cada caso.

miento, veamos cuantos vectores soporte han sido necesarios para su cálculo:

```
>> sv

sv =

     0    95    84
     0     0    98
```

lo que indica que el número de vectores soporte es muy parecido en la construcción de las tres máquinas parciales y ligeramente por debajo del 50% del conjunto de entrenamiento.

Veamos el etiquetado que nuestra máquina realiza sobre el conjunto de entrenamiento. Para ello se toma en el programa "interprete", **datostest = datos**:

```
>> interprete
>> Aent=clasificacion(etientre,etiquetafinal)

Aent =

   200     1     2     3     0
     1   148     1     1     0
     2     2     9     0     0
     3     8     0    31     0
```

Observamos como en este problema se cometen errores en la clasificación del conjunto de entrenamiento, en total 12 que representa un 6% frente al 94% de aciertos. En este caso, la máquina no clasifica el 0% de los datos, es decir, la elección de $\delta = 0'25$ ha "obligado" a la máquina a etiquetar todos los datos, aunque ello ha supuesto el error en algunos de éstos.

Podemos observar dentro de la matriz de clasificación que $\mathbf{Aent}(2,3) = 1$, lo cual indica que el clasificador ha errado en el etiquetado, etiquetando como "2" un

dato cuya etiqueta real es "1". Dentro de la matriz, lo más destacable es que ha etiquetado con "1", ocho datos cuya etiqueta real es "3". Esto se explica si se tiene en cuenta que el conjunto de datos con etiqueta "1" es mucho mayor que los otros dos y tiende a "acaparar" todos los datos.

Por otro lado, ya señalamos en capítulos anteriores, que es posible elegir el parámetro p_1, de la función núcleo de base radial, suficientemente pequeño para conseguir un 100 por 100 de aciertos[(19)] en el conjunto de entrenamiento. Sin embargo, esta elección de p_1, trae como consecuencia una pobre capacidad de generalización, que en este caso se traduciría en un pobre nivel de acierto en el conjunto de test, como así lo hemos podido contrastar en las diferentes implementaciones realizadas.

Veamos ahora el comportamiento con el resto de los datos del conjunto **empresa**. Para ello, consideramos en la función "interprete": **datostest= empresa(201:474,:)** los restantes 274 datos.

```
>> interprete
>> etitest=empresa(201:474,end)';
>> Atest=clasificacion(etitest,etiquetafinal)

Atest =

   274     1     2     3     0
     1   196     1     3    13
     2     7     2     0     7
     3    16     0    23     6
```

Se observa que los resultados no son excesivamente buenos, sobre todo cuando se estudian los datos con etiquetas "2" y "3". En conjunto, el modelo predice correctamente el etiquetado de 221 datos (80′66%), se equivoca en 27 (9′85%) y no eti-

[(19)]Salvo cuando se tengan datos con los mismos valores en las variables explicativas pero con distintas etiquetas.

queta[20] 26 (9′49%). Ciertamente, se supera el nivel mínimo de aciertos exigido (76′58%) pero no se mejora mucho. Aunque, también, se podría ver en el sentido de que la máquina sólo se equivoca un 9′85% de las veces.

Sin embargo, si realizamos un estudio de los resultados por clases se tiene que:

Etiqueta	Aciertos	Fallos	No etiquetados
1	92′02%	01′88%	06′10%
2	12′50%	43′75%	43′75%
3	51′11%	35′56%	13′33%

De estos desarrollos podemos concluir que las Máquinas de Vectores Soporte son sensibles al tamaño relativo de los conjuntos de etiquetas. Claramente, esta característica es inherente en cualquier modelo de análisis discriminante donde el peso del volumen de la información procedente de los datos es determinante.

Aunque es posible pensar que en estos ejemplos con sólo tres etiquetas se puede ajustar la función "interprete" de modo que, con la etiqueta "0" podríamos indicar que la etiqueta a asignar en la función discriminadora f_{ij} es precisamente la etiqueta que no es ni i ni j (por ejemplo, en f_{12} el etiquetado "0" correspondería a la etiqueta "3") con lo que se mejoraría en el sentido de obligar al modelo a asignar alguna etiqueta[21], no es este el sentido que queremos darle a la clase "0"; en ella, se pretende incluir todos los casos de dudosa asignación de una etiqueta concreta con independencia del número de etiquetas que se consideren.

Como ya hemos indicado, una de las características más resaltable de nuestro modelo, respecto a las redes neuronales, es la de poder realizar un análisis más detallado de las salidas ya que en el proceso de construcción de la función discriminante obtenemos un buen conjunto de información intermedia. Por ello, veamos como ha

[20] Nótese que a pesar de dar un valor pequeño al factor de insensibilidad, con objeto de obligar a la máquina a realizar un etiquetado, ésta lleva a cabo un no etiquetado.

[21] Probablemente teniendo en cuenta el tamaño de la primera clase se mejoraría el porcentaje de correctamente clasificados

sido el comportamiento de nuestra máquina en los casos donde se ha llevado a cabo una clasificación errónea.

En primer lugar construimos la matriz de etiquetado y grado de confianza, a la cual denominamos **todo**:

```
>> todo=[etitest;etiquetafinal;etiqueta;confianza];
```

Las columnas de este vector hace referencia al comportamiento de cada uno de los datos del conjunto test. Interpretamos una de estas columnas:

```
>> todo(:,1)

ans =

    1.0000
    1.0000
    1.0000
         0
         0
    0.7772
    0.8515
    0.9856
```

a partir de la tabla 8.3.

Estudiamos los fallos cometidos con los datos correspondientes a la clase "1":

```
>> [w1]=find((etitest(:)~=etiquetafinal(:) &
        etiquetafinal(:)~=0 & etitest(:)==1))

w1 =
```

Tabla 8.3: Interpretación de las columnas de la matriz **todo** obtenida para la multiclasificación.

1	Etiqueta real.
1	Etiqueta predicha en el modelo.
1	Etiqueta asignada por la función discriminante f_{12}.
0	Etiqueta asignada por la función discriminante f_{13}.
0	Etiqueta asignada por la función discriminante f_{23}.
0.7772	Grado de confianza en el etiquetado de f_{12}.
0.8515	Grado de confianza en el etiquetado de f_{13}.
0.9856	Grado de confianza en el etiquetado de f_{23}.

```
 17
 72
209
252
```

Las tres condiciones impuestas a la función "find" (incorporada por defecto en el programa Matlab) son:

- `etitest(:)~=etiquetafinal(:)\rm:` buscamos que la etiqueta real y la etiqueta predicha sean distintas.
- `etiquetafinal(:)~=0` : la etiqueta predicha no es la etiqueta "0".
- `etitest(:)==1` : la etiqueta real es "1".

Sin embargo, esta salida no representan los verdaderos identificadores de los datos errados ya que se han considerado a partir del dato 200 los datos test. Por ello, los verdaderos identificadores de los vectores mal clasificados de esta clase son:

```
>> 200+w1'
```

```
ans =

   217    272    409    452
```

La máquina para dar el etiquetado final ha considerado los resultados siguientes[22]:

```
>> [200+w1';todo(:,w1)]

ans =

  217.0000  272.0000  409.0000  452.0000
    1.0000    1.0000    1.0000    1.0000
    3.0000    3.0000    3.0000    2.0000
         0         0         0         0
         0    3.0000    3.0000    1.0000
    3.0000    3.0000    3.0000    2.0000
    0.5487    0.9860    0.9724    0.7439
    0.9099    0.9743    0.5425    0.9019
    0.9323    0.8083    0.9432    0.9127
```

Veamos como se interpreta esta tabla. En primer lugar se tiene una columna para cada uno de los datos donde la máquina ha errado en su clasificación. Así en este caso, respecto a la etiqueta real "1", se tienen cuatro columnas (se ha cometido cuatro errores). Estudiamos cada una de las columnas:

- Primera columna: El primer número indica que el identificador del vector errado es el número 217. El segundo número muestra cual es la etiqueta real de este vector (etiqueta "1") y el tercer número representa la etiqueta predicha por la máquina (etiqueta "3"). Los dos números siguientes indican que tanto

[22]Incluimos en la primera fila de la matriz **todo** el identificador correspondiente al dato errado.

la máquina f_{12} como la máquina f_{13} no etiquetan este dato (etiqueta "0"). El siguiente número muestra el resultado del etiquetado de la máquina f_{23}, en este caso etiqueta el vector identificado por el número 217 con la etiqueta "3". Los tres números restantes representan la confianza de las máquinas f_{12}, f_{13} y f_{23}, respectivamente.

- Segunda columna: Indica que el identificador del segundo vector errado es el número 272, cuya etiqueta real es "1" y etiqueta predicha por la máquina es "3". A continuación, la máquina f_{12} no etiqueta este dato (etiqueta "0") y las otras dos máquinas f_{13} y f_{23} etiquetan el vector con la etiqueta "3". Los tres números restantes representan la confianza de las máquinas f_{12}, f_{13} y f_{23}, respectivamente.

- Tercera columna: Igual comentario que la columna anterior pero sobre el vector identificado con el número 409.

- Cuarta columna: El cuarto vector mal clasificado es el número 452, cuya etiqueta real es "1" y etiqueta predicha es "2". En este caso, la máquina f_{12} no etiqueta este dato (etiqueta "0") y las otras dos máquinas f_{13} y f_{23} etiquetan el vector con las etiquetas "1" y "2", respectivamente. En este caso, se produce un empate a votos entre dos etiquetas y para llevar a cabo el desempate se recurre al grado de confianza. Así, como la confianza dada por la función f_{23} (novena fila) es del $91'27\%$, superior al $90'19\%$ de confianza de la función f_{13} (octava fila), la etiqueta asignada por nuestra máquina es la "2".

Por otro lado, si estudiamos la tabla anterior por filas, teniendo en cuenta el significado de cada fila dado en la tabla 8.3, observamos como la función f_{12} (cuarta fila) no se equivoca nunca, pero no se "moja", ya que no asigna etiqueta a ninguno de los cuatro vectores. El dicriminador f_{13} (quinta fila) se equivoca en 2 de las 4 veces que asigna etiqueta; y el discriminador f_{23} (sexta fila) es el que peor se ha comportado errando las cuatro veces.

Veamos la situación para los errores cometidos con el conjunto de datos con etiqueta real "2".

```
>> [w2]=find((etitest(:)~=etiquetafinal(:) & etiquetafinal(:)~=0 &
             etitest(:)==2));

>> [200+w2';todo(:,w2)]

ans =

 281.0000  291.0000  305.0000  335.0000  353.0000  386.0000  414.0000
   2.0000    2.0000    2.0000    2.0000    2.0000    2.0000    2.0000
   1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000
   1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000
   1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000
        0         0         0         0         0         0         0
   0.9704    0.9178    0.8160    0.7853    0.9506    0.8699    0.7341
   0.8987    0.6525    0.9488    0.7358    0.9072    0.9145    0.9203
   0.9839    0.9847    0.9302    0.9801    0.9865    0.9849    0.9820
```

Observamos que el comportamiento de las tres funciones discriminadoras parciales es muy similar y las tres señalan equivocadamente a la clase "1" como ganadora (f_{23} de forma indirecta).

Veamos la situación para los errores cometidos con el conjunto de datos con etiqueta real "3".

```
>> [w3]=find((etitest(:)~=etiquetafinal(:) & etiquetafinal(:)~=0 &
             etitest(:)==3));
>> [200+w3';todo(:,w3)]

ans =

 Columns 1 through 7
```

```
231.0000  240.0000  256.0000  274.0000  276.0000  277.0000  286.0000
  3.0000    3.0000    3.0000    3.0000    3.0000    3.0000    3.0000
  1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000
  1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000
  1.0000    1.0000    1.0000         0         0    1.0000         0
       0         0         0         0         0         0         0
  0.7905    0.9066    0.8882    0.9080    0.9080    0.9046    0.5746
  0.6607    0.9142    0.8064    0.7433    0.7433    0.8611    0.9790
  0.9773    0.9864    0.9809    0.6797    0.6797    0.8231    0.7658

Columns 8 through 14

288.0000  336.0000  348.0000  371.0000  389.0000  413.0000  449.0000
  3.0000    3.0000    3.0000    3.0000    3.0000    3.0000    3.0000
  1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000
  1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000
       0         0    1.0000         0    1.0000         0    1.0000
       0         0         0         0         0         0         0
  0.5832    0.5799    0.9068    0.9046    0.6979    0.9046    0.9046
  0.9761    0.9753    0.7775    0.8567    0.6047    0.8567    0.9046
  0.7077    0.7855    0.7832    0.6136    0.9859    0.6136    0.8063

Columns 15 through 16

455.0000  462.0000
  3.0000    3.0000
  1.0000    1.0000
  1.0000    1.0000
  1.0000    1.0000
       0         0
  0.9046    0.9246
```

```
0.9046     0.9318
0.8063     0.9851
```

Se observa un comportamiento del intérprete muy parecido al dado con la etiqueta "2".

8.2.1 Comparativa con el análisis discriminante clásico

Al igual que en el ejemplo de los datos **hatco** hemos contrastado el funcionamiento de nuestra máquina frente al análisis discriminante de Fisher para los datos de **empresa**.

Como anteriormente hemos llevado a cabo el estudio de estos datos teniendo en cuenta y sin tener en cuenta el tamaño de las diferentes clases. También, en este caso, es de esperar de este análisis mejores resultados, que en el ejemplo anterior, en los porcentajes de aciertos de la clasificación ya que el número de datos de trabajo es más grande (474 para un problema de clasificación con tres clases diferentes frente al ejemplo anterior de 100 datos para un problema con cuatro clases).

Al igual que con nuestra máquina, hemos considerado como conjunto de entrenamiento, los 200 primeros datos y como conjunto test los restantes 274. Los resultados del análisis discriminante sin tener en cuenta el tamaño de las clases son los siguientes:

Entrenamiento

200	1	2	3
1	118	18	14
2	1	10	0
3	2	0	37

Test

274	1	2	3
1	170	28	15
2	1	15	0
3	2	1	42

Este modelo clasifica correctamente, dentro del conjunto de entrenamiento, el 82'5% (165 de 200) y en el conjunto test, el 82'8% de los datos (227 de 274).

Los resultados del análisis discriminante teniendo en cuenta el tamaño de las clases son los siguientes:

Entrenamiento			
200	1	2	3
1	141	7	2
2	5	6	0
3	19	0	20

Test			
274	1	2	3
1	196	13	4
2	6	10	0
3	21	0	24

Este modelo clasifica correctamente, dentro del conjunto de entrenamiento, el 83'5% (167 de 200) y en el conjunto test, el 83'9% de los datos (230 de 274). De esta forma, si solo se tiene en cuenta el porcentaje de aciertos, este segundo análisis (teniendo en cuenta el tamaño de las clases) mejora el anterior (sin tener en cuenta el tamaño de las clases).

De esta manera, si comparamos el resultado de este segundo análisis con nuestra máquina observamos que dentro del conjunto de entrenamiento, nuestra máquina se comporta mucho mejor, ya que clasifica correctamente el 94% de los datos frente al 83'5% del análisis discriminante. En cuanto, al comportamiento sobre el conjunto de test, nuestra máquina se muestra ligeramente peor[(23)] ya que clasifica correctamente el 80'7% frente al 83'9% del análisis discriminante. Sin embargo, si consideramos el número de errores, el análisis discriminante presenta un porcentaje de fallos del 16'1% frente al 9'9% de nuestra máquina.

Es cierto, que se puede construir un modelo que no falle nunca, basta con que no emita ningún voto. Pero no es menos cierto que el porcentaje de aciertos de nuestra máquina es bastante alto y el hecho de no llevar a cabo el etiquetado de una serie de datos es sintomático de la dificultad que presentan estos datos, y por ello, se debe entender esta postura de la máquina como un aviso de su difícil clasificación y de la necesidad de un examen puntual de ellos, con objeto de obtener un conjunto de reglas que permita optimizar el resultado final del modelo de clasificación.

[(23)] Si se toma como indicador exclusivamente el porcentaje de aciertos.

Hemos de destacar que esta ha sido una de las características que nosotros hemos decidido incluir dentro del modelo. Característica que nos parece muy importante ya que el modelo crea, dentro del conjunto de ensayo, un subconjunto de vectores cuya clasificación es difícil.

Por supuesto, si en la construcción de la máquina decidimos tomar $\delta = 0$, obligamos a ésta a etiquetar todos los vectores, con lo cual no tendríamos el problema de tener vectores sin etiquetar pero, como consecuencia, aumentaría mucho el porcentaje de errores. Con objeto de ver esta situación vamos a construir una nueva máquina en la siguiente sección, que si bien no etiqueta todos los vectores si la obligamos a que el número de vectores etiquetados sea más grande que en la primera máquina.

8.2.2 Construcción de una máquina con datos tipificados

Como ya se indicó en la elección de la máquina anterior no se utilizó ningún tipo de manipulación de datos y los parámetros fueron elegidos sin llevar a cabo un análisis detallado.

En esta sección se presentan las ordenes así como los resultados de la implementación de la máquina multiclasificadora de vectores soporte, que resulta cuando sobre los datos de **empresa** se realiza una tipificación ya que es conocido que las SVMs, en general, mejoran sus porcentajes de aciertos cuando los datos se encuentran tipificados.

```
>> clear all
>> empresa
>> dd=empresa(1:200,:);
>> medias=mean(dd);
>> desv_tip=std(dd);
>> datos=[dd(:,1),(dd(:,2)-medias(2))/desv_tip(2),
          (dd(:,3)-medias(3))/desv_tip(3),dd(:,4)];
```

Nótese que para llevar a cabo la tipificación hemos utilizado la información sobre la media y la desviación típica exclusivamente de los 200 vectores de entrenamiento, sin hacer uso en ningún momento de información referente al conjunto de test.

La elección de los parámetros es:

- Seguimos con un núcleo rbf y valor del parámetro $p_1 = 1$ ya que cuando los datos están tipificados (medias=0 y desviación típica=1) esta es la elección habitual puesto que la constante p_1 en la función de base radial coincide con la varianza en la función de densidad de probabilidad del modelo normal.
- Consideramos $\delta = 0'10$, con objeto de tener una región, aunque pequeña, donde agrupar los vectores de ensayo que presentan una muy difícil clasificación.
- $C_1 = 5$ y $C_2 = 3$, como en la máquina anterior.

Con esta elección de los parámetros se consigue que la región de no etiquetado disminuya, ya que en la elección realizada en la sección 8.2, la frontera de la región era

$$\delta^* = \frac{5 \cdot 1 + 3 \cdot 0'25}{5+3} = 0'71875$$

y en este caso es:

$$\delta^* = \frac{5 \cdot 1 + 3 \cdot 0'10}{5+3} = 0'6625$$

Construimos la máquina:

```
>> [sv,galfa,gbeta]=discrimina(datos);

 Número de etiquetas ....: 3
 Número de funciones discriminantes dicotómicas ...:  3
-----------------------------------------------
Calculando la función discriminante ....: 1.2
-----------------------------------------------
Calculando la función discriminante ....: 1.3
```

```
-----------------------------------------------
Calculando la función discriminante ....: 2.3
-----------------------------------------------
El programa se ha completado en 13.1 segundos

 >> sv =

     0    51    36
     0     0    58

 >> interprete
 >> clasificacion(dd(:,4),etiquetafinal)

 ans =

   200     1     2     3     0
     1   139     1    10     0
     2     7     4     0     0
     3     4     0    35     0

 >> test=[empresa(201:474,1),(empresa(201:474,2)-medias(2))/desv_tip(2),
         (empresa(201:474,3)-medias(3))/desv_tip(3),empresa(201:474,4)];

 >> interprete
 >> clasificacion(empresa(201:474,4),etiquetafinal)

ans =

   274     1     2     3     0
     1   196     3    10     4
     2    12     2     0     2
```

```
3       7       0       38       0
```

En este caso observamos como el número de vectores soporte, necesarios en el proceso de clasificación, ha disminuido por debajo de la mitad respecto a la anterior máquina. También ha disminuido ostensiblemente el tiempo de ejecución del programa, por debajo de una décima parte.

Por otro lado, observamos que el comportamiento de la máquina dentro del conjunto de entrenamiento es peor ya que clasifica correctamente el 89% de los datos (178 de 200), clasifica mal el 11% (22 de 200) y no etiqueta el 0% (0 de 200).

Del estudio del conjunto test se tiene que clasifica correctamente el 86'13% de los datos (236 de 274), luego mejora tanto a la anterior máquina como a los resultados del análisis discriminante. En cuanto al número de no etiquetados es 2'19%, ha disminuido pero el forzar la máquina ha tenido como consecuencia que haya aumentado el número de errores, 32 de 274, que representa el 11'68%. En la tabla 8.4 se presenta un resumen de los porcentajes obtenidos con los distintos modelos.

Tabla 8.4: Resumen de los resultados obtenidos en la clasificación por los distintos modelos utilizados. Las iniciales A. D. corresponde con Análisis discriminante donde el número hacer referencia a (1) si no se tiene en cuenta y (2) si se tiene en cuenta el tamaño de las clases. La máquina SVM (1) se refiere a los datos sin tipificar y SVM (2) a los datos tipificados.

		A. D. (1)	A. D. (2)	SVM (1)	SVM (2)
Entrenamiento	Aciertos (%)	82'50	83'50	94'00	89'00
	Fallos (%)	17'50	16'50	6'00	11'00
	No etiquetados (%)	0'00	0'00	0'00	0'00
Test	Aciertos (%)	82'85	83'94	80'66	86'13
	Fallos (%)	17'15	16'06	9'85	11'68
	No etiquetados (%)	0'00	0'00	9'49	2'19

8.3 Comentario final

La principal diferencia de las Máquinas de Vectores Soporte, al igual que las redes neuronales, con respecto a las técnicas de análisis disriminante tradicionales y los métodos de regresión logística: logit, probit, tobit,... es la ausencia de contrastes de inferencia estadística. Sin embargo, no debemos considerar las SVMs como algo menos riguroso que las anteriores técnicas, sino más bien como una variación del enfoque metodológico.

Las Máquinas de Vectores Soporte están ganando un uso cada vez más extendido en las áreas aplicadas que en las académicas[(24)], ya que aunque obtengan muy buenos resultados predictivos, lo que es necesario en aplicaciones, se quedan cortas en las áreas académicas necesitadas de explicación.

Por todo ello, pensamos que este trabajo puede abrir puertas para la incorporación de esta técnica siguiendo un esquema comprensible de los diferentes conceptos en él incluidos.

Por otro lado, las continuas investigaciones abiertas en este tema deben animar a ambas comunidades a apreciar más las Máquinas de Vectores Soporte.

Los investigadores deberían considerar la aplicación de las SVMs para los problemas de clasificación, especialmente cuando se hace más énfasis en la precisión de la clasificación y no en la interpretación del valor teórico.

La capacidad de las SVMs para manejar relaciones complejas, particularmente aquellas de naturaleza no lineal, ofrece un instrumento analítico de gran capacidad para los problemas de clasificación, pero también se están consiguiendo grandes logros en los problemas de regresión, análisis de componentes principales, estimación, ... Esta flexibilidad proporciona la base de una mayor esperanza en lograr resultados útiles en otros muchos problemas de predicción y clasificación.

[(24)] Actualmente se imparte en algún que otro programa de doctorado como caso particular de red neuronal, a pesar de que ciertamente no lo es.

CAPÍTULO 9

CONCLUSIONES

El estudio llevado a cabo sobre el problema general de aprendizaje a partir de ejemplos ha sido descrito como un problema de minimización de un funcional riesgo sobre un conjunto finito de datos. Para su resolución se diseña una máquina de aprendizaje cuyo objetivo es la elección de un modelo dentro de un espacio de hipótesis, que esté cerca (con respecto a alguna medida) de la función subyacente en el espacio objetivo. De este estudio se concluye que:

- Según sea la naturaleza de los datos de salida, el problema general de aprendizaje proporciona distintas tareas de aprendizaje (reconocimiento de patrones, clasificación ordinal, regresión, estimación de densidades, ...).

- Sin embargo, el principal problema que se plantea en estos problemas es la elección a priori del conjunto de funciones $\mathcal{F}$ (espacio de hipótesis). Lo ideal sería que la función que resuelve el problema de aprendizaje se encuentre dentro de este conjunto pero esto no será así en general, a menos que se le exija a la solución algún tipo de propiedad (a través de un operador), con lo que se entra en el campo del conocimiento a priori y se enlaza con las técnicas de regularización (capítulo 6).

Aunque, las técnicas de regularización representan una buena herramienta de trabajo, ya que es posible obtener a través de ellas funciones núcleos vía las funciones de Green, que es un línea de investigación abierta, presentan el inconveniente de obtener sus propiedades asintóticas y, además, la solución se expresa en términos de un elevado número de vectores, por lo que resulta poco práctica.

Esta es precisamente la razón por la que un gran número de investigadores se han interesado por las máquinas de vectores soporte. Éstas están basadas en el principio de minimización del riesgo estructural que se configura como el método inductivo más adecuado en la consecución de soluciones que permitan generalizar sobre nuevos conjuntos de entradas a partir de un conjunto finito de entrenamiento; y estas soluciones se expresan a través de un pequeño conjunto de vectores, los vectores soporte.

Así del estudio de la capacidad de generalización de la máquina de vectores soporte aplicadas a los problemas de clasificación concluimos que:

- Siguen el principio de minimización del riesgo empírico, y las cotas sobre el riesgo que se obtienen dependen de la razón entre el número de observaciones y la dimensión Vapnik-Chervonenkis del conjunto de funciones. Por tanto, un tema de estudio abierto es la determinación de la dimensión VC para determinados conjuntos de funciones $\mathcal{F}$ que permitan obtener cotas de generalización que no dependen de la distribución conjunta de los datos.

- Una elección de la función núcleo es necesaria para poder obtener cotas sobre las generalizaciones que sean topológicamente apropiadas.

- Hay que tener cuidado, ya que la introducción de los núcleos aumenta significativamente la potencia de las máquinas a la vez que retienen la linealidad que asegura que los aprendizajes resulten comprensibles. Sin embargo, el incremento de la flexibilidad incrementa el riesgo de sobreajuste que aumentan la posibilidad de plantear un problemas mal condicionado debido al número de grados de libertad.

- Aplicando las SVMs con diferentes núcleos, los resultados empíricos conducen a resultados muy similares de precisión en la clasificación y en los conjuntos de vectores soporte. De esta forma, el conjunto de vectores soporte es el que caracteriza el problema dado, de manera que es independiente en cierto grado, del tipo de núcleo usado.

Del estudio de la función núcleo se concluye que:

- Con objeto de tener un control adecuado de la flexibilidad del espacio característico inducido por el núcleo, se requiere una teoría de generalización, la cual sea capaz de describir con precisión que factores han de ser controlados en las máquinas de aprendizaje con objeto de garantizar unas buenas generalizaciones. De aquí que el estudio de los núcleos sea un campo de investigación muy activo.
- El estudio de estas funciones como cuantificadora de similitudes es un tema abierto ya que el tener una forma de medir la similitud entre objetos heterogéneos permite no solo llevar a cabo un análisis cluster, sino el poder establecer una medida que permita estudiar la "proximidad" dentro de un espacio vectorial de objetos que inicialmente no presentan ninguna estructura.
- Del estudio de los núcleo se ha derivado la función núcleo similitud en la que se ha dejado un camino abierto para estudiar su relación con la función de distribución y de densidad de un vector aleatorio.
- También se ha dejado abierta la posibilidad de realizar un estudio longitudinal de las líneas de investigación de una determinada temática, sin más que estudiar como se comportan las similitudes en diferentes años.

En los problemas de multiclasificación, los cuales se basan generalmente en la construcción de problemas de clasificación dicotómicos, hemos concluido que:

- Aquellas máquinas basadas en un único problema de optimización, donde intervienen todos los vectores de entrenamiento juntos con todas las etiquetas,

no resultan prácticas en los problemas de toma de decisión, ya que no es posible obtener información intermedia del proceso de clasificación.

- Los problemas de clasificación basados en un esquema de descomposición y reconstrucción son más adecuados. Sin embargo, los dos esquemas básicos: 1-v-r y 1-v-1 presentan inconvenientes que son deseables de subsanar. De igual modo, las máquinas ℓ-SVCR resuelven el problema de no incorporación de toda la información recogida en los datos dentro de cada una de las máquinas de vectores soporte dicotómicas de la arquitectura 1-v-1, pero sin embargo, presentan otras limitaciones que nosotros superamos construyendo una nueva máquina.

La máquina diseñada por nosotros resuelve, de manera sencilla, el difícil problema de distinguir empates entre etiquetas, dentro de un esquema de votación, asignando a cada etiqueta un valor promedio de unos grados de confianza proporcionado por las funciones discriminadoras parciales. De esta forma, en caso de empate, asignamos como salida, aquella etiqueta que tenga un mayor grado de confianza en promedio. Esta característica permite dar una interpretación probabilística a las SVMs y, además, en la construcción de la máquina, no solo se indica el resultado final del proceso de clasificación sino que además, se proporciona un conjunto de salidas que ayudan al investigador a resolver el problema de forma más eficiente, puesto que la información última es más rica. Así, la salida de la máquina para cada vector de entrada es:

- La etiqueta predicha.
- Grado de confianza depositado en la salida.

Por otro lado, si bien es cierto que en la mayoría de los casos, la elección de los parámetros dentro de una máquina dependen en mayor medida de la habilidad del investigador, dentro de nuestra máquina hemos obtenido un conjunto de reglas que hacen más fácil la interpretación de cada uno de los parámetros, además de

establecer una relación entre todos ellos muy interesante desde la perspectiva de las aplicaciones.

Aún así, ya que la función que debe ser aproximada es desconocida, la búsqueda de los parámetros óptimos que proporcionan unos mejores resultados para unos datos en concreto esta cargada de una gran incertidumbre debido al ruido que presentan los datos así como el número limitado de datos de que se dispone.

Como ejemplos de aplicación de estos criterios se han estudiado dos problemas, con datos, de multiclasificación donde se ha prestado una mayor atención a la presentación de la dinámica de trabajo con esta técnica, que a la relevancia del proceso de clasificación en sí mismo.

PARTE IV

Apéndices

APÉNDICE A

ESPACIOS DE HILBERT CON NÚCLEO REPRODUCTOR

En los desarrollos del capítulo 6 se trabajó con un tipo especial de espacios de Hilbert (espacios métricos y completos) a los cuales se les denominan espacios de Hilbert con núcleo reproductor. Este capítulo esta dedicado integramente al estudio de estos tipos de espacios.

Se comienza con la definición de espacios de Hilbert, de espacios de Hilbert con núcleo reproductor y a continuación se desarrollan las propiedades fundamentales que éstos poseen. Se estudia como se comportan estos espacios sobre clases de funciones de dimensión finita y en la sección siguiente se verá un teorema de completitud de estas clases de funciones en espacios de Hilbert. La restricción de un núcleo reproductor sobre un subespacio de funciones conforma la sección siguiente y a continuación se demuestra como se comporta la suma de núcleos reproductores. Se demostrará que el producto de núcleos reproductores es otro núcleo reproductor, lo que resulta fundamental para poder construir núcleos en espacios de dimensión superior a partir de núcleos unidimensionales.

Se finaliza este apéndice con un ejemplo muy ilustrativo que permite relacionar

un espacio de Hilbert núcleo reproductor con una función de Green, además de poder ver como se construye un núcleo de Mercer asociado con las funciones splines.

A.1 Definición de los núcleos reproductores

El marco de trabajo de los núcleos reproductores son los espacios de Hilbert de funciones, pero inicialmente damos la definición general de espacio de Hilbert.

Definición A.1.1 (de espacio de Hilbert) *Un espacio de Hilbert $\mathcal{H}$ es un conjunto, cuyos elementos que se denotan por $u, v, \cdots$, se denominan vectores, y poseen las siguientes propiedades:*

1. *$\mathcal{H}$ es un espacio vectorial con la operación "+" (suma de vectores) y el producto por un escalar de un cuerpo $\mathcal{K}$ (complejo o real).*
2. *$\mathcal{H}$ tiene definido un producto escalar $\langle u, v\rangle_{\mathcal{H}} = \langle u, v\rangle$ que cumple:*
   - *(a)* $\langle \alpha\, u, v\rangle = \alpha\, \langle u, v\rangle, \qquad \forall\, u, v \in \mathcal{H},\ \forall\, \alpha \in \mathcal{K}$
   - *(b)* $\langle u + v, w\rangle = \langle u, w\rangle + \langle v, w\rangle, \qquad \forall\, u, v, w \in \mathcal{H}$
   - *(c)* $\langle u, v\rangle = \langle v, u\rangle, \qquad \forall\, u, v \in \mathcal{H}$
   - *(d)* $\langle u, u\rangle > 0 \Leftrightarrow u \neq 0, \qquad \forall\, u \in \mathcal{H}$
3. *$\mathcal{H}$ es un espacio métrico completo bajo la norma $\|u\| = \langle u, u\rangle^{1/2}$, es decir, para toda sucesión $\{u_n\}$ de $\mathcal{H}$ tal que[(1)] $\|u_n - u_m\| \to 0$ cuando $m, n \to \infty$ se cumple que existe un único vector $u \in \mathcal{H}$ tal que $\|u_n - u\| \to 0$ cuando $n \to \infty$.*

Nota A.1.2 *Los desarrollos de esta sección se pueden realizar dentro del cuerpo de los números complejos, sin embargo, salvo que expresamente se indique lo contrario, se supondrá que el cuerpo sobre el que se está trabajando es el conjunto de los números reales, es decir, se consideran espacios de Hilbert reales.* ▲

[(1)] A este tipo de sucesiones se les denominan sucesiones de Cauchy.

Puesto que dentro de los espacios de Hilbert, en este trabajo se esta interesado por aquellos formados a partir de clases de funciones, es conveniente dar la definición de espacio de Hilbert referida a clases de funciones.

Definición A.1.3 *Sea una clase lineal*[(2)] *de funciones reales* $f(x)$ *de variable definida en un conjunto* E*,*

$$\mathcal{F} = \{f \,/\, f : E \to I\!R\}$$

dotada de un producto escalar. Se considera la norma

$$\|f\|^2 = \langle f, f \rangle_{\mathcal{F}}$$

donde por $\langle f, g \rangle_{\mathcal{F}}$ *se denota el producto escalar definido sobre* $\mathcal{F}$*. Si el espacio* $(\mathcal{F}, \|.\|)$ *es completo se dirá que* $\mathcal{F}$ *es un espacio de Hilbert.*

Dentro de los espacios de Hilbert de funciones tienen especial importancia un tipo determinado de función núcleo de dos variables, la cual se define de la siguiente forma.

Definición A.1.4 (de núcleo reproductor) *Sea* $\mathcal{F}$ *una clase de funciones definidas en un conjunto* E*, que determina un espacio de Hilbert. La función de dos variables*

$$k : E \times E \to I\!R$$

se dirá que es un núcleo reproductor (del inglés – reproductor kernel–, abreviadamente r.k.) de $\mathcal{F}$ *si satisface las siguientes propiedades:*

1. $\forall\, y \in E$ *fijado,* $k(x, y) \in \mathcal{F}$ *como función de* $x \in E$*.*

2. (**Propiedad reproductora**)*:* $\forall y \in E, \quad \forall\, f \in \mathcal{F}$*,*

$$f(y) = \langle f(x), k(x, y) \rangle_x \tag{A.1}$$

[(2)]No se debe confundir con las funciones lineales, una clase de funciones es lineal si cualquier combinación lineal de funciones de la clase es otro elemento de la clase.

Nota A.1.5 *El subíndice x en el producto escalar significa que éste es aplicado a las funciones respecto de la variable x, es decir, ya que $f(\cdot)$ y $k(\cdot, y) \in \mathcal{F}$ se tiene*

$$\langle f(x), k(x,y)\rangle_x \stackrel{def}{=} \langle f(\cdot), k(\cdot, y)\rangle_{\mathcal{F}}.$$

▲

Nota A.1.6 *Es importante indicar que no todo espacio de Hilbert es un espacio con núcleo reproductor, es decir, no es cierto que en todos los espacios de Hilbert de funciones exista un núcleo reproductor, ya que por ejemplo, si se considera el espacio de las funciones de cuadrado integrable en $[0,1]$:*

$$L_2[0,1] = \{f : [0,1] \to I\!R \text{ tal que } \int_0^1 f^2(x)\, dx < +\infty\}$$

es fácil comprobar que es un espacio de Hilbert con producto escalar definido como

$$\langle f, g\rangle = \int_0^1 f(x)\, g(x)\, dx$$

pero, sin embargo, no tiene un núcleo reproductor como se demostrará en la nota (A.2.3). ▲

A.2 Propiedades de los núcleos reproductores

En lo sucesivo y siguiendo la notación anteriormente introducida, por $\mathcal{F}$ se denotará una clase lineal de funciones $f(x)$ definidas en un conjunto E (no necesariamente un conjunto de números reales) que forma un espacio de Hilbert con la norma $\|f\|$ y producto escalar $\langle f_1, f_2\rangle$. Por $k(x,y)$ se denota el correspondiente núcleo reproductor, y a veces consideraremos $k_x(y) = k_y(x) = k(x,y)$ con objeto de operar más fácilmente.

Propiedad 1 *(Unicidad) Si existe un núcleo reproductor, éste es único.*

Demostración. La demostración sigue la estrategia de suponer que existen dos núcleos reproductores y llegar a que son el mismo. Si existe otro núcleo reproductor

$$k' : E \times E \to \mathbb{R},$$

el cual cumple la propiedad reproductora: $f(y) = \langle f(x), k'(x,y)\rangle$. Entonces para todo $y \in E$ fijado se tiene:

$$\begin{aligned} \|k(x,y) - k'(x,y)\|^2 &= \langle k(x,y) - k'(x,y), k(x,y) - k'(x,y)\rangle_x \\ &= \langle k(x,y) - k'(x,y), k(x,y)\rangle_x - \langle k(x,y) - k'(x,y), k'(x,y)\rangle_x \\ &= (k_y(y) - k'_y(y)) - (k_y(y) - k'_y(y)) = 0 \end{aligned}$$

donde la última igualdad se sigue a partir de la propiedad reproductora de las dos funciones.
Por otro lado, de la definición de espacio de Hilbert se sigue que si $\langle f, f\rangle = 0$ entonces $f = 0$ y de aquí que $k(x,y) - k'(x,y) = 0$, $\forall x, y \in E$ y se ha demostrado que el núcleo, si existe, es único. ■

Definición A.2.1 *Un operador $P : \mathcal{F} \to I\!R$ se dice que esta acotado si para cualquier $f \in \mathcal{F}$ se cumple que:*

$$|Pf| \leq M \ \|f\|$$

donde la constante $M \in I\!R^+$ no depende de f.

Propiedad 2 *(Existencia) Para que exista un núcleo reproductor $k(x,y)$ es necesario y suficiente que para cada $y \in E$, el operador evaluador*

$$P_y : \mathcal{F} \to I\!R$$

definido como

$$P_y f = f(y)$$

sea un operador acotado.

Demostración. $\Rightarrow$) Si $k(x,y)$ existe, entonces aplicando la propiedad reproductora y la desigualdad de Cauchy-Schwarz se tiene que $\forall f \in \mathcal{F}$ e $y \in E$ fijado:

$$\begin{aligned} |P_y f| = |f(y)| = |\langle f(x), k(x,y)\rangle| &\leq \|f\| \langle k(x,y), k(x,y)\rangle^{1/2} = k(y,y)^{1/2} \|f\| \\ &= M_y \|f\| \end{aligned}$$

de donde se sigue que $P_y f = f(y)$ está acotado ya que si $y \in E$ está fijado el valor de $M_y = k(y,y)$ es necesariamente finito.
$\Leftarrow$) Por otro lado, si P_y es un operador acotado, entonces aplicando el teorema de representación de Riesz[3] existe una función $g_y(x) \in \mathcal{F}$ tal que $f(y) = \langle f(x), g_y(x)\rangle$ y definiendo $k(x,y) = g_y(x)$ se tiene que k es un núcleo reproductor. ■

Nota A.2.2 *Para cada $y \in E$ a la función $g_y(\cdot) : E \to \mathbb{R}$ obtenida a partir del teorema de representación de Riesz se le llama* **función evaluador** *de y en $\mathcal{F}$.* ▲

Nota A.2.3 *Esta caracterización de los núcleos reproductores permite demostrar lo apuntado en la nota (A.1.6).*
Sea la función $f(x) = \dfrac{1}{\sqrt[4]{x}}$ y se tiene:

$$\|f\|^2 = \int_0^1 f^2(x)\,dx = \int_0^1 \left(\frac{1}{\sqrt[4]{x}}\right)^2 dx = \int_0^1 \frac{1}{\sqrt{x}}\,dx = \left[2\sqrt{x}\right]_0^1 = 2$$

luego $f(x) \in L_2[0,1]$ y sin embargo tomando $y = 0$ se sigue que el operador $P_0 f$ no existe para $f(x)$ ya que $f(0) = \dfrac{1}{0}$. ▲

Propiedad 3 *En los espacios de Hilbert con núcleo reproductor se tiene que la convergencia en norma implica la convergencia puntual.*

Demostración. Sea $y \in E$ fijado. Si $f_n \to f$ en la norma del espacio de Hilbert, entonces dado $\varepsilon > 0$, consideramos $\varepsilon' = \varepsilon/A$ con $A = k(y,y)^{1/2}$, y se tiene la

[3] Se puede encontrar una demostración en [Rud79].

existencia de $n_0 \in \mathbb{N}$ tal que $\forall\, n \geq n_0$ se cumple $\|f - f_n\| < \varepsilon'$. Como el operador evaluación es un operador acotado se sigue

$$\begin{aligned} |f(y) - f_n(y)| &\leq \|f - f_n\| \, (k(y,y))^{1/2} \\ &\leq \varepsilon' \, A = \varepsilon \end{aligned}$$

lo que significa que $f_n(y) \to f(y), \quad \forall y \in E$ y se tiene la convergencia puntual. ■

Definición A.2.4 (de función definida positiva) *Una función $R : E \times E \to I\!R$ se dice definida positiva*[4] *si para todo $a_1, a_2, ..., a_n \in I\!R$, e $y_1, y_2, ..., y_n \in E$ se tiene que:*

$$\sum_{i,j=1}^{n} a_i \, a_j \, R(y_i, y_j) \geq 0. \tag{A.2}$$

Nota A.2.5 *Si dado un conjunto de n elementos $\{x_1, x_2, \cdots, x_n\} \in E$ se considera la matriz $\mathbf{R} = \{R(x_i, x_j)\}_{i,j=1}^{n}$ entonces para cualquier conjunto de números reales $\{a_1, a_2, \cdots, a_n\}$ se tiene por la definición que*

$$a' \mathbf{R} a \geq 0$$

donde[5] *$a' = (a_1, a_2, \cdots, a_n) \in I\!R^n$, lo que significa que la matriz $\mathbf{R}$ es definida positiva.* ▲

Nota A.2.6 *Si $R(\cdot, \cdot)$ es una función definida positiva, entonces se puede encontrar siempre una familia $\{X(t),\, t \in E\}$ de variables aleatorias gaussianas de media cero con función de covarianza $\mathbf{R}$, es decir, $X(t) \sim N(0, R)$ donde*

$$E[X(s), X(t)] = R(s,t), \qquad \forall s, t \in E.$$

La existencia y definición de esta familia de variables aleatorias en el caso continuo esta garantizada por el teorema de consistencia de Kolmogorov (Ver por ejemplo en [Tod92]). ▲

[4] La definición de definida positiva, utilizada en este trabajo, es la dada en el sentido de E. H. Moore.

[5] a' es el vector traspuesto de a, a veces se denota también por a^t.

Propiedad 4 *(Positividad). El núcleo reproductor $k(x,y)$ de un espacio de Hilbert es una función definida positiva.*

Demostración. Es evidente sin más que ver que la expresión (A.2) coincide con la norma al cuadrado de la función $f(x) = \sum_{i=1}^{n} k(x, y_i)\, a_i$, para todo $a_1, a_2, ..., a_n \in \mathbb{R}$, e $y_1, y_2, ..., y_n \in E$. Veámoslo:

$$\begin{aligned} 0 \leq \left\| \sum_{i=1}^{n} k(x, y_i)\, a_i \right\|^2 &= \langle \sum_{i=1}^{n} k(x, y_i)\, a_i, \sum_{j=1}^{n} k(x, y_j)\, a_j \rangle \\ &= \sum_{i,j=1}^{n} a_i a_j \, \langle k(x, y_i), k(x, y_j) \rangle \\ &= \sum_{i,j=1}^{n} a_i a_j k(y_i, y_j) \end{aligned}$$

■

De esta propiedad se sigue:

1. $k(x,x) \geq 0$ para todo $x \in E$, sin más que tomar $a_1 = 1$ e $y_1 = x$ en la definición A.2.4.

2. $k(x,y) = k(y,x)$ para todo $x, y \in E$ por ser simétrico el producto escalar.

3. $|k(x,y)|^2 \leq k(x,x) \cdot k(y,y)$, para todo $x, y \in E$ por la desigualdad[(6)] de Cauchy-Schwarz

Esta última propiedad admite su recíproca. Dentro de la teoría de los espacios de Hilbert núcleos reproductor estos dos últimos resultados son tan importantes que se le puede dar forma de teorema.

[(6)] Desigualdad de Cauchy-Schwarz: $\langle x, y \rangle \leq \|x\| \, \|y\|$.

Teorema A.2.7 *A todo espacio de Hilbert núcleo reproductor le corresponde una única función definida positiva y recíprocamente, dada una función definida positiva $k : E \times E \longrightarrow I\!R$, se puede construir un espacio de Hilbert núcleo reproductor de funciones reales definidas en E donde k es el núcleo reproductor.*

Demostración. $\Rightarrow$) Se sigue a partir de las propiedades anteriores.

$\Leftarrow$) Sea $k : E \times E \longrightarrow \mathbb{R}$ una función definida positiva. El espacio de Hilbert núcleo reproductor $\mathcal{F}$ se construye como sigue:

Para $y \in E$ fijado se considera la función

$$k_y : E \longrightarrow \mathbb{R} \quad \text{tal que} \quad k_y(x) = k(x, y), \qquad \forall x \in E$$

y el conjunto de funciones

$$\mathcal{F}_1 = \left\{ \sum_i a_i k_{y_i}(\cdot) \quad / \quad a_i \in \mathbb{R}, \;\; y_i \in E \quad \text{sumas finitas} \right\},$$

donde por suma finita, se quiere indicar que todas las funciones de $\mathcal{F}_1$ se obtienen como una suma finita de funciones de la forma $k_y(\cdot)$. Claramente, por construcción, se cumple que $\mathcal{F}_1$ es un conjunto lineal de funciones reales definidas sobre E, puesto que cualquier combinación lineal de funciones de $\mathcal{F}_1$ determina una suma finita de la forma dada en $\mathcal{F}_1$, es decir la operación suma es una operación interna en $\mathcal{F}_1$ y se verifica las propiedades asociativa, conmutativa, elemento neutro, ...

Se define un producto escalar en $\mathcal{F}_1$ como sigue:

$$\langle \sum_i a_i k_{y_i}, \sum_j b_j k_{y_j} \rangle = \sum_{i,j} a_i b_j \langle k_{y_i}, k_{y_j} \rangle$$

donde

$$\langle k_x, k_y \rangle = k(x, y), \qquad \forall x, y \in E.$$

Ciertamente es producto escalar por ser k una función definida positiva.

Además para todo $y \in E$ se tiene que k_y es el evaluador (ver nota A.2.2) ya que se verifica $\forall f \in \mathcal{F}_1$:

$$\langle k_y, f \rangle = \langle k_y, \sum_i a_i k_{y_i} \rangle = \sum_i a_i \langle k_y, k_{y_i} \rangle = \sum_i a_i k_{y_i}(y) = f(y). \qquad (A.3)$$

Como la convergencia en norma implica la convergencia puntual ya que para la demostración de este resultado solo se necesitaba que se cumpliese la propiedad reproductora y ésta se tiene de (A.3), se sigue que para cualquier sucesión de Cauchy en $\mathcal{F}_1$ existe una función f tal que $\forall y \in E$ existe $f(y)$. Por tanto si se considera un nuevo conjunto de funciones $\mathcal{F}$ añadiendo a $\mathcal{F}_1$ todos los límites de sucesiones de Cauchy, se tiene que $\mathcal{F}$ es un espacio de Hilbert núcleo reproductor, ya que fijado $y \in E$ la sucesión $\{f_n\} \in \mathcal{F}_1$ verifica:

$$\lim_{n\to\infty} f_n(y) = \lim_{n\to\infty} \langle f_n(x), k(x,y)\rangle = \langle \lim_{n\to\infty} f_n(x), k(x,y)\rangle = \langle f(x), k(x,y)\rangle = f(y)$$

de donde se tiene que el operador evaluación esta acotado ya que toda sucesión convergente está acotada y de aquí

$$\|f_n\| \leq A \Rightarrow |\langle f_n(x), k(x,y)\rangle| \leq \|f_n\|\, k(y,y)^{1/2} \leq A \cdot M_y \quad \forall n \in \mathbb{N}.$$

■

Propiedad 5 *(Proyectividad) Si la clase de funciones $\mathcal{F}$ con núcleo reproductor $k(x,y)$ es un subespacio de un espacio de Hilbert $\mathcal{H}$ entonces la fórmula*

$$f(y) = \langle h(x), k(x,y)\rangle_x$$

proporciona la proyección de $h \in \mathcal{H}$ sobre $\mathcal{F}$.

Demostración. En efecto, si $\mathcal{H} = \mathcal{F} \oplus \mathcal{F}'$ donde $\mathcal{F}'$ es el complemento ortogonal de $\mathcal{F}$ en $\mathcal{H}$ entonces cualquier función $h \in \mathcal{H}$ se puede escribir de manera única en la forma: $h = f + g$, donde $f \in \mathcal{F}$ y $g \in \mathcal{F}'$. De aquí se tiene que como $k(x,y)$ pertenece a $\mathcal{F}$ como función de x, se cumple $\langle g(x), k(x,y)\rangle = 0$ y por tanto se tiene que:

$$\langle h(x), k(x,y)\rangle = \langle f(x) + g(x), k(x,y)\rangle = \langle f(x), k(x,y)\rangle = f(y)$$

por la propiedad reproductora. ■

Propiedad 6 *Si el espacio de Hilbert $\mathcal{F}$ posee un núcleo reproductor $k(x,y)$, entonces es el mismo para cualquier subespacio lineal cerrado $\mathcal{F}'$ de $\mathcal{F}$.*

Demostración. Esta propiedad es evidente ya que si para cada $y \in E$ fijado, se considera la función evaluador $P_y f = f(y)$, éste es un operador acotado dentro del espacio $\mathcal{F}$, y por tanto, también lo será dentro de cualquier subespacio $\mathcal{F}'$ de $\mathcal{F}$ lo que significa que $\mathcal{F}'$ es un espacio de Hilbert núcleo reproductor con núcleo reproductor k'; pero por otro lado por la unicidad de los núcleos reproductores se tiene que $k' = k$ en $\mathcal{F}'$. ■

Propiedad 7 *Si $\mathcal{F}$ posee un núcleo reproductor $k(x,y)$ y si $\{g_n\}$ es un sistema ortogonal en $\mathcal{F}$, entonces para cualquier sucesión de números $\{\alpha_n\}$ que cumplan:*

$$\sum_{n=1}^{\infty} |\alpha_n|^2 < \infty$$

se tiene que:

$$\sum_{n=1}^{\infty} |\alpha_n||g_n(x)| \leq k(x,x)^{1/2} \left(\sum_{n=1}^{\infty} |\alpha_n|^2 \right)^{1/2}.$$

Demostración. En efecto, para un valor $y \in E$ fijado, los coeficientes de Fourier de la función $k(x,y)$ (como función de x) para el sistema ortogonal $\{g_n\}$ son:

$$\langle k(x,y), g_n(x) \rangle = \langle g_n(x), k(x,y) \rangle = g_n(y).$$

En consecuencia

$$k(x,y) = \sum_{n=1}^{+\infty} \langle k(x,y), g_n(x) \rangle g_n(x) = \sum_{n=1}^{+\infty} g_n(y)\, g_n(x)$$

lo que implica que

$$k(y,y) = \sum_{n=1}^{+\infty} g_n^2(y).$$

Este último resultado nos asegura que la serie $\sum_{n=1}^{+\infty} g_n^2(y)$ es finita puesto que el núcleo reproductor esta definido para todo $x, y \in \mathbb{R}$, y por otro lado de la desigualdad de Cauchy-Schwarz se sigue:

$$\begin{aligned}\sum_{n=1}^{\infty} |\alpha_n||g_n(x)| &\leq \left(\sum_{n=1}^{\infty} |\alpha_n|^2\right)^{1/2} \left(\sum_{n=1}^{\infty} |g_n(x)|^2\right)^{1/2} \\ &\leq k(x,x)^{1/2} \left(\sum_{n=1}^{\infty} |\alpha_n|^2\right)^{1/2}\end{aligned}$$

y de aquí se tiene el resultado de la propiedad. ■

Propiedad 8 *Cualquier operador lineal acotado sobre un espacio de Hilbert núcleo reproductor $\mathcal{F}$ puede ser representado a partir del núcleo reproductor.*

Demostración. Sea un operador lineal acotado $P : \mathcal{F} \to \mathbb{R}$. Aplicando el teorema de representación de Riesz, existe una función evaluador $\eta \in \mathcal{F}$ que verifica:

$$P(f) = \langle \eta, f \rangle.$$

Por otro lado aplicando la propiedad reproductora se verifica:

$$\eta(x) = \langle \eta, k_x \rangle = P(k_x)$$

luego

$$P(f) = \langle \eta(x), f(x) \rangle = \langle P(k_x), f(x) \rangle \tag{A.4}$$

lo que significa que cualquier operador lineal acotado en $\mathcal{F}$ puede ser definido a partir del núcleo reproductor, esto significa que si se tiene definido el operador sobre el núcleo se tiene definido el operador para todas las funciones de $\mathcal{F}$. ■

Ejemplo A.1 *Sea $\mathcal{F}$ un espacio de Hilbert con núcleo reproductor $k(x, y)$ y producto escalar $\langle f, g \rangle = \int f(x)g(x)dx$.*

Si se considera el operador $P(f) = \int w(y)\, f(y)\, dy$ *con la función* $w \in \mathcal{F}$ *fijada. Se tiene entonces que la función evaluador es* $\eta(x) = \int w(y)\, k(x,y)\, dy$*, y aplicando (A.4):*

$$P(f) = \int \left(\int w(y)\, k(x,y)\, dy \right) f(x)\, dx.$$

Si se considera el operador $P(f) = f'(x_0)$*, con* $x_0 \in E$ *un valor concreto, entonces* $\eta(x) = \left[\dfrac{\partial k(x,y)}{\partial y} \right]_{y=x_0}$ *y aplicando (A.4):*

$$P(f) = \int \left[\frac{\partial k(x,y)}{\partial y} \right]_{y=x_0} f(x)\, dx$$

lo que significa que:

$$f'(x_0) = \int \left[\frac{\partial k(x,y)}{\partial y} \right]_{y=x_0} f(x)\, dx$$

▲

A.3 Núcleos reproductores sobre clases de dimensión finita

Sea $\mathcal{F}$ una clase lineal de funciones de dimensión finita n y $\{w_1(x), \cdots, w_n(x)\}$ una base de $\mathcal{F}$ (como espacio vectorial). Entonces se tiene que toda función $f(x) \in \mathcal{F}$ se puede representar de manera única como:

$$f(x) = \sum_{k=1}^{n} a_k\, w_k(x), \qquad a_k \in \mathbb{R}, \quad k = 1, \cdots, n. \tag{A.5}$$

En virtud de esta representación de una función como combinación lineal de los elementos de una base se tiene que la forma más general de expresar cualquier norma cuadrática en $\mathcal{F}$ viene dada por:

$$\|f\|^2 = \sum_{i,j=1}^{n} \alpha_{ij}\, a_i\, a_j, \qquad \text{donde } \alpha_{ij} \in \mathbb{R}, \quad \forall i, j = 1, 2 \cdots, n \tag{A.6}$$

y el producto escalar asociado:

$$\langle f, g\rangle = \sum_{ij}^{n} \alpha_{ij}\, a_i\, b_j \quad \text{donde} \quad g(x) = \sum_{i=1}^{n} b_j\, w_j(x) \tag{A.7}$$

y claramente se tiene que los coeficientes $\alpha_{ij} = \langle w_i, w_j\rangle$ sin más que tomar en la igualdad (A.7) las funciones $f(x) = w_i(x)$ y $g(x) = w_j(x)$ como caso particular. Además la matriz $\Lambda = \{\alpha_{ij}\}$ es la matriz de Gram del sistema $\{w_k\}_{k=1}^{n}$, y ésta siempre posee inversa, ya que es posible aplicar el método de ortonormalización de Gram-Schmidt a la matriz Λ, obteniendo otra con determinante no nulo (regular). Si se denota a la matriz inversa de Λ por $\Lambda^{-1} = \{\beta_{ij}\}_{i,j=1}^{n}$, la cual es simétrica por serlo Λ, se sigue que:

$$\sum_{j=1}^{n} \alpha_{ij}\, \beta_{jk} = \delta_{ik}, \tag{A.8}$$

y se tiene que la función definida como:

$$k(x, y) = \sum_{i,j=1}^{n} \beta_{ij}\, w_i(x)\, w_j(y) \tag{A.9}$$

es el núcleo reproductor de la clase de dimensión finita $\mathcal{F}$ con la norma (A.6).

La demostración de este hecho se sigue de:

$$\begin{aligned}
\langle k(x,y), f(x)\rangle_x &= \langle \sum_{i,j=1}^{n} \beta_{ij}\, w_i(x)\, w_j(y), \sum_{k=1}^{n} a_k\, w_k(x)\rangle_x \\
&= \sum_{i,j,k=1}^{n} a_k\, \beta_{ij}\, w_j(y)\, \langle w_i(x), w_k(x)\rangle_x \\
&= \sum_{i,j,k=1}^{n} a_k\, w_j(y)\, \beta_{ij}\, \alpha_{ik} \qquad \text{y ya que } \beta_{ij} \text{ es simétrica} \\
&= \sum_{j,k=1}^{n} a_k\, w_j(y) \sum_{i=1}^{n} \beta_{ji}\, \alpha_{ik} \\
&= \sum_{j,k=1}^{n} a_k\, w_j(y)\, \delta_{jk} \qquad \text{por (A.8)} \\
&= \sum_{j=1}^{n} a_j\, w_j(y) \quad = \quad f(y)
\end{aligned}$$

y además se tiene que la matriz $\{\beta_{ij}\}_{i,j=1}^n$ es definida positiva, por ser la inversa de la matriz Λ que lo es, ya que $a'\Lambda a = \|f\|^2 \geq 0$, con $f(x) = \sum_{i=1}^n a_i\, w_i(x)$ y $a' = (a_1, \cdots, a_n)$ (ver [Har97], pág. 214).

Todos estos resultados proporcionan el siguiente teorema:

Teorema A.3.1 *Una función $k(x, y)$ es el núcleo reproductor de una clase de funciones de dimensión finita n si y solo si es de la forma*

$$k(x, y) = \sum_{i,j=1}^{n} \beta_{ij}\, w_i(x)\, w_j(y)$$

con una matriz definida positiva $\{\beta_{ij}\}$ y las funciones $\{w_k(x)\}_{k=1}^n$ linealmente independientes. La correspondiente clase $\mathcal{F}$ es entonces generada por las funciones $\{w_k(x)\}_{k=1}^n$, las funciones $f \in \mathcal{F}$ están dadas por (A.5) y su correspondiente norma por (A.6), donde $\{\alpha_{ij}\}$ es la matriz inversa de $\{\beta_{ij}\}$.

A.4 Completitud de una clase de funciones

En las aplicaciones es frecuente considerar clases de funciones que forman un espacio vectorial y poseen un producto escalar pero no es completo en el sentido que existen sucesiones de Cauchy que no convergen a una función de la clase (se dice que no cumplen la propiedad de completitud). En estos casos surgen dos problemas:

- El problema de completar la clase para obtener una clase de funciones que tenga estructura de espacio de Hilbert, y
- Decidir, antes de completar la clase, si la clase completa va a poseer o no un núcleo reproductor.

Normalmente para completar una determinada clase de funciones lo que se hace es añadirle unos nuevos elementos, los límites de todas las sucesiones de Cauchy, obteniendo de esta forma otro espacio que ya es completo puesto que contiene a $\mathcal{F}$

y además, por construcción, $\mathcal{F}$ es un subconjunto denso en él. Sin embargo este nuevo espacio, en general, no forma un espacio de Hilbert de funciones con núcleo reproductor ya que esta forma de completar destruye las propiedades de continuidad entre los valores de la función y la convergencia en el espacio, es decir, si se tiene una sucesión de Cauchy $\{f_n\}$ lo que se hace es incluir la función $f = \lim f_n$ en la clase que completa a $\mathcal{F}$, pero el problema surge cuando no se garantiza que $\forall y \in E$ exista el valor[7] $f(y)$, lo cual trae como consecuencia que el operador $P_y f = f(y)$ no este acotado y de aquí se tendría por la propiedad (2) la no existencia de un núcleo reproductor.

Cuando se completa una clase lineal de funciones $\mathcal{F}$ para evitar el inconveniente anterior, se añaden todas aquellas funciones tales que los valores de la función f de la clase completa en un punto dependan continuamente de f (se dirá que se está realizando una completitud funcional). De aquí, por la propiedad (2) de existencia de un núcleo reproductor, se deduce que la clase completa tiene un núcleo reproductor[8], es decir el problema de la posibilidad de completitud de la clase de funciones y de existencia de núcleo reproductor se unen en uno mismo.

Teorema A.4.1 (de completitud) *Sea $\mathcal{F}$ una clase de funciones que forma un espacio normado no completo. Para que exista una completitud funcional de $\mathcal{F}$ es necesario y suficiente que:*

1. *$\forall y \in E$: el operador lineal $P : \mathcal{F} \to \mathbb{R}$ tal que $P(f) = f(y)$ este acotado; y*

2. *$\forall \{f_m\} \subset \mathcal{F}$ sucesión de Cauchy tal que $f_m(y) \to 0 \quad \forall y \in E$ implica que $\|f_m\| \to 0$.*

Además, si la completitud funcional existe, es única.

[7] Si por ejemplo $f_n(x) = x^n$ se tiene que $\lim f_n(2) = \lim 2^n = +\infty \Rightarrow \nexists f(2)$.

[8] Si f es una función continua en un punto y entonces existe $f(y)$ y por tanto el operador lineal está acotado.

Demostración. $\Rightarrow$) La primera condición es inmediata por el teorema de existencia de un núcleo reproductor.
La necesidad de la segunda condición se sigue del hecho que las sucesiones de Cauchy $\{f_n\}$ de $\mathcal{F}$ presentan una convergencia fuerte (convergencia en norma) en el espacio completo hacia una función f; y la función f es el límite de f_n en cada punto $y \in E$ (la convergencia en norma implica la convergencia puntual en los espacios de Hilbert núcleo reproductor). En consecuencia, $f \equiv 0$ y la norma de f_n converge a la norma de f la cual es igual a 0.

$\Leftarrow$) Sea $\{f_n\} \subset \mathcal{F}$ una sucesión de Cauchy. Para cualquier $y \in E$ se denota por M_y la cota del operador funcional $P(f) = f(y)$, es decir

$$|f(y)| \leq M_y \|f\|, \qquad \forall f \in \mathcal{F}. \tag{A.10}$$

De donde se tiene: $|f_n(y) - f_m(y)| \leq M_y \|f_n - f_m\|$ y por tanto $\{f_n(y)\}$ es una sucesión de Cauchy en $\mathbb{R}$, y converge a un número al cual se le denota por $f(y)$. De esta forma la sucesión de Cauchy $\{f_n\}$ define una función f hacia la cual converge $\forall y \in E$.

Sea la clase de todas las funciones f, límites de las sucesiones de Cauchy $\{f_n\} \subset \mathcal{F}$. Es inmediato ver que es una clase lineal (cualquier combinación lineal de funciones de la clase pertenece a la clase) de funciones y que contiene a $\mathcal{F}$. Se considera en esta clase, que se denotará por $\overline{\mathcal{F}}$, la norma (se ve en el teorema que ciertamente lo es):

$$\|f\|_1 = \lim \|f_n\|, \quad \forall \{f_n\} \subset \mathcal{F} \text{ de Cauchy} \quad / \, f_n(y) \to f(y) \quad \forall y \in E. \tag{A.11}$$

Esta norma no depende de la elección de la sucesión de Cauchy, es decir se encuentra bien definida. Sea $\{f'_n\}$ otra sucesión de Cauchy convergente a la función $f(y)$, $\forall y \in E$, entonces $|f_n(y) - f'_n(y)| \to 0$ y por la segunda condición se tiene que $\|f_n - f'_n\| \to 0$ y por tanto

$$|\lim \|f'_n\| - \lim \|f_n\|| = \lim |\|f_n\| - \|f'_n\|| \leq \lim \|f_n - f'_n\| = 0.$$

Por otro lado, es fácil ver que $\|f\|_1^2$ es una norma positiva en la clase $\overline{\mathcal{F}}$, ya que

es cero para $f \equiv 0$ y positiva si $f \neq 0$ por (A.10). Resta probar que $\overline{\mathcal{F}}$ es completo y que contiene a $\mathcal{F}$ como subespacio denso.

Es inmediato probar que $\mathcal{F} \subset \overline{\mathcal{F}}$ ya que para cada elemento de $f \in \mathcal{F}$, las normas $\|\cdot\|, \|\cdot\|_1$ coinciden sin más que tomar $f_n = f, \forall n \in \mathbb{N}$ y además toda función $f \in \overline{\mathcal{F}}$ es, por definición, el límite de una sucesión de Cauchy $\{f_n\} \subset \mathcal{F}$. Se sigue que f es el límite fuerte de f_n en $\overline{\mathcal{F}}$ ya que se tiene de (A.11)

$$\lim_{n\to\infty} \|f - f_n\|_1 = \lim_{n\to\infty} \lim_{m\to\infty} \|f_m - f_n\|$$

Falta probar la completitud de la clase $\overline{\mathcal{F}}$. Sea $\{f_n\} \subset \overline{\mathcal{F}}$ una sucesión de Cauchy. Como $\mathcal{F}$ es denso en $\overline{\mathcal{F}}$, se puede encontrar una sucesión de Cauchy $\{f'_{nm}\} \subset \mathcal{F}$ tal que $\lim_{m\to\infty} \|f'_{nm} - f_n\|_1 = 0$. Ahora la sucesión $\{f'_{nn}\}$ converge a una función $f \in \overline{\mathcal{F}}$, esta convergencia es en principio una convergencia puntual en todo E, pero a la vista de los anteriores argumentos se tiene que f'_{nn} también converge fuertemente a f en el espacio $\overline{\mathcal{F}}$. Se sigue entonces que $f_n \to f$ en norma (convergencia fuerte). La unicidad de la clase completa es clara ya que en la clase completa una función f debe ser necesariamente el límite de una sucesión de Cauchy $\{f_n\} \subset \mathcal{F}$.

Ya que el núcleo reproductor debe existir para la clase completa, esto implica que f es un límite de cierta sucesión de Cauchy $\{f_n\}$ lo cual significa que pertenece a la clase $\overline{\mathcal{F}}$. Como la norma de f tiene que ser el límite de $\|f_n\|$, necesariamente coincide con $\|f\|_1$. Es también claro que para toda función $f \in \overline{\mathcal{F}}$ debe pertenecer a la clase completa. En suma se tiene que cualquier completitud funcional de $\mathcal{F}$ debe coincidir con $\overline{\mathcal{F}}$ y tener la misma norma y producto escalar que $\overline{\mathcal{F}}$. ■

En [Aro50] se demuestra que la segunda condición del teorema no puede ser excluida, ya que, si no se da, significa que las funciones de la clase incompleta están definidas en un conjunto E demasiado pequeño, pero aun en estos casos se puede actuar de la siguiente forma. Dada una clase incompleta $\mathcal{F}$ se considera una completitud de esta clase añadiendo los elementos que se indican en el teorema anterior. Con esta completitud se tiene un espacio de Hilbert $\mathcal{H}$. A cada elemento de $\mathcal{H}$ le corresponde una función $f(x)$ bien definida sobre E. Pero el problema ahora

es que a diferentes elementos de $\mathcal{H}$ le puede corresponder la misma función f. Como $\forall y \in E$ el operador $P(f) = f(y)$ es lineal y acotado en $\mathcal{H}$ se tiene que $f(y) = (\overline{f}, h_y)$, donde $\overline{f}$ es cualquiera de los elementos de $\mathcal{H}$ correspondiente a la función f. Como existen elementos de $\mathcal{H}$ diferentes del elemento cero a los cuales les corresponden la función idénticamente nula en E, es claro que el conjunto de elementos h_y no esta completo en el espacio de Hilbert $\mathcal{H}$. Al conjunto de elementos h_y se le puede entonces añadir un conjunto adicional de elementos tales que se obtenga un conjunto completo en $\mathcal{H}$. Este conjunto adicional se le denota por E'. Ahora se extienden las funciones de la clase $\mathcal{F}$ en el conjunto $E \cup E'$ definiendo $\forall y \in E' : f(h) = (f, h)$. Esta clase de funciones, extendidas en $E \cup E'$, satisface la condición segunda.

A.5 La restricción de un núcleo reproductor

Teorema A.5.1 *Si k es el núcleo reproductor de una clase $\mathcal{F}$ de funciones definidas en un conjunto E con la norma $\|\cdot\|$, entonces la restricción de k a un subconjunto $E_1 \subset E$ es el núcleo reproductor de la clase $\mathcal{F}_1$ de todas las restricciones de las funciones de $\mathcal{F}$ en el conjunto E_1. Además se tiene que $\forall f_1 \in \mathcal{F}_1$, la norma $\|f_1\|$ es la mínima de $\|f\|$, $\quad \forall f \in \mathcal{F}$ cuya restricción en E_1 es f_1, es decir*

$$\|f_1\| = \min_{f \in \mathcal{F}} \{\|f\| \,, \, / \, f(x) = f_1(x), \, \forall x \in E_1\}$$

Demostración. Sea el subespacio lineal

$$\mathcal{F}_0 = \{f \in \mathcal{F} \, / f(x) = 0, \, \forall x \in E_1\} \subset \mathcal{F}.$$

Este subespacio es completo ya que si se tiene una sucesión de Cauchy $\{f_n\}$ de funciones de $\mathcal{F}_0$ se tiene, que como la clase $\mathcal{F}$ es completa, existe una función $f \in \mathcal{F}$ tal que $f = \lim_{n\to\infty} f_n$, y, como la convergencia en norma implica la convergencia puntual en estos espacios, se verifica que $f(x) = \lim_{n\to\infty} f_n(x)$ para cualquier $x \in E$, en particular si $x \in E_1$ se tiene que, como para todo n es $f_n(x) = 0$, se verifica $f(x) = 0$, $\forall x \in E_1$ lo que significa que $f \in \mathcal{F}_0$.

Sea el $\mathcal{F}'$ el complemento ortogonal de $\mathcal{F}_0$, el cual por construcción también es completo. Como tanto $\mathcal{F}_0$ como $\mathcal{F}'$ son subespacios completos de un espacio de Hilbert núcleo reproductor se verifica que ellos también son espacios de Hilbert núcleo reproductor. Sean k_0 y k' los núcleos reproductores de $\mathcal{F}_0$ y $\mathcal{F}'$ respectivamente.

Cualquier función $f \in \mathcal{F}$ se puede representar de manera única en la forma $f = f_0 + f'$ donde $f_0 \in \mathcal{F}_0$ y $f' \in \mathcal{F}'$. Aplicando la propiedad (5) se tiene que para cualquier $y \in E$ fijado:

$$f_0(y) = \langle f(x), k_0(x,y)\rangle$$
$$f'(y) = \langle f(x), k'(x,y)\rangle$$

lo que implica que

$$\begin{aligned} f(y) &= \langle f(x), k(x,y)\rangle = f_0(y) + f'(y) \\ &= \langle f(x), k_0(x,y)\rangle + \langle f(x), k'(x,y)\rangle \\ &= \langle f(x), k_0(x,y) + k'(x,y)\rangle \end{aligned}$$

y por la unicidad del núcleo reproductor se tiene que:

$$k(x,y) = k_0(x,y) + k'(x,y), \qquad \forall x, y \in E. \tag{A.12}$$

Como $k_0(x,y) \in \mathcal{F}_0$, $\forall y$ fijo se tiene que $k_0(x,y) = 0, \quad \forall x \in E_1$, de donde:

$$k(x,y) = k'(x,y) \quad \forall x \in E_1. \tag{A.13}$$

Si se considera la clase $\mathcal{F}_1$ de todas las restricciones de $\mathcal{F}$ en el conjunto E_1, se tiene que si dos funciones $f, g \in \mathcal{F}$ tienen la misma restricción $f_1(x)$ en E_1 lo que implica que $f(x) - g(x) = 0, \forall\, x \in E_1$ y de aquí $f - g \in \mathcal{F}_0$.

Recíprocamente, si $f - g \in \mathcal{F}_0$ entonces $f = g$ en E_1, es claro que todas las funciones $f \in \mathcal{F}$ las cuales tienen la misma restricción f_1 en E_1 tienen la misma proyección f_1' sobre $\mathcal{F}'$ y la restricción de f_1' en E_1 es igual también a f_1. Además se deduce claramente que de todas estas funciones, f_1' es la que tiene menor norma, por tanto por la definición del teorema se puede escribir

$$\|f_1\|_1 = \|f_1'\|. \tag{A.14}$$

Así se establece una aplicación entre $\mathcal{F}_1$ y $\mathcal{F}'$ que a cada $f_1 \in \mathcal{F}_1$ le asocia $f' \in \mathcal{F}'$ se tiene una congruencia[9] entre los espacios $\mathcal{F}_1$ con la norma $\|\cdot\|_1$ y $\mathcal{F}'$ con la norma $\|\cdot\|$.

Se demuestra también que k es el núcleo reproductor de la clase $\mathcal{F}_1$ con la norma $\|\cdot\|_1$ restringido a E_1. Esto se deduce como sigue: si para cada $f_1 \in \mathcal{F}_1$ se toma la correspondiente función $f_1' \in \mathcal{F}'$ entonces $\forall y \in E_1$:

$$f_1(y) = f_1'(y) = \langle f_1'(x), k'(x,y)\rangle = \langle f_1(x), k_1(x,y)\rangle_1$$

donde $k_1(x,y)$ es la restricción de $k(x,y)$ (considerada como función de x) en el conjunto E_1. Por simetría y por (A.13) se tiene que $k(x,y) = k'(x,y)\,, \quad \forall y \in E_1$ y con esto se prueba que la restricción $k_1(x,y)$ de $k'(x,y)$ coincide con la restricción de k en el conjunto E_1. ■

Si la clase de funciones $\mathcal{F}_0$ se reduce a la función nula, es decir, $\mathcal{F}_0 = \{f_0\}$ donde $f_0(x) = 0$ para todo x, entonces la norma $\|\cdot\|_1$ es muy simple, ya que en este caso $\mathcal{F}' = \mathcal{F}$ y cada $f \in \mathcal{F}_1$ es la restricción de f si y solo si la función $f \in \mathcal{F}$ y además $\|f\|_1 = \|f\|$ para las funciones f con restricción f_1.

A.6 Suma de núcleos reproductores

Teorema A.6.1 *Sea $k_i(x,y)$ el núcleo reproductor del espacio de Hilbert núcleo reproductor de funciones $\mathcal{F}_i$ con norma $\|\cdot\|_i$, $\forall i = 1, \cdots, n$ entonces la función $k(x,y) = \sum_{i=1}^{n} k_i(x,y)$ es el núcleo reproductor del espacio de Hilbert núcleo reproductor de funciones $\mathcal{F}$ formado por todas las funciones $f = \sum_{i=1}^{n} f_i$ con $f_i \in \mathcal{F}_i$ y la norma definida por*

$$\|f\|^2 = \text{mín}[\|f_1\|^2 + \cdots + \|f_n\|^2] \tag{A.15}$$

[9] Una congruencia es una correspondencia isométrica biyectiva, es decir, es una correspondencia que respecta la métrica de los espacios y a la vez es inyectiva y sobreyectiva.

donde el mínimo se refiere a todas las posibles descomposiciones de f *en la forma* $\sum_{i=1}^{n} f_i$.

Demostración. La demostración se realizará para $n = 2$ y de ésta se verá que la generalización al caso n es trivial.

Lo primero que se debe ver es cual es la clase $\mathcal{F}$ y posteriormente comprobar que (A.15) es la norma correspondiente a $k(x, y)$. Sea el espacio[(10)] de Hilbert $\mathcal{H}$ formado por todos los pares $\{f_1, f_2\}$ con $f_i \in \mathcal{F}_i$. La norma considerada en este espacio viene dada por:

$$\|\{f_1, f_2\}\|^2 = \|f_1\|_1^2 + \|f_2\|_2^2.$$

Sea la clase de funciones $\mathcal{F}_0 = \{f \ / \ f \in \mathcal{F}_1, f \in \mathcal{F}_2\}$ ($\mathcal{F}_0 = \mathcal{F}_1 \bigcap \mathcal{F}_2$) y sea $\mathcal{H}_0$ el conjunto formado por los pares de la forma $\{f, -f\}$ con $f \in \mathcal{F}_0$. Es claro que $\mathcal{H}_0$ es un subespacio lineal y además es cerrado ya que si $\{f_n, -f_n\} \to \{f', f''\}$ entonces f_n y $-f_n$ convergen en norma a f' y f'' respectivamente. En consecuencia f_n converge puntualmente a f' y $-f_n$ a f'', lo que significa que $f'' = -f'$ y $f', f'' \in \mathcal{F}_0$.

Como $\mathcal{H}_0$ es un subespacio cerrado de $\mathcal{H}$ se considera el subespacio complementario $\mathcal{H}'$ por lo cual $\mathcal{H} = \mathcal{H}_0 \oplus \mathcal{H}'$. Se define la aplicación tal que a cada elemento de $\mathcal{H}$, $\{f', f''\}$ le hace corresponder la función $f(x) = f'(x) + f''(x)$. Claramente se tiene que es una aplicación lineal que transforma el espacio $\mathcal{H}$ en una clase lineal de funciones $\mathcal{F}$. Los elementos de $\mathcal{H}$ los cuales se transforman por esta aplicación en la función cero son los elementos de $\mathcal{H}_0$. Por tanto, esta aplicación transforma $\mathcal{H}'$ en una inyección dentro de $\mathcal{F}$. La aplicación inversa de ésta transforma cualquier función $f \in \mathcal{F}$ distinta de la función nula en un elemento $\{g'(f), g''(f)\} \in \mathcal{H}'$. Se define la norma en $\mathcal{F}$ por:

$$\|f\|^2 = \|\{g'(f), g''(f)\}\|^2 = \|g'(f)\|_1^2 + \|g''(f)\|_2^2.$$

Ahora se va a demostrar que a la clase $\mathcal{F}$ con esta norma le corresponde el núcleo reproductor $K = k_1 + k_2$ y para ello se tiene que ver que:

[(10)] La demostración de que es espacio de Hilbert es fácil.

1. $k(x,y) \in \mathcal{F}$ como función de x con y fijado, es decir $\{k_1(x,y), k_2(x,y)\} \in \mathcal{H}$.

2. Para y fijado, $k'(x,y) = g'(k(x,y))$ y $k''(x,y) = g''(k(x,y))$.

Si $\forall f \in \mathcal{F}$ se denota $f' = g'(f)$ y $f'' = g''(f)$ entonces $f(y) = f'(y) + f''(y)$, $k'(x,y) + k''(x,y) = k(x,y) = k_1(x,y) + k_2(x,y)$ y por tanto

$$k''(x,y) - k_2(x,y) = -[k'(x,y) - k_1(x,y)] \Rightarrow$$

$$\{k_1(x,y) - k'(x,y), k_2(x,y) - k''(x,y)\} \in \mathcal{H}_0.$$

De donde se tiene que

$$\begin{aligned} f(y) &= f'(y) + f''(y) \\ &= \langle f'(x), k_1(x,y)\rangle_1 + \langle f''(x), k_2(x,y)\rangle_2 \\ &= \langle \{f', f''\}, \{k_1(x,y), k_2(x,y)\}\rangle \\ &= \langle \{f', f''\}, \{k'(x,y), k''(x,y)\}\rangle \\ &\quad + \langle \{f', f''\}, \{k_1(x,y) - k'(x,y), k_2(x,y) - k''(x,y)\}\rangle. \end{aligned}$$

El último producto escalar es cero ya que $\{k_1(x,y) - k'(x,y), k_2(x,y) - k''(x,y)\} \in \mathcal{H}_0$ y $\{f', f''\} \in \mathcal{H}'$ y estos espacios son ortogonales. El primer producto escalar del último miembro es por definición igual a $\langle f(x), k(x,y)\rangle$ lo que prueba la propiedad reproductora de $k(x,y)$.

Se define en el teorema la clase $\mathcal{F}$ como la clase de todas las funciones $f(x) = f_1(x) + f_2(x)$ con $f_i \in \mathcal{F}_i$, y norma dada por (A.15). Para probar la equivalencia de esta definición y todo lo anterior solo se tiene que recordar que a $f(x)$ le corresponde un elemento $\{f', f''\} \in \mathcal{H}$ y también otro $\{g'(f), g''(f)\} \in \mathcal{H}'$, es decir $f = f_1 + f_2 = g'(f) + g''(f)$; de donde $f_2 - g''(f) = -(f_1 - g'(f))$ luego $\{f_1 - g'(f), f_2 - g''(f)\} \in \mathcal{H}_0$ y

$$\|f_1\|_1^2 + \|f_2\|_2^2 = \|\{f_1, f_2\}\|^2 = \|\{g'(f), g''(f)\}\|^2 + \|\{f_1 - g'(f), f_2 - g''(f)\}\|^2$$

y para esta expresión obviamente el valor mínimo se da si y solo si $f_1 = g'(f)$, $f_2 = g''(f)$ y este valor viene dado entonces por la norma $\|\{g'(f), g''(f)\}\|^2$ la cual es por definición $\|f\|^2$. ■

A.7 Producto de núcleos reproductores

Sean los espacios de Hilbert núcleo reproductor de funciones $\mathcal{F}_1$ y $\mathcal{F}_2$, con normas $\|\cdot\|_1 = \|\cdot\|_{\mathcal{F}_1}$ y $\|\cdot\|_2 = \|\cdot\|_{\mathcal{F}_2}$ y núcleos reproductores k_1 y k_2, respectivamente. Se construye el producto directo[11] $\mathcal{F}'$ de los espacios de Hilbert $\mathcal{F}_1$ y $\mathcal{F}_2$, $\mathcal{F}' = \mathcal{F}_1 \otimes \mathcal{F}_2$ de la siguiente forma:
Sea el conjunto producto $E' = E \otimes E$ de todos los pares $\{x_1, x_2\}$ con $x_i \in E$, $i = 1, 2$. Sobre el conjunto E' se considera la clase de todas las funciones $f'(x_1, x_2)$ representables en la forma:

$$f'(x_1, x_2) = \sum_{j=1}^{n} f_1^{(j)}(x_1) f_2^{(j)}(x_2), \tag{A.16}$$

con $f_1^{(j)} \in \mathcal{F}_1$ y $f_2^{(j)} \in \mathcal{F}_2$ para $j = 1, 2, \cdots, n$. Como producto escalar de dos funciones se define:

$$\langle f', g' \rangle = \sum_{j=1}^{n} \sum_{\ell=1}^{m} \langle f_1^{(j)}, g_1^{(\ell)} \rangle \langle f_2^{(j)}, g_2^{(\ell)} \rangle, \tag{A.17}$$

donde m es el número de términos en la representación de g'.

Se ha de demostrar que ciertamente (A.17) es un producto escalar. Para ello, se ha de ver en primer lugar que esta bien definido, ya que una misma función f' puede admitir diferentes representaciones de la forma (A.16), es decir hay que demostrar que el producto escalar, $\langle f', g' \rangle$ es independiente de la particular representación elegida de f' y g' para que se encuentre bien definido. Pero esto es inmediato ya que la expresión del producto escalar (A.17) se puede expresar en la forma:

$$\langle f', g' \rangle = \sum_{\ell=1}^{m} \langle \langle f'(x_1, x_2), g_1^{(\ell)}(x_1) \rangle_1, g_2^{(\ell)}(x_2) \rangle_2 \tag{A.18}$$

lo cual demuestra que $\langle f', g' \rangle$ es independiente de la particular elección de f'. De manera similar se demuestra que también es independiente de la particular elección de g'.

[11] Para la noción de producto directo de espacios de Hilbert abstractos se puede ver [NM36].

De la propia definición se tiene que ciertamente el producto escalar definido en (A.17) es una forma bilineal, es decir

$$\langle \alpha_1 f' + \alpha_2 f'', \beta_1 g' + \beta_2 g'' \rangle = \alpha_1\beta_1\langle f', g'\rangle + \alpha_1\beta_2\langle f', g''\rangle + \alpha_2\beta_1\langle f'', g'\rangle + \alpha_2\beta_2\langle f'', g''\rangle$$

Para completar que ciertamente es un producto escalar queda ver que es definida positiva, es decir $\langle f', f'\rangle \geq 0$ para cualquier f' definida por (A.16); y además el producto escalar es nulo cuando $f' = 0$. Para demostrar esto, se toma una representación cualquiera de f' del tipo (A.16) y sean los conjuntos $\left\{f_1^{(j)}\right\}$ y $\left\{f_2^{(j)}\right\}$ en los espacios $\mathcal{F}_1$ y $\mathcal{F}_2$ las funciones que representan a f'. Estos conjuntos se ortonormalizan utilizando el método de Gram-Schmidt y se obtienen dos nuevos conjuntos ortonormales $\left\{f_1^{(j,1)}\right\}$ y $\left\{f_2^{(\ell,1)}\right\}$ donde $j = 1, \cdots, n_1$, $\ell = 1, \cdots, n_2$.
De esta forma, cada función $\left\{f_i^{(k')}\right\}$ se puede representar como una combinación lineal de las funciones ortonormales $f_i^{(k',1)}$, con lo que se consigue tener una representación de f' como una suma doble:

$$f'(x_1, x_2) = \sum_{j=1}^{n_1}\sum_{\ell=1}^{n_2} \alpha_{j,\ell} f_1^{(j,1)}(x_1) f_2^{(\ell,1)}(x_2), \tag{A.19}$$

donde $\alpha_{j,\ell} \in \mathbb{R}$. Entonces se tiene la siguiente expresión para el producto escalar $\langle f', f'\rangle$ dado en (A.17):

$$\begin{aligned} \langle f', f'\rangle &= \sum_{j=1}^{n_1}\sum_{\ell=1}^{n_2}\sum_{j'=1}^{n_1}\sum_{\ell'=1}^{n_2} \alpha_{j,\ell}\alpha_{j',\ell'}\langle f_1^{(j,1)}, f_1^{(j',1)}\rangle_1 \langle f_2^{(\ell,1)}, f_2^{(\ell',1)}\rangle_2 \\ &= \sum_{j=1}^{n_1}\sum_{\ell=1}^{n_2} |\alpha_{j,\ell}|^2 . \end{aligned} \tag{A.20}$$

Utilizando esta nueva representación queda claro que $\langle f', f'\rangle \geq 0$ e igual a cero solo cuando todos los $\alpha_{k,\ell} = 0$, es decir cuando $f' = 0$.

La clase de todas las funciones f' del tipo (A.16) no forma, en general, un espacio de Hilbert ya que puede no ser completo. Para completar esta clase con respecto a la norma[12] $\|\cdot\|'$, se considera una sucesión ortonormal completa $\left\{g_i^{(k)}\right\}$ en el

[12] La norma que se sigue del anterior producto escalar sobre la clase de todas las funciones representables en la forma A.16.

espacio $\mathcal{F}_i$, $i = 1, 2$. Evidentemente la sucesión $\left\{g_1^{(j)}(x_1) \cdot g_2^{(j)}(x_2)\right\}$ esta compuesta de funciones del tipo (A.16) (ya que coincide con una de estas representaciones con $n = 1$) y es ortonormal con respecto a la norma $\|\cdot\|'$ ya que:

$$\langle g_1^{(j)} \cdot g_2^{(j)}, g_1^{(\ell)} \cdot g_2^{(\ell)} \rangle = \langle g_1^{(j)}, g_1^{(\ell)} \rangle_1 \cdot \langle g_2^{(j)}, g_2^{(\ell)} \rangle_2 = \delta_{j\ell} \cdot \delta_{j\ell} = \delta_{j\ell}.$$

Si se considera el conjunto de todas las funciones g' representables en la forma:

$$g'(x_1, x_2) = \sum_{j=1}^{\infty} \sum_{\ell=1}^{\infty} \alpha_{j,\ell}\, g_1^{(j)}(x_1) g_2^{(\ell)}(x_2) \tag{A.21}$$

donde

$$\langle g', g' \rangle = \sum_{j=1}^{\infty} \sum_{\ell=1}^{\infty} |\alpha_{j,\ell}|^2 < \infty, , \qquad \text{con } \alpha_{j\ell} \in \mathbb{R}. \tag{A.22}$$

Claramente si la suma en (A.21) es finita[13] entonces es del tipo (A.16) sin más que considerar por ejemplo:

$$f_1^{(j)} = \alpha_{j\ell}\, g_1^{(j)} \text{ y } f_2^{(j)} = g_2^{(j)},$$

para aquellos índice cuyo término es no nulo; y la norma $\|\cdot\|'$ para tales sumas finitas coincide con la norma introducida en (A.22).

Se demuestra en primer lugar que cada una de las sumas del tipo (A.21) es absolutamente convergente para cada par (x_1, x_2). Esto es verdad ya que como la clase $\mathcal{F}_1$ posee núcleo reproductor k_1, y teniendo en cuenta la propiedad 7 se sigue que:

$$\sum_{j=1}^{\infty} |\alpha_{j,\ell}| \left|g_1^{(j)}(x_1)\right| \leq [k_1(x_1, x_1)]^{1/2} \left[\sum_{j=1}^{\infty} |\alpha_{j,\ell}|^2\right]^{1/2}. \tag{A.23}$$

Entonces,

$$\begin{aligned} \sum_{j=1}^{\infty} \sum_{\ell=1}^{\infty} |\alpha_{j,\ell}| \left|g_1^{(j)}(x_1)\right| \left|g_2^{(\ell)}(x_2)\right| &\leq \sum_{\ell=1}^{\infty} \left|g_2^{(\ell)}(x_2)\right| [k_1(x_1, x_1)]^{1/2} \left[\sum_{j=1}^{\infty} |\alpha_{j,\ell}|^2\right]^{1/2} \\ &\leq [k_1(x_1, x_1)]^{1/2} [k_2(x_2, x_2)]^{1/2} \left[\sum_{j=1}^{\infty} \sum_{\ell=1}^{\infty} |\alpha_{j,\ell}|^2\right]^{1/2} \end{aligned} \tag{A.24}$$

[13] Existe un número finito de términos no nulos.

ya que como la clase $\mathcal{F}_2$ es un espacio de Hilbert con núcleo reproductor k_2 se cumple también la desigualdad (A.23).

Se ha demostrado que la clase de todas las funciones g' de la forma (A.21) forma un espacio de Hilbert, y además es isomórfo al espacio de la sucesión doble $\{\alpha_{j,\ell}\}$ cumpliendo (A.22), es decir se puede establecer una correspondencia biunívoca[14] (biyectiva) entre ambos espacios que conserva las operaciones.

La desigualdad (A.24) que se ha obtenido permite además, para una función g' del tipo (A.21), tener que:

$$|g'(y_1, y_2)| < k_1(y_1, y_1)^{1/2}\, k_2(y_2, y_2)^{1/2}\ \|g'\|', \tag{A.25}$$

lo que significa que el espacio de funciones de tipo (A.21) posee un núcleo reproductor, ya que (A.25) significa que fijado $y = (y_1, y_2) \in E'$, el operador evaluador $P_y = P_{(y_1,y_2)} = g'(y_1, y_2)$ esta acotado:

$$|P_y g'| \leq M_y\ \|g'\|', \qquad \text{donde } M_y = k_1(y_1, y_1)^{1/2}\, k_2(y_2, y_2)^{1/2}.$$

Queda probar que este espacio es la completitud de la clase formada por todas las funciones f' del tipo (A.16) con norma $\|\cdot\|'$. Ya que el conjunto de las sumas finitas del tipo (A.21) es casi denso en el espacio de todas las funciones del tipo (A.21) es suficiente probar que cada función del tipo (A.16) es del tipo (A.21). Para ello es suficiente ver que cada función del tipo (A.16) puede ser aproximada tanto como se desee (con respecto a la norma $\|\cdot\|'$) por sumas finitas del tipo (A.21). En una representación (A.16) de la función f', se puede aproximar cada $f_i^{(j)}$ por una combinación lineal finita $h_i^{(j)}$ de funciones $g_i^{(\ell)}$ tal que $\left\|h_i^{(j)}\right\|_i \leq \left\|f_i^{(j)}\right\|_i$, $\left\|f_i^{(j)} - h_i^{(j)}\right\|_i \leq \varepsilon$. Antes de continuar en este punto, vamos a demostrar que para cada función f' y cualquiera que sea su representación (A.16) se verifica la desigualdad:

$$\|f'\|' \leq \sum_{j=1}^{n} \left\|f_1^{(j)}\right\|_1 \cdot \left\|f_2^{(j)}\right\|_2.$$

[14] Definición de isomorfismo.

En efecto,

$$\begin{aligned}\|f'\|'^2 = \langle f', f'\rangle &= \sum_{j=1}^{n}\sum_{\ell=1}^{n}\langle f_1^{(j)}, f_1^{(\ell)}\rangle_1 \langle f_2^{(j)}, f_2^{(\ell)}\rangle_2 \\ &\leq \sum_{j=1}^{n}\sum_{\ell=1}^{n}\left\|f_1^{(j)}\right\|_1\left\|f_1^{(\ell)}\right\|_1\left\|f_2^{(j)}\right\|_2\left\|f_2^{(\ell)}\right\|_2 \\ &= \left[\sum_{j=1}^{n}\left\|f_1^{(j)}\right\|_1\left\|f_2^{(j)}\right\|_2\right]^2\end{aligned}$$

Continuando con la demostración de la aproximación, se considera las funciones:

$$\begin{aligned}h'(x_1,x_2) &= \sum_{j=1}^{n} h_1^{(j)}(x_1) f_2^{(j)}(x_2), \\ g'(x_1,x_2) &= \sum_{j=1}^{n} h_1^{(j)}(x_1) h_2^{(k)}(x_2).\end{aligned}$$

Es claro que h' es de tipo (A.16) y que g' es a la vez de tipo (A.16) y (A.21), lo cual se ve claramente sin más que tener en cuenta que la función $h_i^{(j)}$ es una combinación lineal de $g_i^{(j)}$. Denotando por M el máximo de todas las funciones $\left\|f_i^{(j)}\right\|_i$, se obtiene:

$$\begin{aligned}\|f'-g'\|' &\leq \|f'-h'\|' + \|h'-g'\|' \\ \|f'-h'\|' &= \sum_{j=1}^{n}\left\|\left(f_1^{(j)}(x_1) - h_1^{(j)}(x_1)\right) f_2^{(j)}(x_2)\right\|' \\ &\leq \sum_{j=1}^{n}\left\|f_1^{(j)} - h_1^{(j)}\right\|_1 \cdot \left\|f_2^{(j)}\right\|_2 \leq \sum_{j=1}^{n} M\varepsilon = nM\varepsilon \\ \|h'-g'\|' &= \sum_{j=1}^{n}\left\|h_1^{(j)}(x_1)\left(f_2^{(j)}(x_2) - h_2^{(j)}(x_2)\right)\right\|' \\ &\leq \sum_{j=1}^{n}\left\|h_1^{(j)}\right\|_1 \cdot \left\|f_2^{(j)} - h_2^{(j)}\right\|_2 \leq \sum_{j=1}^{n} M\varepsilon = nM\varepsilon\end{aligned}$$

Finalmente se tiene:

$$\|f'-g'\| \leq 2nM\varepsilon.$$

La clase de todas las funciones del tipo (A.21) con la norma dada en (A.22) forma el producto directo $\mathcal{F}' = \mathcal{F}_1 \otimes \mathcal{F}_2$. Como se ha obtenido por completitud funcional

de la clase de funciones del tipo (A.16), esta clase será independiente de la elección del sistema ortonormal $\left\{g_1^{(k)}\right\}$ y $\left\{g_2^{(k)}\right\}$, y la clase $\mathcal{F}'$ es también independiente de la elección de estos sistemas (se puede ver la unicidad de la completitud funcional en la sección A.4).

De todo lo anterior se tiene el siguiente teorema:

Teorema A.7.1 *Sean $\mathcal{F}_1$ y $\mathcal{F}_2$ dos espacios de Hilbert núcleo reproductor con núcleos reproductores k_1 y k_2 respectivamente. Entonces el producto directo $\mathcal{F}' = \mathcal{F}_1 \otimes \mathcal{F}_2$ es un espacio de Hilbert núcleo reproductor con núcleo reproductor:*

$$k'(x_1, x_2, y_1, y_2) = k_1(x_1, y_1)\, k_2(x_2, y_2).$$

Demostración. La demostración es casi inmediata. En primer lugar, como función de x_1, x_2, k' es de la forma dada en (A.16) y por tanto pertenece a $\mathcal{F}'$. En segundo lugar, para cada función g' de la forma (A.21) se tiene:

$$\begin{aligned} g'(y_1, y_2) &= \sum\sum \alpha_{j,\ell} \langle g_1^{(j)}, k_1(x_1, y_1)\rangle_1 \, \langle g_2^{(j)}, k_2(x_2, y_2)\rangle_2 \\ &= \langle g'(x_1, x_2), k'(x_1, x_2, y_1, y_2)\rangle \end{aligned}$$

lo cual completa la demostración. ■

A.8 Ejemplo de núcleo reproductor

Este ejemplo ya ha sido mencionado en el capítulo 6 cuando se trataba el tema de sistemas de regularización (ver sección 6.3), como ejemplo de función de Green. En este ejemplo se construye un espacio de Hilbert núcleo reproductor que está asociado a un operador determinado.

Para ello se comienza definiendo una familia de funciones la cual proporcionará un marco de trabajo que resulta muy útil en muchos de los diferentes análisis teóricos sobre funciones regulares.

Definición A.8.1 *Se llama espacio de Sobolev, y se denota por W_m a la familia de funciones reales de variable real:*

$$W_m = \left\{ f : [0,1] \to I\!R \,/\, f \in C^{m-1}([0,1]) \ y \ f^{m)} \in L_2([0,1]) \right\} \tag{A.26}$$

donde

$$C^{m-1}([0,1]) = \left\{ f : [0,1] \to I\!R \,/ f, f', \cdots, f^{m-1)} \ \textit{son absolutamente continuas} \right\}$$

y

$$L_2([0,1]) = \left\{ f : [0,1] \to I\!R \,/ \int_0^1 f^2(x)dx < \infty \right\}.$$

Es posible encontrar distintos caminos para construir una norma sobre el espacio de Sobolev W_m. La que se va a construir en esta sección fue dada en [KW71] y esta asociada con un núcleo reproductor que es muy útil desde el punto de vista del cálculo numérico y de las matemáticas de los ordenadores.

Aplicando el teorema de Taylor con resto integral, se puede expresar cualquier función $f \in W_m$ de manera única en la forma (ver por ejemplo en [Apo79] página 342):

$$f(x) = \sum_{k=0}^{m-1} \frac{f^{k)}(0)}{k!} x^k + \int_0^1 \frac{(x-y)_+^{m-1}}{(m-1)!} f^{m)}(y)dy \tag{A.27}$$

Considerando una nueva familia de funciones

$$W_m^0 = \{ f \in W_m \,/\, f^{k)}(0) = 0, \ k = 0, 1, \cdots, m-1 \},$$

la aplicación del Teorema de Taylor a las funciones de esta familia conduce a:

$$f(x) = \int_0^1 \frac{(x-y)_+^{m-1}}{(m-1)!} f^{m)}(y)dy = \int_0^1 G_m(x,y) f^{m)}(y)dy \tag{A.28}$$

donde

$$G_m(x,y) = \frac{(x-y)_+^{m-1}}{(m-1)!} \tag{A.29}$$

para las funciones $f(x) \in W_m^0$.

Definición A.8.2 *A la función $G_m(x,y)$ se le llama función de Green del problema $D^{m)} f = g$, con $f \in W_m^0$; y donde $D^{m)}$ denota la derivada m-ésima.*

Se tiene el siguiente lema:

Lema A.8.3 *El espacio de funciones W_m^0 es un espacio de Hilbert con norma cuadrática:*

$$\|f\|^2 = \int_0^1 \left(f^{m)}(x)\right)^2 dx \tag{A.30}$$

Demostración. Hay que ver una a una las tres propiedades que debe cumplir una norma, en primer lugar

- $\|\alpha f\| = |\alpha| \, \|f\|$, $\forall \alpha \in \mathbb{R}$, lo que resulta evidente,

y las demás propiedades se comprueba directamente.

- $\|f\| = 0 \Leftrightarrow f \equiv 0$. $\|f\|^2 = 0 \Leftrightarrow \int_0^1 (f^{m)}(x))^2 dx = 0 \Leftrightarrow f^{m)}(x) = 0 \; \forall x \in [0,1]$ ya que $(f^{m)}(x))^2$ es una función positiva para todo $x \in [0,1]$; y por otro lado, como la función $f \in W_m^0 \Rightarrow f \equiv 0$ por la fórmula de Taylor (A.27).

- $\|f+g\| \leq \|f\| + \|g\|$. Esta desigualdad se tiene de

  $$\|f+g\|^2 = \|f\|^2 + \|g\|^2 + 2\langle f, g\rangle \leq \|f\|^2 + \|g\|^2 + 2\,\|f\| \cdot \|g\| = (\|f\| + \|g\|)^2$$

  donde la desigualdad $\langle f, g\rangle \leq \|f\| \cdot \|g\|$ es cierta ya que la desigualdad de Cauchy-Schwarz se cumple para la norma $\|f\| = \int f(x)\, dx$ sea cual sea f.

■

Nota A.8.4 *De la definición de $G_m(x,y)$, es evidente que la función escrita en la forma $G_{m,y}(x)$ (como función de x con un valor $y \geq 0$ fijo) pertenece a W_m^0 ya que*

$$\frac{dG_{m,y}^{k)}}{dx^k} = \frac{(x-y)_+^{m-k-1}}{(m-k-1)!}$$

de donde

$$\frac{dG_{m,y}^{k)}}{dx^k}(0) = \frac{(0-y)_+^{m-k-1}}{(m-k-1)!} = 0$$

ya que $(0-y) \leq 0$ *lo que implica que* $(0-y)_+ = 0$*, y se verifica, claramente, que sus derivadas hasta orden* $m-1$ *son absolutamente continuas, ya que son funciones definidas en dos trozos, en uno nulo y en el otro por un polinomio con punto de enlace (nodo)* $y \in [0,1]$ *donde coinciden las derivadas de los dos trozos hasta el orden* $m-1$*; y la derivada m-ésima es de cuadrado integrable ya que* $\dfrac{dG_{m,y}^{m)}}{dx^m}(x) = 0$ *por ser una función definida a trozo polinómica de grado menor que* m*.* ▲

Lema A.8.5 *El espacio* W_m^0 *es un espacio de Hilbert núcleo reproductor con núcleo reproductor dado por:*

$$k(x,y) = \int_0^1 G_m(y,u)\, G_m(x,u)\, du \tag{A.31}$$

Demostración. En primer lugar se ve que la función de evaluación $P_y(f) = f(y)$ esta acotada. Sea $f \in W_m^0$ entonces:

$$\begin{aligned} |f(y)| &= \left| \int_0^1 G_m(y,x) f^{m)}(x) dx \right| \\ &\leq \sqrt{\int_0^1 G_m^2(y,x) dx} \sqrt{\int_0^1 \left(f^{m)}(x)\right)^2 dx} = M_y \ \|f\| \end{aligned}$$

aplicando la desigualdad de Cauchy-Schwarz y que $G_m \in W_m^0$ (ya que toda función continua en un compacto esta acotada y de aquí la integral de G_m existe y es finita).

Para demostrar que $k(x,y)$ es un núcleo reproductor de W_m^0 hay que demostrar que:

1. $k_y(\cdot) = k(\cdot, y) \in W_m^0$ para $y \in [0,1]$ fijado; y
2. cumple la propiedad reproductora.

Puesto que

$$k_y(x) = \int_0^1 G_m(y,u)\, G_m(x,u)\, du$$

se tiene que k_y es una función de clase $m-1$, es decir es continua y derivable hasta orden $(m-1)$ $\forall y \in [0,1]$, y además

$$\left(\frac{\partial^m}{\partial x^m} k_y\right)(x) = G_m(y,x)$$

y de aquí por el teorema de Taylor (A.27) se tiene que $k_y \in W_m^0$.
La propiedad reproductora es fácil de comprobar ya que se tiene:

$$\langle k_y, f\rangle = \int_0^1 \frac{\partial^m k_y(x)}{\partial x^m}\, f^{m)}(x)\, dx = \int_0^1 G_m(y,x)\, f^{m)}(x)\, dx = f(y).$$

■

Sea un nuevo espacio, $\mathcal{H}_0$ el espacio m-dimensional de los polinomios de grado menor o igual que $m-1$. Dentro de este espacio se considera la base[15] $\{\phi_1, \phi_2, \cdots, \phi_m\}$ donde $\phi_\ell(x) = \dfrac{x^{\ell-1}}{(\ell-1)!}$ para $\ell = 1, 2, \cdots, m$.

Nota A.8.6 *Se cumple que $D^{m)}(f) = 0$ para toda función $f \in \mathcal{H}_0$, es decir la derivada m-ésima de cualquier función de $\mathcal{H}_0$ es nula. Este resultado es evidente ya que las funciones de $\mathcal{H}_0$ son polinomios de grado menor que m, pero aún más se cumple que:*

$$\left(D^{j-1)}\phi_\ell\right)(0) = \begin{cases} 1 & si \quad j = \ell \\ 0 & si \quad j \neq l \end{cases} \qquad j, \ell = 1, 2, \cdots, m \tag{A.32}$$

puesto que

$$D^{j-1)}\phi_\ell(x) = \frac{x^{\ell-j}}{(\ell-j)!}.$$

▲

Al espacio $\mathcal{H}_0$ se le dota de la norma cuadrática:

$$\|\phi\|^2 = \sum_{k=0}^{m-1} \left[(D^{k)}\phi)(0)\right]^2, \qquad \forall \phi \in \mathcal{H}_0. \tag{A.33}$$

Con esta norma se tiene que:

[15] Que ciertamente es base ya que es fácil ver utilizando las propiedades de la matriz de Vandermonde.

1. $\|\phi_\ell\| = \sum_{k=0}^{m-1} \left[(D^{k)}\phi_\ell)(0)\right]^2 = 1$ para $\ell = 1, \cdots, m-1$ por la igualdad (A.32).

2. Si $i \neq j$ se tiene $(\phi_i, \phi_j) = \sum_{k=0}^{m-1} \phi_i^{(k}(0)\, \phi_j^{(k}(0) = 0$ por la igualdad (A.32) y la definición de $\phi_\ell(x),\ \ell = 1, 2, \cdots, m$.

Por tanto se tiene que el espacio $\mathcal{H}_0$ es un espacio de funciones de Hilbert m-dimensional con base $\{\phi_\ell, \quad \ell = 1, 2, \cdots, m\}$ ortonormal y, por el teorema (A.3.1), se sigue que, como $\{\alpha_{ij}\} = \{(\phi_i, \phi_j)\}$ es la identidad, entonces $\{\beta_{ij}\}$ (inversa de la matriz $\{\alpha_{ij}\}$) es la identidad y el núcleo reproductor queda:

$$k(x, y) = \sum_{\ell=1}^{m} \phi_\ell(x)\, \phi_\ell(y). \tag{A.34}$$

De todo lo anterior se tiene que, dado $f \in W_m$, existe un único desarrollo de Taylor de orden m:

$$f(x) = \sum_{k=0}^{m-1} \frac{f^{k)}(0)}{k!} x^k + \int_0^1 \frac{(x-y)_+^{m-1}}{(m-1)!} f^{m)}(y) dy$$

lo que significa que cualquier función $f \in W_m$ admite una descomposición única en la forma:

$$f = f_0 + f_1, \qquad \text{con } f_0 \in \mathcal{H}_0 \text{ y } f_1 \in W_m^0.$$

Además por construcción se tiene:

$$\int_0^1 (D^{m)} f_0)(x)\, dx = 0 \quad y \quad \sum_{k=0}^{m-1} \left[(D^{k)} f_1)(0)\right]^2 = 0.$$

De esta manera denotando a W_m^0 como $\mathcal{H}_1$, se puede afirmar que $W_m = \mathcal{H}_0 \oplus \mathcal{H}_1$ con la norma

$$\|f\|^2 = \sum_{j=0}^{m-1} \left[(D^{j)} f)(0)\right]^2 + \int_0^1 (D^{m)} f)^2(x)\, dx \tag{A.35}$$

ya que los espacios $\mathcal{H}_0$ y $\mathcal{H}_1$ son espacios ortogonales con esta norma y además se cumple del teorema (A.6.1) que el núcleo reproductor de este espacio es:

$$k(x, y) = \sum_{\ell=1}^{m} \phi_\ell(x)\, \phi_\ell(y) \ + \ \int_0^1 G_m(x, u)\, G_m(y, u)\, du \tag{A.36}$$

Un importante aspecto geométrico, que se utiliza en algunos desarrollos, es que el operador funcional

$$J_m : W_m \to \mathbb{R}$$

definido por

$$J_m(f) = \int_0^1 (f^{m)}(x)))^2 \, dx$$

puede escribirse como $J_m(f) = \|P_1 f\|^2_{W_m}$ donde $P_1 f$ es la proyección ortogonal de f sobre W_m^0.

APÉNDICE B

COLECCIÓN DE PROGRAMAS ELABORADOS

En este apéndice aparecen recogidos todos los programas que hemos necesitado elaborar para completar los desarrollos y ejemplos que aparecen en los capítulos 7 y 8. Todos ellos han sido elaborado utilizando como soporte informático el paquete Matlab versión 5.3 (R11), bajo sistema operativo Windows 98.

En este capítulo, además de mostrar las ordenes que conforman los programas, indicamos como es su funcionamiento y señalamos todas aquellas características que consideramos necesarias para una mejor comprensión de este trabajo.

B.1 Programas del capítulo 7

Para la realización de este capítulo hemos necesitado diseñar cinco programas que nos permitiesen calcular la matriz de similitudes y las consiguientes representaciones gráficas. Hemos añadido uno más que nos permite calcular la distancia entre diferentes items, todo ello a partir de la función núcleo similitud.

B.1.1 Programa Simil

Este programa nos permite obtener, a partir de la matriz de frecuencias absolutas bidimensional (con el mismo número de entradas) y el número total de observaciones, la matriz de similitudes y las probabilidades[1] de cada suceso.

Para que el programa funcione correctamente, la matriz de frecuencias debe ser una matriz cuadrada, donde el valor máximo por fila y columna se encuentra en la diagonal principal[2]. Si la matriz no es simétrica, el programa ajusta automáticamente la matriz tomando el máximo de los dos elementos implicados.

El programa es el siguiente:

```
function [sim,pr] = Simil(X,N)

% Simil Similitud entre item
%
%  Usage: [similitud,probabilidad] = Simil(X,N)
%
%  Parameters: X      - Matriz de frecuencias
%              N      - Número total de datos
%              sim    - Matriz de similitudes
%              pr     - Matriz de probabilidades
%
%  Author: Luis González Abril (luisgon@us.es)

% chequea si el número de argumentos es correcto
  if (nargin <1 | nargin>2)
    help Simil
```

[1] Utilizando la interpretación frecuencialista de la probabilidad cuando el tamaño de la muestra es grande.

[2] Algo que debe ser siempre cierto, si no se ha producido ningún error en el recuento.

```
  else
    fprintf('Similitud entre items\n')
    fprintf('_____________________________\n')
    n = size(X,1);
    % Construcción de la matriz de trabajo
    fprintf('Construyendo matriz de trabajo ...\n');
    if (nargin<2) X1=X;
    else
    X1 = zeros(n,n);
    for i=1:n
       for j=1:n
          X1(i,j) = max(X(i,j),X(j,i))/N;
       end
    end
   , end
, end
% Comprobación de la matriz de trabajo
    fprintf('Comprobación de la matriz de trabajo ...\n');
    for i=1:n,
       for j=1:n,
          if X1(i,j)>max(X1(i,i))
             i;j;
             fprintf('Error en el elemento : %d %d\n',i,j);
          ,end
       end
    end
 pr=X1;
% Construcción de la matriz de similitudes
 fprintf('Matriz de similitudes\n'), sim = zeros(n,n);
    for i=1:n
       for j=1:n
```

```
          sim(i,j) = X1(i,j)-X1(i,i)*X1(j,j);
       end
    end
```

B.1.2 Programa Graf

Este programa nos permite obtener una representación gráfica del tipo dado en la figura 7.10, a partir del vector de probabilidades, la matriz de similitudes y el suceso que se toma como referencia en la función k_A.

El programa es el siguiente:

```
function Graf=Graf(probabilidad,similitud,otro)

% Simil Similitud entre item
%
%  Usage: Graf(probabilidad,similitud,otro)
%
%  Parametros: probabilidad - Matriz de probabilidades
%              similitud    - Matriz de similitudes
%              otro         - Item de referencia
%
%  Author: Luis González Abril (luisgon@us.es)

 n=size(probabilidad,1);
 prob=zeros(n,1);
 for i=1:n
   if probabilidad(i,i)>0.5
      prob(i)=1-probabilidad(i,i);
   else
      prob(i)=probabilidad(i,i);
      end
```

```
end

xx=zeros(n,1);
for i=1:n
  for j=1:n
     xx(i)=xx(i)+(prob(i)-prob(j))^2;
  end
end

if otro==0
  [xm,im]=min(xx);
else
  im=otro;
end

   xx1=0:0.001:1;
   nn=size(xx1,2);
   yy1=zeros(nn,2);
for i=1:nn
  if xx1(i)<prob(im)
     yy1(i,1)=xx1(i)*(1-prob(im));
  else
     yy1(i,1)=prob(im)*(1-xx1(i));
  end
end

for i=1:nn
  if xx1(i)<1-prob(im)
     yy1(i,2)=-prob(im)*xx1(i);
  else
     yy1(i,2)=(xx1(i)-1)*(1-prob(im));
```

```
   end
 end

 for i=1:n
    proba(i)=probabilidad(i,i);
   end

 hold on
% Quitar el símbolo %, en las dos siguientes líneas,
% si se quiere aparezca como titulo el item  de referencia.

%titulo='item nº: ';
%titulo=[titulo, char(48+im)];
axis([0 1 -1.2*prob(im)*(1-prob(im)) 1.2*prob(im)*(1-prob(im))])
% Relleno el dominio con clor gris claro
fill(xx1,yy1,[.9 .9 .9])%
dd=plot(xx1,yy1,'b-',proba,similitud(im,:),'k.');
set(dd,'MarkerSize',15);
%title(titulo)
```

B.1.3 Programa matrizgraf

Este programa obtiene de una sola vez, a partir de una matriz de gráficos, todas las gráficas realizadas con el programa Graf, tomando como referencia en cada caso un suceso distinto hasta completarlos todos.

El programa ajusta automáticamente el número de filas y columnas necesarias para poder tener una mejor visualización de los gráficos. Con este programa se ha obtenido la figura 7.11

El programa es el siguiente:

```
function matrizgraf= matrizgraf(probabilidad,similitud)

% Simil Similitud entre item
%
%  Usage: matrizgraf(probabilidad,similitud)
%
%  Parameters: probabilidad - Matriz de probabilidades
%              similitud    - Matriz de similitudes
%
%  Author: Luis González Abril (luisgon@us.es)

 n11=size(probabilidad,1);
 n12=ceil(sqrt(n11));
 for i=1:n11
   subplot(n12,n12,i),Graf(probabilidad,similitud,i)
 end
```

B.1.4 Programa grafsimi

Este programa nos permite obtener en un solo gráfico, a partir del vector de probabilidades y la matriz de similitudes todas las similitudes junto con su probabilidades.

Con este programa se ha obtenido la figura 7.12, pero si los sucesos han sido ordenados de menor a mayor utilizando como medida las probabilidades, con el parámetro "dd" incorporado en la función podemos representar exclusivamente los "dd" primeros sucesos. De esta forma, con dd= 5 hemos obtenido la figura 7.13, y de esta forma hemos podido realizar una especie de zoom dentro de la figura 7.12.

El programa es el siguiente:

```
function grafsimi = grafsimi(prob,simil,dd)
```

```
% Simil Similitud entre item
%
%  Usage: [] = grafsimi(prob,simil,dd)
%
%  Parameters: prob   - Matriz de probabilidades
%              sim    - Matriz de similitudes
%              dd     - Solo considera los dd primeros items
%
%  Author: Luis González Abril (luisgon@us.es)

 for i=1:dd
   for j=1:dd
      pr(i,j)=prob(i,j);
      sim(i,j)=simil(i,j);
   end
 end

 vv=max(pr); vvs=max(sim);
 maxpr=max(vv); maxs=max(vvs);
 x=0:maxpr/100:maxpr;
 close
 hold on
 axis([0,51*maxpr/50,-21*maxs/20,21*maxs/20])
 title('Representación gráfica de similitudes')
 xlabel('Probabilidades') ylabel('Similitudes')

 plot(x,x.*(1-x)) plot(x,-x.*(1-x)) plot(x,x.*0,'k-')
 for i=1:dd
   et1=48+i ;
   plot(vv(i),[sim(i,i); -sim(i,i)],'r-x')
```

```
   vvY(i)=sim(i,i)+sim(dd,dd)/30;
   text(vv(i),vvY(i),char([et1]))
   for j=i+1:dd
      et2=48+j;
      plot(vv(i),sim(i,j),'kx')
      text(vv(i)+maxpr/50,sim(i,j),char([et2]))
   end
 end
 hold off
```

B.1.5 Programa Dist

Este programa no lo hemos utilizado pero se utilizaría para calcular a partir de las similitudes una distancia entre sucesos.

El programa es el siguiente:

```
 function Dist = Dist(X)

% Distancia entre items
%
%  Usage: Dist = Dist(X)
%
%  Parameters: X      - Matriz de similitudes
%
%  Author: Luis González Abril (luisgon@us.es)

  if (nargin <1 | nargin>1) % check correct number of arguments
    help Simil
  else

    fprintf('Distancia entre items\n')
```

```
    fprintf('_____________________________\n')
    n = size(X,1);
    Dist = zeros(n,n);
      for i=1:n
         for j=1:n
          Dist(i,j) = sqrt(X(i,i)+X(j,j)-2*X(i,j));
         end
      end
  end
```

B.2 Programas del capítulo 8

Para la realización de este capítulo hemos necesitado realizar una serie de programas que nos permitiesen construir una máquina de vectores soporte que cumpla las características que a lo largo, principalmente, del capítulo 5 se ha diseñado.

El programa Matlab incorpora dentro de su rutina diferentes programas que resuelven problemas de optimización (lineal, cuadráticos, con o sin restricciones, ...). Sin embargo, nosotros hemos utilizado el programa **mcirwls**, el cual esta diseñado específicamente para resolver los problemas de optimización que surgen en los problemas de clasificación con tres etiquetas, siendo extremadamente rápido y flexible.

B.2.1 Programa preparadatos

Este programa realiza un tratamiento de los datos con objeto de obtener como salida, la entrada necesaria para la ejecución del programa de optimización **mcirwls**.

El programa es el siguiente:

```
function [ddx,ddy]=preparadatos(datos,k1,k2)
```

```
% Uso:Programa que realiza la agrupación de las etiquetas en las
%     clase -1, 0 y 1. La salida esta preparada para ejecutar el
%     programa de multiclasificación mcirwls diseñado por Cecilio
%     Angulo (U.P.C.)
%
% Orden: [ddx,ddy]=preparadatos(datos,k1,k2)
%
% Parámetros de entrada
%  datos: Conjunto de datos a estudiar.
%   k1:   Etiqueta correspondiente a la clasificación 1.
%   k2:   Etiqueta correspondiente a la clasificación -1.
%
% Salidas
%  ddx:   Vectores inputs ordenadas por etiquetas 1, -1, 0
%  ddy:   Etiquetas en la forma [1 ... 1 -1 ... -1 0 ... 0]
%
% Cuidado:
%   Los datos aparecen por filas.
%   Las etiquetas originales se deben encontrar en la última columna.
%   Los identificadores se situan en la primera columna.
%
%  Autor: Luis González Abril (luisgon@us.es)

% Construimos las matrices auxilares, correspondientes a las etiquetas,
%  a los datos y las etiquetas ordenadas en la forma 1, -1, 0
 datos1=[]; datos_1=[]; datos0=[];
 datos1y=[];datos_1y=[];datos0y=[];
% Se ve cual es la última columna (referente a las etiquetas)
 nfinal=length(datos(1,:));

% Se construye las entradas para el programa mcirwls
```

```
 for i=1:length(datos)
   if datos(i,nfinal)==k1
      datos1=[datos1; datos(i,2:(nfinal-1))];
      datos1y=[datos1y;1];
   elseif datos(i,nfinal)==k2
      datos_1=[datos_1;datos(i,2:(nfinal-1))];
      datos_1y=[datos_1y;-1];
   else
      datos0=[datos0;datos(i,2:(nfinal-1))];
      datos0y=[datos0y;0];
   end
 end

% Salidas del programa
% Valores inputs ordenadas segun etiquetas 1, -1 y 0
 ddx=[datos1; datos_1; datos0];
% Etiquetas de los datos anteriores
 ddy=[datos1y; datos_1y; datos0y];
% Fin del programa.
```

B.2.2 Programa salida

Este programa proporciona a partir de un conjunto de nuevos inputs, los outputs (las etiquetas) que la máquina de vectores soporte multiclasificadora necesita. Además, al incluir los parámetros presentes en el proceso de optimización de las SVMs, conseguimos unas probabilidades (grados de confianza) que quedan definidas a partir de los desarrollos teóricos del capítulo 5.

El programa es el siguiente:

```
function
  [etiqueta,confianza]=salida(ddx,ddxx,alfa,beta,nucleo,C1,C2,delta)
```

```
% Uso:Programa que proporciona las etiquetas y los grados de
%    confianza a partir de una función discriminadora obtenida
%    mediante el procedimiento de las Máquinas de Vectores Soporte.
%
% Orden: [etiqueta]=salida(ddx,ddxx,alfa,beta,nucleo,C1,C2,delta)
%
% Parámetros de entrada
%   ddx:  Datos inputs obtenidos de datosfinales.
%  ddxx:  Conjunto de validación.
%  alfa:  Vector de los lagrangianos.
%  beta:  Término independiente de función discriminadora.
%   ker:  Núcleo utilizado 'rbf', 'poli' (no olvidar la variable
%         global p1).
%   C1:   Valor de la ponderación dada a la suma de los errores
%         de las dos clases que se discriminan.
%   C2:   Valor de la ponderación dada a las restantes clases.
%  delta: Valor del factor de insensibilidad.
%
%   Salidas
%  etiqueta:  Etiqueta que asigna la función discriminadora
%  confianza: Grado de confianza en la clasificación.
%
%  Autor: Luis González Abril (luisgon@us.es)

%% COMENZAMOS LA CONSTRUCCIÓN DEL SALIDA

% Para poder continuar adecuadamente debemos realizar este tratamiento
% utilizando matrices auxiliares, que nos permite poder trabajar con
% la transposición de matrices.
 fbeta=beta(:);
```

```
 for i=1:1; beta_teo(i)=fbeta(i);end
 falfa=alfa(:);
 for i=1:length(falfa); alfa_teo(i)=falfa(i);end

% Calculamos el valor teórico asignado al conjunto test.
 valorteorico=[];
 valorteorico=alfa_teo* svkernel1(nucleo,ddx,ddxx)+beta_teo;

% Debemos asignarle una etiqueta en función del criterio elegido
% según las verosimilititudes.

 confianza=[]; etiqueta=[];
 for i=1:length(ddxx)
   theta=valorteorico(i);
   vtheta=exp(-C1*max(1-theta,0))+exp(-C1*max(1+theta,0))
          +exp(-C2*max(abs(theta)-delta,0));
   p1=exp(-C1*max(1-theta,0))/vtheta;
   pm1=exp(-C1*max(1+theta,0))/vtheta;
   p0=exp(-C2*max(abs(theta)-delta,0))/vtheta;

   if p1>=p0;
      etiqueta(i)=1;
      confianza(i)=p1;
   elseif pm1>=p0;
      etiqueta(i)=-1;
       confianza(i)=pm1;
   else
      etiqueta(i)=0;
       confianza(i)=p0;
  end
end
```

```
% Fin del programa
```

B.2.3 Programa clasificacion

Con este programa se obtiene la matriz de clasificación resultante de comparar dos vectores de etiquetas. El primer vector debe ser el vector de etiquetado real, y el segundo el de etiquetado predicho.

Si se realiza con una única entrada, proporciona la distribución del etiquetado.

El programa es el siguiente:

```
function [salida2]=clasificacion(aa,bb)

% Uso:Programa que proporciona la matriz de clasificación, donde
%    además se indica el número de no etiquetados.
%
% Orden: [salida2]=clasificacion(aa,bb)
%
% Parámetros de entrada
%   aa:  Es un vector fila con las etiquetas reales.
%   bb:  Es un vector fila con las etiquetas predicha.
%
%  Salidas
%  salida2:  Es la matriz de clasificación cuyo elemento (1,1)
%         indica el número de datos estudiados. La primera columna
%         representa las etiquetas reales y la primera fila las
%         etiquetas predichas
% (*) Si solo se utiliza un vector aa entonces la diagonal nos da el
%     número de vectores que tienen una etiqueta determinada.
%
%  Autor: Luis González Abril (luisgon@us.es)
```

```
 if nargin == 1
   bb=aa;
 end

% Cuerpo del programa
 imax=max(aa); % Valor de la mayor etiqueta
 for i=1:imax    % Inicialización de la matriz de  clasificacion
   for j=1:imax+1
      A(i,j)=0;
   end
 end

% Calculamos los coeficiente de la matriz de clasificación.
long=length(aa);    % Tamaño del vector
for i=1:long
   if bb(i)==0
      A(aa(i),imax+1)=A(aa(i),imax+1)+1;
   else
      A(aa(i),bb(i))=A(aa(i),bb(i))+1;
   end
end

% Ordenamos adecuadamente las matrices para conseguir una mejor
% interpretación.
 et=1:imax;
 et1=[long,et,0];
 salida=[et',A];
 salida2=[et1;salida];
% Fin del programa
```

B.2.4 Programa discrimina

Con este programa se obtienen los coeficientes de todas las funciones discriminadoras parciales, así como el número de vectores soporte necesarios en cada una de ellas. Una explicación detallada de su funcionamiento se encuentra en el capítulo 8 de implementación.

El programa es el siguiente:

```
function [sv,galfa,gbeta]=discrimina(datos)

% Uso: [galfa,gbeta]=discrimina(datos)
%   Parámetro de entrada
%    datos:  Conjunto de datos a estudiar.
%
%   Salidas
%      sv:   Matriz del número de vectores soporte para cada una
%            de las SVMs necesarias.
%   galfa:   Hipermatriz que proporciona los multiplicadores de
%            Lagrange de todas las máquinas SVMs necesarias.
%   gbeta:   Hipermatriz que proporciona los valores b
%            de todas las máquinas SVMs necesarias.
%
% Los datos aparecen por filas.
% Las etiquetas originales se deben encontrar en la última columna.
% Los identificadores se situan en la primera columna.
% La salida galfa(i,j,k) se debe entender como sigue:
%   --(i,j) hace referencia a la función discriminante f_ij.
%   -- k hace referencia a las distintas componentes de los
%      multiplicadores de Lagrage de f_ij.
% La salida gbeta(i,j,1) se debe entender como sigue:
%   --(i,j) hace referencia a la función discriminante f_ij.
```

```
%   -- k hace referencia al valor b de f_ij.

% Valor del factor de insensibilidad
global delta
delta=0;

% Valor de la ponderación dada a la suma de los errores de
% las dos clases que se discrimina
global C1
C1=5;

% Valor de la ponderación dada a las restantes clases
global C2
C2=5;

% Función núcleo
global nucleo
nucleo='rbf';

% Parámetro de la función núcleo
global p1
p1=1;

% Limpiamos la pantalla de Matlab
clc

% Calculamos el número de etiquetas.
% Con: length(datos(1,:)) vemos el número de columnas de datos
imax=max(datos(:,length(datos(1,:))));
fprintf('Número de etiquetas ....:%2.0f\n', imax)
fprintf('Número de funciones discriminantes dicotómicas ....
```

```
          :%3.0f\n', imax*(imax-1)/2)

% Construimos un bucle que determine todos los coeficientes
% necesarios para la elaboración de todas las máquinas SVMs
% (funciones discriminante) necesarias para llevar a cabo
% la multiclasificación.
fprintf('_____________________________________________\n')
% Nos permite evaluar el tiempo consumido en este proceso
st = cputime;
 for i=1:imax-1
   for j=i+1:imax
      a=i+0.1*j;% Para tener una adecuada expresión de la siguiente linea
      fprintf('Calculando la función discriminante ....:%4.1f\n', a)
     fprintf('_____________________________________________\n')
      [ddx,ddy]=preparadatos(datos,i,j);
      [nsv,alfa,beta]=mcirwls(ddx,ddy,nucleo,C1,C2,delta);
      sv(i,j)=nsv;
      galfa(i,j,:)=alfa;
      gbeta(i,j,1)=beta;
   end
 end
 fprintf('El programa se ha completado en %4.1f segundos \n', cputime-st)
% Fin del programa
```

B.2.5 Programa interprete

Realiza la interpretación de todas las salidas intermedias de las máquinas de vectores soporte parciales. Una explicación detallada de su funcionamiento se encuentra en el capítulo 8 de implementación.

El programa es el siguiente:

```
% Introducimos el conjunto de vectores de entrenamiento.
datos=datos;

% Introducimos el conjunto de vectores test
datostest=datostest(:,2:end-1);

% Calculamos el número de etiquetas
imax=max(datos(:,end));

% Comienza el grueso del programa
etiqueta=[]; confianza=[];
 for i=1:imax-1
   for j=i+1:imax
      [ddx,ddy]=preparadatos(datos,i,j);
      [etiqueta1,confianza1]=salida(ddx,datostest,galfa(i,j,:),
                           gbeta(i,j,1),'rbf',5,5,0);
      etiqueta2=[];
            for k=1:length(etiqueta1)
         if etiqueta1(k)==1
            etiqueta2(k)=i;
         elseif etiqueta1(k)==-1
            etiqueta2(k)=j;
         else
            etiqueta2(k)=0;
         end
       end
       % Matriz que nos proporciona todas las etiquetas de la
       % funciones discriminantes parciales
       etiqueta=[etiqueta;etiqueta2];
       % Matriz que nos proporciona todas las respectivas confianzas
      confianza=[confianza;confianza1];
```

```
   end
end

% Construcción del interprete final a partir de las etiquetas
% anteriores y las confianzas
etiquetafinal=[]; confianzafinal=[];
 for i=1:length(etiqueta)
   b=[];con=[];
   for j=1:imax
      % Contamos el número de veces que aparecen cada etiquetas
      aa=find(etiqueta(:,i)==j);
      b(j)=length(aa);
      % Calculamos la confianza media de cada etiqueta
      if b(j)==0
         con(j)=0;
      else
         con(j)=sum(confianza(find(etiqueta(:,i)==j),i))/b(j);
      end
      % Calculamos el número de etiquetas que aparece con mayor
      % frecuencia.
      [xm,im]=max(b);
      if xm==0 % Si no hay etiquetas, no predice.
         etiquetafinal(i)=0;
         confianzafinal(i)=100;
      else
         imx=find(b==xm);
          conta=length(imx);
           if conta==1
           etiquetafinal(i)=im;
            confianzafinal(i)=con(im);
         else
```

```
          xmax=max(con(imx));
            aca=find(con==xmax);
            etiquetafinal(i)=aca(1);
            confianzafinal(i)=xmax;
          end
       end
      end
end
% fin del programa
```

B.2.6 Programa walea

Este es un pequeño programa pero muy útil para obtener una muestra aleatoria estratificada de un conjunto de índices (en este caso 100).

```
w1=00+extract(30,21);
w2=30+extract(30,21);
w3=60+extract(20,14);
w4=80+extract(20,14);
walea=[w1,w2,w3,w4];
```

ÍNDICE DE TÉRMINOS

BIBLIOGRAFÍA DEL TRABAJO

[ABR64] M. A. Aizerman, E. M. Braverman, and L. I. Rozonoér. Theoretical foundations of the potential function method in pattern recognition learning. *Automation and Remote Control*, 1(25):821–837, 1964.

[AC01] C. Angulo and A. Català. Ordinal regression with k-svcr machines. *IWAN*, 2001.

[Ang01] C. Angulo. *Aprendizaje con máquinas núcleos en entornos de multiclasificación*. Tesis doctoral, Universidad Politécnica de Cataluña, Abril 2001.

[AP95] J. M. Alba and J. L. Pérez. La teoría de la decisión. Technical report, Dpto. Economía Aplicada I. Universidad de Sevilla, 1995.

[Apo79] T. Apostol. *Calculus*, volume 1. Ed. Reverter, S.A., 2ª edition, 1979.

[Aro50] N. Aronszajn. Theory of reproducing kernels. *Trans. Amer. Math. Soc.*, 1(68):337–404, 1950.

[BGV92] B. E. Boser, I. M. Guyon, and V. Vapnik. A training algorithm for optimal margin classifiers. In Fifth Annual Workshop on Computational Learning Theory, Pittsburgh, ACM, 1992.

[Bis95] C. M. Bishop. *Neural networks for pattern recognition*. Clarendon Press, Oxford, 1995.

[Bur96] C. Burges. A tutorial on support vector machines for pattern recognition. *Knowledge Discovery and Data Mining*, 1996.

[Bur98] C. Burges. Simplified support vector decision rules. *Proceedings of the Thirteenth International Conference on Machine Learning*, 1998.

[Cam00] L. Le Cam. *La Statistique Mathematique Depuis 1950, en Development of Mathematics 1950-2000.* Ed. Birkåuser, 2000.

[CH53] R. Courant and D. Hilbert. *Methods of mathematical physics.* Interscience, 1953.

[Cor95] C. Cortes. *Prediction of generalization ability in learning machine.* PhD thesis, Department of Computer Science, University of Rochester, 1995.

[CSS01] B. Cabrer, A. Sancho, and G. Serrano. *Microeconometría y decisión.* Ed. Piramide, 2001.

[CST00] N. Cristianini and J. Shawe-Taylor. *An introduction to Support Vector Machines and other kernel-based learning methods.* Cambridge University, 2000.

[CV95] C. Cortes and V. Vapnik. Support vector networks. *Machine Learning*, 20:273–297, 1995.

[DGL96] L. Devroye, L. Györfi, and G. Lugosi. *A probabilistic theory of pattern recognition.* Springer-Verlag, New York, Inc, 1996.

[Fle87] R. Fletcher. *Practical methods of optimization.* John Wiley and Sons, Inc, 2 edition, 1987.

[GB92] S. German and E. Bienenstock. Neural networks and the bias / variance dilemma. *Neural Computation, 4:1-58*, 1992.

[GJP93] F. Girosi, M. Jones, and T. Poggio. Priors, stabilizers and basis functions: From regularization to radial, tensor and additive splines. *A.I. Memo No. 1430, MIT*, 1993.

[Gon98] L. González. Muestreo e informática. simulación. el método de monte carlo. *Epsilon*, 14(1)(40):55–64, 1998.

[Gon00] L. González. *Teoría del aprendizaje estadístico de la regresión. Máquinas de regresión de vector base.* Tesina, Facultad de Ciencias Económicas y Empresariales, Universidad de Sevilla, Diciembre 2000.

[Gun98] S. Gunn. Support vector machines for classification and regression. Technical report, ISIS, May 1998.

[GVB+92] I. Guyon, V. Vapnik, B. Boser, L. Bottou, and S. Solla. Structural risk minimization for character recognition. *Advances in Neural Information Processing Systems, 4:471-479*, 1992.

[Har97] D. A. Harville. *Matrix algebra from a statistician's perspective.* Springer-Verlag, New York, 1997.

[HATB00] J. Hair, R. Anderson, R. Tatham, and W. Black. *Análisis Multivariante.* Prentice Hall, quinta edition, 2000.

[Hay94] S. Haykin. *Neural networks: a comprehensive foundation.* Macmillan, New York, 1994.

[Her01] L. Hernández. *Técnicas de Taxonomía numérica.* Número 18 de Cuadernos de Estadística. Ed. La Muralla, 2001.

[HT90] T. Hastie and R. Tibshirani. *Generalized additive models.* Volumen 43: Monographs on Statistics and Applied Probability. Chapman & Hall, London, 1990.

[Kol65] A. Kolmogorov. Three approaches to the quantitative definition of information. *Problems of information transmission*, 1:1–7, 1965.

[Kon86] H. Konig. *Eigenvalue distribution of compact operators.* Birkhauser, Basel, 1986.

[Kre99] U. Kressel. Pairwise classification and support vector machine. *In B. Schölkopf, C. Burgues and A. Smola, editors*, Advances in Kernel Methods: Support Vector Learning. MIT Press. Cambridge, MA, 1999.

[KW71] G. Kimeldorf and G. Wahba. Some results on tchebycheffian spline function. *J. Math. Anal. Appl.*, 1(33):82–95, 1971.

[LWZL01] Y. Lin, G. Wahba, H. Zhang, and Y. Lee. Statistical properties and adaptive tuning of support vector machine. Technical report, Departament Statistics. University of Wisconsin, June 2001.

[MA99] E. Mayoraz and E. Alpaydin. *Support vector machines for multi-class clasification*. In Proceeding of the 5th International Work-Conference on Artificial and Natural Neural Networks IWANN'99, 1999.

[McC83] G. P. McCormick. *Non Linear Programing: Theory, Algorithms and Applications*. John Wiley and Sons, Inc, 1983.

[Mer09] J. Mercer. Function of positive and negative type and their connection with the theory of integral equations. *Philos. Trans. Royal society London*, A(209):415–446, 1909.

[MP92] D. Montgomery and E. Peck. *Introduction to Linear Regression Analysis*. John Wiley and Sons, Inc. 2nd edición, 1992.

[NM36] J. Neymann and F. Murray. On rings of operators. *Annals of Mathematics*, 37:116–229, 1936.

[Par62] E. Parzen. An approach to times series analysis. *Ann. Math. Statist.*, 1(32):951–989, 1962.

[Pla99] J. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods, 1999.

[Pér01] C. Pérez. *Técnicas Estadísticas con SPSS*. Ed. Prentice Hall, 2001.

[Rao84] C.R. Rao. *Estadística y Verdad. Aprovechando el azar*. Universitas- 73, 1984.

[RN55] F. Riesz and B. S. Nagy. *Functional Analysis*. Frederick Ungar Publishing Co., 1955.

[Rud79] W. Rudin. *Análisis real y complejo*. Ed. Alhambra, 1979.

[SBV95] B. Schölkopf, C. Burges, and V. Vapnik. Extracting support data for a given data. Proceeding , First International Conference on Knowledge Discovery Data Mining, AAAI Press, Menlo Park, CA, 1995.

[Skh00] B. Skhölkopf. Statistical learning and kernel methods. Technical Report MSR-TR-2000-23, Microsoft Research Limited, Febrero 2000.

[SMB+99] B. Schölkopf, S. Mika, C. Burges, P. Knirsch, K. Müller, G. Ratsch, and A. Smola. Input space vs. feature space in kernel-based methods. *IEEE Transactions on neural networks*, 1999.

[Smo98] A. J. Smola. *Learning with Kernels*. PhD thesis, Vom Fachbereich 13-Informatik der Technischen Universitat Berlin, 1998.

[Sol00] P. Sollich. Bayesian methods for support vector machines: Evidence and predictive class probabilities. Kluwer Academic Publidhers, 2000.

[SS73] P. H. Sneath and R. Sokal. *Numerical Taxonomy. The principles and practice of numerical classification*. San Francisco: W.H: Freeman and Co., 1973.

[SS98] A. J. Smola and B. Schölkopf. On a kernel-based method for pattern recognition, regression, aproximation and operator inversion. *Algorithmica*, 1(22):211–231, 1998.

[SSB+97] B. Schölkopf, K. Sung, C. Burges, F. Girosi, P. Niyogi, T. Poggio, and V. Vapnik. Comparing support vector machines witn gaussian kernels to radial basis function classifiers. *IEEE Trans. Sign. Processing*, 45:2758–2765, 1997.

[SWG+96] M. Stitson, J. Weston, A. Gammerman, V. Vovk, and V. Vapnik. Theory of support vector machines. Informe Técnico. Bajado de http://svm.first.gmd.de/, 1996.

[SWSST99] B. Schölkopf, R. C. Williamson, A. J. Smola, and J. Shawe-Taylor. Sv estimation of a distribution's support. *Aceptado en NIPS'99*, 1999.

[TA77] A.N. Tikhonov and V. Y. Arseni. *Solution off Ill-Posed Problems*. Winston, Washington, 1977.

[Tip00] M. Tipping. The relevance vector machine, 2000.

[Tod92] P. Todorovic. *An Introduction to Stochastic Processes and Their Applications*. Springer-Verlag, New York, 1992.

[Vap82] V. Vapnik. *Estimation of Dependences Based on empirical Data*. Springer-Velag, Berlin, 1982.

[Vap95] V. Vapnik. *The Nature of Statistical Learning Theory*. John Wiley & Sons, Inc, 1995.

[Vap98] V. Vapnik. *Statistical Learning Theory*. John Wiley & Sons, Inc, 1998.

[VC71] V. Vapnik and A. Y. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probab. and its Applications*, 16(2):264–280, 1971.

[VGS97] V. Vapnik, S. Golowich, and A. Smola. Support vector method for function aproximation, regression estimation, and signal processing. *Advances in Neural Information Processing Systems*, 1(9):281–287, 1997.

[Wah90] G. Wahba. *Spline Models for Observational Data*. Philadelphia. Pensylvania. SIAM., 1990.

[Wah98] G. Wahba. Support vector machines, reproducing kernels hilbert space and the gacv. In Proceeding of the 1997 NIPS Worshop on Support Vector Machines, 1998.

[WW83] E. Wegman and Y. Wright. Splines in statistics. *J.A.S.A.*, 1(78):351–366, 1983.

[WW98] J. Weston and C. Watkins. Multi-class support vector machines. Technical Report CSD-TR-98-04, Royal Holloway, University of London, Egham, UK, 1998.