

Trabajo Fin de Máster
Máster Universitario en Organización Industrial y
Gestión de Empresas

Mejora de la Eficiencia Energética de una Planta
Fotovoltaica con la Detección Temprana de Fallos
Mediante el Uso de Modelos Predictivos de Redes
Neuronales Artificiales

Autor: Jesús Ferrero Bermejo

Tutor: Adolfo Crespo Márquez

Co-Tutor: Juan F. Gómez Fernández

Dep. de Organización Industrial
Escuela Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2014



Trabajo Fin de Máster
Máster Universitario en Organización Industrial y Gestión de Empresas.

Mejora de la Eficiencia Energética de una Planta Fotovoltaica con la Detección Temprana de Fallos Mediante el Uso de Modelos Predictivos de Redes Neuronales Artificiales

Autor:

Jesús Ferrero Bermejo

Tutor:

Adolfo Crespo Márquez

Profesor titular

Co-Tutor:

Juan F. Gómez Fernández

Profesor Asociado

Departamento de Organización Industrial

Escuela Superior de Ingeniería

Universidad de Sevilla

Sevilla, 2014

Trabajo Fin de Máster: Mejora de la Eficiencia Energética de una Planta Fotovoltaica con la Detección Temprana de Fallos Mediante el Uso de Modelos Predictivos de Redes Neuronales Artificiales

Autor: Jesús Ferrero Bermejo
Tutor: Adolfo Crespo Márquez
Co-Tutor: Juan Gómez Fernández

El tribunal nombrado para juzgar el Proyecto arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2014

El Secretario del Tribunal

A mi familia

En particular a mis padres

En concreto a mi mujer

En especial a mis hijas.

Agradecimientos

A lo largo de toda mi vida han existido muchos periodos de formación, por lo que en primer lugar quisiera agradecer a todos aquellos profesores que han dedicado su vida y su tiempo a la formación de numerosos alumnos, cuya aportación es fundamental. Por tanto quiero hacer especial mención a la labor realizada por todos los docentes de cualquier ámbito.

En segundo lugar quisiera agradecer a todas aquellas personas que han sido fundamentales para la realización de este Máster. Por una parte Salvador Caro sin el cual no hubiese comenzado y compañero de tantas batallas; por otra parte Fernando Olivencia por su inestimable colaboración; por otra el profesor de la Facultad de Matemáticas Rafael Pino por su ayuda en la explicación de conceptos de Redes Neuronales y en la ayuda con el código para la resolución del problema y por último por el apoyo, colaboración, ánimo e inestimable ayuda de mis dos tutores Adolfo Crespo y Juan Francisco Gómez.

En tercer lugar agradecer a mis padres por su ayuda y sacrificio en la no fácil tarea de criar y educar a 5 hijos, los cuales siempre nos han aportado valores en la vida fundamentales para afrontar retos con honestidad y valentía. Es el mayor de los legados que sin duda pueden dejar en esta vida y del que deben estar orgullosos.

También quisiera agradecer a mi mujer su ayuda y comprensión en todas esas largas jornadas de dedicación en las que aparte de la soledad en la que la he tenido que dejar por la dedicación al máster (añadido a las ya largas jornadas laborales), no sólo no se ha quejado sino que ha sido un bastión fundamental dándome siempre fuerzas, apoyo y cariño. Muchas gracias Esther por tu cariño, por tu ánimo, por tu profundo amor y por estar ahí siempre que te he necesitado. Sin duda eres mi vida.

Finalmente mi especial dedicatoria a esos dos angelitos nacidos este año, María y Paola las cuáles desde su inocencia inspiran en mi toda la ternura y fortaleza que necesito en la vida para hacerme feliz y afrontar con fuerza cualquier reto. Espero algún día puedan leer este trabajo y se sientan orgullosas de su padre.

Resumen

Un importante número de empresas españolas han decidido invertir en los últimos años en instalaciones de generación de energías renovables, debido fundamentalmente a la rentabilidad de dichas instalaciones.

Dicha rentabilidad ha sido motivada por incentivos que el Gobierno español ha promocionado a través de decretos (R.D. 661/2007 de 25 de Mayo) que primaban considerablemente la generación de “energías limpias”. En concreto en Septiembre del 2008 comenzaron sus puestas en marcha numerosas Instalaciones Fotovoltaicas cuya energía eléctrica generada vertían a la red eléctrica española.

Estas instalaciones se construían en los denominados Parques Fotovoltaicos o Solares los cuales se agrupaban en instalaciones de 100 Kw, que denominaremos en adelante Huertos Solares. La inversión realizada por cada uno de estos Huertos Solares suponía una media de 660.000 € (dato que utilizaremos para todo el proyecto aunque dicha cantidad lógicamente puede variar dependiendo de la tecnología adoptada), por lo que en general cualquier instalación de este tipo por pequeña que sea, supone una considerable inversión.

El modelo de negocio inicial para estas instalaciones tenía previsto un periodo de 12 años para amortizar la inversión con una vida media de 25 años, lo que suponía una rentabilidad aproximada del 15%. A partir de dicha fecha podrían seguir produciendo los Parques Fotovoltaicos vendiendo la energía a precio de mercado sin incentivo alguno. Sin embargo, en este trabajo no nos vamos a centrar en el modelo de negocio, sino en cómo mejorar la producción lo que hará que lógicamente mejore la rentabilidad.

El largo periodo de vida de estas instalaciones, junto con posibles modificaciones en los incentivos regulados por Decreto (de hecho ya ha ocurrido) hace necesario tener modelos predictivos de producción adaptables en cada momento que nos permitan rehacer el modelo de negocio para conocer con exactitud (o al menos del modo más aproximado posible) el resultado de la importante inversión.

Conocido el modelo ideal de predicción de producción de la Instalación Fotovoltaica, éste nos permitiría por una parte rehacer el modelo de negocio y por otra comparar con la producción real, pero no nos aportaría más que una predicción que al fin y al cabo no modificaría la producción real, la cual es realmente lo que al propietario o inversor le interesaría poder mejorar. Es por ello que en este trabajo pretendemos dar un paso más adelante como aportación al conocimiento, obteniendo no sólo el modelo de predicción ideal, sino utilizándolo para mejorar la producción real de la instalación. Dada la gran inversión de un Parque Solar cualquier mejora en la producción mediante el modelo supondría una importante ganancia en euros.

Para la obtención del modelo ideal de producción existen diversas técnicas matemáticas, como el clásico modelo de regresión matemática que nos permitiría obtener resultados con intervalos de confianza y parámetros totalmente definidos para el modelo. Sin embargo, el patrón de comportamiento de producción no sigue relación directa o indirecta con el conjunto de variables principales que influyen directamente en la producción, de ahí que pudiendo ser el mejor método, en el caso particular no se adapta y no aporta una solución fiable. Es por ello que revisada la bibliografía el modelo seleccionado para el estudio por considerar que se adapta muy bien al patrón de comportamiento es el de las Redes Neuronales Artificiales.

Obtenido el patrón de comportamiento ideal de producción del Parque Solar, a través del entrenamiento de una red Neuronal con datos históricos de producción horaria a lo largo de un año (filtrados en ausencia de fallo, de modo que el patrón sea el ideal) obtendremos una comparativa con la producción real en cada momento.

Una vez realizada esta comparación, en el presente trabajo nos centraremos en los posibles fallos del Parque Solar y en concreto en aquellos fallos que afecten al menos a un Huerto Solar completo. De los fallos seleccionados volveremos a centrarnos en aquellos fallos que nos permitan obtener la caída o disminución en la producción con suficiente antelación a la detección del fallo por el Sistema.

Con un posterior análisis de criticidad y un sistema que defina la condición de alarma podremos implementar en el Scada un sistema de alerta temprana que permita avisar al personal de mantenimiento de la planta con objeto de reparar la avería o fallo, y evitar por tanto pérdidas innecesarias o visto de otro modo mejorar la producción o beneficio del Parque Solar.

Este Trabajo Fin de Master, desarrollado con el grupo de Sistemas Inteligentes de Mantenimiento de la ETSI de Sevilla, se enmarca dentro del proyecto de investigación del citado grupo de título SMARTSOLAR, del programa OPN – INNPACTO (Ref IPT-2011-1282-920000), del Ministerio de Ciencia e Innovación.

Abstract

An important number of Spanish companies have decided to invest in renewable energies generation, basically due to the profitability of these facilities.

This profitability has been caused by incentives which has been promoted by the Spanish Government across decrees (R.D 661/2007 from 25 of May) that reward the generation of “clean energies”. Concretely, in Septiembre of 2008, the Government started to run many Photovoltaic Facilities which generated electric power they were spilling to the electrical Spanish network.

These facilities were constructed in parks called The Photovoltaic Solar Parks, which gathered at facilities of 100 Kw, which we will name in forward Solar Gardens. The investment carry out per each of these Solar Gardens supposed an average of 600.000 €, which supposes a considerable investment.

The model of initial business for these facilities had foreseen a period of 12 years to amortize the investment with an average life of 25 years, which supposed an approximate profitability of 15 %. From the mentioned date they might continue producing the Photovoltaic Parks selling the energy to price of market without any incentive. However, in this work we do not go to centring in the model of business, but how we can improve the production what will do it improves the profitability.

The long period of life of these facilities, together with possible modifications in the incentives which are regulated by Decree (in fact already it has existed) it does necessarily to have predictive models of production which are adaptables in every moment and this model of business allows us to re-do it in order to know the result of the important investment precisely.

Known the ideal production's prediction's model of the Photovoltaic Installation, this would allow us, on the one hand, to re-do the model of and for comparing other one with the royal production, but it us would not contribute any more than a prediction that in the end it would not modify the royal production, which is really what the owner or investor would be interested in being able to improve. It is for it that in this work we try to give a step hereinafter as contribution to the knowledge, obtaining not only the model of ideal prediction, but using it to improve the royal production of the installation. Given the great investment of a Solar Park any improvement in the production by means of the model would suppose an important profit in Euros.

For the obtaining of the ideal model of production diverse mathematical technologies exist, as the classic model of mathematical regression who would allow us to obtain results with intervals of confidence and parameters totally defined for the model. Nevertheless, the standard of behaviour of production does not follow direct relation with the set of principal variables that influence directly in the production, of there that being able to be the best method, in the particular case it does not adapt and contributes a trustworthy solution. It is for it that checked the bibliografía the model selected for the study for thinking that he adapts very well to the standard of behaviour (to include bibliographical references that support it) is that of the Networks Neuronales Artificiales

Obtained the ideal standard of behaviour of production of the Solar Park, across the training of a network Neuronal with historical information of hourly production throughout one year (leaked in absence of failure, so that the boss is the ideal one) we will obtain the comparative one with the royal production in every moment.

Once realized this comparison, in the present work we will centre on the possible failures of the Solar Park and in I make concrete in those failures that they affect at least to a Solar complete Garden. Of the selected failures we will return to centre on those failures that allow us to obtain the fall or decrease in the production with sufficient anticipation to the detection of the failure for the System.

Una vez realizada esta comparación, en el presente trabajo nos centraremos en los posibles fallos del Parque Solar y en concreto en aquellos fallos que afecten al menos a un Huerto Solar completo. De los fallos seleccionados volveremos a centrarnos en aquellos fallos que nos permitan obtener la caída o disminución en la producción con suficiente antelación a la detección del fallo por el Sistema.

With a later analysis of criticidad and a system that defines the condition of alarm we will be able to implement in the Scada a system of early alert that it allows to warn to the personnel of maintenance of the plant in order to repair the breakdown or failure, and to avoid therefore losses unnecessary or I dress differently to improve the production or benefit of the Solar Park.

Con un posterior análisis de criticidad y un sistema que defina la condición de alarma podremos implementar en el Scada un sistema de alerta temprana que permita avisar al personal de mantenimiento de la planta con objeto de reparar la avería o fallo, y evitar por tanto pérdidas innecesarias o visto de otro modo mejorar la producción o beneficio del Parque Solar.

Finally we generalize the case of study so that following a methodology of work allow us to obtain similar conclusions in other industrial processes.

Agradecimientos	i
Resumen	iii
Abstract	v
Índice	vii
Índice de Tablas	ix
Índice de Figuras	xi
Notación	xv
1 OBJETO DE ESTUDIO	17
1.1 <i>Introducción al Proyecto</i>	17
1.2 <i>Objetivo General</i>	18
1.3 <i>Objetivos Específicos.</i>	19
1.4 <i>Justificación del Problema Seleccionado</i>	20
1.5 <i>Sumario del Proyecto</i>	20
2 ANTECEDENTES	23
2.1 <i>Introducción a las RNA</i>	23
2.1.1 <i>Historia de las Redes Neuronales Artificiales</i>	24
2.1.2 <i>Elementos Básicos que conforman una RNA</i>	27
2.1.3 <i>Tipos de Redes en función del número de capas.</i>	33
2.1.4 <i>Tipos de Redes en función del entrenamiento.</i>	35
2.1.5 <i>Clasificación de Redes.</i>	37
2.1.6 <i>Justificación del Modelo de Red Neuronal Artificial Seleccionada.</i>	38
2.1.7 <i>Algoritmo Teórico de las Redes Neuronales Backpropagation</i>	41
2.1.8 <i>Algoritmo para Minimizar el ECM</i>	45
3 APLICACIÓN PRÁCTICA CON EJEMPLO NUMÉRICO DE BACPROPAGATION	48
3.1 <i>Ejemplo Numérico de Aplicación de la Red Neuronal Backpropagation</i>	48
4 DESARROLLO. OBJETIVO GENERAL	57
4.1 <i>Caso de Estudio</i>	57
4.2 <i>Metodología</i>	58
4.3 <i>Planteamiento</i>	61
4.3.1 <i>Selección de Variables</i>	63
4.3.2 <i>Tratamiento de los datos.</i>	68
4.3.3 <i>Normalización de los Datos.</i>	70
4.3.4 <i>Software para la resolución: código. Primeros Resultados.</i>	72
4.3.5 <i>Arquitectura Seleccionada.</i>	81
4.3.6 <i>Entrenamiento y testeo.</i>	83
4.3.7 <i>Solución al problema en R</i>	89
4.4 <i>Objetivo General: Modelo Ideal de Producción</i>	92
5 DESARROLLO. OBJETIVO ESPECÍFICO	93
5.1 <i>Introducción a los Objetivos Específicos.</i>	93

5.2	<i>Algoritmo para la Detección de Fallos.</i>	95
5.2.1	<i>Priorización para la condición de Alarma.</i>	97
5.2.2	<i>Condición de Alarma</i>	99
5.3	<i>Fallo Modelo para Estudio</i>	100
5.4	<i>Comparación Modelo Ideal – Modelo Real en el fallo seleccionado.</i>	105
6	RESULTADOS	111
6.1	<i>Presentación de Resultados</i>	111
6.2	<i>Generalización del Modelo.</i>	114
6.3	<i>Discusión</i>	115
7	CONCLUSIONES	117
7.1	<i>Síntesis de las Conclusiones. Resultados</i>	117
7.2	<i>Experiencia en el Trabajo.</i>	119
8	LÍNEAS DE INVESTIGACIÓN FUTURA	121
	Referencias	123
	Participación en proyectos de investigación	125
	Méritos asociados al proyecto	126

ÍNDICE DE TABLAS

Tabla 1-1. Clasificación de las principales RNA conocidas	38
Tabla 1-2. Datos del Centro de Transformación	48
Tabla 2-3. Conjunto de valores de entrada para entrenamiento.	50
Tabla 3-4. Inicialización de los valores de los pesos. Todos a 1.	51
Tabla 4-5. Incremento para ajuste de los pesos de la capa de salida.	53
Tabla 5-6. Incremento para ajuste de los pesos de la capa oculta.	53
Tabla 6-7. Valores de los nuevos pesos tras primer patrón de entrenamiento.	54
Tabla 7-8. Serie de Valores para el entrenamiento a realizar como ejemplo.	54
Tabla 8-9. Tabla Resumen valores de los pesos tras los correspondientes entrenamientos.	55
Tabla 9-10. Mejoras en los errores tras los Entrenamientos.	55
Tabla 1-11. Configuración Estándar de un Centro de Transformación de la Instalación Fotovoltaica.	57
Tabla 2-12. Configuración de los 15 Centros de Transformación de la Instalación Fotovoltaica.	63
Tabla 3-13. Configuración del Centro de Transformación CT objeto de estudio seleccionado.	63
Tabla 4-14. Tabla con valores ejemplos utilizados para el estudio sin manipular.	67
Tabla 5-15. Tabla con valores normalizados.	71
Tabla 6-16. Tabla ejemplo con datos para carga en software.	74
Tabla 7-17. Resultado con valores obtenidos en R.	75
Tabla 8-18. Resultado con valores obtenidos en R.	76
Tabla 9-18. Resultado de R con número de células capa oculta	81
Tabla 10-19. Resumen valores entradas y salidas	90
Tabla 11-20. Pesos Arquitectura o Red Óptima. Entradas con Capa Oculta.	90
Tabla 12-21. Pesos Arquitectura o Red Óptima. Capa Oculta con Salida.	90
Tabla 1-22. Definición del Impacto	95
Tabla 2-23. Definición del Severidad.	97
Tabla 3-24. Definición de Criticidad.	98
Tabla 4-25. Definición de Prioridad.	98
Tabla 5-26. Prioridad – Tiempos de Respuesta y Resolución	99
Tabla 6-27. Definición Condición de Alarma	100
Tabla 7-28. Relación Prioridad – Condición de Alarma - Tiempos	100
Tabla 8-29. Tipos de Fallo por Impacto	101
Tabla 9-30. Fallos Seleccionados para el estudio.	102
Tabla 10-31. Producción en días previos al fallo por el Scada.	103
Tabla 11-32. Condición de Alarma.	105

Tabla 12-33. Relación Prioridad – Condición de Alarma - Tiempos.	105
Tabla 13-34. Fallo concreto seleccionado para el estudio.	106
Tabla 14-35. Datos ejemplos cargados en Software.	107
Tabla 15-36. Predicción frente a Producción. Inclusión condición de alarma.	108
Tabla 16-37. Predicción frente a Producción. Activación en contador 4 de alarma.	109
Tabla 17-38. Tiempos de resolución tras salto de alarma.	109

ÍNDICE DE FIGURAS

Figura 1-1. Sumario del Proyecto	22
Figura 1-2. Sinapsis Neuronal	23
Figura 2-3. Evolución Histórica de las RNA	26
Figura 3-4. Elementos que conforman una RNA	27
Figura 4-5. Neurona Artificial	28
Figura 5-6. Conexiones entre Neuronas	29
Figura 6-7. Función de Activación Escalón o Paso	30
Figura 7-8. Función de Activación Signo	31
Figura 8-9. Función de Activación Sigmoide o Logística.	31
Figura 9-10. Función de Activación Hiperbólica	32
Figura 10-11. Comportamiento Básico Red Neuronal	33
Figura 11-12. RNA Monocapa	34
Figura 12-13 RNA Multicapa.	35
Figura 13-14. RNA con Entrenamiento Supervisado	36
Figura 14-15. RNA con Entrenamiento NO Supervisado	36
Figura 15-16. Arquitectura Red Bacpropagation	42
Figura 16-17. Gráfico de Patrón de Comportamiento con pocos valores.	47
Figura 1-18. Topología de Red para Comprensión de las RNA con ejemplo numérico.	49
Figura 1-19. Imagen Aérea Instalación Fotovoltaica objeto de Estudio.	57
Figura 2-20. Metodología Utilizada en el Estudio	60
Figura 3-21. Metodología Utilizada en el Estudio. Selección de Variables	65
Figura 4-22. Metodología Utilizada en el Estudio. Selección de Variables: Horas de Funcionamiento.	66
Figura 5-23. Metodología Utilizada en el Estudio. Tratamiento y Validación de los Datos.	69
Figura 6-24. Gráfico de Caja y Bigote.	78
Figura 7-25. Gráfico mejor solución células capa oculta y valor de penalización del ECM.	82
Figura 8-26. Arquitectura Red Seleccionada.	83
Figura 9-27. Nube de puntos frente a la recta de valores reales en el Entrenamiento.	84
Figura 10-28. Comparación entrenamiento valores predecidos frente a valores reales.	85
Figura 11-29. Nube de puntos frente a la recta de valores reales en el Test.	86
Figura 12-30. Comparación test valores predecidos frente a valores reales.	87
Figura 13-31. Arquitectura Seleccionada Red Óptima.	89
Figura 1-32. Gráfico comparación producción Real frente a Predicción.	94
Figura 2-33. Gráfico de Producción diaria Inversor 1. Muestra de Fallo.	103
Figura 3-34. Gráfico de Producción diaria de Inversor 3. Muestra de Fallo.	104

Figura 4-35. Gráfico Producción Real Frente a Predicción en mes de Fallo.	108
Figura 5-36. Gráfico Prducción Real frente a Predicción en momento de detección del fallo por la RNA.	109
Figura 13-37. Arquitectura Seleccionada Red Óptima.	112
Figura 13-38. Metodología General.	114

Notación

a.C.	Antes de Cristo
a_j^0	Valor de la neurona j-ésima capa “o” al aplicarle la función de activación.
Adaline	Adaptative Linear Elements
Aij	Contador de lectura ij, relacionado con huerto solar.
ART	Adaptative Resonance Theory Networks ART
b	Término independiente o bias.
RNA	Redes Neuronales Artificiales
ANN	Artificial Neural Networks
BAM	Bi-Directional Associative Memory BAM
BSB	Brain State in Box
°C	Grados centígrados.
c.c.	En caso contrario
CT	Centro de Transformación
DBD	Delta Bar Delta
DNNA	Digital Neural Networks Architecture
DRS	Directed Random Search
FLN	Functional-link Networks
°K	Grados Kelvin.
min(j)	Mínimo Valor que toman los valores de la variable j.
max(j)	Máximo Valor que toman los valores de la variable j.
MIN-MAX	Técnica de Normalización de Datos.
n_j^0	Valor de la neurona j-ésima de la capa “o”
R.D.	Real Decreto
Kw	Kilovatio
Kw/h	Kilovatio / hora.
LVQ	Learninr Vector Quantization
Mw	Megavatio
ECM	Error Cuadrático Medio
F	Funcion de Activación
KVA	Kilo voltio amperio
MSE	Mean square error
MTTF	Tiempo medio hasta el fallo.
N(i)	Neurona i-ésima
PNN	Probabilistic Neural Network
R	Software libre utilizado para la resolución del problema.
RCM	Reliability centered maintenance
SOM	Self Organizing Mpas SOM
SPR	Spatio-Temporal-Pattern Recognition.

T_k	Salida real o salida deseada.
x_i	Valor de la variable de entrada i-ésima.
w_{ji}^0	Peso de la conexión neurona i-ésima con neurona j-ésima capa "o"
W/m ²	Medida de la radiación horizontal global en vatio / metro cuadrado.
Y_i	Predicción de producción con la RNA
Z_k	Salida calculada
:	Tal que
<	Menor o igual
>	Mayor o igual
\	Backslash
\sum	Sumatorio
\Leftrightarrow	Si y sólo si
Δ	Incremento
∇	Decremento

1 OBJETO DE ESTUDIO

1.1 Introducción al Proyecto

El mantenimiento de cualquier equipo cada día cobra más importancia en las empresas, pues todas ellas están orientadas a ofrecer servicios a clientes. No solo es importante el hecho de que un activo tenga una larga durabilidad para amortizar su inversión, sino que cobra mucha más importancia el hecho de que no se produzcan averías y que funcionen a pleno rendimiento en todo momento.

Por tanto, requisitos como la fiabilidad son más demandados en todas las empresas pues aportan garantía de funcionamiento y calidad hacia sus clientes posicionando de este modo a la empresa en el mercado de un modo mucho más competitivo.

Los activos de una empresa no solo deben funcionar durante mucho tiempo, sino que deben funcionar sin averías ya que cualquier parada o mal funcionamiento genera costes incalculables.

Todo ello nos lleva a considerar no solo los mantenimientos clásicos de equipos sino que asume una especial relevancia el prevenir fallos, de ahí el auge dentro del mantenimiento del uso de técnicas predictivas que eviten este tipo de paradas o cortes de servicios tan costosos.

El objetivo de cualquier industria o empresa es el de generar beneficios, para lo que se apoyan en una serie de medios que les permitan obtener el resultado deseado. Por tanto sería muy interesante el poder conocer la predicción futura de producción de dicha empresa sea el producto final que sea. En el presente trabajo pretendemos conocer la predicción de producción de Energía Eléctrica de una planta fotovoltaica de 6,1 Mw ubicada en el término municipal de Fuente Palmera, provincia de Córdoba.

Para la resolución del problema inicialmente se plantean dos posibles escenarios.

El primero de ellos consiste en la predicción de la producción a través de un modelo tradicional de regresión estadística, el cual nos proporcionaría una ecuación matemática. Para la obtención del resultado de dicha ecuación existen una serie de valores o variables de entrada, las cuales sin duda dependen de aspectos meteorológicos por una parte, por otra dependen del tiempo de funcionamiento de la instalación ya que con el tiempo existe una degradación de componentes que aminora la producción de energía y por otra parte depende del estado de ciertos elementos instalados y sus condiciones de contorno. Independientemente de la selección de variables de entrada no podemos obviar los aspectos anteriormente descritos lo que hace que de por si descartemos este método para la obtención de la predicción.

En un modelo estadístico de regresión para la obtención de un resultado con ciertas garantías debe existir una relación o dependencia entre las variables de entrada y la variable de salida. En el caso particular de estudio esta situación no se produce puesto que por ejemplo diferentes horas de funcionamiento de la instalación pueden obtener como resultado la misma producción; o bien la misma temperatura en distintas épocas puede ocasionar que nos dé resultados totalmente opuestos en la producción. Mucho más distante puede ser el resultado si combinamos las diferentes variables de entrada.

Otro inconveniente de este modelo para la resolución de problemas industriales de este tipo es la constante necesidad de actualización de los datos de entrada con los datos obtenidos para actualizar el modelo de predicción, lo que nos obliga a resolver constantemente el problema de un modo no trivial lo que lo hace poco operativo. Por tanto debemos decantarnos por otro método de resolución.

El segundo de ellos consiste en la resolución del problema a través del uso de Redes Neuronales Artificiales. Esta es la técnica seleccionada para la resolución del caso en el presente trabajo y que vamos a desarrollar y justificar en detalle.

Una vez definido el problema particular a resolver y la técnica seleccionada para ello, el esquema de trabajo será el que a continuación se describe:

- Descripción de los objetivos perseguidos con el estudio; tanto el objetivo general como los objetivos específicos.
- Justificar la elección del problema seleccionado en relación a las materias cursadas.
- Desarrollo del trabajo:
 - o Revisión histórica en detalle de las Redes Neuronales Artificiales.
 - o Justificación de la Red Seleccionada.
 - o Caso de Estudio.
 - o Metodología para la resolución.
 - o Planteamiento detallado y resolución del problema.
 - o Arquitectura del modelo ideal de producción.
 - o Aplicación del modelo ideal de producción a la mejora de la producción.

1.2 Objetivo General

El Objetivo general del estudio es del predecir la producción de energía eléctrica de la Instalación Fotovoltaica, obteniendo un modelo ideal de producción. El modelo ideal de producción será obtenido utilizando como datos de entrada para el aprendizaje de la Red Neuronal Artificial seleccionada los datos históricos de producción de la planta.

Para ello centraremos el estudio en un huerto solar de 100 kw tipo en el que existen sensores instalados, que nos han permitido obtener en el Scada todos los valores de las variables de entrada del modelo que posteriormente definiremos. El modelo de producción obtenido se considera ideal debido a que es un modelo de producción en ausencia de fallo.

Objetivo General: “Obtención del Modelo Ideal de Producción”

1.3 Objetivos Específicos.

El objetivo específico es la utilización del Modelo Ideal de Producción para la obtención de conclusiones particulares que permitan obtener mejoras en la producción de la planta fotovoltaica o para cualquier otra cuestión relacionada que afecta a otras áreas. En sí la búsqueda de dichos objetivos es la parte más interesante del Proyecto en el sentido de que es una nueva aportación al conocimiento a través del uso de técnicas novedosas.

Algunos de los objetivos específicos que podríamos obtener a través del conocimiento del Modelo Ideal de Producción son los siguientes:

- Mejora de la Eficiencia Energética.
- Generalización a cualquier Instalación Fotovoltaica.
- Generalización del Modelo a multitud de procesos industriales
- Creación de Algoritmo para la detección de Fallos con degradación en la producción previa a la detección del fallo por el Scada.
- Aplicación en la mejora del Mantenimiento.
- Mejora del Modelo de Negocio con la previsión de resultados económicos según el modelo Ideal de Producción.
- Mejora económica al obtener mayor producción.
- Anticipación a fallos.
- Validación de la detección de alarmas del sistema de monitorización de la Planta.
- Etc.

Puesto que los objetivos específicos son muchos, nos vamos a centrar en uno de ellos a través del cual se pueden deducir otros de los considerados. El Objetivo específico en el que nos vamos a centrar es en el de la Mejora de la Eficiencia Energética de la Planta tal y como indica el propio nombre del Proyecto.

Para ello compararemos el Modelo Ideal de Producción obtenido con la producción real lo que nos permitirá detectar desviaciones significativas en la producción para poder anticiparnos a fallos en la planta, con la generación de alarmas que activarán al personal de mantenimiento para la resolución de dicho fallo. Ello evitará la pérdida de producción en un número determinado de horas según el fallo lo que redundará en una mayor generación económica.

Finalmente, podremos modelar fallos y en concreto nos centraremos en uno de ellos a modo de ejemplo genérico, obteniendo datos concretos sobre ganancia en la producción con la implantación de este método.

Por tanto, el objetivo específico perseguido (como el fundamental) en el presente trabajo es el de la “Mejora de la Eficiencia Energética de la Planta Fotovoltaica con la detección temprana de fallos”

La detección temprana de fallos supondrá un adelanto en la detección del fallo sobre la alarma que produce el sistema de monitorización de la planta. Por ello todo fallo detectado en el mismo momento por el Scada y por nuestro sistema es descartado para el estudio, puesto que no supondría ventaja alguna sobre el sistema utilizado en la actualidad. La dificultad de este apartado radica por tanto en poder detectar algún fallo que produzca pérdida en la producción de al menos un Huerto Solar sin que por ello se produzca la alarma en el Scada hasta pasado al menos algún tiempo considerable (en nuestro caso consideramos dicho período como un mínimo de 24 horas de anticipación), de modo que efectivamente con la detección temprana de dicha caída en la producción detectemos anomalías y por tanto podamos reparar con la ganancia de tiempo y producción.

El fallo no sólo debe ser seleccionado sino que además para darlo como válido tendrá que producirse en diversos inversores y en diferentes ocasiones puesto que un fallo puntual no volverá a repetirse y por tanto tampoco podrá volver a ser detectado. Se trata de que la conclusión obtenida sea implantada de modo real en la Instalación Fotovoltaica con objeto de que una vez probado en dicha Instalación y verificada su validez el estudio sea ampliado a todas las plantas que actualmente son mantenidas por la empresa.

Seleccionado al menos uno de esos fallos podremos modelarlo, de modo que suponga para nuestro sistema una mejora en la eficiencia de la producción. Finalmente con el sistema de alarma que definiremos, de producirse la alarma y su correspondiente resolución de la avería a través de los técnicos de mantenimiento, podremos garantizar que se mejora la producción al adelantarnos al fallo pero no podremos concluir que sea un fallo determinado el que iba a producirse con la alarma producida con el sistema implantado.

1.4 Justificación del Problema Seleccionado

Diversas materias de estudio despiertan en mí un gran interés. Por una parte las estadísticas en general y en particular la predicción de resultados, y por otra la fiabilidad y mantenimiento de equipos que supongan mejoras en la producción de una industria. La utilización de ambas técnicas bien modeladas pueden suponer un nuevo modo de entender el mantenimiento en la industria, puesto que se trataría no de reparar ante la ocurrencia de un fallo, sino que podríamos de este modo anticiparnos al fallo mejorando en general la eficiencia en la producción de cualquier industria a la que se le apliquen este tipo de técnicas de mantenimientos predictivos o futuribles.

Este interés en dichas materias unido a la posibilidad de poder acceder a datos fiables y de calidad de una Planta Fotovoltaica han sido motivación fundamental para la elección del tema tratado en el presente trabajo. Dicha motivación aumenta con el resultado obtenido el cual se va a implementar en la Planta objeto de estudio, pudiendo por tanto realizar una gran aportación en la planificación del mantenimiento y en la generación de beneficios de la empresa.

1.5 Sumario del Proyecto

A continuación exponemos los principales capítulos en los que se desarrolla el presente Proyecto Fin de Máster.

Capítulo 1.- Objeto de Estudio.

En este capítulo realizo una introducción al problema planteado, cuyo fin particular consiste en la Mejora de la Eficiencia Energética de una Planta Fotovoltaica con la Detección Temprana de Fallos Mediante el Uso de Modelos Predictivos de Redes Neuronales Artificiales. Para ello inicialmente se requiere la obtención de un Modelo de Predicción Ideal en ausencia de fallos que nos permita a través de su comparación con la producción real la detección de fallos y por tanto su resolución y mejora en la producción. Todo ello nos lleva de por sí a la obtención de los diferentes objetivos específicos planteados.

Para la resolución del Problema cuya entidad es considerable por la complejidad de las variables consideradas y el volumen de datos tratados que obliga a manejar materias diferentes, se ha utilizado una metodología innovadora y que no es otra que la de la utilización de Redes Neuronales Artificiales.

Capítulo 2.- Antecedentes.

En este capítulo realizamos un recorrido histórico a través del cual surge la Inteligencia Artificial y con ella las Redes Neuronales Artificiales, las cuales han tenido sus altibajos y de hecho continúan teniendo hoy en día sus detractores. Revisada la historia y evolución de las Redes Neuronales Artificiales, exponemos en detalle los elementos que componen dichas redes y a través de ellos se realizan las primeras clasificaciones, llegando a exponer un listado de las redes más conocidas en la actualidad. Finalmente y dentro de ese conjunto de Redes Neuronales Artificiales, seleccionamos para el estudio la Red Backpropagation justificando su elección. Puesto que es la red seleccionada a continuación explicamos su funcionamiento teórico, detallando el algoritmo y la minimización del Error Cuadrático Medio, condición para la obtención del modelo ideal o mejor red entrenada que se ajusta al patrón de comportamiento deseado.

Capítulo 3.- Aplicación Práctica con ejemplo Numérico de Backpropagation

En este capítulo y dada la dificultad de comprensión y abstracción de las Redes Neuronales Artificiales aportamos un ejemplo numérico detallado que facilita su comprensión y justifica como con el entrenamiento la red se adapta perfectamente a un determinado patrón. Igualmente este capítulo esboza la gran capacidad de las RNA para resolver problemas, por ello realizamos un ejemplo totalmente diferente al caso de estudio del presente Proyecto.

Capítulo 4.- Desarrollo. Objetivo General.

En este capítulo nos centramos en la consecución del Objetivo General del Proyecto que no es otro que la consecución del Modelo Ideal de Producción a través del uso de la RNA Backpropagation de una Planta Fotovoltaica de 6,1 Mw ubicada en la provincia de Córdoba y de la disponemos de datos de producción, variables y fallos desde su puesta en marcha en Septiembre del 2008. Para la obtención del modelo ideal creamos una arquitectura de red modelada a través de una serie de variables de entrada que son las que inciden de forma clara en la producción de la Instalación. Con dicha arquitectura obtenemos los datos necesarios para el entrenamiento de la red de las variables seleccionadas y con un determinado software resolvemos el problema obteniendo los pesos ideales los cuales nos aportan el Modelo Ideal de Red.

Capítulo 5.- Desarrollo. Objetivos Específicos

Son diversos los objetivos específicos que se alcanzan con el modelo (pudiendo incluso obtenerse muchos más), centrándonos en este caso en la mejora de la eficiencia energética. Para ello se define un algoritmo a través del cual y con las condiciones definidas, nos permite encontrar fallos en la planta fotovoltaica que producen degradación en las producciones previas a la detección del fallo por el sistema de monitorización de la instalación. Esa degradación sin embargo si es detectada con la comparación con el modelo ideal de producción lo que nos lleva a obtener beneficios en la producción con todo lo que conlleva.

Este capítulo ha supuesto un importante riesgo debido a que si no llegamos a encontrar ningún fallo de las características definidas en este apartado, el modelo ideal no nos hubiera servido más que para comparar con la producción real sin obtener por ello ganancia alguna. Es por tanto que es este punto el que hace una aportación al conocimiento con la mejora expuesta en los objetivos específicos planteados.

Capítulo 6.- Resultados

En este capítulo incluimos los resultados obtenidos en el Proyecto como resumen de todo ello. Incluimos tanto el modelo ideal de producción obtenido como la ganancia que se obtiene con su utilización en lo que hemos denominado Mejora de la Eficiencia Energética de la Instalación con el uso de Redes Neuronales Artificiales. Referenciamos del mismo modo los objetivos específicos obtenidos y generalizamos el modelo, de modo que se ha creado una metodología de trabajo que permite implementar esta técnica en multitud de procesos industriales. Por último incluimos una discusión en el capítulo por la controversia que a veces provoca la utilización de las RNA.

Capítulo 7.- Conclusiones

Exponemos de modo sistemático la principal conclusión obtenida con el trabajo aportando mi visión personal sobre la experiencia que ha supuesto esta investigación.

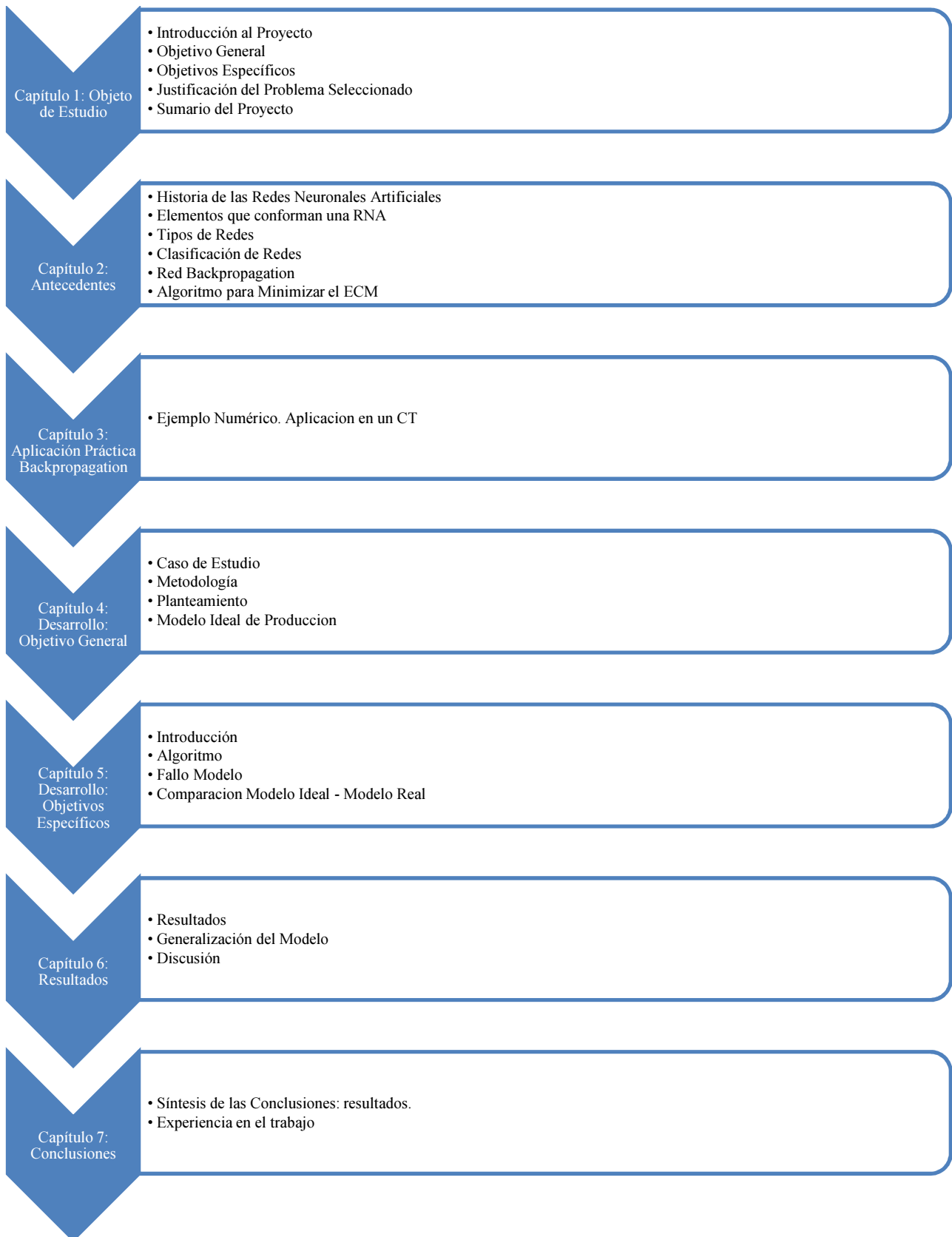


Figura 1-1. Sumario del Proyecto

2 ANTECEDENTES

2.1 Introducción a las RNA

El hombre a lo largo de la historia siempre ha querido mejorar, inventando máquinas y técnicas que facilitan y mejoran la vida de las personas. Muchos de los esfuerzos han ido encaminados a la construcción de máquinas que permitieran realizar operaciones de manera automática de modo que ahorraran cálculos pesados realizados manualmente.

La primera computadora electrónica fue construida por ENIAC en el año 1946, año a partir del cual ha evolucionado con una progresión auténticamente espectacular. Todas estas máquinas permitían resolver algoritmos realmente complicados de un modo relativamente rápido ahorrando tiempo y permitiendo evolucionar a la tecnología gracias al apoyo de estas computadoras.

Por otra parte, en la vida cotidiana existen multitud de problemas que no se pueden resolver mediante algoritmos, como por ejemplo el reconocimiento del rostro de una persona o multitud de problemas existentes en la vida real. Sin embargo, este tipo de problemas tiene un nexo común y es que los hombres resolvemos fácilmente gracias a la experiencia. A partir de este planteamiento básico, los científicos se plantean resolver problemas o implementar en máquinas algoritmos basados en el cerebro humano, lo que nos permitiría resolver otro tipo de problemas no modelables mediante algoritmos y de dónde surge por tanto la denominada Inteligencia Artificial.

Esta disciplina comienza a desarrollarse por científicos con el objeto de resolver cuestiones basándose en la inteligencia humana o más en concreto en el modo que resuelve el cerebro humano. De ahí que una Red Neuronal no sea más que un artificio para reproducir una parte del cerebro humano que permita tener parte de sus cualidades, como pueden ser el aprendizaje a partir de la experiencia, la capacidad de memorizar o la capacidad de unir hechos o ideas para obtener nuevos resultados. La idea en si es realmente interesante, sin embargo complicada de implementar pues no hay más que ver la cantidad de “elementos” que dispone el cerebro humano.

La siguiente figura muestra una simple unión de la unidad básica del cerebro humano, una neurona y sus uniones con otras:

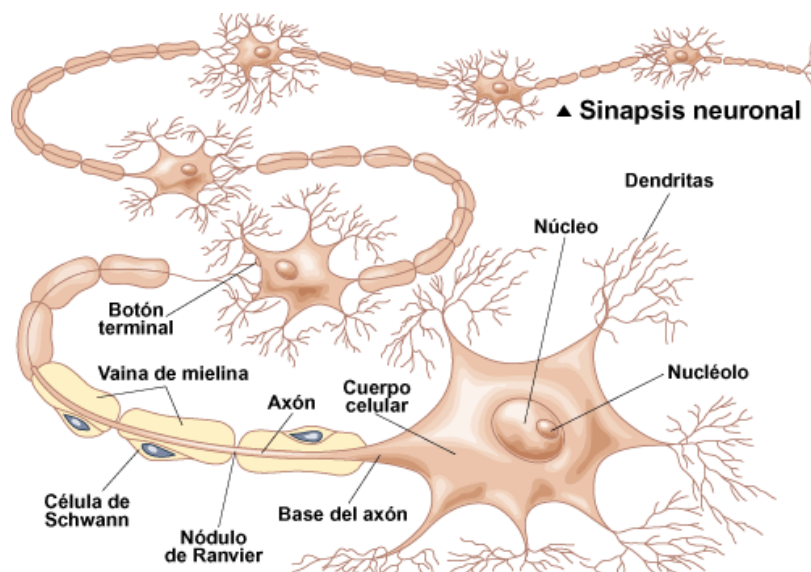


Figura 1-2. Sinapsis Neuronal

(“Fuente: www.definicionabc.es”)

Como datos básicos a considerar que nos da idea de la complejidad del cerebro humano y por tanto la dificultad de simularlo, sean los siguientes:

- Peso del Cerebro Humano: 1.300 grs
- Número de dendritas: entre 10.000 y 100.000
- Número de Neuronas: 30.000 millones
- Número de interconexiones: 100 trillones.

Biológicamente el procesamiento de la información en el cerebro se produce a través de las diferentes transmisiones entre las neuronas, transmisión que se produce a través de las conexiones entre ellas o las denominadas sinapsis.

El modo básico de procesar la información es el siguiente: una neurona se activa a través de la información de las entradas o estímulos del exterior que recibe y cuando ésta alcanza un determinado nivel entonces la neurona se activa pasando de nuevo otra señal procesada a través del axón. Esta explicación simple del funcionamiento del cerebro nos permitirá entender mejor el funcionamiento de las redes neuronales artificiales.

Por tanto, las redes neuronales artificiales se caracterizan por: múltiples unidades que conectadas entre sí intercambian información, tienen la capacidad de aprender de la experiencia y por tanto de aprender patrones lo que les permite mejorar los resultados a través del entrenamiento.

En adelante denominaremos por sus siglas a las redes neuronales artificiales, RNA, o bien ANN por su nomenclatura en inglés Artificial Neural Networks.

2.1.1 Historia de las Redes Neuronales Artificiales

Aunque los diferentes autores coinciden que en 1943 es cuando se dieron a conocer los primeros conceptos o base científica de la computación neuronal, existen escritos que esbozan las primeras ideas sobre el comportamiento del pensamiento e inteligencia del cerebro humano. Sirvan ellos como antesala a la explicación histórica del concepto más formal de inteligencia artificial pues a nivel conceptual o de simple pensamiento han existido a lo largo de la historia desde años antes de Cristo.

Los primeros escritos históricos fueron realizados por filósofos, como por ejemplo Platón (427 – 347 a. C) y Aristóteles (348 – 422 a. C) que ya vislumbraban alguna explicación teórica entre cerebro y pensamiento. Estas ideas además han sido desarrolladas y mantenidas posteriormente por Descartes (1569 – 1650) y filósofos empiristas del siglo XVIII. Por tanto, el concepto como tal no es nuevo, sino que lo que se ha aportado al conocimiento científico es el rigor apoyado en los medios tecnológicos que permiten dicho desarrollo.

Los primeros científicos a los que realmente se les ha reconocido los primeros conceptos teóricos sobre computación neuronal fueron el *neurobiólogo Warren McCulloch* y el *estadístico Walter Pitts*, los cuales publicaron en 1943 el artículo “*A logical calculus of Ideas Imminent in Nervous Activity*” cuyo contenido expresaba la forma de trabajar de las neuronas, modelando una red neuronal a través de circuitos eléctricos.

Este artículo ha sido la base para conocimientos posteriores:

- Ordenadores Digitales de John Von Neuman
- La Inteligencia Artificial por Marvin Minsky
- Red Neuronal Perceptron de Frank Rosenblatt que explicaba el funcionamiento del ojo.

En 1949 Donal Hebb publicó el libro “La organización y el comportamiento” que intentaba explicar desde la psicología, como eran los procesos de aprendizaje del ser humano. Cada vez que una persona aprendía algo nuevo una neurona era activada, por lo que claramente expresa relación aprendizaje – red neuronal (que como hemos visto es una de las características de una red neuronal, aprende a través de la experiencia). Los trabajos de este autor son la base de la Teoría de las Redes Neuronales.

Otro avance se produjo en 1950, año en el que Karl Lashley descubrió que la información que llega al cerebro no se guarda como tal, sino que es distribuida. Es decir, aparece otro de los principios básicos, que es la propagación de la información a través de la red.

Como hemos comentado anteriormente, en 1957 Frank Rosenblatt inventó la conocida y aún utilizada hoy en día Red Perceptron, cuya aplicación está en el reconocimiento de ciertos patrones. Basado en la experiencia esta red podría aprender e incluso a partir de dichos patrones reconocer otros, aunque lógicamente tenía sus propias limitaciones pues básicamente resolvía problemas o reconocía patrones que se podían en cierto modo clasificar como modelos lineales.

La siguiente Red Neuronal Artificial inventada fue la red Adaline (Adaptative Linear Elements) en el año 1960 por Bernard Widrow y Marcial Hoff. Ha sido la primera red empleada en la resolución de un problema real, en concreto se ha utilizado durante mucho tiempo para eliminar los ecos de las líneas telefónicas.

En 1961 el científico alemán Karl Steinbeck desarrolló el primer modelo matemático de “memoria asociativa” con la red denominada Die Lernmatrix, aunque fue un trabajo que no tuvo mucho auge y se quedó en dicho modelo sin prosperar.

Posteriormente ya en 1967 el científico Stephen Grossberg inventó la red “Avalancha” que era indicada en la resolución de patrones de reconocimiento del habla y para el aprendizaje de brazos de un robot. Incorpora el concepto de variables continuas en el tiempo.

Debido que hasta las fechas descritas no existía una base matemática de soporte de las redes neuronales, surgieron críticas al respecto hacia dichas redes que paralizaron su estudio de forma bastante radical. Causantes entre otros de ello, fueron Marvin Minsky y Seymour Papert quienes en 1969 publicaron el libro Perceptrons, demostrando que la red perceptron era muy débil debido a que la mayoría de los procesos de la naturaleza seguían funciones no lineales que no eran capaces de resolver dicha red. Aunque investigadores siguieron trabajando en la investigación y avance de las redes neuronales artificiales, supuso un frenazo considerable en dichas investigaciones. Hasta 1982 los avances fueron lentos en este campo.

Como hitos significativos hasta el relanzamiento de las investigaciones destacamos los siguientes:

- James Anderson descubrió el modelo “Asociador Lineal” así como su desarrollo con el BSB (Brain State in Box). Dicho modelo suponía una relación lineal de las neuronas obteniendo como resultado la suma de sus entradas.
- Ya en 1974 se produjo un importante descubrimiento con una idea de aprendizaje nuevo que consistía en la propagación hacia atrás o retro propagación. Dicha base sería desarrollada posteriormente en lo que supondría el relanzamiento del estudio y aplicación de las redes neuronales artificiales.
- En 1977 Stephen Grossberg desarrolló la Teoría de Resonancia Adaptada, la cual añade dos nuevas propiedades a las redes, la memoria a corto y a largo plazo. Cualidades que como en todas las anteriores simulan al comportamiento del cerebro humano.

- En este mismo año el ingeniero electrónico Stephen Grossberg desarrolló un modelo basado en el modelo de James Anderson pero incluyendo la independencia.
- En 1980 ya casi de nuevo en el relanzamiento de las RNA, Kunihiro Fukushima creó una red neuronal para el reconocimiento de patrones visuales.
- En 1985 se produjo de nuevo el auge e interés por las RNA con la publicación del libro “Computación Neuronal de Decisiones en Problemas de Optimización” del científico John Hopfield.

A partir de dicha fecha, se produjo como hemos comentando el relanzamiento de estudios de las RNA, publicándose constantemente artículos así como surgiendo aplicaciones en el mundo empresarial. Es un campo apasionante aún por desarrollar que puede aportar gran valor a la comunidad científica y al mundo empresarial pues numerosas son las aplicaciones y grandes las cualidades y capacidades del cerebro humano por imitar por computadoras.

En 1986 los científicos David Rumelhart y G Hinton desarrollaron el algoritmo de la Red Backpropagation, algoritmo de aprendizaje que debido a la propagación hacia atrás permite con el entrenamiento reconocer patrones, lo que la convierte en una red muy utilizada para el cálculo de modelos predictivos (Lapedes y Farber (1987)). Dicha retro propagación aporta una base matemática que la convierte en una potente herramienta para simular procesos pudiendo calcular valores medios de separación entre los valores obtenidos por dicha red y los valores deseados o reales. La proximidad de estos valores es lo que puede aportar que el comportamiento de la red tras el aprendizaje sea válido para el proceso a predecir permitiendo conocer parámetros como la correlación entre ellos, lo que explica la variación media de los valores.

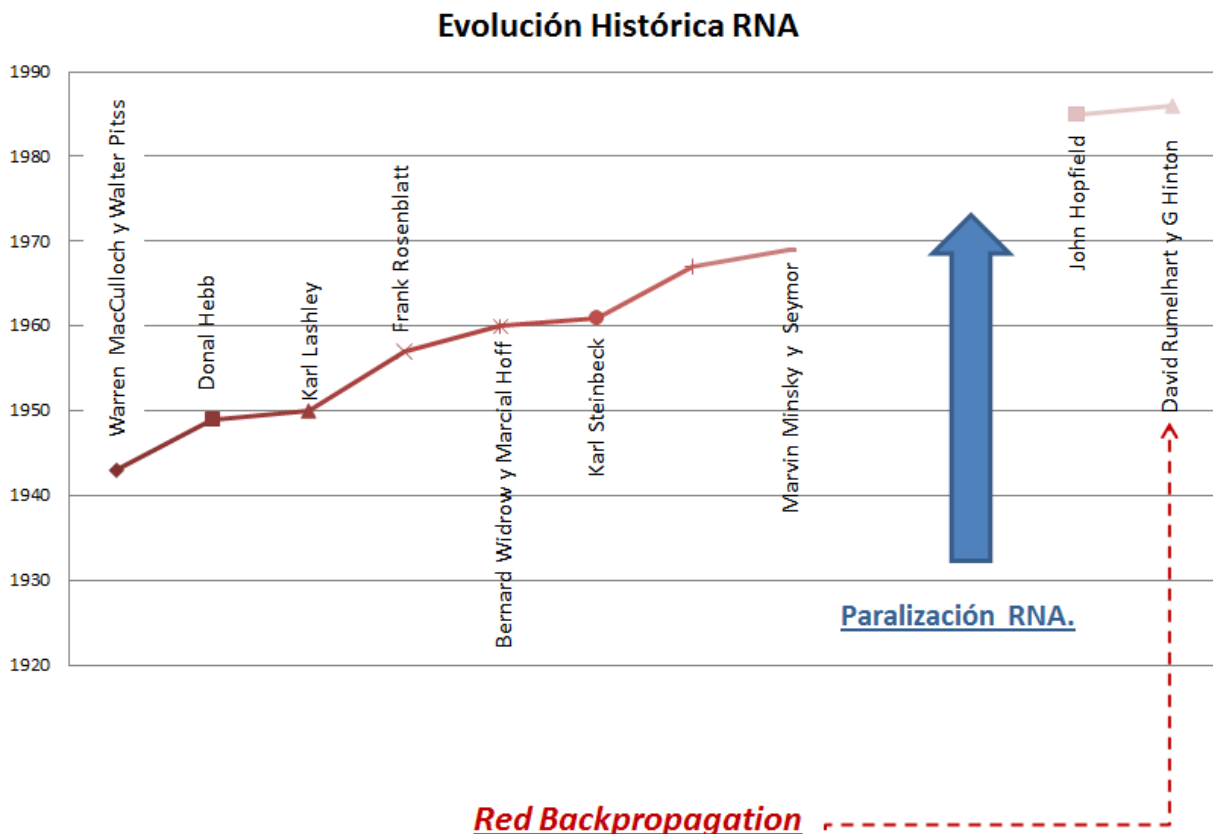


Figura 2-3. Evolución Histórica de las RNA

2.1.2 Elementos Básicos que conforman una RNA

Una vez realizada la revisión histórica de las Redes Neuronales Artificiales, en este apartado vamos a indicar cuáles son los elementos básicos que componen una red neuronal, independientemente del tipo de red. A partir de la composición de dichos elementos y con diversas técnicas se construirían todas las redes que tienen estos elementos en común.

Las Redes Neuronales Artificiales intentan simular el comportamiento del cerebro humano, por lo que a partir del comportamiento de éste podemos deducir el comportamiento de las redes. Como elemento fundamental en las RNA tenemos las neuronas, a las que les llega un estímulo: dichas neuronas se pueden localizar en diferentes capas. La primera de ellas es aquella en la que la neurona recibe la información sensorial u información inicial que no procede de otras neuronas, por lo que las neuronas de este tipo son las que denominamos neuronas de la capa de entrada. Son aquellas que reciben la información de entrada o inicial y que es procesada en diversas etapas hasta obtener una salida o resultado. Por ejemplo es el estímulo que recibimos con la visión y que es procesado hasta reconocer un rostro, o es una variable como la temperatura ambiente y que tras procesar en varias etapas obtenemos como resultado la producción de energía eléctrica de una instalación fotovoltaica. Son las denominadas neuronas de la capa de entrada.

Previo a la obtención de un resultado o salida, procesamos la información en diversas etapas intermedias, etapas que no son más que cálculos y que al no ser útil salvo como medio, denominaremos a dichas neuronas, neuronas de las capas ocultas. Pueden existir una o múltiples capas ocultas, lo cual aportará una primera clasificación de la RNA.

Finalmente y tras el procesamiento intermedio la información llega a las denominadas neuronas de la capa de salida ya que procesan la información final obteniendo el resultado o valor. El esquema genérico por tanto de una red neuronal general será el siguiente:

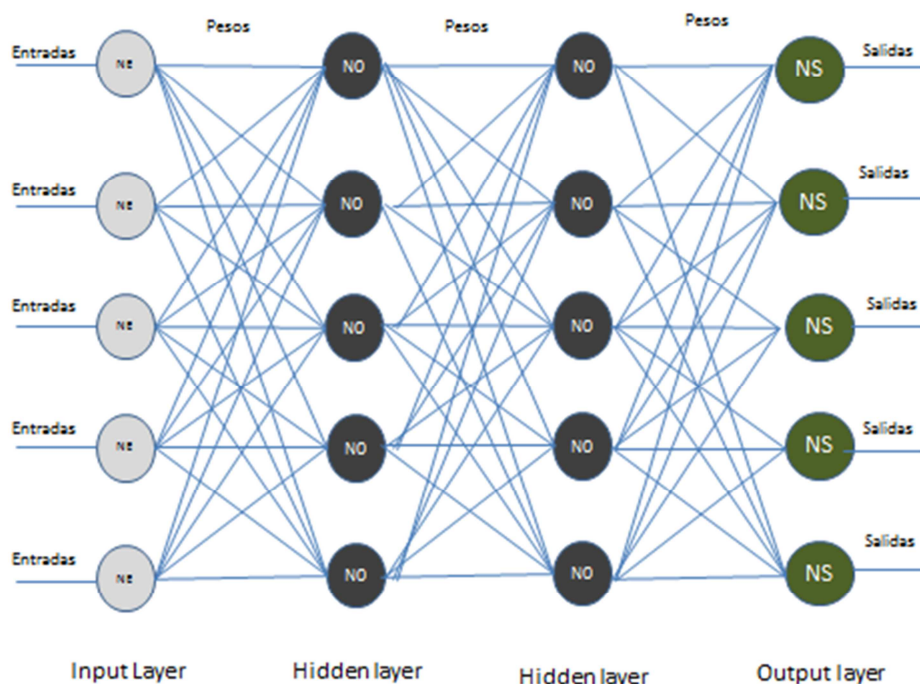


Figura 3-4. Elementos que conforman una RNA

Como resumen, las redes neuronales están compuestas por:

- Capa de Entrada.
- Capas Ocultas o Intermedias: puede existir una o múltiples.
- Capa de Salida.

En general una Red Neuronal no es más que un modelo de procesar distribuidamente en forma de grafo dirigido.

Por tanto, una vez explicada la estructura general de una Red Neuronal, exponemos la composición y funcionamiento del elemento básico: la Neurona Artificial. La combinación y distribución de todas las neuronas es lo que compone y da forma a las diferentes redes neuronales.

El funcionamiento de cada una de estas neuronas viene determinado por el siguiente gráfico:

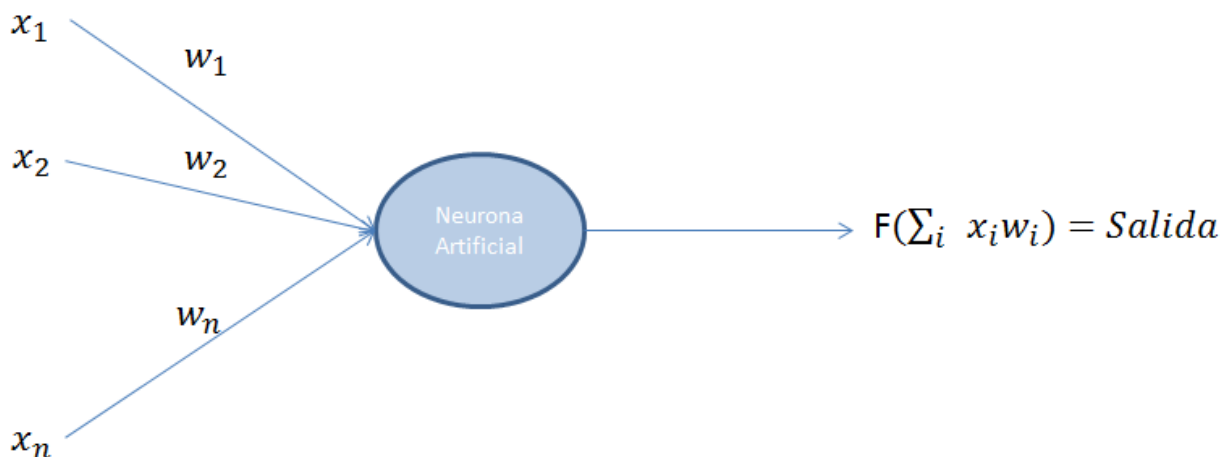


Figura 4-5. Neurona Artificial

Siendo F la función de Activación que se aplica al resultado de la salida de la neurona, la cual suele ser el sumatorio de las entradas recibidas por sus correspondientes pesos.

En dicho gráfico se observa por tanto los componentes básicos de una Red Neuronal y que son los siguientes:

- Neurona: tiene por defecto un estado, el cual puede variar en función de la información recibida que es procesada a través de una función de activación.
- Conexiones entre todas las neuronas. Dichas conexiones tienen asociado un peso que no es más que la tasa de aprendizaje o peso (importancia) que se le da a una determinada variable. Dichos pesos van modificándose con el entrenamiento.
- Función de Transferencia: función de activación que aplica sobre el conjunto de entradas que llegan a la neurona, transformando el estado actual en una señal de salida. Existen diferentes funciones de activación que explicaremos más adelante.

A continuación explicamos un poco más en detalle cada uno de estos elementos que componen la Red Neuronal.

2.1.2.1 Neuronas

Las Redes Neuronales están compuestas por un conjunto de neuronas, que se denotan habitualmente de modo consecutivo, siendo la notación que utilizaremos

- $N(i), i= 1, \dots\dots\dots$

Las neuronas están distribuidas a lo largo de la Red Neuronal en diferentes capas, bien en la de Entrada, bien en la o las capas ocultas o bien en la Capa de Salida. La función de dichas neuronas no es más que la de procesar la información recibida a través de las conexiones y pesos de la capa anterior, devolviendo una salida que se transmite bien a la capa siguiente o bien como resultado final.

Las Neuronas tienen un estado de activación a los que se les denomina normalmente Reposo o Excitación, el cual viene determinado por el conjunto de entradas que recibe por su correspondiente peso. Es decir, la suma ponderada de las entradas que recibe una determinada neurona constituye el nivel de activación de la neurona.

En función por tanto del nivel de activación en el que se encuentra una neurona, transmitirá la correspondiente señal de salida (lógicamente también dependiente de la función de activación)

2.1.2.2 Conexiones

Las RNA tienen un conjunto de conexiones que son aquellas que unen entre sí a las diferentes neuronas de las diferentes capas que componen la red. Estas conexiones tienen asociadas unos determinados pesos que son justamente los que hacen que la red vaya adquiriendo conocimiento a través del aprendizaje y el entrenamiento. Por tanto los pesos se van modificando a través de la “experiencia”.

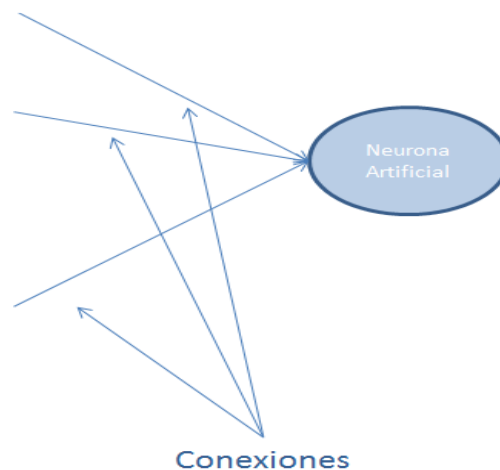


Figura 5-6. Conexiones entre Neuronas

La señal de salida de una determinada neurona es el efecto de añadir o sumar las diferentes conexiones entrantes por sus respectivos pesos, es lo que se denominada regla de propagación.

$$(\sum_i x_i w_i) = y_i \quad (1)$$

Finalmente a dicha salida se le aplica la función de activación, que se explica en el siguiente apartado.

2.1.2.3 Función de Activación.

La función o regla de activación es aquella que modifica el estado de activación de una determinada neurona de la red, produciendo un nuevo estado de activación. Dicho nuevo estado resulta de aplicar la función a la señal de salida (obteniendo una nueva salida que sería la que se aplica a la neurona de la siguiente capa)

$$F(\sum_i x_i w_i) = F(y_i) = \text{Salida} \quad (2)$$

Denotamos por tanto con F a la función de activación, denominada en la bibliografía función de activación o función de transferencia. Exponemos una de las principales clasificaciones que existen de dichas funciones de activación o transferencia:

Función de Activación Escalón o de paso:

Viene determinada por la siguiente función:

$$F(y_i) = 1 \text{ si } (y_i) \geq 0 \quad \text{o} \quad F(y_i) = 0 \text{ en caso contrario.} \quad (3)$$

La representación gráfica de dicha función es la que a continuación se denota, existiendo como su nombre indica un escalón en función de la positividad o negatividad del resultado obtenido tras la aplicación de la función de activación.

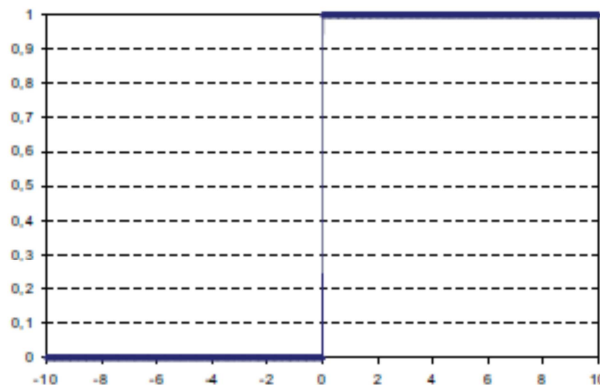


Figura 6-7. Función de Activación Escalón o Paso

Función de Activación Signo

Esta función de activación es análoga a la anterior con la única diferencia de que los valores obtenidos tras la aplicación de la función de activación son el 1 y el -1.

$$F(y_i) = 1 \text{ si } (y_i) \geq 0 \quad \text{o} \quad F(y_i) = -1 \text{ c. c.} \quad (4)$$

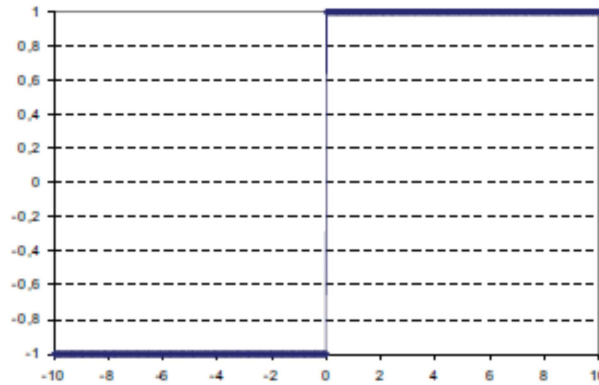


Figura 7-8. Función de Activación Signo

Función de Activación Sigmoidea o logística

Esta función de activación viene denotada por la siguiente función de activación:

$$F(y_i) = \frac{1}{e^{-y_i} + 1} \quad (5)$$

La representación gráfica es la que se muestra a continuación, en la cual como se puede observar el resultado obtenido con dicha función varía entre 0 y 1.

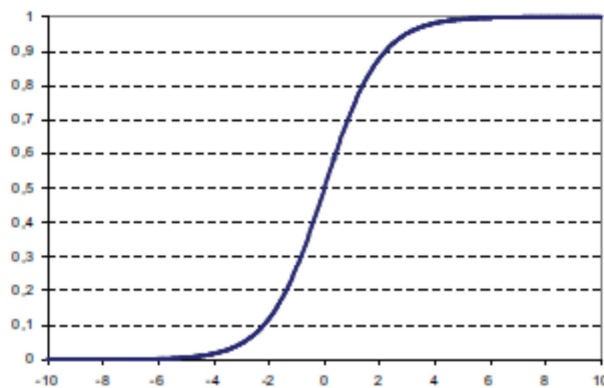


Figura 8-9. Función de Activación Sigmoide o Logística.

Función de Activación Hiperbólica

Esta función de activación viene denotada por la siguiente función de activación:

$$F(y_i) = \frac{1 - e^{-2y_i}}{1 + e^{-2y_i}} \quad (6)$$

La representación gráfica es la siguiente, en la cual como se puede observar el resultado obtenido con dicha función varía entre -1 y 1 (implica que la salida del estudio realizado pudiera tomar valores negativos. En el caso particular que vamos a estudiar este hecho de por sí elimina la posibilidad de utilizar dicha función de activación puesto que la salida sólo toma valores positivos)

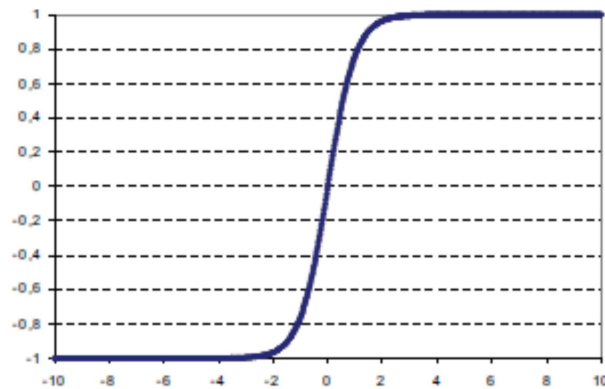


Figura 9-10. Función de Activación Hiperbólica

Nota: en todas las gráficas anteriores hemos indicado en el eje de abcisas los valores entre 10 y -10, los cuáles se han tomado como escala como una simple referencia para el gráfico pero se ha de notar que es a modo de ejemplo puesto que por norma general podrán otros valores más amplios (o no) en la escala.

Función de Activación Identidad

Notar finalmente como posible función de activación la función de identidad, puesto que a veces igualmente se utiliza. La notación simple de dicha función es la siguiente:

$$F(y_i) = y_i \quad (7)$$

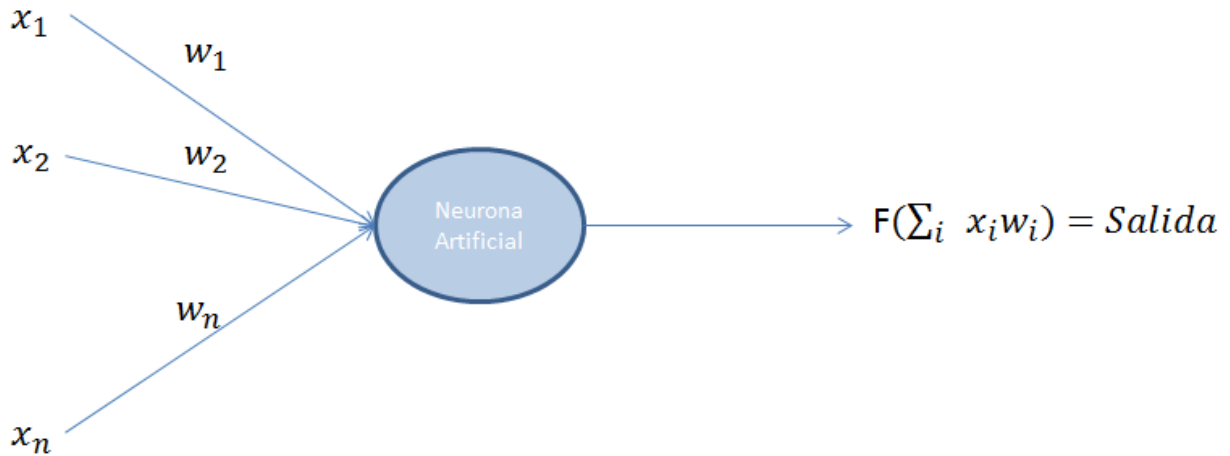


Figura 10-11. Comportamiento Básico Red Neuronal

Por tanto, conocidos los elementos básicos que componen una red neuronal (neuronas, conexiones y función de activación) estamos en disposición de componerlos de modo que podamos construir redes neuronales artificiales. Dentro de dicha composición y por seguir un orden lógico vamos a ver una primera clasificación de las redes en función del número de capas y una segunda clasificación en función del tipo de entrenamiento. Con todo ello estaríamos en disposición de realizar una clasificación de las principales redes existentes y base de muchos estudios y aplicaciones.

2.1.3 Tipos de Redes en función del número de capas.

La primera clasificación que podemos realizar de la Redes Neuronales es por el número de capas que componen la propia red, por tanto la clasificación realizada es:

- RNA monocapas.
- RNA multicapas.

Una red imita el comportamiento del cerebro humano, por tanto la combinación de neuronas y capas es lo que aporta el enorme potencial de computación de las redes al existir múltiples conexiones entre todas ellas. Es por ello que quizás no tenga mucho sentido el trabajar con una red de una sola capa, sin embargo es conveniente explicarlas puesto que las redes multicapas son composiciones de diferentes capas o redes monocapas.

El siguiente gráfico muestra una red neuronal artificial de una sola capa:

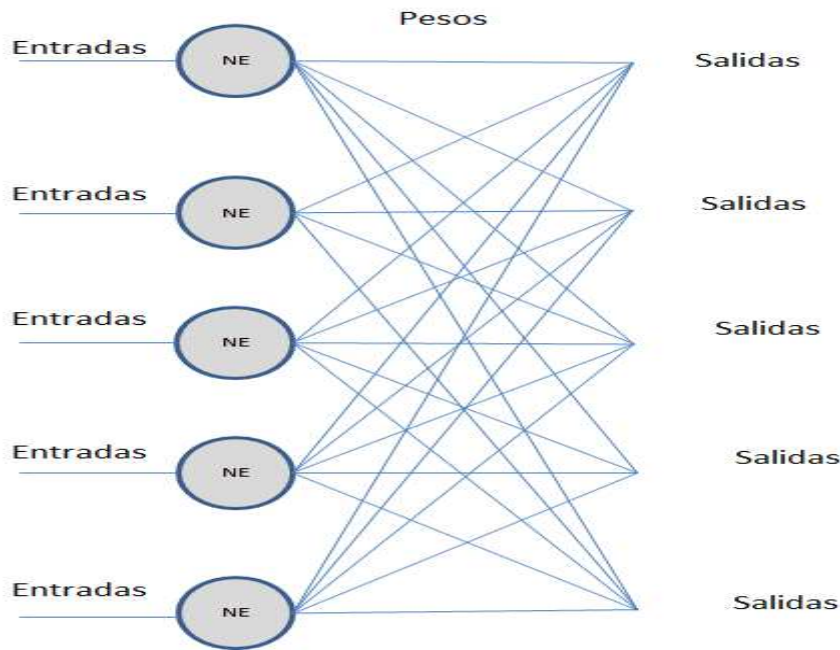


Figura 11-12. RNA Monocapa

Ejemplos de redes neuronales artificiales de una sólo capa son la perceptron y la adaline.

El funcionamiento de una red neuronal de una sólo capa es realmente parecido al que hemos explicado con una sola neurona. En este caso la capa está compuesta de un conjunto de neuronas, las cuales reciben una variable de entrada o estímulo; estas entradas se conectan con la salida a través del peso, de modo que la combinación lineal de entradas por sus correspondientes pesos a la salida es activada con la función de transferencia o activación obteniendo directamente la salida. El ejemplo al ser una generalización combina todas las neuronas de la capa con la salida, aunque pueden no existir algunas de ellas, simplemente se exponen todas de modo teórico o genérico.

En cuando al las Redes Neuronales Artificiales multicapas, no son más que una combinación de diferentes capas, teniendo un comportamiento similar al de las redes monocapas. En este caso la salida de una capa es la entrada de la capa siguiente, de modo que la información se va propagando entre las diferentes capas hasta obtener la salida final. Las redes multicapas lógicamente son mucho más potentes que las monocapas debido justamente a la propagación de la información entre las diferentes capas que aportan cualidades que no tienen las monocapas. Hoy en días éstas son las redes que se utilizan más por su aplicación, como ejemplo de algunas de ellas son la Perceptron Multicapa.

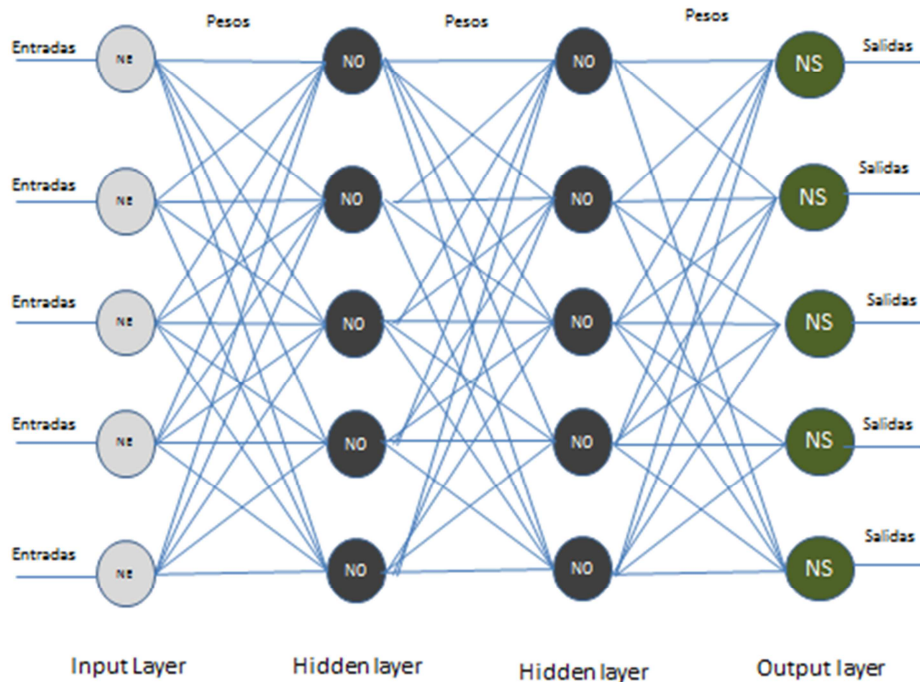


Figura 12-13 RNA Multicapa.

Finalmente indicar que existen otras clasificaciones de las redes en cuanto a su topología, sin embargo nos hemos centrado en la principal clasificación, puesto que para conocer la procedencia de la red que utilizaremos en nuestro caso de estudio es suficiente.

2.1.4 Tipos de Redes en función del entrenamiento.

Otra clasificación interesante de resaltar es en función del tipo de entrenamiento o aprendizaje, Al igual que en el apartado anterior, existen clasificaciones más amplias o complejas que incluyen redes híbridas, aprendizajes reforzados, etc. Sin embargo no vamos a estudiarlas en detalle puesto que lo que pretendemos es aportar que es una forma de clasificación (la red que propondré para el estudio será con Entrenamiento Supervisado).

La clasificación de redes que indicamos en el presente trabajo, en función del entrenamiento es la siguiente:

- Entrenamiento Supervisado.
- Entrenamiento no Supervisado.

Con una red Neuronal pretendemos simular ciertos comportamientos o patrones para lo que entrenamos la red. Para ello, introducimos las variables de entrada en la red y obtenemos una determinada salida, la cuál es la que pretendemos simular u obtener de modo que se parezca lo máximo posible a la salida real. Por tanto se pretende minimizar la diferencia entre el valor obtenido con la red y el valor real (similar a lo que se pretende conseguir con la regresión). Nuestro objetivo es intentar minimizar esas diferencias de modo que la predicción o modelo que obtengamos con la Red Neuronal sea lo más parecido posible con los valores reales, de ahí que el objetivo del entrenamiento sea justamente el de enseñar a la red para que se aproxime lo máximo posible a la realidad.

El entrenamiento Supervisado lo que hace es comparar el valor de la salida real con el valor obtenido para ajustar los pesos de la Red de modo que con los sucesivos entrenamientos vayamos ajustando las diferencias entre dichos valores para que se haga lo más pequeña posible. Dicha diferencia podría ser válida en un determinado umbral o diferencias entre esos errores, de modo que podemos incluso acotarlos.

En el Entrenamiento Supervisado por tanto existen algoritmos que pretenden minimizar las diferencias entre valor real y valor obtenido. Para ello se aplican de modo cíclico parejas de valores para el entrenamiento utilizando las diferencias obtenidas para ajustar los pesos, de ahí que también sean conocidos como coeficientes de aprendizaje. Aquí podemos observar por tanto la característica de la capacidad de aprendizaje de la red.

Como ejemplo, tenemos la Red Bacpropagation que utiliza Entrenamiento Supervisado.

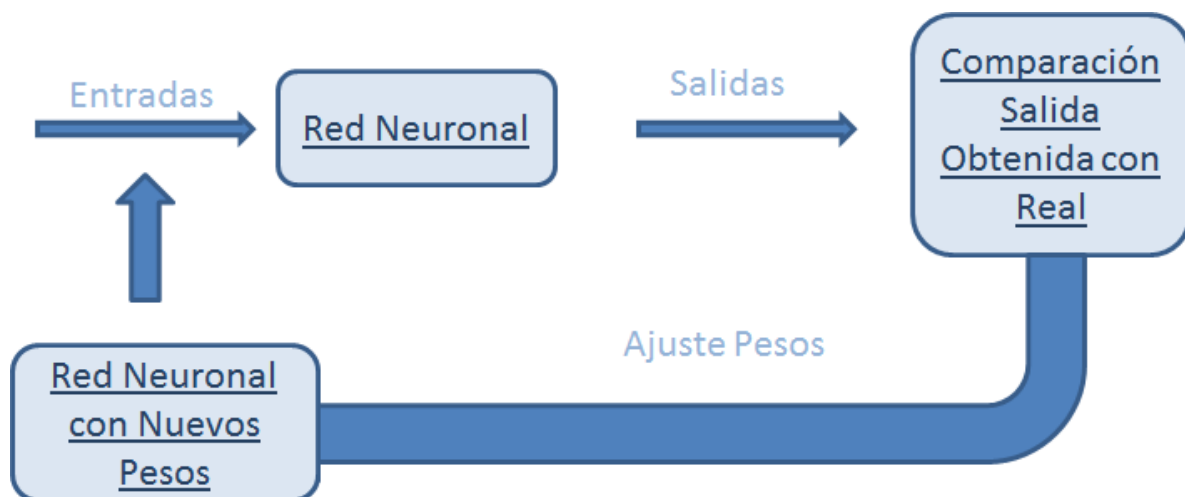


Figura 13-14. RNA con Entrenamiento Supervisado

Entrenamiento No Supervisado: en este caso sólo tenemos valores de entrada para el entrenamiento obteniendo finalmente agrupaciones de valores. Aunque quizás el entrenamiento supervisado es más razonable tiene sus críticas puesto que los fenómenos biológicos de la naturaleza no siguen la lógica anterior.

Como ejemplo tenemos la Self Organizing Mpas (SOM).

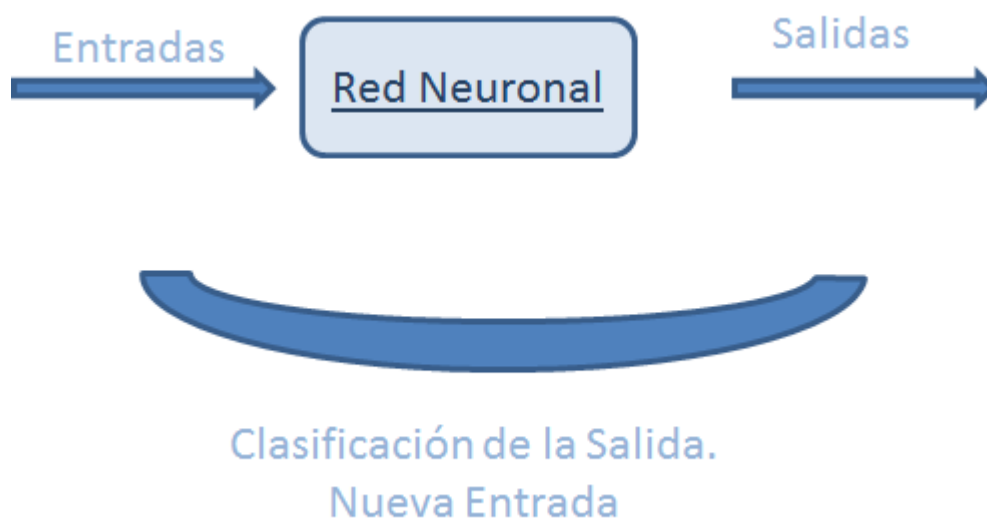


Figura 14-15. RNA con Entrenamiento NO Supervisado

Conocidos los pilares básicos o procedencia histórica de las Redes Neuronales Artificiales, conocidos los elementos que componen toda red y conocidos los tipos de redes en función de su número de capas y el tipo de entrenamiento, estamos ya en disposición de realizar una clasificación de las Redes más conocidas que existen en la actualidad.

2.1.5 Clasificación de Redes.

A continuación exponemos una tabla con las RNA más conocidas, clasificadas por: Nombre de la Red, Autor, Año, Grupo, Observaciones y Referencia.

<u>Nombre de la Red</u>	<u>Creador</u>	<u>Año</u>	<u>Grupo</u>	<u>Observaciones</u>
Adaline y Madaline	Bernad Widrow	1960	Predicción	Reconocimiento de Patrones
Adaptative Resonance Theory Networks (ART)	Carpenter, Grossberg	1960-1986	Conceptualización	Reconocimiento y predicción de Patrones.
Back-Propagation	Rumelhart y Parker	1985	Predicción	Predicción de patrones. Mejora la perceptron
Bi-Directional Associative Memory (BAM) Networks	Bart Kosko	1987	Asociación	Está basada en la red ART.
The Boltzman Machine	Ackley, Hinton y Sejnowski	1985	Asociación	Parecida a la Holpfield
Brain State in a Box	James Anderson	1970-86	Asociación	Es una red asociativa lineal
Cascade-Correlation Networks	Fahhman y Lebiere	1990	Asociación	Capas Ocultas en Cascada
Counter-Propagation	Hecht-Nielsen	1987	Asociación	Clasificación Adaptativa de Patrones
Delta Bar Delta (DBD) Networks	Jacobb	1988	Clasificación	Métodos Heurísticos para acelerar la convergencia
Digital Neural Networks Architecture (DNNA)	Neural Semiconductor Inc.	1990	Predicción	Implementación Hardware de la función Sigmoid

Directed Random Search Networks (DRS)	Maytas y Solis	1965-1981	Clasificación	Técnica de valores aleatorios en el ajuste de pesos.
Functional-link Networks (FLN)	Pao	1989	Clasificación	Versión de la Backpropagation
Hamming Networks	Lippman	1987	Asociación	Clasifica vectores binarios con la distancia de Hamming
Hopfield Networks	Hopfield	1982	Optimización	
Learninr Vector Quantization (LVQ) Networks	Kohonen	1988	Clasificación	
Perceptron Networks	Rosenblatt	1950	Predicción	Primer Modelo del Sistema Neuronal Artificial
Probabilistic Neural Network (PNN)	Spetcht	1988	Asociacion	Clasifica patrones con métodos estadísticos
Recirculation Networks	Hinton y McClelland	1988	Filtrado	Variante Back-Propagation
Self-organizing Maps (SOM)	Kohonen	1979-1982	Conceptualización	Aprendizaje sin supervisión
Spatio-Temporal-Pattern Recognition (SPR)	Grossberg	1960-1970	Asociación	Clasificación invariante en el tiempo y espacio

Tabla 1-1. Clasificación de las principales RNA conocidas

Una vez realizada la clasificación de las principales ARN existentes actualmente, vamos a explicar en detalle el funcionamiento de la red seleccionada para el estudio con un ejemplo práctico de su utilización que nos haga entender como la red mejora con el entrenamiento. Previamente justificamos su elección.

2.1.6 Justificación del Modelo de Red Neuronal Artificial Seleccionada.

Tal y como hemos expresado en el presente documento, en los últimos años se ha despertado mucho interés en la comunidad científica en los modelos predictivos con el objeto de poder conocer con antelación producciones de diversos procesos industriales. La idea va enfocada a poder aplicar técnicas futuribles en el mantenimiento para poder aportar mayor fiabilidad en el uso de los activos.

En concreto, por su auge y alto coste se ha prestado especial atención en las instalaciones de generación de energías limpias, y más en concreto en este trabajo nos vamos a centrar en una instalación fotovoltaica. Dichas instalaciones necesitan de una red de sensores distribuidos para la adquisición de series temporales de datos que deberían ser analizados estadísticamente para obtener conclusiones. Este análisis conlleva un costoso mantenimiento al alcance de muy pocas empresas por envergadura y medios.

Es justamente por ello que en el trabajo objeto de estudio pretendemos utilizar otras tecnologías que nos permitan obtener nuevos modelos de fiabilidad. Por ello presentamos un modelo basado en redes neuronales artificiales, el cual nos va a permitir disponer de un análisis actualizado del estado de los activos que nos permitirá anticiparnos al fallo pudiendo automatizar dicho modelo en el Scada ya implementado. Con avisos generados por nuestro “Modelo Ideal de Producción “ obtenido a través de la RNA y los datos históricos de la Instalación, vamos a poder generar alarmas dinámicamente y a tiempo real que eviten una posible degradación del activo. Nuestro modelo va a estimar la producción de un inversor de potencia, el cual nos permitirá detectar degradaciones tempranas que desencadenarán en actividades de mantenimiento.

Las energías renovables presentan una alta dependencia de las condiciones climatológicas que lógicamente son aleatorias y por tanto no predecibles a medio y largo plazo. Ello supone que existe una gran variabilidad en la producción y por tanto en el cumplimiento de los objetivos de producción para alcanzar las cuotas necesarias de producción para amortizar la inversión.

En cuanto a la previsión de la producción debido a las condiciones ambientales, se pueden realizar diferentes estimaciones:

- Teniendo en cuenta la previsión climatológica: se realizan con modelos físicos, con curvas de potencia de irradiación solar de modo que se puedan realizar predicciones en función de la irradiación.
- Sin tener en cuenta la previsión meteorológica: se obtienen utilizando modelos estadísticos basados en datos históricos, como el que presenta el Ministerio de Industria de España a través de una herramienta de predicción sin cálculos meteorológicos.

Por todo ello, hoy en día para poder implementar en algún sistema la detección de fallos en Instalaciones Fotovoltaicas es necesario disponer de un gran número de variables, como corrientes en los paneles solares, tensiones de las baterías, etc. junto con datos ambientales como radiación y temperatura (Orioli y Di Gangi 2013; Kostylev y Pavlovski 2011). Existen otras propuestas que tratan de reducir la complejidad en la detección de fallos utilizando pocas variables y análisis estadísticos complejos (Guasch et al 2003; Olivencia et al 2010).

En este Proyecto me centro en la aplicación de Redes Neuronales Artificiales para poder obtener patrones de comportamiento que nos permitan justamente aplicarlos para la anticipación de fallos y por tanto para la mejora de la producción.

Las RNAson herramientas matemáticas utilizadas para la resolución de problemas complejos del mundo real, especialmente problemas de clasificación y problemas de predicción. Las RNA tal y como hemos explicado en apartados anteriores pretenden emular comportamientos del cerebro humano heredando cualidades como la capacidad de aprender, la experiencia y la capacidad de generalizar comportamientos anteriores. Para ello se construyen redes de neuronas artificiales cuya misión es procesar la información recibida y trasladarla a través de conexiones cuando la información es activada mediante una función de transferencia, procesando de este modo la información a modo de grafo dirigido (Miller 1990).

La arquitectura de las RNA está compuesta por diferentes capas (entrada, ocultas y salida) cuya principal característica es la capacidad de procesar la información en entornos no lineales, con la capacidad de aprendizaje y generalización (Basheer y Hajmeer 2000; Zhang et al 1998). En comparación con métodos basados en modelos tradicionales, las RNA son modelos que se auto-adaptan a los patrones de comportamiento, basándose en datos históricos conocidos a través de los cuáles se pueden implementar en ordenadores, aprendiendo la red de dichos datos históricos que permitan captar datos o relaciones ocultas difíciles de obtener por otros métodos.

Por otra parte, las RNA proporcionan una fuerte tolerancia a fallos, debido al almacenamiento de información de forma redundante. Por ello, las RNA son muy adecuadas para la resolución de problemas de difícil solución con falta de suficientes datos para ello (Malcolm et al. 1999, Curry et al. 2000, Kuo 2011). En este sentido, Lapedes y Farber (1987) han demostrado que la Red Neuronal Backpropagation excede en un orden de magnitud al método lineal convencional y a los métodos polinómicos. Es por ello, mi interés en utilizar las RNA para el análisis de los datos de producción de la Instalación Fotovoltaica objeto de estudio para la consecución de los objetivos perseguidos.

La aplicación de estas técnicas de utilización de RNA en el sector de las energías renovables, y más en concretos en el sector de los sistemas fotovoltaicos han estado desarrollándose continuamente en los últimos años (Mellit y Kalogirou 2008; Kalogirou 2001):

- Meteorological data forecasting (Mellit et al. 2005; Yacef 2012).
- Meteorological data forecasting (Mellit et al. 2005; Yacef 2012).
- PV systems modelling, simulation and control (Hiyama 1997).

Existen trabajos previos a este Proyecto en los que se intenta pronosticar la producción de la energía eléctrica generada por una instalación fotovoltaica a través de RNA (Ashraf y Chandra 2004; Mellit y Shaari 2009), sin embargo en este Proyecto se describe un nuevo algoritmo para la detección temprana de fallos, lo que supone una nueva aportación al conocimiento al utilizar el modelo ideal de producción precedido con las RNA para obtener mejoras en la eficiencia energética de la instalación objeto de estudio. Dicha mejora se traducirá en el programa de mantenimiento que resulta de la aplicación de programas RCM (Reliability centered maintenance)

El Mantenimiento Centrado en la Fiabilidad (RCM) es la metodología más utilizada para estudiar los programas de mantenimientos de activos en un contexto operacional dado (Moubray 1997), la cuantificación de los riesgos (Rausand y Hoyland 2004) y la evaluación de las medidas correctivas para detectar, evitar o prevenir fallos (Campbell y Jardine 2001). Atribuibles a las condiciones de funcionamiento son las condiciones del contorno y los factores ambientales junto con los de la propia operación. Igualmente inciden en los posibles fallos los defectos propios ocultos de los activos (Crespo 2007; Pham y Wang, 1996).

El modelo de predicción que voy a presentar en el Proyecto tiene características innovadoras en comparación con trabajos anteriores existentes en la literatura. El modelo de RNA que vamos a utilizar no solo considera variables de contorno como la radiación y la temperatura exterior, sino también condiciones de ciertos activos como la temperatura interna del inversor o las horas de funcionamiento de la instalación que afectan lógicamente a todos los elementos que la componen. A través de una detección temprana de la degradación en la producción podremos detectar alguna anomalía o fallo en el funcionamiento de la instalación, pudiendo valorar cuantitativamente el riesgo o costo por dicha degradación. Es importante para prevenir, poder conocer el riesgo asociado a tener un fallo pues incluso podría incrementar considerablemente el valor de compra del equipo (Wilson, 1986),

En el caso de estudio el análisis realizado se ha llevado a cabo sobre equipos críticos para anticiparnos al fallo, lo que nos aportará un gran beneficio o valor puesto que es una forma de proteger la gran inversión realizada. Para la detección de fallos a futuro utilizo la Red Neuronal Backpropagation la cual se entrena con datos históricos conocidos y filtrados.

La Red Backpropagation es un mecanismo de aprendizaje muy popular para la solución de problemas de predicción en redes perceptrón multicapa (Wervos 1974; Rumelhart et al 1986.), en la que se requiere diferenciabilidad en la capa de salida. Requisito que se cumple al utilizar la función Sigmoide como función de activación.

La investigación realizada en el Proyecto de acuerdo con Hornik et al. (1989), trata de centrarse en el principio de funcionamiento de las redes neuronales artificiales, las cuales se basan en la experiencia a través del entrenamiento. Para ello hemos contado con datos históricos de 5 años (puesta en producción de la instalación en Septiembre del 2008).

2.1.7 Algoritmo Teórico de las Redes Neuronales Backpropagation

Las redes perceptron multicapa han sido redes muy conocidas para la solución de problemas de predicción y de clasificación. La Red Backpropagation se basa en dichas redes multicapas a las que les incorpora la propagación hacia atrás lo que supone una importante mejora lo que hace resurgir el estudio de las RNA.

Puesto que dicha Red es la que vamos a utilizar para la solución de nuestro problema, vamos a entrar en el detalle explicando la teoría de su funcionamiento así como aportando un caso práctico (3 APLICACIÓN PRACTICA CON EJEMPLO NUMÉRICO DE BACKPROPAGATION) para ver en detalle su funcionamiento y mejora con el entrenamiento.

Para la explicación del algoritmo y dada la dificultad teórica o abstracción que supone este tipo de técnicas matemáticas, vamos a realizarlo utilizando una red con un sólo capa oculta y una variable en la capa de salida. Ello no supone en absoluto pérdida de rigor matemático, sino que lo que se pretende es simplificar la notación matemática con objeto de que sea más fácilmente comprensible.

El algoritmo de Resolución de una Red Backpropagation consta de 5 pasos que se repiten de forma cíclica siendo dichos pasos (Hastie et al 2001; Rutkowski 2008) los que se denotan a continuación:

Pasos Hacia Delante:

- 1.- Selecciona un valor de Entrada desde el Conjunto de Entrenamiento.
- 2.- Aplicar dicho valor a la entrada y calcular la salida.

Pasos Hacia Atrás:

- 3.- Calcular el error entre la salida calculada en el paso 2 y la salida deseada (la salida real conocida) de dicha entrada.
- 4.- Ajustar los pesos para que el error cometido entre la salida calculada y la salida deseada sea disminuido.
- 5.- Repetir los pasos 1 a 5 para todas las entradas del conjunto de entrenamiento, hasta que el error global sea aceptablemente bajo.

En si y explicado de este modo es simple aunque conlleva gran abstracción. Lo que realizamos con dicho algoritmo es utilizar los valores conocidos de variables de entrada y su correspondiente salida de algún proceso para entrenar a la red de modo que con sucesivos entrenamientos dicha red se ajuste lo máximo posible al patrón de comportamiento del proceso en cuestión. O lo que es lo mismo, intentamos aproximar lo máximo posible el valor calculado con la red con el valor real o salida.

Esta explicación es la que desarrollamos a continuación con su correspondiente formulación matemática. En primer lugar incluimos un gráfico con una Red Neuronal a la que vamos a aplicar el algoritmo de la Red backpropagation. Dicho gráfico corresponde con el que vamos a utilizar en nuestro modelo de estudio pero que nos sirve perfectamente para explicar el algoritmo.

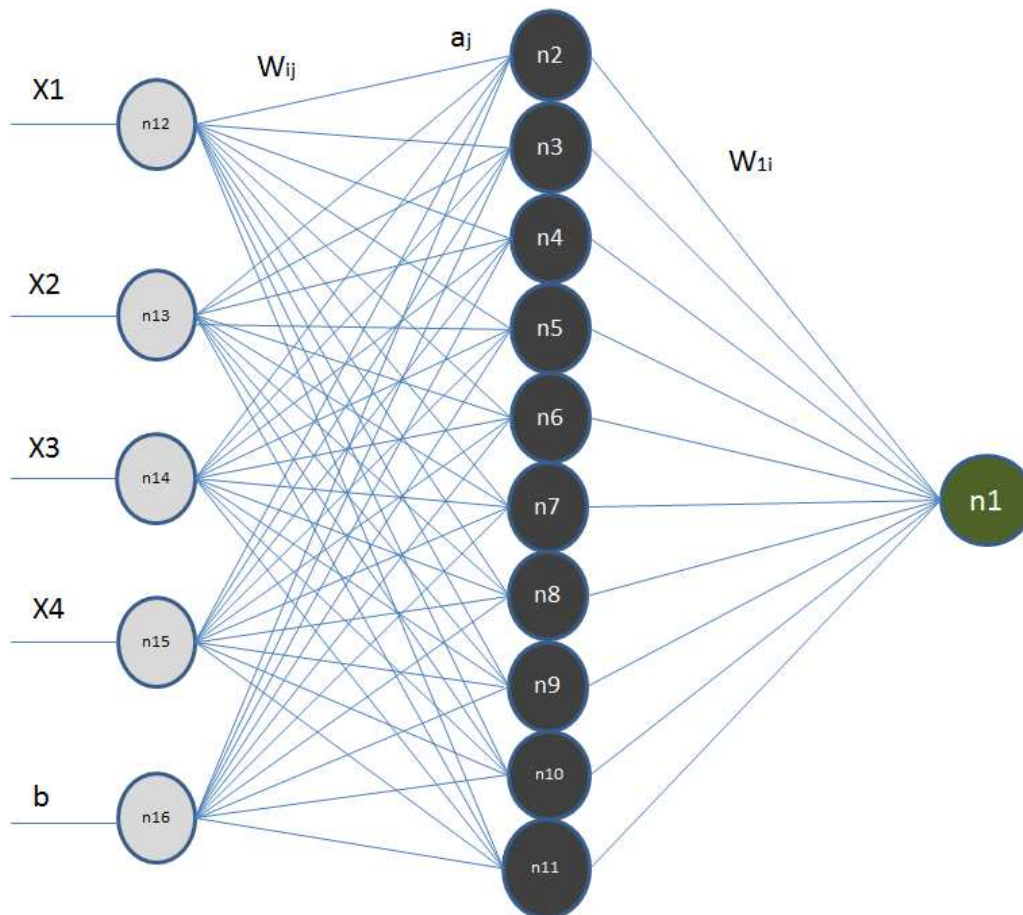


Figura 15-16. Arquitectura Red Bacpropagation

La notación a emplear es la siguiente:

x_i = valor de la variable de entrada. Son valores conocidos que se utilizan inicialmente para entrenar la red. Cada conjunto de x_i de entradas en la red se denomina un patrón de entrenamiento. Cada patrón tiene su correspondiente salida conocida, real o como se suele denominar deseada puesto que es la que se pretenda alcanzar. El subíndice i corresponde al número de variables de entrada, en el caso particular del gráfico anterior el subíndice i tomaría valores de 1 a 4. Ello implica que un patrón de entrenamiento viene determinado en este caso por 4 valores en cada entrada o patrón. De modo general $i = 1..q$

b = término independiente, ganancia o bias que se utilizar para que la red se ajuste mejor. Dicho término se podría generalizar incluso considerando que es realmente el último x_i del patrón de entrenamiento tomando el valor 1. Se suele realizar por simplicidad en la notación.

Podríamos generalizarlo utilizando la notación x_{ij} , de modo que el valor j controlara el patrón de entrenamiento. De este modo cada patrón j es un vector que contiene un valor para cada neurona de la capa de entrada, las cuales se recorren con el índice i . Como hemos indicado anteriormente no vamos a generalizar la formulación puesto que lo que se pretende es explicar el funcionamiento de la red de un modo claro para que sea aplicable a nuestro estudio sin perdernos en la propia formulación.

Un patrón de entrenamiento aplicado al gráfico anterior sería: $x_{i1} = \langle x_{11} | x_{21} | x_{31} | 1 \rangle$

Para cada patrón de entrenamiento disponemos de un valor conocido, salida real o salida deseada, la cual vamos a denotar por T_k . Aplicando el algoritmo:

Paso 1.- Selecciona un valor de Entrada desde el Conjunto de Entrenamiento.

Tomamos $x_{ij} = \langle x_{1j}|x_{2j}|x_{3j}|1 \rangle$ siendo $i= 1 \dots$ hasta el número de entrenamientos del que dispongamos o la red se haya ajustado lo suficientemente bien como para que no sea necesario seguir entrenándola. Por simplicidad en la notación utilizaremos x_i conociendo que es un vector cuyo número de componentes depende del número de células de la capa de entrada.

Paso 2.- Aplicar dicho valor a la entrada y calcular la salida.

Aplicado dicho valor obtenemos un valor, que denominamos salida calculada la cual denotamos por Z_{ij} . Para llegar a obtener este valor hemos realizado previamente diversos pasos. Conocemos a través de los x_i los valores de las neuronas de las capas de entrada, el valor final Z_k es el valor calculado sobre la última neurona o neurona de la capa de salida. Para llegar a dicho valor previamente realizamos:

Sea q el número de componentes del vector de entrada, y como variable de salida sólo existe 1. Como número de neuronas de la capa oculta (1 sólo capa oculta) sea el valor m . Con el superíndice hacemos referencia a la capa en la que nos encontramos, por lo que utilizaremos el “0” para la capa oculta y la “s” para la capa de salida. Por último indicar que con la “f” hacemos referencia a al función de transferencia o activación.

$$n_j^0 = \sum_{i=1}^q (w_{ji}^0 * x_i) + b; j = 1 \dots m; \quad (8)$$

$$a_j^0 = f^0(n_j^0); \quad (9)$$

Una vez calculados los valores aplicados sobre las entradas de las neuronas de la capa oculta y conociendo los pesos que aplican entre las neuronas de la capa oculta y la neurona de la capa de salida estamos en disposición de obtener el valor final.

Es de hacer constar que aplicamos un superíndice en la función de activación puesto que dicha función puede ser diferente la que se aplica en la capa oculta que la que se aplica en la capa de salida por lo que es por ello necesario distinguirla,

$$n_k^s = \sum_{j=1}^m (w_{kj}^s * a_j^0); k = 1 \text{ (sólo una célula en la capa oculta en el caso particular de estudio)}; \quad (10)$$

Finalmente aplicamos la función de activación a la neurona de la capa de salida y obtenemos el valor obtenido o valor calculado.

$$Z_k = f^s(n_k^s); \quad (11)$$

Con estos dos pasos hemos calculado la salida de la red con un patrón de entrenamiento. El siguiente paso es utilizar la salida deseada y compararla con la obtenida para mejorar la red, para ello propagamos hacia atrás y aplicamos:

Paso 3.- Calcular el error entre la salida calculada en el paso 2 y la salida deseada (la salida real conocida) de dicha entrada.

$$T_k - Z_k \quad (12)$$

Este es el error para un solo entrenamiento, para el cálculo del error se utiliza el error cuadrático medio cuya fórmula es la siguiente:

$$ECM = \frac{1}{n} \sum_{i=1}^n (T_k - Z_k)^2 \quad (13)$$

Para un conjunto de n entrenamientos.

Notamos por ECM error cuadrático medio, o bien utilizamos su nomenclatura en inglés MSE = Mean square error.

Paso 4.- Ajustar los pesos para que el error cometido entre la salida calculada y la salida deseada sea disminuido.

La gran aportación de esta red es justamente la propagación hacia atrás que es la que supuso el relanzamiento en el año 1985 de la Redes Neuronales Artificiales, puesto que aporta una base matemática más fuerte al ajuste de los errores pudiendo de este modo entrenar a la red para poder ajustar los valores obtenidos con los valores deseados, premisa fundamental para obtener buenos resultados.

En cuanto al ajuste de los pesos debemos hacer una distinción, entre ajuste de pesos de la capa oculta y ajuste de pesos de la capa de salida.

Paso 4.1 Ajuste de los pesos de la capa de salida.

Los nuevos pesos de la capa oculta vienen determinados por la siguiente fórmula:

$$\Delta w_{kj}^s = \delta_k * Z_k \quad (14)$$

Siendo:

$$\delta_k = Z_k * (1 - Z_k) (T_k - Z_k) \quad (15)$$

Paso 4.2 Ajuste de los pesos de la capa oculta.

Los nuevos pesos de la capa oculta vienen determinados por la siguiente fórmula:

$$\Delta w_{ji}^0 = \delta_k * Z_k \quad (16)$$

Siendo en este caso el δ_k :

$$\delta_k = Z_k * (1 - Z_k) (\sum_k \delta_k w_{kj}^0) \quad (17)$$

Paso 5.- Repetir los pasos 1 a 5 para todas las entradas del conjunto de entrenamiento, hasta que el error global sea aceptablemente bajo.

$$ECM = \frac{1}{n} \sum_{i=1}^n (T_k - Z_k)^2 \quad (18)$$

2.1.8 Algoritmo para Minimizar el ECM

Cómo hemos explicado en todos los apartados anteriores y visto en concreto en el algoritmo de la Red Backpropagation, con la propagación hacia atrás intentamos ajustar los pesos de la red, de modo que vayamos aproximando lo máximo posible el valor deseado con el valor obtenido. Ello nos da una idea de hasta cuando debemos entrenar o iterar la red o definir el umbral válido para las diferencias en media de los valores obtenidos con los valores deseados.

Por tanto, esto nos indica que debemos intentar minimizar la función del error cuadrático medio para encontrar la solución óptima al problema que nos permita encontrar los pesos óptimos de la red que mejor se ajuste al comportamiento del proceso en estudio. Es decir, nos encontramos con el siguiente problema de optimización cuya función objetivo es la siguiente:

$$Min \frac{1}{n} \sum_{i=1}^n (T_k - Z_k)^2 \quad (19)$$

La solución ideal es aquella que nos aporta el mínimo global de la función objetivo, sin embargo en la mayoría de los problemas complejos con los que nos podemos encontrar la función de activación que calcula los valores obtenidos o Z_k es la función logística o la función hiperbólica cuyas funciones vienen definidas por las siguientes fórmulas:

$$F(y_i) = \frac{1}{e^{-y_i} + 1} \quad (20)$$

$$F(y_i) = \frac{1 - e^{-2y_i}}{1 + e^{-2y_i}} \quad (21)$$

Esto nos indica claramente que nos encontramos con un problema de optimización NO lineal cuya solución óptima es realmente compleja de obtener. De hecho, no existe un algoritmo que nos de la solución óptima del problema o mínimo global de la función.

Es por ello que debemos utilizar algoritmos heurísticos que nos aporten una buena solución al problema pero sin tener garantías de ser la mejor de ellas. Entre los algoritmos utilizados para la solución de dicho problema los más conocidos son los algoritmos de Cuasi-Newton y Levenberg-Marquart, sin poder indicar cuál de ellos nos da la mejor solución al ser ambos heurísticos. Por tanto la decisión de utilizar uno u otro depende del software utilizado, en concreto el software más conocido es Matlab el cuál tiene implementado el algoritmo de Levenberg-Marquart. En nuestro caso particular el software utilizado para la resolución del problema de estudio es R, y el motivo no es otro que el hecho de ser software libre. Dicho software utiliza como algoritmo de resolución el método de cuasi-newton que es aplicado a través de la función nnet. Como conclusión para la resolución del problema utilizamos:

- Software Libre: “R”
- Algoritmo: Cuasi – Newton al que le aplicaremos una penalización.
- Función de Activación: Sigmoide o logística.

Hemos indicado que utilizamos función de error con penalización, implica por tanto que al minimizar el Error Cuadrático Medio le aplicamos o sumamos una penalización, que viene determinada por la siguiente fórmula:

$$ECM = \frac{1}{n} \sum_{i=1}^n (T_k - Z_k)^2 \quad (22)$$

$$ECM + Penalización = \frac{1}{n} \sum_{i=1}^n (T_k - Z_k)^2 + \gamma \sum_{i=1}^M w_i^2 \quad (23)$$

El motivo por el que utilizaremos en la resolución del problema esta penalización es por el hecho de evitar los denominados sobreajustes. Esto ocurre cuando estamos intentando modelar o ajustar algún patrón de comportamiento con pocos datos, lo que puede hacer la red es que memorice el patrón y por tanto pierda capacidad de generalización.

Por ejemplo el siguiente gráfico contiene muy pocos datos, lo que implica que si quisiéramos modelar el comportamiento a través de una red neuronal, dicha red seguro que lo memorizaría, de ahí que tenga sentido el incorporar este tipo de penalizaciones.

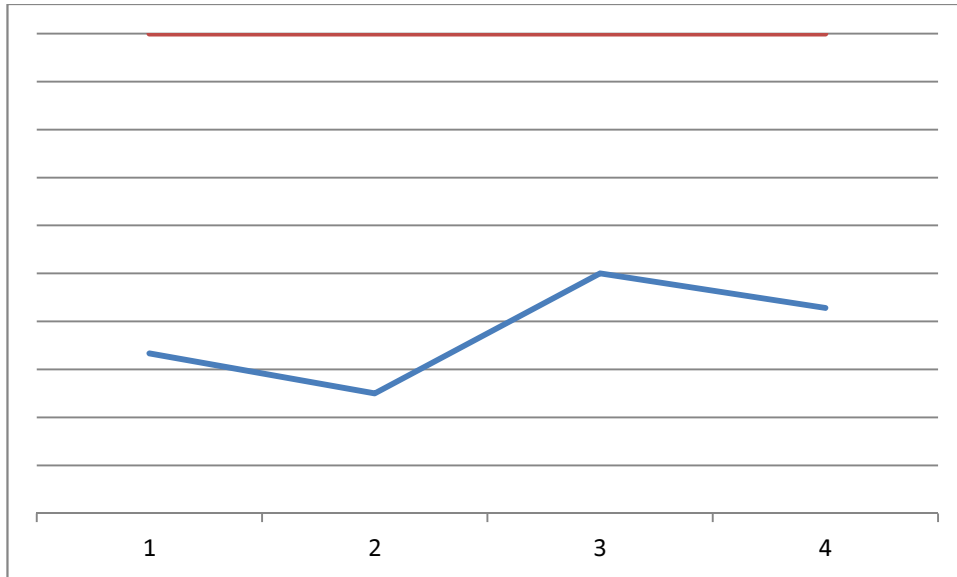


Figura 16-17. Gráfico de Patrón de Comportamiento con pocos valores.

Sin embargo, en casos es los que disponemos de muchos datos para el entrenamiento y son patrones complicados de modelar lo lógico es que tras aplicar el algoritmo la solución nos indique que el factor de penalización es 0, sin embargo y por seguridad no está de más el incluirlo para evitar dichos sobreajustes.

Una vez explicada la teoría de la Red Neuronal Backpropagation con el algoritmo de resolución así como los algoritmos para minimizar el ECM y que por tanto nos den la condición de parada del algoritmo estamos en disposición de resolver el caso de estudio que vamos a considerar en el presente trabajo.

Sin embargo, y tras explicar el algoritmo no deja de tener cierta abstracción y complejidad su comportamiento por lo que antes de proceder a resolver el caso de estudio con el software, variables y datos muchos más complejos vamos a aportar un ejemplo numérico que nos permita comprender en detalle el comportamiento del funcionamiento y entrenamiento de una red neuronal explicándolo paso a paso.

A parte nos sirve dicho ejemplo para comprender en cierto modo la capacidad de modelar muchos procesos industriales, lo que nos da idea de la magnitud de problemas que se podrían afrontar con un buen planteamiento y estudio de base.

3 APLICACIÓN PRÁCTICA CON EJEMPLO NUMÉRICO DE BACPROPAGATION

3.1 Ejemplo Numérico de Aplicación de la Red Neuronal Backpropagation

Disponemos de un centro de transformación con una potencia de salida de 1000 KVA que puede dar servicio a un conjunto residencial ubicado en la provincia de Sevilla en el que residen 200 familias. Es importante conocer las condiciones de contorno en las que opera dicho centro de transformación puesto que una parada por avería en dicho centro dejaría sin luz a 200 hogares con la consiguiente penalización para la compañía eléctrica. Los datos históricos que disponemos de dicho centro de transformación son los siguientes:

MTTF (x5000)	Mantenimiento Preventivo	Condiciones de Operación (KVAx100)	Condiciones del Entorno (°Cx10)
0,2	0	8	6
0,4	1	7	2
0,3	1	9	5

Tabla 1-2. Datos del Centro de Transformación

Siendo MTTF = tiempo medio hasta el fallo.

En cuanto al mantenimiento preventivo consideramos:

0: no se ha realizado mantenimiento preventivo antes de producirse la avería.

1: si se ha realizado mantenimiento preventivo antes de producirse la avería.

Se desea poder predecir el tiempo medio hasta el fallo de la posible siguiente avería, con objeto de informar a la compañía eléctrica para que realice un mantenimiento preventivo anterior que evite dicha parada por avería. De este modo conseguiríamos evitar gastos innecesarios por paradas e igualmente gastos innecesarios en otros mantenimientos anteriores si no son necesarios realizarlos.

La red neuronal que vamos a construir tendrá la siguiente topología de red:

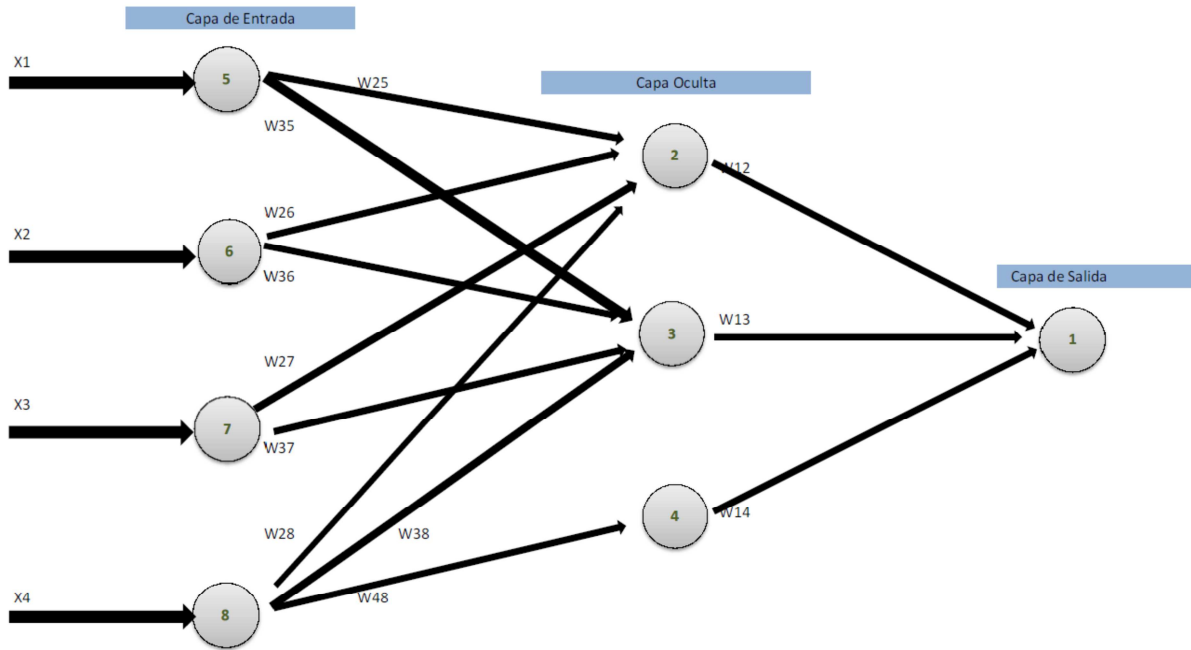


Figura 1-18. Topología de Red para Comprensión de las RNA con ejemplo numérico.

Disponemos por tanto de una capa de entrada, una capa oculta y una capa de salida.

En la capa de entrada las variables:

- X1: variable booleana que toma los valores 0 o 1. El valor 1 indica que se ha realizado un mantenimiento preventivo al Centro de Transformación en el periodo de funcionamiento normal del equipo previo al fallo. El valor 0 por tanto indica que no se le ha realizado mantenimiento alguno.
- X2: variable real que indica la potencia en KVA a la que está trabajando el Centro de Transformación. Dicha variable al tomar valores altos la utilizamos dividida por 100 para simplificar el cálculo e intentar que la red aproxime en menos entrenamientos al valor deseado.
- X3: variable real que indica las condiciones del contorno en la que está trabajando el Centro de Transformación, en este caso temperatura en °C. Al igual que con la variable anterior dividimos los valores entre 10.
- Z(i): son los valores calculados de la neurona i. Puesto que la neurona de salida es la 1, el valor de Z(1) será el valor calculado final que compararemos con el valor deseado.
- T(i): valor deseado conocido a priori y que son los que nos sirven para entrenar a la red y ajustar los pesos. En nuestro caso este valor deseado es el MTTF(i): tiempo medio hasta el fallo. Dichos valores los normalizamos entre 0 y 1 dividiendo el valor real entre 5000, puesto que la función de activación utilizada es una función que toma cualquier valor real y nos da como resultado un valor entre 0 y 1.

- Sigmoide: es la función de activación que utilizamos para el cálculo del valor deseado $Z(i)$
- $W(i,j)$: peso de la salida i y la neurona j .

El objetivo que pretendemos por tanto es entrenar la red neuronal con una serie de valores históricos conocidos para poder predecir futuras averías en el Centro de Transformación. Conocida dicha predicción (incluso con un margen de error aceptado que podemos asumir) debemos planificar un mantenimiento preventivo del equipo con cierto margen de error sobre el valor calculado que evite paradas innecesarias por averías con el consiguiente ahorro de costes. Igualmente evitamos realizar mantenimientos preventivos innecesarios.

Observar que una vez realizados los cálculos y la posterior predicción todos los datos que inicialmente hemos normalizado debemos volver a realizar la transformación para llevarlo a su valor original. Este es un error habitual que no debemos obviar pues nos puede llevar a hacer predicciones totalmente erróneas.

Para comprender la utilización del algoritmo y al eficacia del comportamiento y mejora de la red neuronal vamos a explicar la utilización del algoritmo siguiéndolo paso a paso tal y como se ha descrito en el apartado anterior.

*PASOS**HACIA**DELANTE*

Paso 1: Seleccionamos un vector de entrada desde el conjunto de entrenamiento.

Valor Deseado				
T	X1	X2	X3	X4
MTTF	Mantenimiento Preventivo	Condiciones de Operación (KVAx100)	Condiciones del Entorno ($^{\circ}\text{C} \times 10$)	Término Independiente
0,2	0	8	6	1

Tabla 2-3. Conjunto de valores de entrada para entrenamiento.

Los pesos de la red a inicio no son conocidos, de hecho dichos pesos son los que se irán modificando en el algoritmo formando parte por tanto del aprendizaje a través de la experiencia. Por tanto es habitual inicializar como norma todos los pesos a 1, tal y como lo hacemos en el ejemplo y en el caso de estudio que abordaremos en detalle más adelante.

Los pesos iniciales considerados son todos iguales a 1:

Pesos / Valor Inicial					
w(1,2)	1	w(2,5)	1	w(4,8)	1
w(1,3)	1	w(3,5)	1		
w(1,4)	1	w(2,6)	1		
		w(3,6)	1		
		w(2,7)	1		
		w(3,7)	1		
		w(2,8)	1		
		w(3,8)	1		

Tabla 3-4. Inicialización de los valores de los pesos. Todos a 1.

Paso 2: Aplicamos esta entrada a la red y calculamos la salida.

Neurona 2:

$$N(2) = W(2,5) \cdot X(1) + W(2,6) \cdot X(2) + W(2,7) \cdot X(3) + W(2,8) \cdot X(4) \cdot$$

$$N(2) = 15$$

Neurona 3:

$$N(3) = W(3,5) \cdot X(1) + W(3,6) \cdot X(2) + W(3,7) \cdot X(3) + W(3,8) \cdot X(4)$$

$$N(3) = 15$$

Con estos valores calculamos la salida de dichas neuronas, es decir, obtenemos la salida de las neuronas de la capa oculta, utilizando la función de activación. La función de activación utilizada es la Sigmoide por la propia similitud del caso que vamos a abordar, cuya fórmula como recordatorio es la siguiente:

$$F(y_i) = \frac{1}{e^{-y_i} + 1} \quad (20)$$

Los valores de salida de las neuronas de la capa oculta tras la aplicación de la función de activación son los siguientes:

$$Z(2) = 0.9999969$$

$$Z(3) = 0.9999969$$

$$Z(4) = 1 \text{ (valor del término independiente)}$$

Utilizamos estos valores con sus correspondientes pesos para calcular finalmente el valor de salida de la neurona 1.

Neurona (1):

$$N(1) = W(1,2) \cdot Z(2) + W(1,3) \cdot Z(3) + W(1,4) \cdot Z(4)$$

$$N(1) = 2.9999939$$

Finalmente aplicamos la función de activación Sigmoide a la neurona 1 para obtener el valor calculado para el primer vector del conjunto de entrenamiento. Obtenemos

$$\underline{Z(1) = 0.9525741}$$

Observar que en ambas capas hemos utilizado la función de activación logística o Sigmoide, sin embargo no es obligatorio que sea la misma tal y como explicamos en el algoritmo. De hecho en el caso de estudio utilizamos la Sigmoide en la capa oculta y la identidad en la capa de salida.

PASOS HACIA ATRÁS

Una vez obtenido el valor calculado de la salida de la neurona 1, utilizamos el valor deseado conocido para obtener el error cometido y de este modo corregir los pesos de la red. Por tanto a cada entrenamiento obtenemos una mejora de los pesos que nos aproxima poco a poco el valor calculado con el valor deseado.

Paso 3: Calcular el error entre la salida calculada y la salida deseada de la entrada usada. La fórmula utilizada para el cálculo del ECM es la siguiente:

$$ECM = \frac{1}{n} \sum_{i=1}^n (T_k - Z_k)^2 \quad (22)$$

Obtenemos como resultado $E(1) = (0.2 - 0.9525741)^2$.

$$\underline{E(1) = 0.56}$$

Paso 4. Ajustar los pesos para que el error cometido entre la salida calculada y la salida deseada sea disminuido.

Para ello distinguimos entre la capa de salida y la capa oculta.

Para la capa de salida utilizamos la siguiente fórmula para conocer el incremento de los pesos:

$$\Delta w_{kj}^s = \delta_k * Z_k \quad (14)$$

Siendo:

$$\delta_k = Z_k * (1 - Z_k) (T_k - Z_k) \quad (15)$$

En nuestro caso la neurona de salida es la neurona 1, por tanto realizamos los cálculos.

Delta (1)	-0,033998803
$\Delta W(1,2)$	-0,033998792
$\Delta W(1,3)$	-0,033998792
$\Delta W(1,4)$	-0,033998803

Tabla 4-5. Incremento para ajuste de los pesos de la capa de salida.

Para la capa oculta utilizamos las siguientes fórmulas

$$\Delta w_{ji}^0 = \delta_k * Z_k \quad (16)$$

Siendo en este caso el δ_k :

$$\delta_k = Z_k * (1 - Z_k) (\sum_k \delta_k w_{kj}^0) \quad (17)$$

Al igual que en la neurona de salida consideramos gamma igual a 1. Realizando cálculos obtenemos los siguientes resultados sobre las neuronas de la capa oculta, en nuestro ejemplo las neuronas 2 y 3.

Delta (2)	-1,04003E-08
Delta (3)	-1,04003E-08
$\Delta W(2,5)$	0
$\Delta W(3,5)$	0
$\Delta W(2,6)$	0
$\Delta W(3,6)$	0
$\Delta W(2,7)$	0
$\Delta W(3,7)$	0
$\Delta W(2,8)$	-1,04003E-08
$\Delta W(3,8)$	-1,04003E-08

Tabla 5-6. Incremento para ajuste de los pesos de la capa oculta.

Con el incremento de todos los pesos obtenemos finalmente los nuevos pesos los cuales nos servirán para ajustar mejor el valor obtenido al valor deseado.

Los pesos nuevos son los siguientes:

w(1,2)	0,96600121
w(1,3)	0,96600121
w(1,4)	0,9660012
w(2,5)	1
w(3,5)	1
w(2,6)	1
w(3,6)	1
w(2,7)	1
w(3,7)	1
w(2,8)	0,99999999
w(3,8)	0,99999999

Tabla 6-7. Valores de los nuevos pesos tras primer patrón de entrenamiento.

Paso 5. Repetir los pasos 1 a 5 para todas las entradas del conjunto de entrenamiento, hasta que el error global sea aceptablemente bajo.

Con la serie de datos iniciales:

MTTF (x5000)	Mantenimiento Preventivo	Condiciones de Operación (KV Ax100)	Condiciones del Entorno (°Cx10)
0,2	0	8	6
0,4	1	7	2
0,3	1	9	5

Tabla 7-8. Serie de Valores para el entrenamiento a realizar como ejemplo.

Podemos realizar tres entrenamientos para esta serie, por tanto realizando todos los pasos anteriores en cada uno de ellos exponemos a modo de resumen la siguiente tabla que nos aporta los valores de los pesos y su correspondiente error.

Pesos	Entrenamiento 1	Entrenamiento 2	Entrenamiento 3
w(1,2)	1	0,966001208	0,938875378
w(1,3)	1	0,966001208	0,938875378
w(1,4)	1	0,966001197	0,938874915
w(2,5)	1	1	1
w(3,5)	1	1	1
w(2,6)	1	1	1
w(3,6)	1	1	1
w(2,7)	1	1	1
w(3,7)	1	1	1
w(2,8)	1	0,99999999	0,999999535
w(3,8)	1	0,99999999	0,999999535

Tabla 8-9. Tabla Resumen valores de los pesos tras los correspondientes entrenamientos.

	Entrenamiento 1	Entrenamiento 2	Entrenamiento 3
Error Entrenamiento	0,5663	0,433	0,426
Salida Deseada	0,952574099	0,947746062	0,943567649
Salida Calculada	0,2	0,4	0,3

Tabla 9-10. Mejoras en los errores tras los Entrenamientos.

Como hemos indicado finalmente tendríamos que volver a deshacer la normalización inicial de valores para conocer el valor real obtenido. Tras el entrenamiento observamos que el error disminuye lo que hace mejorar el comportamiento de la red.

Puesto que el objetivo de la red es conseguir la mejora o acercamiento a la salida real con los nuevos pesos podríamos repetir el cálculo de la salida en cualquiera de los entrenamientos anteriores y observaríamos como la salida deseada está ahora algo más próxima que en el caso inicial con los pesos iguales a 1.

En todo caso no insistimos es realizar más cálculos ni aportar más valores puesto que lo que se pretende en este apartado es dar un ejemplo numérico para poder entender el ejemplo del algoritmo así como la mejora de la red con el entrenamiento.

4 DESARROLLO. OBJETIVO GENERAL

4.1 Caso de Estudio

EL grupo de empresas Magtel al cual pertenezco, es propietario entre otras de una Instalación Fotovoltaica de 6,1Mw de potencia nominal ubicada en el término municipal de Fuente Palmera, provincia de Córdoba. Dicha instalación se encuentra a pleno rendimiento desde Septiembre del año 2008, lo que implica que disponemos de suficiente información y datos recogidos para poder tener capacidad de analizarlos. Esta es la base fundamental para poder realizar el estudio objeto del presente trabajo. Toda la información recogida queda registrada en el Scada de la Instalación Fotovoltaica, del que hemos obtenido todos los datos necesarios de las variables analizadas.



Figura 1-19. Imagen Aérea Instalación Fotovoltaica objeto de Estudio.

La Instalación Fotovoltaica tal y como hemos indicado tiene una potencia nominal de 6,1 Mw que se encuentran repartidos en huertos solares de 100 Kw. La planta dispone de un total de 37.180 paneles fotovoltaicos de diferentes potencias configurados en función de cada uno de dichos huertos.

Cada uno de estos huertos solares está configurado en torno a un inversor, el cual convierte la corriente continua en alterna para finalmente poder verter la energía generada en un centro de transformación que en este caso ya sería propiedad de la compañía eléctrica y es en el que se mide la energía generada con contadores por cada uno de los huertos solares. Un centro de transformación recoge la energía de un máximo de 5 inversores, siendo la configuración estándar de uno de esos centros de transformación la que se describe a continuación:

CT	ID	kWn	Ref. Módulo	Nº Total Módulos	Nº String	Nº Módulos x String
CT15	A8-1	100	IS-220	528	44	12
	A8-2	100	IS-220	528	44	12
	A8-3	100	IS-220	528	44	12
	A8-4	100	IS-220	528	44	12
	A8-5	100	IS-220	528	44	12

Tabla 1-11. Configuración Estándar de un Centro de Transformación de la Instalación Fotovoltaica.

Cada uno de estos huertos fotovoltaicos genera anualmente (datos del año 2012) una media de 210.000 Kwh/año según radiación solar de la zona en la que se encuentra y considerando la degradación de dichas instalaciones. El precio de venta de un Kwh es de 0.4886 € (como dato de referencia para el estudio pues la prima depende del año), lo que supone una generación económica anual de cada uno de estos huertos de 100 Kw de 102.606 €, lo que multiplicado por 61 instalaciones de esta Planta en concreto supone una generación en venta de energía anual de 6.258.966 €.

Es por todo ello que se hace primordial el disponer de herramientas que nos permitan anticiparnos a los fallos para evitar paradas innecesarias por el importante coste económico que supondría. En este sentido cobra especial importancia un correcto mantenimiento de la planta, mantenimiento al que se le deben aplicar nuevas técnicas que permitan anticiparnos, utilizando para ello técnicas predictivas o futuribles. Ya no sólo es importante mantener y reparar en corto espacio de tiempo, sino que evitar paradas con la anticipación al fallo puede suponer una gran ganancia económica. Ello justifica el hecho de realizar una considerable inversión y esfuerzo en este tipo de estudios que permitan implantar modelos matemáticos avanzados.

4.2 Metodología

Existen diversas metodologías para predecir, en el presente trabajo hemos optado por utilizar los modelos de Redes Neuronales Artificiales al ser modelos que se adaptan muy bien al comportamiento de la producción de la Instalación (tal y como se ha justificado en el apartado 2.1.6)

Las Redes Neuronales Artificiales se adaptan bien a ciertos patrones de comportamiento que no son modelables por otras técnicas matemáticas debido a la variación existente entre las variables de entrada y las variables de salida. Además son modelos adaptables en función del entrenamiento que hace que la red mejore y se adapte mucho mejor, y en nuestro caso particular muy aplicable como veremos en los resultados obtenidos, entre otros porque tenemos un importante histórico de datos filtrados que permite a la red disponer de una importante entrada de valores para que esté bien entrenada.

Dentro del conjunto de RNA existentes y tal y como se ha descrito en apartados anteriores la red elegida es la Red Backpropagation debido a que es la red más adecuada para la predicción de valores en función de datos históricos conocidos, no siendo más que un mecanismo de aprendizaje de una Red Perceptron Multicapa (Wervos 1974; Rumelhart et al 1986.).

Una vez seleccionada la red el método para la minimización del Error Cuadrático Medio es el método de Cuasi_Newton al que le incluiremos un sumando de penalización para evitar el sobreajuste

$$\text{ECM} + \text{Penalización} = \frac{1}{n} \sum_{i=1}^n (T_k - Z_k)^2 + \gamma \sum_{i=1}^M w_i^2 \quad (23)$$

Se dice que existe sobreajuste en la red cuando el ajuste es tan bueno que perdemos capacidad de generalización. Este fenómeno ocurre habitualmente cuando estamos intentando simular un patrón de comportamiento con muy pocos valores, por lo que la red lo que intenta es ajustar tanto que se aprende de memoria el patrón repitiendo los valores en las circunstancias futuras análogas. De ahí el perder generalización y el incluir el factor de penalización.

En el caso de que el conjunto de datos del que dispongamos para entrenar la red sea lo suficientemente grande y la variabilidad entre los resultados o salidas sea lo suficientemente importante, lo lógico es que el factor de penalización sea cero a la hora de resolver el problema puesto que dicho patrón no es posible memorizarlo al existir disparidad de resultados. En el caso particular de estudio es lo que sucederá como veremos más adelante.

Existen varios métodos para la minimización del ECM, ninguno de ellos obtiene con garantías el mínimo global por lo que dentro de los más reconocidos el motivo por el cual he seleccionado este método no es otro que el que utiliza el software elegido para la resolución del algoritmo. Es decir, utilizamos el método de Cuasi-Newton con factor de penalización y software R.

Con la conocida y explicada Red Neuronal Backpropagation y el algoritmo de minimización del error cuadrático medio de cuasi-newton al que aplicaremos en el software R un coeficiente de penalización para evitar los sobreajustes, vamos a entrenar la red en ausencia de fallos obteniendo un modelo ideal de producción. Dicho modelo Ideal es el Objetivo General que pretendíamos alcanzar en el presente estudio, modelo que utilizaremos en la siguiente fase con objeto de mejorar la eficiencia en la producción de la planta, como el principal de los objetivos específicos que pretendemos alcanzar.

El modelo ideal de producción vendrá determinado por la Red Neuronal Artificial, la cual contendrá exactamente un número de células concreto en la capa oculta así como conoceremos los pesos de todas las conexiones existentes entre todas las neuronas de las diferentes capas que configurarán la red. Dicho modelo quedará almacenado para posteriormente predecir valores de producción en función de los valores de entrada que le demos a la red. La aportación más significativa a este trabajo es justamente la de la utilización de este modelo para mejorar la producción de la planta, lo que abordaremos en posteriores apartados.

Expuesta la metodología para la solución del problema planteado, a continuación aportamos los datos considerados para el entrenamiento de la red:

- Datos de producción horaria desde las 8:00 h hasta las 17:00 h. Es decir, por cada variable de entrada en la red disponemos de 10 valores diarios. Se ha considerado dicho horario con objeto de disponer de una producción importante en todas las horas independientemente de la época del año. No tendría sentido por ejemplo considerar las 20:00 h puesto que lo lógico es que en el periodo estival la producción sea 0. EL esfuerzo del proyecto va encaminado en optimizar la producción, esfuerzo muy importante que no sería acorde para la mejora de un número de menor de Kw.
- Datos de producción de Junio del 2011 a Julio de 2012. Es decir, tratamos 395 días para disponer de un ciclo completo de producción repitiendo el primer mes del siguiente año. Lo que supone que para el entrenamiento de la red disponemos de 3950 valores por cada variable de entrada, valor suficientemente importante para que la red esté muy bien entrenada. Nos aporta garantía a la hora de obtener el modelo ideal de producción, aunque el manejo y tratamiento de datos es muy considerable y pesado de inicio.
- De dichos datos consideramos el 75% de la red para entrenamiento y el 25% para testeo
- Aún obteniendo muy buenos resultados en los valores de la red como veremos, para ratificar aún más los resultados y dado que conocemos los valores de producción hasta hoy en día, hemos utilizado los meses de Julio y Agosto del 2012 para testear el resultado haciendo una predicción de valores con el modelo ideal de producción y comparándolos con los valores reales de producción del huerto solar considerado en dichas fechas.

- El periodo para la comparación debido al conocimiento de la existencia del fallo de estudio, ha sido Septiembre y Octubre del 2012. De ahí que el entrenamiento se haya realizado en fechas previas para que la red esté entrenada según el modelo más adaptado o próximo a la fecha de estudio. Lo ideal de la red es tenerla entrenada del modo más continuo posible, algo que con la implementación creada es muy sencillo de realizar.
- Los datos seleccionados son en un comportamiento normal de la planta y por tanto en ausencia de fallos. Debido a que han existido fallos en el período considerado, se ha realizado un filtrado previo de todos los datos de estudio. Caso de existir un fallo se ha sustituido por un valor en condiciones normales considerando la media de producción entre los dos días anteriores y los dos días posteriores a dicha producción.
- Para la obtención de los valores de las temperaturas se obtienen cuatro valores cada hora, considerando en el estudio la media horaria de dichos valores.
- La escala para la temperatura son grados Kelvin. Se ha realizado el entrenamiento y comparación con grados centígrados obteniendo peores resultados. Al ser valores mayores existe mayor diferencia y por tanto mayor sensibilidad detectada en los cambios.
- La red igualmente se ha entrenado utilizando la exponencial de las temperaturas por la propia sensibilidad de la exponencial, sin embargo son valores excesivamente altos que no aportan mejora alguna al modelo.
- Los datos para el entrenamiento han sido normalizados previamente tal y como se describe en detalle apartado posterior (4.3.3).
- El modelo presentado para la planta de estudio es generalizable para cualquier otra planta de generación de energía fotovoltaica.

El siguiente gráfico denota de forma esquemática la metodología utilizada:

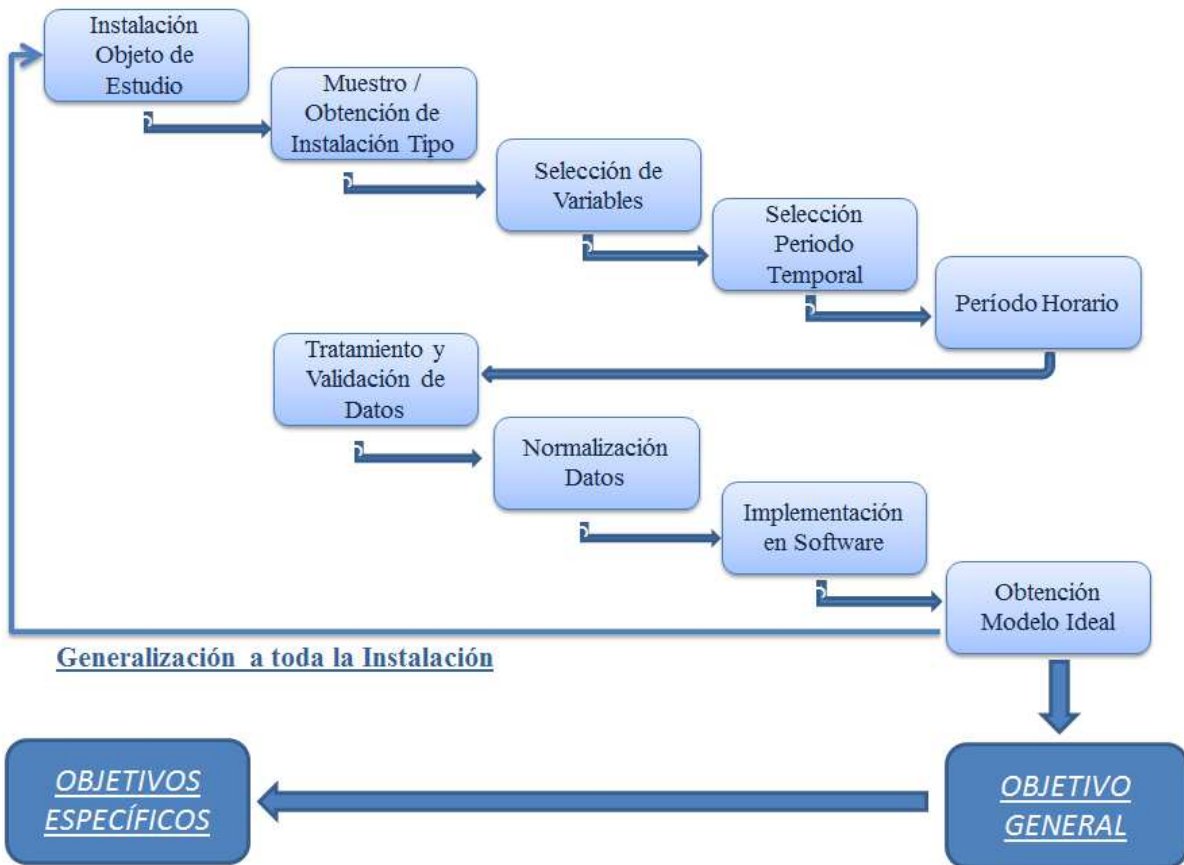


Figura 2-20. Metodología Utilizada en el Estudio

Una vez expuesta la metodología a utilizar y las variables y datos necesarios para la obtención del Modelo Ideal de Producción, a continuación exponemos el detalle.

4.3 Planteamiento

La empresa propietaria de la Instalación Fotovoltaica que es objeto del estudio, no sólo mantiene su propia instalación, sino que además se dedica al mantenimiento de Instalaciones Fotovoltaicas de otras propiedades ubicadas en distintas zonas geográficas de España. En el propio mantenimiento se garantiza una producción mínima pues de lo contrario se penalizaría sobre el propio contrato, lo que hace ser muy consciente de la importancia del correcto mantenimiento de una Instalación Fotovoltaica.

Conocer en detalle de dichos mantenimientos se plantea la resolución del presente problema con objeto de poder mejorar la eficiencia en la producción en cualquier Instalación Fotovoltaica aplicando métodos matemáticos avanzados, con cierto coste en su estudio pero que dadas las grandes inversiones que se realizan sin duda merece la pena invertir en ello.

La Instalación objeto de estudio tal y como se ha indicado, está en producción desde Septiembre del 2008 obteniendo buenos resultados en su rendimiento. Sin embargo el hecho de ser instalaciones con un largo periodo de vida requieren de una importante amortización y por tanto un conocimiento de la caja que se va a generar justifica la necesidad de disponer de algún método que pueda predecir la producción en un periodo de al menos un año. Forman parte de los activos de las empresas y por tanto de sus cuentas y resultados, de ahí la necesidad de poder proyectar dichos activos al menos vista un año renovable en el tiempo.

Para poder predecir hemos justificado el uso de las Redes Neuronales Artificiales que con el histórico de datos manejado nos aportará muy buenos resultados. Una de las grandes ventajas de las Redes Neuronales una vez implantado el modelo de predicción, es que es fácilmente moldeable adaptando la red de un modo fácil y económico en función de los datos que iremos obteniendo de producción lo que hace que la red pueda estar constantemente actualizada en el tiempo. De ahí la necesidad de incorporar como variable de entrada el número de horas de funcionamiento, puesto que nos aporta mayor sensibilidad ya que con dicha variable estamos considerando las pérdidas de producción debido a la degradación de los componentes con el tiempo de funcionamiento.

La instalación se encuentra ubicada en una parcela de 30 hectáreas en el término municipal de Fuente Palmera, Córdoba. Dicha instalación está compuesta como hemos indicado, por 61 huertos solares de 100 kW cada uno de ellos. Para el estudio se ha realizado un muestreo previo de la producción mensual de cada uno de esos huertos solares en los años 2009 y 2010, obteniendo producciones muy similares en todos ellos mes a mes. Al tener todas configuraciones similares y estar ubicados en la misma zona geográfica no existen diferencias significativas en la producción, por lo que el estudio realizado se centra en una de dichas instalaciones de 100 Kw pudiendo ser generalizada al resto de instalaciones.

En instalaciones menores no es aconsejable realizar el estudio por la variabilidad existente en configuraciones y paneles así como por el coste que supondría.

La relación inversión – coste del estudio debe ser positiva para realizarlo, por lo que la instalación de 100 Kw es la óptima para ello disponiendo de datos de calidad y pudiendo ser fácilmente generalizable.

La siguiente tabla muestra datos de configuración de la instalación, la cual está distribuida en 15 centros de transformación:

CT	ID	kWn	Ref. Módulo	Nº Total Módulos	Nº String	Nº Módulos x String
CT1	A1-1	100	YL170Wp	616	22	28
	A1-2	100	YL170Wp	616	22	28
	A1-3	100	YL170Wp	616	22	28
	A1-4	100	IBC-180M	616	22	28
CT2	A2-5	100	IBC-180M	616	22	28
	A2-6	100	IBC-180M	616	22	28
	A2-7	100	IBC-180M	616	22	28
	A2-8	100	TSM-175	660	44	15
CT3	A2-1	100	IS-170	660	44	15
	A2-2	100	IS-170	660	44	15
	A2-3	100	IS-170	660	44	15
	A2-4	100	IS-170	660	44	15
CT4	A7-5	100	TSM-170	210	14	15
			TSM-175	450	30	15
	A7-6	100	TSM-175	660	44	15
	A7-7	100	TSM-175	660	44	15
A7-8	100	BP 3170	660	44	15	
CT5	A4-1	100	IS-170	660	44	15
	A4-2	100	IS-160	660	44	15
	A4-3	100	IS-160	660	44	15
	A4-4	100	IS-170	660	44	15
CT6	A3-1	100	TSM-220	484	22	22
	A3-2	100	TSM-220	484	22	22
	A3-3	100	IS-170	660	44	15
	A3-4	100	IS-170	660	44	15
CT7	A7-1	100	BP 3170	660	44	15
	A7-2	100	BP 3170	660	44	15
	A7-3	100	TSM-170	660	44	15
	A7-4	100	TSM-170	660	44	15
CT8	A4-7	100	IS-160	390	26	15
			IS-170	270	18	15
	A4-8	100	IS-170	660	44	15
	A4-5	100	TSM-170	660	44	15
	A4-6	100	TSM-170	660	44	15
CT9	A1-5	100	IBC-225TE	484	22	22
	A1-6	100	IBC-225TE	484	22	22
	A1-7	100	IBC-225TE	484	22	22
	A1-8	100	IBC-225TE	484	22	22
CT10	A3-5	100	IS-160	660	44	15
	A3-6	100	IS-160	660	44	15
	A3-7	100	IS-160	660	44	15
	A3-8	100	IS-170	660	44	15
CT11	A6-1	100	IS-170	660	44	15
	A6-2	100	IS-170	660	44	15
	A6-3	100	ISF-200	528	22	24
	A6-4	100	ISF-200	528	22	24
CT12	A6-5	100	AIDE-180	660	44	15
	A6-6	100	AIDE-180	660	44	15
	A6-7	100	AIDE-180	660	44	15
	A6-8	100	IS-220	528	44	12

CT	ID	kWn	Ref. Módulo	Nº Total Módulos	Nº String	Nº Módulos x String
CT13	A5-1	100	IS-220	528	44	12
	A5-2	100	IS-220	528	44	12
	A5-3	100	IS-220	528	44	12
	A5-4	100	IS-220	528	44	12
CT14	A5-5	100	IS-220	528	44	12
	A5-6	100	IS-170	660	44	15
	A5-7	100	IS-170	660	44	15
	A5-8	100	IS-170	660	44	15
CT15	A8-1	100	IS-220	528	44	12
	A8-2	100	IS-220	528	44	12
	A8-3	100	IS-220	528	44	12
	A8-4	100	IS-220	528	44	12
	A8-5	100	IS-220	528	44	12

Tabla 2-12. Configuración de los 15 Centros de Transformación de la Instalación Fotovoltaica.

Al verificar que los datos de producción de todos ellos son similares mes a mes, consideramos para el estudio un inversor en particular el cuál es posteriormente generalizable puesto que el comportamiento como hemos indicado es muy parecido en todos los inversores. La zona geográfica en la que están ubicados todos es la misma, no existiendo diferencias entre las variables que vamos a considerar en los diferentes huertos solares.

CT	ID	kWn	Ref. Módulo	Nº Total Módulos	Nº String	Nº Módulos x String
CT15	A8-1	100	IS-220	528	44	12
	A8-2	100	IS-220	528	44	12
	A8-3	100	IS-220	528	44	12
	A8-4	100	IS-220	528	44	12
	A8-5	100	IS-220	528	44	12

Tabla 3-13. Configuración del Centro de Transformación CT objeto de estudio seleccionado.

El inversor seleccionado para el estudio es el inversor 5 del centro de transformación 15, cuya nomenclatura es el A8-5 tal y como explicamos en el apartado siguiente. La tabla anterior vuelve a mostrar la configuración de dicho inversor.

4.3.1 Selección de Variables

En este apartado comenzamos con el primer punto a considerar para la creación del modelo que nos permita entrenar a la red Neuronal y obtener el Modelo Ideal de Producción como Objetivo General perseguido.

Una vez seleccionado el modelo de Red Neuronal, el paso lógico es modelar la topología para lo que necesitamos elegir las variables que van a intervenir en el modelo. Este punto inicial es primordial, por lo que es importante prestar especial atención a la elección de las variables que intervienen puesto que una mala elección nos llevará al traste todo el trabajo realizado.

Por la definición propia de las Redes Neuronales Artificiales existen variables que interfieren en la Capa de Entrada y variables que interfieren en la Capa de Salida, el resto son cálculos intermedios que realiza la propia red a través de las neuronas de las capas ocultas. Es obvio por tanto que debemos tener muy claras las variables que inciden en dichas Capas.

En cuanto a la variable de la Capa de Salida es evidente, puesto que estamos modelando la predicción de Producción de Energía, por lo que la variable de Salida no es otra que la producción.

La notación que vamos a utilizar para dicha variable de salida será la siguiente:

- Y_i : Predicción de la Producción de Energía obtenida con el modelo de Red Neuronal.
- i : fecha de la producción en formato horario.

La producción considerada del inversor 5 del Centro de Transformación 15.

Mayor dificultad tiene el elegir las variables de la Capa de Entrada, que son aquellas variables que inciden directamente en la producción de la Instalación Fotovoltaica. En un modelo de Red Neuronal es importante la sencillez y simplicidad del modelo, puesto que un modelo sencillo explica de un modo mucho más ajustado la predicción del modelo ajustándose mejor al patrón de comportamiento. Obviamente tiene un menor coste de computación.

Si embargo, no por ello debemos dejar de considerar variables que incidan en su comportamiento. Esta explicación justifica que este punto es clave en el éxito del modelo, de ahí el dedicar mucho tiempo y esfuerzo a realizar pruebas y consultar con expertos en Mantenimiento en este tipo de instalaciones.

En cuanto a la selección de la muestra en la planta para el estudio, disponemos de 61 contadores de lectura de producción, cuya notación es la siguiente:

A_{ij} : $i= 1..8$, $j= 1..8$. Los 7 primeros i tienen sus correspondientes 8 j lo que hace 56 contadores y el último i , es decir, el 8 sólo tiene 5 j haciendo el total de los 61 contadores.

En cuanto a los sensores instalados cuya información queda recogida en el Scada, se ubicaron en los Huertos Solares correspondientes a los contadores A1.2, A7.3 y A8.5. Dichos sensores instalados en puntos distantes de la Planta para poder verificar si la distancia entre ellos aportaba diferencias en la producción.

Tras verificar los datos de producción en el periodo de estudio se observa que no existen diferencias significativas, por lo que los datos de estudio obtenido de uno de ellos es una muestra suficientemente buena para poder generalizar el resultado del estudio a toda la planta. En nuestro caso el estudio lo vamos a realizar sobre el huerto denotado A8-5 cuya energía generada es vertida al Centro de Transformación denotado con el número 15.

El siguiente gráfico denota el punto de estudio en el que nos encontramos:



Figura 3-21. Metodología Utilizada en el Estudio. Selección de Variables

En el Huerto Solar objeto del inversor de estudio y con la instrumentación establecida se han medido los siguientes parámetros:

- Radiación Global Horizontal: medimos con esta variable la radiación en superficie horizontal sobre toda la extensión del panel, cuya medida viene dada como irradiancia en superficie. Consideramos por tanto dicha radiación como un valor representativo de la radiación en toda la planta, que dado el propio valor de la variable y las ubicaciones consideradas es un valor generalizable de modo totalmente razonable.
- Temperatura Ambiente: es el valor de la temperatura ambiente considerada para toda la planta, El valor tomado por tanto es representativo y generalizable a toda la Instalación.
- Temperatura en el Panel: esta temperatura es muy poco representativa ya que varía de un panel a otro y el rendimiento de los paneles depende de dicha temperatura siendo diferente el comportamiento para diferentes paneles incluso del mismo fabricante y potencia. Esta variable queda por tanto descartada.
- Temperatura Interior del Inversor: esta variable incide considerablemente en el comportamiento del inversor, siendo éste el que recoge la energía de un Huerto Solar de 100 Kw. Es por ello que es otra de las variables primordiales a considerar.
- Horas de funcionamiento: el rendimiento de la instalación depende de las horas de funcionamiento que llevan acumulados los diferentes componentes, existiendo cierta degradación a lo largo del tiempo. Es importante considerar una variable que acumule dichas horas para que la propia red detecte la bajada de producción en el tiempo debido a la acumulación de horas. Si no tuviésemos en consideración dicha variable la red podría detectar caídas en la producción que nos hiciera pensar en algún fallo, siendo realmente la producción normal consecuencia de la antigüedad propia de la Instalación Fotovoltaica.

Como conclusión, las variables seleccionadas y con las que trabajaremos en todo el estudio son las siguientes:

- RADIACIÓN GLOBAL HORIZONTAL
- TEMPERATURA AMBIENTE
- TEMPERATURA INTERIOR DEL INVERSOR
- HORAS DE FUNCIONAMIENTO (acumulamos horas en el período de estudio, es decir, de 8:00 h a 17:00)

El siguiente gráfico denota las variables seleccionadas, las cuáles influenciadas por las horas de funcionamiento de la instalación y tras aplicarlas a la Red Neuronal Artificial nos aporta la predicción de la

Produccion:

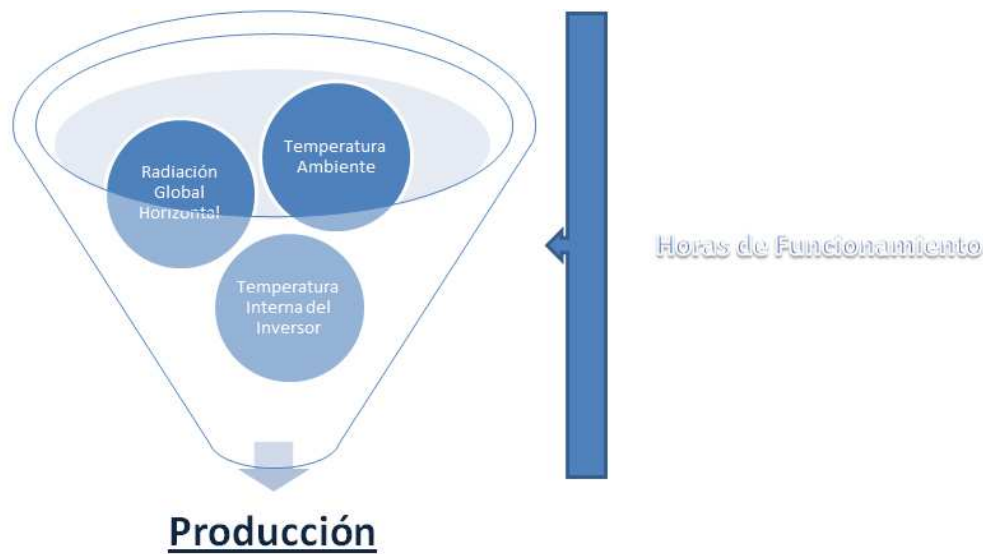


Figura 4-22. Metodología Utilizada en el Estudio. Selección de Variables: Horas de Funcionamiento.

Son variables asepticas en el sentido de que no dependen del comportamiento de un elemento particular, sino que son variables generales lo que nos permite obtener comportamientos generales.

En todo caso y previo a la selección de variables definidas se han realizado pruebas y entrenamientos con distintos conjuntos de variables, llegando a la conclusión de que las seleccionadas son las que se amoldan mejor al comportamiento de producción de la planta.

Otro componente a considerar en la elección de las variables es su valor, es decir, la escala considerada para cada una de las variables elegidas. Aunque en el siguiente apartado veremos diferentes técnicas de normalización de valores que nos permitirá obtener resultados más rápidos con el entrenamiento de la red, dicha normalización depende lógicamente de los valores iniciales a considerar. En general, indicar que valores muy próximos en una determinada variable dificultará obtener conclusiones ante un cambio debido a que son pequeñas las variaciones en sus valores. Sin embargo valores excesivamente grandes serán dificultosos para trabajar y tampoco darán buenos resultados pues pueden llegar a distorsionar las conclusiones al obtener valores alejados que nos den lugar a considerar anomalías realmente no existentes (se han realizado entrenamientos considerando la exponencial de los valores obtenidos inicialmente para intentar obtener mayor sensibilidad en los cambios, sin embargo los valores son excesivamente grandes no teniendo buenos resultados con la normalización por lo que se ha desechado). Como conclusión es importante elegir escalas con diferencias entre los valores que aporten sensibilidad a los cambios en su justa medida.

La escala de medida para cada una de las variables seleccionadas ha sido la siguiente:

- Temperatura Ambiente: grados Kelvin. Al existir una mayor variabilidad en grados Kelvin que en grados centígrados consideramos esta escala por ser más sensible a los cambios lo que nos permitirá más fácilmente detectar desviaciones. Al tener los valores de esta variable en el Scada en grados centígrados, los transformamos en grados Kelvin sumando a dichos valores 273.15 (0°C equivalen por tanto a $273,15^{\circ} \text{K}$). Los valores que disponemos en el estudio oscilan en el intervalo [273 , 333]
- Temperatura Interior del inversor: por analogía con la anterior utilizamos la misma escala. Los valores que disponemos en el estudio oscilan en el intervalo [290 , 330]
- Radiación Global Horizontal: es la magnitud utilizada para medir la potencia que incide en la superficie del panel. Medida en W/m^2 , es decir vatio/metro cuadrado. Esta medida es la que utilizamos para el estudio, los valores que disponemos en el estudio oscilan en el intervalo [10 , 1291]

- Horas de Funcionamiento: es un número entero secuencial que contabiliza el número de horas de producción de la planta considerando sólo el horario de estudio, es decir las 10 horas diarias desde las 8:00 hasta las 17:00. El valor con el que iniciamos es de 9900 horas (considerando el mismo razonamiento y fecha de inicio Septiembre del 2008) correspondiente a las 8:00 del día 1 de Junio del 2011 y finalizando con 14990 horas de funcionamiento correspondiente a las 15:00 horas del 30 de Junio del 2012.

La variable de Salida es la producción del Huerto Solar medido en Kw/h obteniendo como valores reales en dicho periodo valores en el intervalo [1,99]. Notar que no existe el valor 0 puesto que el modelo considerado es un modelo en ausencia de fallos, por lo que en horario de producción siempre se ha generado electricidad aunque sea mínima.

A continuación exponemos una tabla con las variables consideradas para el estudio, algunas de ellas con su valor inicial sin manipular, como por ejemplo la temperatura interior del inversor inicialmente con valores en °C. Podemos observar claramente que para el estudio estamos ya utilizando los valores en ciertas escalas y filtrados por un inversor concreto.

Fecha	Hora	Inversor	Temp Ext	Radiacion	Temp Int °C	H. Funcionamiento	Producción
1-6-11	8:00	5- CT 15	303	490	40,10	9900	52
1-6-11	9:00	5- CT 15	313	756	39,00	9901	74
1-6-11	10:00	5- CT 15	319	861	42,70	9902	80
1-6-11	11:00	5- CT 15	323	902	41,90	9903	82
1-6-11	12:00	5- CT 15	325	918	43,50	9904	83
1-6-11	13:00	5- CT 15	327	924	44,50	9905	81
1-6-11	14:00	5- CT 15	327	923	45,20	9906	83
1-6-11	15:00	5- CT 15	327	929	46,20	9907	83
1-6-11	16:00	5- CT 15	326	916	43,10	9908	82
1-6-11	17:00	5- CT 15	325	890	42,40	9909	81
7-6-11	8:00	5- CT 15	293	250	37,60	9960	25
7-6-11	9:00	5- CT 15	306	609	37,50	9961	61
7-6-11	10:00	5- CT 15	313	777	39,70	9962	73
7-6-11	11:00	5- CT 15	318	930	39,00	9963	87
7-6-11	12:00	5- CT 15	314	626	40,00	9964	79
7-6-11	13:00	5- CT 15	320	881	41,40	9965	76
7-6-11	14:00	5- CT 15	319	999	39,20	9966	88
7-6-11	15:00	5- CT 15	314	825	41,00	9967	75
7-6-11	16:00	5- CT 15	318	975	41,40	9968	78
7-6-11	17:00	5- CT 15	319	966	40,90	9969	80

Tabla 4-14. Tabla con valores ejemplos utilizados para el estudio sin manipular.

Junto con todas las variables consideradas, se incluye la producción puesto que todos estos datos iniciales son utilizados para entrenar a la red.

Una vez seleccionadas las variables para el estudio así como todos los datos de cada una de ellas, el siguiente paso sería el filtrar y tratar dichos datos, puesto que es fundamental validarlos para poder obtener resultados fiables.

4.3.2 Tratamiento de los datos.

Esta fase del Proyecto es fundamental, puesto que el objetivo es disponer de un conjunto de datos de calidad pues son la base del resultado que vayamos a obtener y posteriormente implementar. Errores en dichos datos supondrá obtener resultados erróneos o incoherentes lo que supondría desechar el trabajo al no obtener conclusiones válidas,

Es por ello que esta fase es primordial ya que los datos son esenciales, tienen gran valor para la I+D+i. EL problema es que el trabajo es muy tedioso, conlleva tiempo y dedicación y posteriormente se deben mantener con la misma calidad para que las predicciones futuras sean válidas. Como contraprestación positiva, la validez de los datos nos permitirá obtener conclusiones que aporten beneficios (desde el punto de vista de la mejora de la eficiencia de la producción) y serán la base de posibles futuros estudios para la mejora de la Instalación en cualquier sentido.

Magtel dispone de un sistema de adquisición de datos que se compone de dos partes. Por una parte la instrumentación necesaria para la obtención de los datos y por otra el almacenamiento de éstos. Ambas partes deben mantenerse de modo que deben ser coherentes, pues de lo contrario perdemos fiabilidad.

El objetivo por tanto de esta fase de proyecto se centra en dos aspectos fundamentales:

- El filtrado de la Calidad de los Datos.
- Homogeneizar la base de datos: periodo, muestra y referencia temporal.

Para homogeneizar los datos:

- Período de Obtención de Datos: Junio 2011 – Julio 2012
- Muestra: Inversor 5 del Centro de Transformación 15.
- Referencia Temporal: datos horarios desde las 8:00 h hasta las 17:00.

En cuanto al filtrado de la Calidad de los datos generamos un algoritmo para validar los datos obtenidos. Dicho algoritmo consiste es:

- Obtener los datos de todas las variables de Entrada.
- Filtrar los datos por el Inversor Seleccionado.
- Ordenación de los datos por fecha y hora.
- Limitar las fechas al periodo de estudio.
- Filtrar los Datos según el horario deseado para homogeneizar.
- Ordenar los datos por variables de Estudio.
- Para cada variable de estudio homogeneizar las escalas.

- Por cada variable de estudio establecer umbrales.
- Si un valor supera el umbral máximo o mínimo establecer una alarma.
- Si no lo supera el valor es validado.

- Si hay alarma. Es poco frecuente un valor anormal, aunque ello no implica que sea válido, Para ello se consulta con el operador de mantenimiento o experto. Entonces sucede:
 - o Nos indica que es correcto: se valida.
 - o Nos indica que no es correcto, está mal medido. Tomamos como valor de referencia la media entre el valor de dicha variable en esa hora de los dos días anteriores y los dos días posteriores.

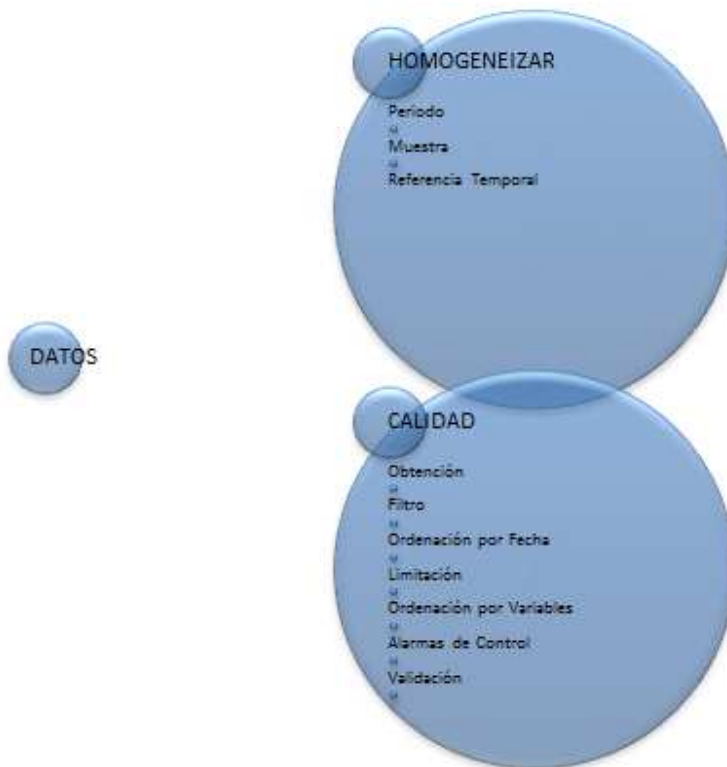


Figura 5-23. Metodología Utilizada en el Estudio. Tratamiento y Validación de los Datos.

El gráfico anterior muestra la metodología utilizada para el Tratamiento y Validación de los datos.

Una vez ordenados y validados los datos acordes a las variables seleccionadas, el siguiente paso es utilizar algún software para entrenar la Red y obtener el Modelo Ideal de Producción. Para realizarlo, es conveniente de modo previo, normalizar los datos puesto que se converge mucho más rápido hacia el resultado obteniendo por tanto beneficios en tiempo y en costes de computación.

4.3.3 Normalización de los Datos.

Existen diversas técnicas de normalización de datos, una de ellas es la que vamos a utilizar. De este modo los datos se concentran en un rango de valores entre 0 y 1 que hacen que su tratamiento sea más cómodo y rápido. La normalización de los datos implica que una vez obtenidos los resultados volvamos a eliminar dicha normalización, algo que a veces no se realiza cometiendo importantes errores.

No nos vamos a extender en las diferentes técnicas existentes para normalizar, simplemente indicar algunas de las que existen compartiendo de modo general el siguiente patrón para entrenar la Red:

- 1.- Normalizar los valores para el Entrenamiento.
- 2.- Normalizar los valores reales o conocidos de la salida de modo que nos permita comparar en la misma escala, el valor conocido o deseado con el valor obtenido con la Red Neuronal.
- 3.- Utilizar los datos normalizados para entrenar la red.
- 4.- Eliminar la normalización para la obtención de valores como el Error Cuadrático Medio o las predicciones realizadas con la red entrenada.

Es posible utilizar diferentes técnicas de normalización de modo que utilicemos una para los valores del entrenamiento y otra para los valores reales o conocidos de la salida.

Algunas técnicas de normalización de datos son las siguientes:

- Normalización Estadística o Z-score Normalization
- Normalización MIN – MAX.
- Normalización Sigmoidea.

Las técnicas utilizadas para la normalización en nuestro estudio han sido las siguientes:

- Para los datos conocidos o salidas reales: al conocer que los valores de producción obtenidos en el rango de estudio han sido entre valores entre 0 y 100 la normalización es muy simple dividiendo en este caso todos los valores por 100. De este modo simplificamos cálculos.
- Para los datos de todas las variables de entrada utilizamos la normalización MIN – MAX. Dado un valor de entrada cualquiera lo normalizamos obteniendo como resultado el siguiente valor:

$$X(\text{normalizado})_i^j = \frac{x_i^j - \min(j)}{\max(j) - \min(j)} \quad (24)$$

Siendo:

x_i^j = Valor i-ésimo de la variable j.

$\min(j)$ = Mínimo Valor que toman los valores de la variable j.

$\max(j)$ = Máximo Valor que toman los valores de la variable j

Como ejemplo de la normalización utilizada exponemos los valores de la tabla expuesta anteriormente (Tabla 14. Tabla con valores ejemplos utilizados para el estudio sin manipular) pero con dichos valores normalizados

Fecha	Hora	Inversor	Temp Ext	Radiacion	Temp Int °C	H. Funcionamiento	Producción
01/06/2011	8:00	5- CT 15	0,29	0,32	0,30	0,00	0,52
01/06/2011	9:00	5- CT 15	0,59	0,68	0,17	0,01	0,74
01/06/2011	10:00	5- CT 15	0,76	0,82	0,60	0,03	0,8
01/06/2011	11:00	5- CT 15	0,88	0,87	0,51	0,04	0,82
01/06/2011	12:00	5- CT 15	0,94	0,89	0,69	0,06	0,83
01/06/2011	13:00	5- CT 15	1,00	0,90	0,80	0,07	0,81
01/06/2011	14:00	5- CT 15	1,00	0,90	0,89	0,09	0,83
01/06/2011	15:00	5- CT 15	1,00	0,91	1,00	0,10	0,83
01/06/2011	16:00	5- CT 15	0,97	0,89	0,64	0,12	0,82
01/06/2011	17:00	5- CT 15	0,94	0,85	0,56	0,13	0,81
07/06/2011	8:00	5- CT 15	0,00	0,00	0,01	0,87	0,25
07/06/2011	9:00	5- CT 15	0,38	0,48	0,00	0,88	0,61
07/06/2011	10:00	5- CT 15	0,59	0,70	0,25	0,90	0,73
07/06/2011	11:00	5- CT 15	0,74	0,91	0,17	0,91	0,87
07/06/2011	12:00	5- CT 15	0,62	0,50	0,29	0,93	0,79
07/06/2011	13:00	5- CT 15	0,79	0,84	0,45	0,94	0,76
07/06/2011	14:00	5- CT 15	0,76	1,00	0,20	0,96	0,88
07/06/2011	15:00	5- CT 15	0,62	0,77	0,40	0,97	0,75
07/06/2011	16:00	5- CT 15	0,74	0,97	0,45	0,99	0,78
07/06/2011	17:00	5- CT 15	0,76	0,96	0,39	1,00	0,8

Tabla 5-15. Tabla con valores normalizados.

Notar que en los valores expuestos como ejemplo sólo hemos normalizado, es decir, la variable de temperatura interior del inversor la seguimos dejando con °C que no implica que sea la que hayamos utilizado en el trabajo, simplemente es a modo de ejemplo. Observar igualmente que las horas de funcionamiento son valores que van de 0 a 1 puesto que es un incremental en función de las horas de funcionamiento.

Por último indicar que la normalización de valores la podemos realizar con la tabla de apoyo en la que tenemos los datos o bien simplemente podemos hacerlo con el código que utilicemos para la solución del caso. En nuestro caso particular lo hemos realizado con el código, puesto que evita posibles confusiones en los cambios de escala u olvidos que podamos cometer.

4.3.4 Software para la resolución: código. Primeros Resultados.

Conocido el modelo de Red a utilizar, conocidas las variables tanto de entrada como de salida del modelo para su entrenamiento y filtrados los datos para tener garantías de poder disponer de información de calidad estamos en disposición de resolver el modelo. Para ello existen diversos tipos de software para la resolución del estudio realizado. En nuestro caso particular se ha optado por utilizar el software R básicamente por ser software libre disponiendo por tanto de acceso para su utilización. Para su utilización, descarga y ayuda no hay más que consultar su web: <http://www.r-project.org/>

A continuación exponemos el código utilizado para la resolución del problema en R; llegando en este caso simplemente a resolver el problema generando el resumen del modelo y guardando la red generada para su posterior utilización en la siguiente parte del Proyecto.

El código utilizado es el siguiente:

```
install.packages("xlsx")
install.packages("e1071")
library(xlsx)
datos<- read.xlsx(file="hechos.xlsx",1)
summary(datos)
datos<- datos[complete.cases(datos),] #quitar casos con valores perdidos
summary(datos)
x<- datos[,1:4]
y<- datos[,5]/100
summary(y)
print(n<- nrow(x))
nent<- ceiling(0.75*n)
ntest<- n-nent
indin<- 1:n
indient<- sort(sample(indin,nent))
inditest<- setdiff(indin,indient)
xent<- x[indient,]; xtest<- x[inditest,]
yent<- y[indient]; ytest<- y[inditest]
maximos<- apply(xent,2,max)
minimos<- apply(xent,2,min)
for (j in 1:ncol(x))
{
  xent[j]<- (xent[,j]-minimos[j])/(maximos[j]-minimos[j])
  xtest[j]<- (xtest[,j]-minimos[j])/(maximos[j]-minimos[j])
}
boxplot(xent)
```



```

boxplot(xtest)
library(nnet)
library(e1071)
redtune<- tune(nnet,xent,yent,ranges=list(size=1:10,decay=(0:15)/10),maxit=1000,linout=TRUE)
summary(redtune)
plot(redtune)
redselvc<- redtune$best.model

cat("ECM entrenamiento =", mean((100*yent-(100*predict(redselvc,xent)))^2),"\n")
cat("ECM test =", mean((100*ytest-(100*predict(redselvc,xtest)))^2),"\n")
cat("R2 entrenamiento =", cor(100*yent,(100*predict(redselvc,xent)))^2,"\n")
plot(100*yent,100*predict(redselvc,xent),col="blue", main="Predicciones nnet (entrenamiento)")
lines(100*yent,100*yent,type="l",col="red")
plot(1:nent,100*yent,type="l",main="Entrenamiento")
lines(1:nent,100*predict(redselvc,xent),lty=2,col="red")

cat("R2 test =", cor(100*ytest,(100*predict(redselvc,xtest)))^2,"\n")
plot(100*ytest,100*predict(redselvc,xtest),col="blue", main="Predicciones nnet (test)")
lines(100*ytest,100*ytest,type="l",col="red")
plot(1:ntest,100*ytest,type="l",main="Test")
lines(1:ntest,100*predict(redselvc,xtest),lty=2,col="red")
save(redselvc,file="Redfinal.RData")
summary(redselvc).

```

Una vez expuesto el código, a continuación explicamos el significado de las instrucciones incluidas, puesto que de este modo se pretende por una parte dejar claro su significado con objeto de que pueda volver a ser utilizado y adaptado a algún otro modelo de un modo ya sencillo. Por otra parte nos sirve para ir introduciendo parte de los resultados obtenidos con el modelo.

Con la función “install.packages(“xlsx”)” cargamos el paquete que nos permite leer los datos de las variables definidas anteriormente desde una hoja de cálculo de Excel.

A continuación instalamos la librería (install.packages(“e1071”)) e1071 que contiene la función “tune” que utilizaremos posteriormente, la cual es la que nos sirve para realizar la validación cruzada, es decir, nos calcula el óptimo de nodos o células en la capa oculta (definiremos los límites) y a su vez nos aporta el mejor valor de lambda para evitar los sobreajustes. Es decir nos selecciona la red óptima que minimiza el error cuadrático medio utilizando el algoritmo de Cuasi-Newton.

$$\text{ECM} + \text{Penalización} = \frac{1}{n} \sum_{k=1}^n (T_k - Z_k)^2 + \gamma \sum_{i=1}^M w_i^2 \quad (23)$$

Una vez instaladas las dos librerías lo que realizamos es cargarlas en el sistema para utilizarlas posteriormente, de ahí las funciones “library (xlsx)” y library “(e1071)”.

Posteriormente procedemos a cargar en el sistema todos los datos de las variables de estudio. Estos datos se incluyen en la hoja de cálculo en columnas sin encabezado alguno, simplemente con los datos ordenados directamente. Es muy importante que los datos estén ordenados de modo que cada fila contendrá para una determinada fecha los datos de todas las variables de estudio así como el valor de la salida o producción conocida para esos valores. Son los datos que utilizaremos en la propagación hacia atrás para el ajuste de los pesos.

Como ejemplo la siguiente tabla de datos:

303	490	312,10	9900	52
313	756	311,00	9901	74
319	861	314,70	9902	80
323	902	313,90	9903	82
325	918	315,50	9904	83
327	924	316,50	9905	81
327	923	317,20	9906	83
327	929	318,20	9907	83
326	916	315,10	9908	82
325	890	314,40	9909	81
295	21	312,50	9910	1
296	48	311,20	9911	5
298	111	312,10	9912	13
301	279	312,90	9913	31
308	848	314,50	9914	80
313	887	315,00	9915	88
314	898	315,50	9916	88
314	888	316,60	9917	81
318	891	317,20	9918	84
315	719	318,10	9919	73

Tabla 6-16. Tabla ejemplo con datos para carga en software.

Estos son algunos de los datos iniciales cargados de los miles incluidos en el software, simplemente se pretende mostrar la forma de la carga de datos. No tienen ningún tipo de encabezado, simplemente los datos en una hoja de Excel con un determinado nombre. Es importante que aún cargando de este modo los datos tengamos claramente referenciados el orden de todos los datos para que el entrenamiento sea correcto a la vez que la interpretación de los resultados obtenidos. De los datos de las columnas anteriores el orden respectivo de las columnas es el siguiente: Temperatura Ambiente o Exterior, Radiación, Temperatura interna del inversor y producción. Por fila están ordenados los datos por fecha y hora considerando todo el periodo de estudio.

Es crucial el ser muy escrupuloso en el tratamiento de los datos, existe gran cantidad de información en diversos formatos por lo que nunca debemos perder de vista el origen de todos ellos para poder posteriormente interpretar correctamente los resultados así como realizar predicciones adecuadas en función de dichos valores.

Los datos se cargan leyendo ya directamente del Excel al que hemos nombrado “hechos” y cuyos datos van incluidos en la hoja 1 cuya denominación debe ser literal para que la lectura sea correcta. La lectura queda almacenada en una variable que denominaremos “datos”. La función que ejecuta toda esta acción es la siguiente: “datos<- read.xlsx(file="hechos.xlsx",1)”.

Una vez cargados los datos en la variable “datos” podemos tener un resumen de todos ellos con la función “summary(datos)”. Con esta función podemos verificar por ejemplo que los datos cargados son correctos.

Si la aplicamos sobre el estudio realizado obtenemos el siguiente resultado (lo incluimos en formato tabla para que quede claro el resultado pero no es el formato que nos aporta la aplicación):

Resultados	X303	X490	X312,10	X9900	X52
Mínimo	273	9,5	288,5	9901	1
Cuartil 1	298,2	34,9	309,2	10886	41
Mediana	309,8	786,5	312,6	11770	74
Media	308,7	643,9	312,2	12134	60,46
Cuartil	319,8	903,2	316,6	13974	82
Máximo	333,2	1291	329,1	14990	99

Tabla 7-17. Resultado con valores obtenidos en R.

Nos muestra un primer valor con la X inicial para denotar la variable e identificarla. En el caso que aplica y según se han ordenado corresponden respectivamente con Temperatura Ambiente, Radiación Global Horizontal, Temperatura Interna del Inversor, y Produccion como valor real conocido. Tal y como se ha definido previamente la escala de la temperatura viene definida por grados Kelvin.

Una vez identificadas las variables la información que nos aporta para cada una de las variables introducidas: valor mínimo, valor máximo, percentil 25, percentil 75, media y mediana. Estos valores a parte del resumen como tal, nos pueden servir como verificación de la información recibida por la aplicación. Se comprueba y verifica la validación de dichos datos.

Tras leer todos los datos el paso lógico sería verificar que todos los datos están completos, pues datos vacíos en alguno de los campos podrían distorsionar el resultado, Para ellos utilizamos la siguiente función “datos<-datos[complete.cases(datos),]”, que lo que hace es modificar los datos contenidos en la variable “datos” de modo que elimina aquellas filas que contienen datos vacíos en alguno de sus campos.

A continuación volvemos a resumir los datos con la función summary(datos), de modo que verificamos la variación con respecto al apartado anterior. En el caso particular de estudio el resultado como era de esperar sigue siendo el mismo, puesto que ya habían sido previamente verificados todos ellos. En todo caso no está de más el volver a comprobar de modo mecánico, obteniendo de nuevo por tanto los mismos datos que los de la tabla anterior.

Validada toda la información de las variables de entrada y salida en “datos” el siguiente paso consiste en separar los valores de las variables de entrada de los datos de la variable de salida puesto que su comportamiento en el algoritmo es diferente por lo que es necesario tenerlas identificadas de distinta manera.

Todos los datos de las variables de entrada los almacenamos en una nueva variable (matriz de valores) denotada como x, lo hacemos con la siguiente función “x<- datos[,1:4]”. Por tanto en esta matriz de valores “x” ya tenemos almacenados todos los valores por columnas de las 4 variables de entrada al sistema.

Del mismo modo obtenemos los valores de la columna que contiene los datos de la salida, aunque en esta ocasión y para evitar pasos podemos directamente aprovechar para normalizar dichos valores. Puesto que como hemos visto en el paso anterior el máximo de los valores es 99, sería suficiente con dividir todos ellos entre 100 para tenerlos normalizados. Ambas operaciones las realizamos con la siguiente función

“y<- datos[,5]/100”.

Para verificar que este paso lo hemos realizado correctamente volvemos a resumir los datos obtenidos en este paso con la función “summary(y)”, de modo que obtenemos el mismo resultado que obtuvimos de la última variable al realizar “summary(datos)” pero divididos por 100, por lo que el resultado es correcto.

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
0.0100	0.4100	0.7400	0.6046	0.8200	0.9900

Tabla 8-18. Resultado con valores obtenidos en R.

Puesto que necesitamos dividir el conjunto de valores para realizar el entrenamiento y el testeo del resultado, contabilizamos el número de filas de la matriz de valores x. Para ello utilizamos la función “n<- nrow(x)” guardando en la variable n el número de casos. Si queremos conocerlo y verificar que los ha leído todos del Excel introducido no tenemos más que escribir la instrucción “print(n)” devolviéndonos la aplicación R el valor 3535, que indica lógicamente el número de entradas a la red.

Como hemos comentado en el siguiente paso dividimos el conjunto de valores entre entrenamiento y validación o testeo. Para el entrenamiento consideramos el 75% de los valores generando por tanto la variable (matriz de valores realmente) nent a la que le asignamos dichos valores y lo hacemos con la siguiente función “nent<-ceiling(0.75*n)” puesto que como hemos visto “n” contabilizaba el total de casos. Para el testeo utilizamos el resto de valores, que no es más que la diferencia entre el total y los utilizados para el entrenamiento por tanto utilizamos “ntest<-n-nent”.

Puesto que vamos a necesitar mediante algún bucle recorrer todos los valores tanto del entrenamiento como del testeo, generamos para ello índices que recorran dichos valores. Generamos por tanto un índice total con la función:

```
“indin<- 1:n”
```

De este conjunto total del índice que recorre todos los valores de entrada de la red, extraemos los del entrenamiento y que vienen determinados por la función.

```
“indient<- sort(sample(indin,nent))”
```

Finalmente almacenamos el resto de valores del índice en el entrenamiento que no es más que la diferencia entre el total guardado en indin y el almacenado en indient, para ello utilizamos la siguiente función:

```
“inditest<- setdiff(indin,indient)”
```

Utilizamos los valores de los índices generados para extraer de las variables de entrada y salida los correspondientes valores de entrenamiento y testeo tanto de las 4 variables de entrada como de la salida. Para ello generamos 4 nuevas variables del siguiente modo:

```
xent<- x[indient,]; xtest<- x[inditest,]
```

```
yent<- y[indient]; ytest<- y[inditest]
```

Según hemos definido en el apartado anterior (4.3.3) a continuación vamos a normalizar todos los datos de las variables de entrada. Los valores de la variable de salida ya los normalizamos anteriormente dividiendo todos ellos por 100. Para la normalización utilizamos la técnica de “Normalización MIN – MAX.”

En primer lugar calculamos máximo y mínimo de los valores, utilizando para ello los valores extraídos para el entrenamiento. Dichos valores los vamos a utilizar para normalizar tanto los valores del entrenamiento como los valores del test.

La obtención de dichos valores los almacenamos en las variables máximos y mínimos, con las siguientes funciones:

```
“maximos<- apply(xent,2,max)”
```

```
“minimos<- apply(xent,2,min)”
```

Una vez obtenidos dichos valores, mediante un bucle recorreremos las variables de entrada (utilizando los índices previamente calculados) para normalizar todos los valores con la técnica indicada.

La función que realiza en R dicha operación es la siguiente:

```
for (j in 1:ncol(x))  
{  
  xent[j]<- (xent[,j]-minimos[j])/(maximos[j]-minimos[j])  
  xttest[j]<- (xttest[,j]-minimos[j])/(maximos[j]-minimos[j])  
}
```

Ya tenemos por tanto normalizados todos los valores, tanto los de la variable de salida como las de las variables de entrada distinguidos por entrenamiento y testeo. Como complemento podemos obtener un gráfico de caja y bigote de dichas variables normalizadas, siendo por ejemplo los siguientes:

```
boxplot(xent)
```

```
boxplot(xttest)
```

Como resultado de uno de ellos (ya que ambos son muy similares), exponemos el gráfico obtenido con dicha función:

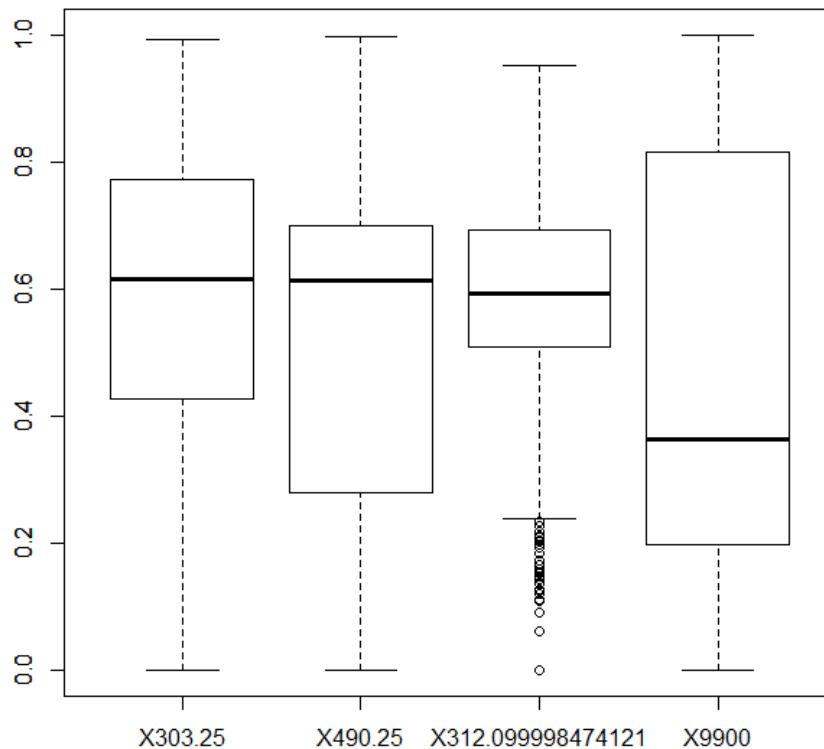


Figura 6-24. Gráfico de Caja y Bigote.

Una vez normalizados los datos, eliminados aquellos que tienen entradas vacías y guardados todos ellos en las diferentes variables, el siguiente paso a realizar es el de cargar la función que nos permita utilizar la red Backpropagation. En el caso del software R para la minimización del error cuadrático medio utiliza por defecto el algoritmo de Cuasi-Newton. Dicha función en R es `nnet` que se carga con la función:

```
“library(nnet)”
```

El siguiente paso es comenzar a entrenar la red, para ello vamos a utilizar la función `tune` con diversos parámetros que explicaremos a continuación, La función utilizada para ello es la siguiente:

```
“redtune<- tune(nnet,xent,yent,ranges=list(size=1:10,decay=(0:15)/10),maxit=1000,linout=TRUE)”
```

Guardamos el resultado obtenido de la red, en variable que denominamos `redtune`. Con la función `tune` indicamos lo siguiente:

- Utilizamos la red Backpropagation con los valores seleccionados para el entrenamiento (`nnet`, `xent`, `yent`)
- Indicamos un rango de valores para determinar el número de células de la capa oculta (`size=1:10`) Lo limitamos a 10 siendo suficiente como máximo número de células de la capa oculta, Se han realizado cálculos incrementando dicho valor perdiendo en este caso tiempo en computación y recursos y no obteniendo mejores resultados.

- Por otra parte, con el rango de valores indicamos que el valor de penalización posible (utilizamos en el algoritmo la fórmula de minimización del ECM con penalización para evitar los sobreajustes) oscila entre 0 y 1,5 ($\text{decay}=(0:15)/10$).
- Con cada par de valores combinados entre el número de células de la capa oculta y el valor de la posible penalización el algoritmo realiza 10 entrenamientos calculando el ECM para cada uno de ellos. De todos los entrenamientos realizados se selecciona aquel que hace menor el ECM lo cual nos permite obtener el número de células de la capa oculta y el valor del parámetro de penalización. En muestras de tamaño suficientemente grande como es el caso de estudio obtenemos siempre valor 0 para el coeficiente de penalización.
- Indicamos igualmente que el número máximo de iteraciones es de 1000 ($\text{maxit}=1000$) para el entrenamiento de la red y la obtención de la óptima. Igualmente se han realizado modificaciones en este parámetro considerando que 1000 es suficiente para la obtención de un buen resultado con este número de valores para el entrenamiento. Valores mayores no aporta mejoras.
- La función de activación utilizada en la capa oculta es la función Sigmoide.
- Con el parámetro “ $\text{linout}=\text{TRUE}$ ” indicamos que la función de activación de la capa de salida es la función identidad. Si queremos utilizar la función de activación Sigmoide también en la capa de salida indicaríamos “ $\text{linout}=\text{FALSE}$ ”. Dicho parámetro también lo hemos utilizado obteniendo penalización en el tiempo de computación pero no mejora alguna en la red.

Una vez realizado el cálculo anterior podemos indicarle al software que nos de el resumen de la de que hemos denominado “ redtune ” con la función “ $\text{summary}(\text{redtune})$ ”. Obtenemos como hemos indicado todos los entrenamientos con cada uno de los pares de valores entrenados y que nos aporta el menor ECM.

Finalmente guardamos en la variable redselvc la red seleccionada para el modelo. Lo realizamos con la función:

```
“redselvc<- redtune$best.model”
```

Los resultados obtenidos los mostramos fácilmente con la función “ $\text{summary}(\text{redselvc})$ ”, que nos da los valores óptimos de los pesos de todas las conexiones entre las diferentes células de la red.

Obtenido el mejor modelo las siguientes instrucciones en R están orientadas a mostrar los resultados obtenidos, es decir, los errores cuadráticos medios, correlación lineal, gráficos y pesos de la red final. Con estos resultados, que exponemos en detalle en los siguientes apartados, alcanzaremos el Objetivo General perseguido en el Proyecto, que no es otro que el “Modelo Ideal de Producción”.

Para la obtención del Error Cuadrático Medio tanto del entrenamiento como del test utilizamos funciones que exponemos a continuación:

```
“cat("ECM entrenamiento =", mean((100*yent-(100*predict(redselvc,xent)))^2),"\n")”
```

```
“cat("ECM test =", mean((100*ytest-(100*predict(redselvc,xtest)))^2),"\n")”
```

Utilizamos la función “ cat ” para la realización de los cálculos, entrecomillando en sus parámetros el nombre con el que denotaremos en el resultado dichos valores calculados. En nuestro caso los denominaremos ECM entrenamiento y ECM test.

Notar que es importante considerar la normalización realizada previamente con los valores de la variable de salida, de modo que desagamos dicha normalización para obtener los valores reales que nos permita comparar con la realidad. El hecho de no hacerlo puede hacernos llegar a cometer errores puesto que obtendríamos valores muy pequeños

Del mismo modo calculamos el coeficiente de correlación (nos indica la bondad del ajuste) entre la variable de salida y la predicción realizada, tanto para las variables del entrenamiento como para las variables del test. La función en R para dicho cálculo es la siguiente:

```
“cat("R2 entrenamiento =", cor(100*yent,(100*predict(redselvc,xent)))^2,"\n")”
“cat("R2 test =", cor(100*ytest,(100*predict(redselvc,xtest)))^2,"\n")”
```

Si queremos obtener información gráfica de los cálculos realizados lo podemos realizar con la función `plot`. Para ello utilizamos las siguientes funciones en R:

Para el entrenamiento:

```
“plot(100*yent,100*predict(redselvc,xent),col="blue", main="Predicciones nnet (entrenamiento)")”
“lines(100*yent,100*yent,type="l",col="red")”
“plot(1:nent,100*yent,type="l",main="Entrenamiento")”
“lines(1:nent,100*predict(redselvc,xent),lty=2,col="red")”
```

Para el test:

```
“plot(100*ytest,100*predict(redselvc,xtest),col="blue", main="Predicciones nnet (test)")”
“lines(100*ytest,100*ytest,type="l",col="red")”
“plot(1:nent,100*ytest,type="l",main="Test")”
“lines(1:nent,100*predict(redselvc,xtest),lty=2,col="red")”
```

Finalmente y para concluir es interesante poder guardar el resultado obtenido con objeto de poder volver a utilizarlo en cualquier otro momento. De este modo utilizaremos el mismo modelo y no tenemos que volver a realizar todas las operaciones con el consabido esfuerzo y coste. Esta es lógicamente por tanto una de las grandes ventajas del software.

La función utilizada para guardar el modelo es la siguiente:

```
“save(redselvc,file="Redfinal.RData")”
```

Finalmente podemos volver a obtener los resultados con la función ya vista:

```
“summary(redselvc)”
```


4.3.5 Arquitectura Seleccionada.

Para obtener el modelo final de arquitectura es necesario solamente indicar el número de células de la capa oculta, puesto que conocemos tanto las variables de entrada como la salida. Para justificar el óptimo de dichas células referenciamos el código utilizado (en este caso exponiendo ya los resultados que justifican la selección de dicho número):

```
“redtune<- tune(nnet,xent,yent,ranges=list(size=1:10,decay=(0:15)/10),maxit=1000,linout=TRUE)”  
“summary(redtune)”
```

Los resultados obtenidos son los siguientes:

sampling method: 10-fold cross validation

- best parameters:

size decay

10 0

- best performance: **0.008235409**

- Detailed performance results:

	size	decay	error	dispersion
1	1	0.0	0.022263263	0.0262877991
2	2	0.0	0.009612097	0.0014211972
3	3	0.0	0.009104120	0.0014170569
4	4	0.0	0.008734193	0.0012396270
5	5	0.0	0.008558731	0.0010181507
6	6	0.0	0.008397384	0.0013050425
7	7	0.0	0.008442925	0.0011277842
8	8	0.0	0.008282985	0.0009222277
9	9	0.0	0.008345413	0.0010343004
10	10	0.0	0.008235409	0.0011427889
11	1	0.1	0.010289422	0.0016469840
12	2	0.1	0.009994826	0.0016829780

Tabla 9-18. Resultado de R con número de células capa oculta

Este algoritmo itera hasta las 160 combinaciones posibles, en decir realiza un entrenamiento por cada una de las combinaciones posibles entre los 1 a 10 (valores enteros) valores del número de células de la capa oculta y entre los 0 a 1,5 (tomados con incrementos de 0,1) valores que pueda tomar la posible penalización del algoritmo de optimización del ECM con penalización., De todos ellos se queda con aquél que tiene un error menor, el cual hemos marcado obteniendo como se ha indicado:

- Células de la capa oculta: 10 (size)
- Valor de la penalización: 0 (decay)
- Best Performance: 0.008235409.

Igualmente podemos obtener un gráfico resumen del cálculo realizado para la obtención de dichos valores. La función utilizada que nos aporta el gráfico mostrado es “plot(redtune)”

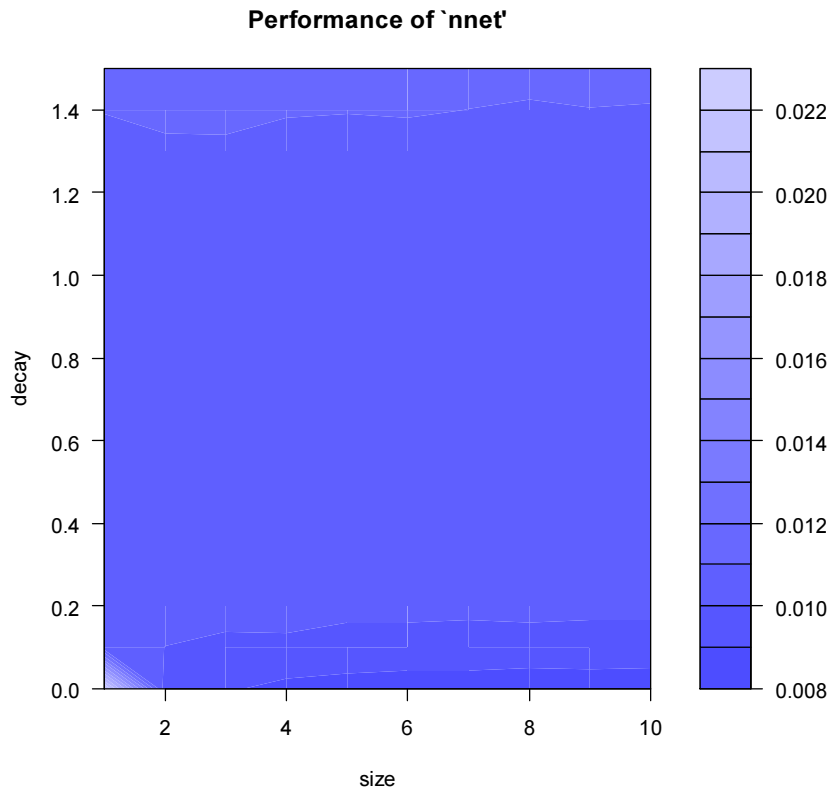


Figura 7-25. Gráfico mejor solución células capa oculta y valor de penalización del ECM.

Con todos estos resultados podemos mostrar la arquitectura final de la Red Neuronal, puesto que conocemos el número de Neuronas de cada una de las capas. Como resumen:

- Neuronas de la Capa de Entrada: 4 que son las variables más la ganancia o bias.
- Neuronas de la Capa Oculta: 10. La que nos aporta menor ECM
- Neuronas de la Capa de Salida: 1, ya conocida que es la producción.

Nos podemos igualmente plantear de inicio si sería aconsejable el incluir más capas ocultas, cuestión complicada de por sí en cuanto a su elección. El incluir más capas ocultas conlleva mucho mayor cálculo, lo que en procesos especialmente complicados de ajustar puede ser razonable, sin embargo son es el caso particular del estudio. En todo caso y por verificar resultados, he realizado pruebas y entrenamientos con mayor número de capas consiguiendo simplemente ralentizar enormemente el proceso sin obtener mejoras significativas en los resultados ya obtenidos, por lo que podemos concluir que la arquitectura seleccionada es la mejor para nuestro estudio. Dicha arquitectura viene determinada por el siguiente gráfico:

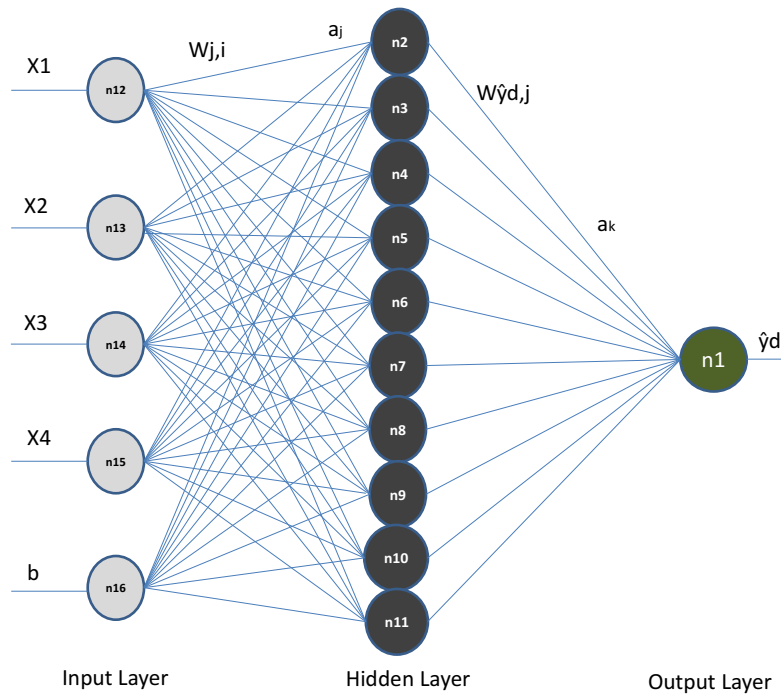


Figura 8-26. Arquitectura Red Seleccionada.

4.3.6 Entrenamiento y testeo.

Para no inundar el documento con exceso de datos numéricos que nos aporta el software utilizado, exponemos como resumen de todos ellos los gráficos resultado tanto del entrenamiento como del testeo realizado, utilizando para ello el 75% y 25% de los valores respectivamente (tal y como ha quedado definido en el código). Los valores normales para testeo y entrenamiento son habitualmente los utilizados, sin embargo y como en todos los pasos anteriores hemos realizado diferentes variaciones para contrastar los resultados, considerando finalmente el 75%, 25% al haber obtenido la mejor solución.

En primer lugar exponemos el gráfico que hemos denominado con el código “Predicciones nnet (entrenamiento):

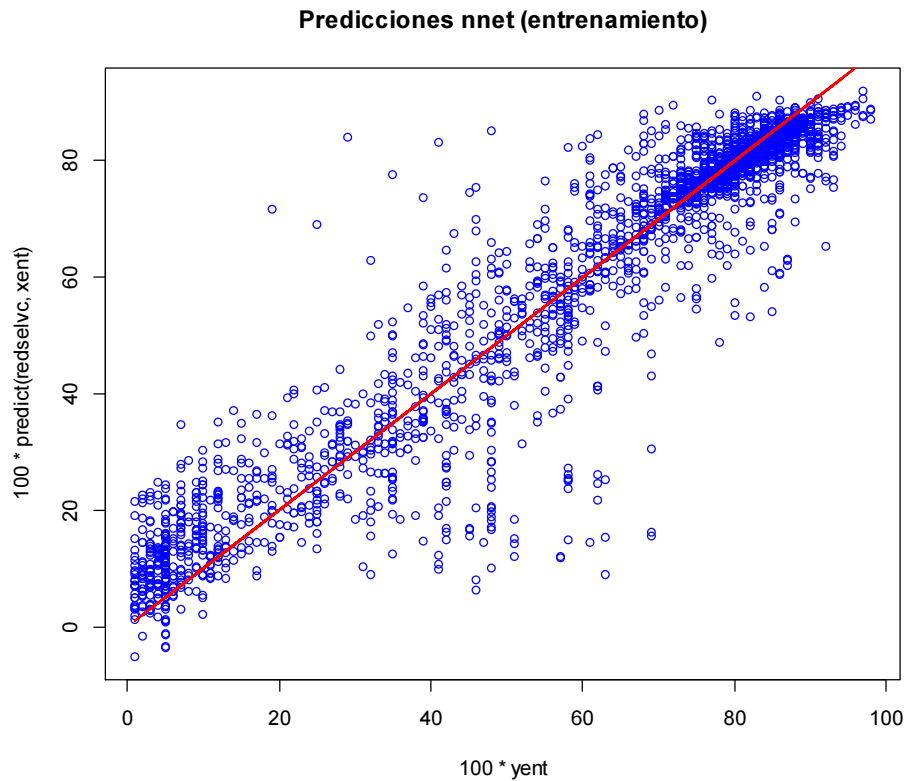


Figura 9-27. Nube de puntos frente a la recta de valores reales en el Entrenamiento.

En dicho gráfico hemos denotado por una parte la nube de puntos en color azul (`plot(100*yent,100*predict(redselvc,xent),col="blue", main="Predicciones nnet (entrenamiento)")`) que confronta los valores de la salida real de producción frente a los valores de la predicción con la red utilizada para el entrenamiento. Como se puede observar todos los valores van multiplicados por 100 para deshacer la normalización.

Por otra parte se ha incluido una línea o recta (`lines(100*yent,100*yent,type="l",col="red")`) en color rojo que nos indica el valor real de producción. De este modo podemos tener una primera aproximación visual del modelo que estamos entrenando, con la proximidad de la nube de puntos a la recta. Observamos que la nube de puntos efectivamente está muy próxima a la recta lo que nos indica que el resultado del modelo es bueno.

En segundo lugar exponemos el gráfico que hemos denominado con el código “Entrenamiento”

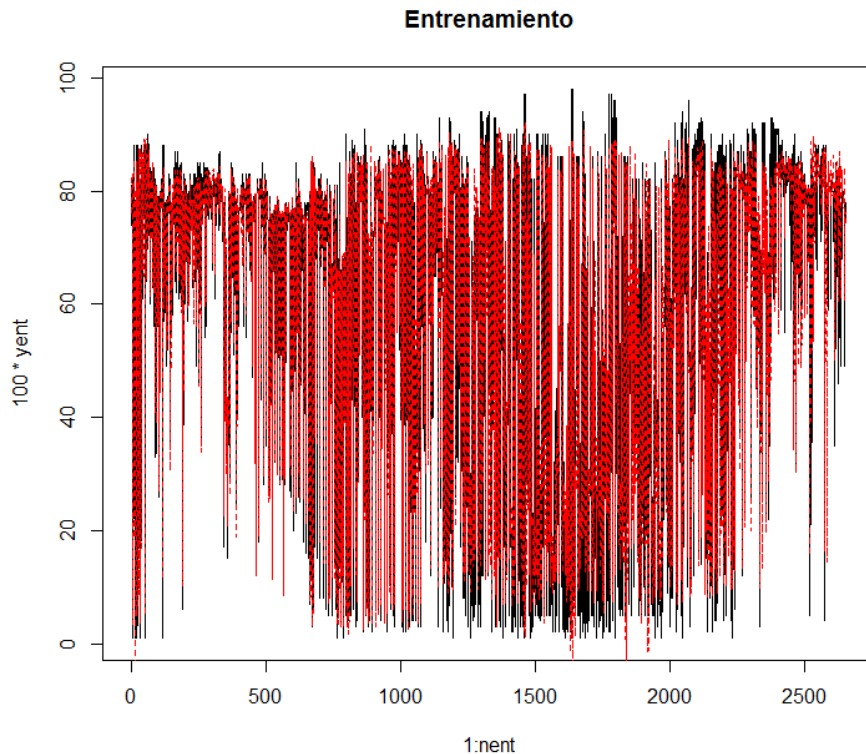


Figura 10-28. Comparación entrenamiento valores predichos frente a valores reales.

En este gráfico comparamos los valores concretos obtenidos con la predicción de los valores del entrenamiento a través de la red en color rojo (75% del total de los datos del entrenamiento), frente a los valores reales de producción en dichos valores que se denotan con las líneas en color negro. Del mismo modo que en el gráfico anterior la cercanía entre ambas líneas que se confunden nos hace indicar que el modelo es bueno ya que ambos valores obtenidos son muy cercanos. Para dibujar el gráfico utilizamos el código denotado en el apartado anterior:

```
plot(1:nent,100*yent,type="l",main="Entrenamiento")
lines(1:nent,100*predict(redselvc,xent),lty=2,col="red")
```

Para finalizar con el entrenamiento exponemos el resultado obtenido de los valores solicitados al software que nos aportan numéricamente la explicación de la aproximación del modelo. Deben ser valores acordes a lógicamente a los expuestos en los gráficos anteriores.

Los valores solicitados con el código han sido el Error Cuadrático Medio y el Coeficiente de correlación entre los valores de la predicción y los valores reales conocidos de la producción. Obtenemos:

ECM (entrenamiento) = 75.21266. O utilizando la desviación el valor obtenido es de 8.67. Lo importante de estos valores es su interpretación para que nos indiquen la exactitud que contienen en el caso de estudio.

El valor 8,67 nos indica que la diferencia en media entre los valores de la producción real y los valores del entrenamiento es justo esa cantidad. Es decir si sobre el gráfico “Predicciones nnet (entrenamiento)” calculamos las distancias de todos los puntos de la nube de puntos a la “línea roja” o valores reales de producción y calculamos su media obtenemos el valor 8.67. Conocemos que los valores de producción horaria que estamos estudiando oscilan entre 1 y 100 por lo con esta sólo información no podemos concluir si el modelo es bueno o es malo. Si la mayoría de los valores de producción fuesen próximos a 1 el resultado obtenido es realmente malo, lo contrario si la mayoría de los valores se aproximan a 100. Es por ello que para tener la información completa y obtener una conclusión razonables calculamos el coeficiente de correlación entre valores reales de producción y valores obtenidos con la red en los valores utilizados para el entrenamiento. El valor obtenido, que hemos denominado R2 es

R^2 entrenamiento = 0.905068.

Este valor lo que nos indica es que queda explicado entre ambos valores una relación del 90.5%, o lo que es lo mismo, existe una variabilidad del 9.5% no explicada por el modelo.

Este valor no es generaliable para obtener una conclusión en diferentes procesos industriales, la validez o no de ello dependerá de la interpretación por los profesionales en cada caso particular.

En el caso de estudio y conociendo además que existe ese porcentaje no explicado o lo que es lo mismo que podemos tener esa diferencia en la predicción en media, podemos aseverar que el modelo es válido.

En cuanto al test realizado para validar el modelo con el 25% de los valores restantes los gráficos obtenidos análogos a los del entrenamiento son los siguientes:

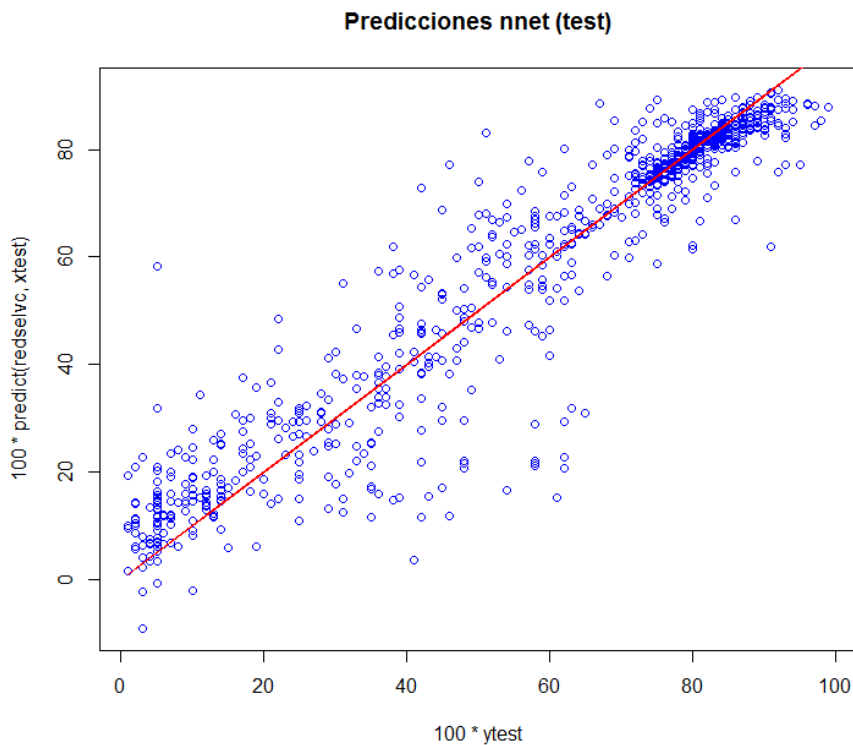


Figura 11-29. Nube de puntos frente a la recta de valores reales en el Test.

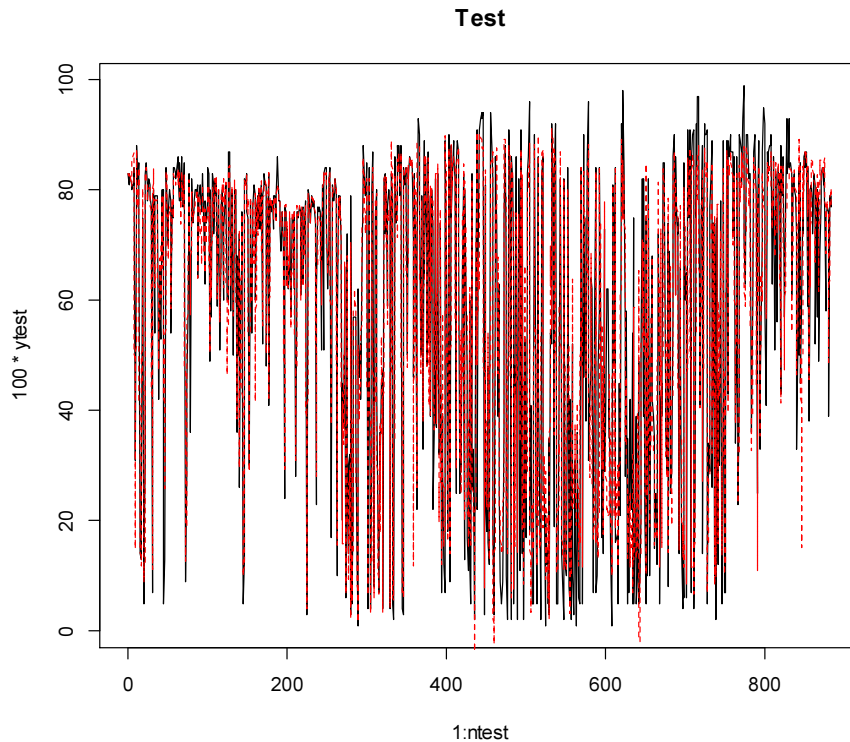


Figura 12-30. Comparación test valores predcidos frente a valores reales.

A parte de dichos gráficos y como en el entrenamiento aportamos los valores del error cuadrático medio y del coeficiente de correlación lineal.

ECM test = 81.24364

R2 test = 0.899802

La explicación de los resultados anteriores es totalmente análoga a la indicada en el caso del entrenamiento por lo que no merece la pena repetir la misma explicación. Simplemente indicar que los resultados son muy similares lo que nos indica que los resultados obtenidos en el entrenamiento son acordes.

Podríamos finalizar por tanto y validar el modelo con los datos obtenidos. Sin embargo y debido a que poseemos datos históricos los vamos a utilizar para verificar o no los datos obtenidos en este mismo apartado.

Según indicamos con el código utilizado para la resolución del modelo con el software R, incluimos una instrucción para guardar el modelo. Dicha instrucción es la siguiente: “save(redselvc,file="Redfinal.RData")”

Podemos por tanto utilizar la Red para realizar predicciones que en el caso en cuestión vamos a utilizar como comprobación de la validación del modelo. Para ello y por proximidad en fechas hemos utilizado los datos de producción de los dos siguientes meses a los utilizados para la creación de la red. Es decir, realizamos una predicción de los meses de Julio y Agosto del 2012, comparamos con los valores reales y verificamos la validación de los resultados obtenidos. En este apartado no incluimos datos de la predicción y posterior comparación de dichos meses puesto que algo muy similar es lo que vamos a realizar posteriormente para la obtención de los objetivos específicos. No se trata en consecuencia de aportar exceso de valores numéricos, en este caso nos ha servido como una nueva validación del modelo.

Para la predicción de valores utilizamos el siguiente código:

```
load(file = "Redfinal.RData")
summary(redselvc)
xprueba<- read.xlsx(file = "entradas.xlsx",1)
pred<-predict(redselvc,xprueba)
cbind(xprueba,pred)
```

Lo que hacemos inicialmente con el código es cargar de nuevo la red cargada, para lo que hemos utilizado la instrucción “load(file = "Redfinal.RData)". Verificamos los datos de dicha red con la instrucción “summary(redselvc)” que nos aporta un resumen.

Posteriormente cargamos a través de una hoja de cálculo Excel o incluso con un archivo de texto plano los valores de las variables de entrada tal y como se han definido correspondientes a los meses de Julio y Agosto del 2012. Lo realizamos con la instrucción “xprueba<- read.xlsx(file = "entradas.xlsx",1)”.

Con los datos y la red cargados, el siguiente paso es predecir para lo que utilizamos la instrucción

“pred<-predict(redselvc,xprueba)”. Finalmente queremos mostrar el resultado de la predicción realizada para lo que utilizamos la instrucción “cbind(xprueba,pred)” que nos devuelve los valores de la producción predecida de 8:00 a 15:00 de los meses de Julio y Agosto del 2012.

Tal y como hemos indicado dicho resultado vuelve a aportarnos validez del modelo. En la predicción que realizaremos en la consecución de los objetivos específicos nos detendremos en aportar valores numéricos en la predicción realizando a su vez la correspondiente interpretación de un modo mucho más detallado.

4.3.7 Solución al problema en R

En los apartados anteriores hemos validado los resultados obtenidos explicando en detalle los gráficos resultantes, error cuadrático medio y coeficiente de correlación. Por otra parte, se ha indicado la arquitectura con el número de células óptimo en la capa oculta. Por tanto para dejar el modelo totalmente definido lo único que quedaría por indicar serían los pesos de las conexiones entre todas las células de las diferentes capas.

La arquitectura que volvemos a indicar es la siguiente:

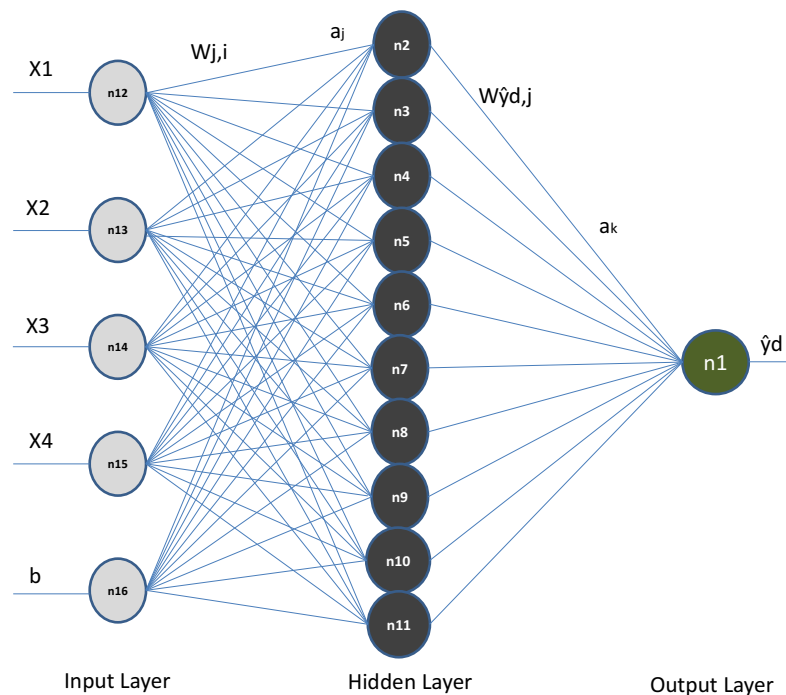


Figura 13-31. Arquitectura Seleccionada Red Óptima.

Con la instrucción en R “summary(redselv)” obtenemos los pesos de todas las conexiones, los cuales aportaremos en una tabla a continuación. En cuanto a la notación que nos ofrece el software en el resultado de los pesos es la siguiente:

i: para los input, es decir la nomenclatura que utiliza el software para las variables de entrada.

b: el término independiente, ganancia o bias. Esta neurona tiene una conexión con cada una de las neuronas de la capa oculta.

h: para las neuronas de la capa oculta.

O: output, es decir, la nomenclatura que utilizar el software para las variables de salida. En nuestro caso particular solo una.

Por tanto si notamos un valor de i3 a h4, indica que dicho valor es el peso de la red entrenada que conecta la neurona 3 de la capa de entrada con la neurona 4 de la capa oculta.

En el caso de estudio y por dejar claro a qué variable corresponde cada uno de esos valores adjuntamos la siguiente relación:

Entrada: i	Variable
1	Temperatura Exterior o Ambiente
2	Radiación
3	Temperatura Interna del Inversor
4	Horas de Funcionamiento
Salida: o	Variable
1	Producción Horaria
Ocultas: h	Variable
1 a 10	10 variables

Tabla 10-19. Resumen valores entradas y salidas

Con la notación anterior que aporta el software indicamos en la siguiente tabla el peso de cada una de las conexiones entre todas las neuronas que componen la Red Neuronal.

		Entradas: i				
		b	1	2	3	4
Capa Oculta: h	1	-2,25	9,55	-7,59	-1,95	2,57
	2	-30,25	-14,29	38,72	12,75	1,11
	3	-11,27	26,54	8,40	8,53	14,56
	4	-19,51	36,61	-14,66	16,71	31,81
	5	-2,80	4,93	1,18	-2,29	6,74
	6	-1,41	2,55	1,99	-2,70	8,47
	7	1,44	-4,38	-1,00	0,64	-1,32
	8	9,96	-22,99	3,22	2,57	-3,88
	9	-1,76	3,14	1,76	-2,60	7,95
	10	1,36	-8,00	-5,08	1,54	-129,00

Tabla 11-20. Pesos Arquitectura o Red Óptima. Entradas con Capa Oculta.

		Capa Oculta: h									
		1	2	3	4	5	6	7	8	9	10
Salida: o	o	-0,41	-0,17	-0,64	-0,25	-4,59	-9,57	-3,92	0,26	13,62	1,35

Tabla 12-21. Pesos Arquitectura o Red Óptima. Capa Oculta con Salida.

Con todos los pesos expuestos obtenemos la Red Neuronal óptima, lo que implica que podemos realizar ahora con ellos cualquier predicción para conocer la producción dada como entrada la temperatura ambiente, la radiación, la temperatura interna del inversor y las horas de funcionamiento.

Este resultado se puede obtener a través del software con la función explicada o en un caso concreto lo podríamos realizar numéricamente tal y como se ha explicado en el ejemplo del apartado 3.1.

Como Resumen final al planteamiento realizado indicar de nuevo los datos utilizados:

Red: Backpropatgation

Algoritmo de Minimización del ECM: Cuasi_Newton con Penalización.

Software: R

Iteraciones para el Entrenamiento: 1000

Valores para el Entrenamiento: 75% del conjunto

Valores para el Testeo: 25%

ECM: 75.21266

R2: 0.905068

Coefficiente de Penalización: 0

Variables de Entrada: 4

Variables de Salida: 1

Número de Células de la Capa Oculta: 10

Periodo de Estudio: Junio 2011 – Julio 2012

Horario de Produccion estudiado: desde las 8:00 hasta las 17:00

4.4 Objetivo General: Modelo Ideal de Producción

Con los datos obtenidos y definida totalmente la Red podemos concluir que hemos alcanzado el Objetivo General del Proyecto, que tal y como se definió a inicio es la “*Obtención del Modelo Ideal de Producción*”.

Con dicho objetivo podríamos habernos planteado el concluir el Proyecto, sin embargo y salvo lo novedoso de toda la metodología utilizada para su resolución se quedaría en una base teórica.

La aportación más interesante por tanto es la utilización de dicha base teórica para la obtención de los denominados Objetivos Específicos del Proyecto. La base es muy potente como para poder obtener nuevas conclusiones en la Planta Objeto de estudio y que como hemos visto ha quedado descrita una metodología de trabajo que va a permitir realizar este mismo procedimiento en numerosos procesos industriales.

En nuestro caso particular y como aportación al conocimiento vamos a ahondar en los mantenimientos futuros utilizando lógicamente la predicción de valores a través del Modelo Ideal.

5 DESARROLLO. OBJETIVO ESPECÍFICO

5.1 Introducción a los Objetivos Específicos.

Tal y como hemos indicado en el punto 1.3, algunos de los objetivos específicos que podríamos obtener a través del Modelo Ideal de Producción son los siguientes:

- Generalización del Modelo a multitud de procesos industriales
- Mejora de la Eficiencia Energética.
- Generalización a cualquier Instalación Fotovoltaica.
- Creación de Algoritmo para la detección de Fallos con degradación en la producción previa a la detección del fallo por el Scada.
- Aplicación en la mejora del Mantenimiento.
- Mejora del Modelo de Negocio con la previsión de resultados económicos según el modelo Ideal de Producción.
- Mejora económica al obtener mayor producción.
- Anticipación a fallos.
- Validación de la detección de alarmas del sistema de monitorización de la Planta.
- Etc.

De todos ellos al ser el más complicado de obtener, nos vamos a centrar en la Mejora de la Eficiencia Energética, pero no por ello se deben dejar de obviar los demás, ya que con dicha mejora el resto de objetivos casi que son consecuencia directa de él. Muchas son las mejoras a aportar con este apartado especialmente porque como se introdujo la inversión de este tipo de instalación es enorme, por lo que cualquier mejora por pequeña que sea supone una cantidad económica realmente considerable. Más interesante es si cabe, que la metodología utilizada en el presente trabajo para la obtención de conclusiones es generalizable a cualquier otro proceso industrial que se pueda asemejar. De hecho en el apartado 3.1 aportamos un ejemplo numérico de otro proceso industrial totalmente diferente al estudiado, lo que da idea de la magnitud de aplicaciones que podemos abordar. Es un campo apasionante el que se abre como línea de investigación futura.

Centrándonos en concreto, esta segunda fase del Proyecto es la que supone una aportación al conocimiento utilizando el Modelo Ideal de Producción de una Instalación Fotovoltaica para mejorar la Eficiencia Energética con la detección temprana de fallos.

La idea general que vamos a desarrollar y utilizar es la siguiente: vamos a comparar la producción real generada por la Instalación Fotovoltaica con la producción ideal que hemos obtenido a través del Modelo Ideal de Producción. A través de dicha comparación vamos a crear mecanismos para que diferencias significativas entre ambas producciones nos generen alarmas de modo que llegado a cierto umbral tenga que actuar el departamento de mantenimiento lo antes posible con objeto de subsanar el fallo existente para que la producción vuelva a la normalidad. Si todo fallo detectado con esta metodología es igualmente detectado con el sistema de monitorización de la Instalación Fotovoltaica el trabajo realmente no tendrá validez a parte de servir para verificar el correcto funcionamiento del sistema de monitorización. Lógicamente no es lo que se pretende, sino que el objetivo claro es la mejora de la producción y ello lo conseguiríamos a través de la detección temprana del fallo con la caída en la producción previa a la propia alarma generada por el Scada.

Para introducir un poco más este concepto de modo que se visualice perfectamente, lo que hacemos es representar gráficamente en los mismos ejes la producción real y la producción ideal, de modo que separaciones significativas entre ambas líneas nos darán idea de la existencia de algún fallo.

Como ejemplo el siguiente gráfico denota dicha comparación pudiendo observar como a finales de Septiembre existe una diferencia significativa entre ambas producciones. En este gráfico comparamos la producción diaria acumulada, pudiendo observar como la predicción (Modelo Ideal de Producción) es muy similar a la producción real (es lo que ocurre en la normalidad, lo que igualmente valida el modelo) salvo a finales de Septiembre, lo que nos hace pensar que puede existir alguna anomalía. Dicho caso lo ampliaremos a continuación, simplemente sirva ahora como idea del desarrollo que se va a exponer.

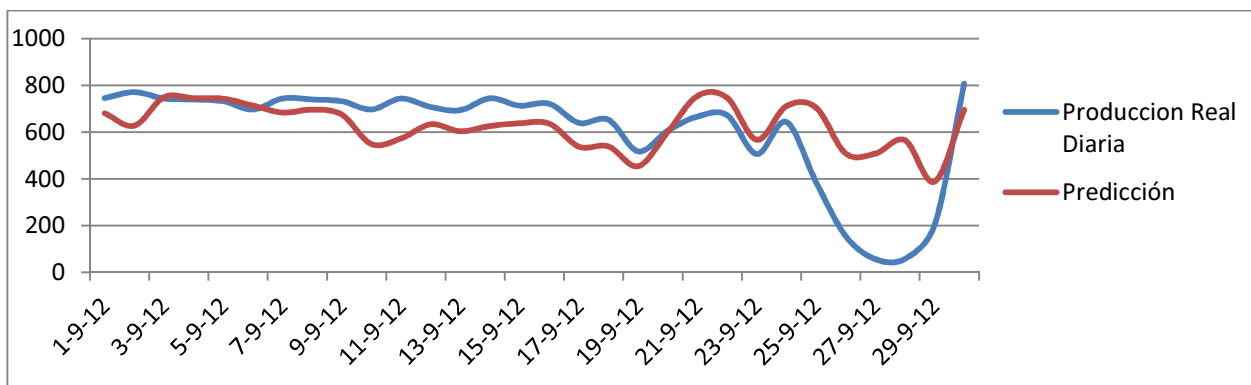


Figura 1-32. Gráfico comparación producción Real frente a Predicción.

El Scada que monitoriza la planta, indica mediante alarmas al operador de mantenimiento que está visualizando la pantalla que existe un fallo en un determinado componente. Lógicamente ante dicho aviso el operador dará la señal de alarma para que el personal de mantenimiento en campo actúe lo antes posible con objeto de volver a restablecer el funcionamiento normal de la Instalación, puesto que cualquier parada supone pérdida de producción y por tanto pérdida económica. Este sería el denominado mantenimiento correctivo, el cual aplica tras la detección de un fallo que como su nombre indica es reactivo ante cualquier incidente lo que no supone mejora alguna en la anticipación y por tanto en la eficiencia.

Muchos de los fallos existentes en cualquier instalación industrial son fallos que se producen en el instante por lo que ante dicho error no existe forma posible de anticipación, por lo que ante ellos la mejor solución es resolverlo lo antes posible valorando las consecuencias coste – parada. Implica que la parada para la reparación puede ser posible que se realice en horas de no producción. Por tanto el mantenimiento correctivo es un mantenimiento que como tal no puede omitirse sino que con el presente trabajo incluimos un complemento para llegar a evitar que se produzcan pérdidas o paradas innecesarias con la detección de ciertas alarmas que vamos a definir posteriormente y que supondrán la anticipación al fallo.

Con el presente proyecto pretendemos anticipar el fallo de modo que no llegue a producirse. Para ello vamos a introducir en los siguientes apartados una serie de conceptos necesarios para llegar a tener claramente definida una condición de alarma.

5.2 Algoritmo para la Detección de Fallos.

En primer lugar debemos identificar claramente el tipo de fallo objeto de estudio y que no es otro que aquél fallo que produzca una degradación progresiva en la producción que nos permita detectar con la suficiente antelación el hecho de que el fallo se va a producir. De por sí, esta es una cuestión nada trivial puesto que la mayoría de los fallos producidos en la planta son fallos que se producen en el instante concreto generando una alarma, ante la cual no queda otra solución que repararlo de inmediato. Por tanto, al igual que en el apartado de “tratamiento de los datos” es necesario estudiar con detenimiento todo el histórico de fallos para categorizarlos y de este modo seleccionar aquéllos susceptibles del estudio.

Para poder definir el algoritmo o procedimiento a utilizar vamos a introducir inicialmente el concepto de Impacto. Con el impacto medimos los elementos a los que afecta el fallo, o lo que es lo mismo traducido a una instalación fotovoltaica: kw que se dejan de producir en función de los elementos afectados.

Con la siguiente tabla categorizamos los impactos posibles en la Instalación Fotovoltaica objeto de Estudio

Impacto	Afección
5	No existe afección como tal, se produce al existir fallos de comunicaciones en momentos puntuales por lo que no se aporta información. La planta sigue funcionando a pleno rendimiento.
4	Un panel fotovoltaico
3	Un seguidor
2	Un conjunto de seguidores (< 50 Kw)
1	Afecta al menos a un inversor. Implica pérdidas de producción superiores a 100 Kw

Tabla 1-22. Definición del Impacto

Una vez definido el concepto de impacto a continuación exponemos el algoritmo utilizado para la detección de los fallos objeto de estudio:

- 1.- Selección de Fallos de la Instalación Fotovoltaica.
- 2.- Categorización de Fallos en función de la afección o impacto.
- 3.- Seleccionar aquellos fallos que afecten a la producción de al menos un huerto solar. Es decir, aquellos con impacto 1, tal y como hemos definido.
- 4.- De dichos fallos (con el apoyo de los operadores expertos en mantenimiento) seleccionar aquellos que pueden ser susceptibles del estudio. Es decir, aquéllos que pudieran permitir su detección temprana con la caída de la producción en días previos a que se produzca.
- 5.- De dichos fallos seleccionar inicialmente aquéllos de los que dispongamos de una muestra suficientemente representativa. Es decir, que se hayan producido en más de un huerto Solar de diferentes Centros de Transformación al que vierten y que se haya repetido en varias ocasiones.
- 6.- Seleccionar uno de los fallos anteriores y localizar el Huerto Solar en el que se ha producido.
- 7.- Seleccionar la fecha concreta del fallo.
- 8.- Ver la producción el día de fallo, los días anteriores y los días posteriores.

- 9.- Si la pérdida de producción sólo se produce el día del fallo volver al paso 6. En este caso no sería posible anticiparnos al fallo.
- 10.- Repetir los pasos 6 a 9 hasta localizar un fallo en el que exista pérdida de producción los días previos a la detección del fallo (por parte del Scada).
- 11.- De dicho fallo seleccionado volver a localizar todos los de dicha categoría. Obtener una muestra de al menos tres fallos en diferentes centros de transformación. Si el fallo es puntual y sólo tenemos un fallo no es representativo para el estudio, por lo que en este caso volver al paso 6.
- 12.- Verificar que en dicha muestra, el comportamiento de los fallos es el mismo. Es decir, en todos existe una caída de producción los días previos a la detección de la alarma por el Scada.
- 13.- Categorizar el fallo (utilizaremos uno como modelo en el Proyecto).
- 14.- Definimos una condición de alarma. Será genérica y la misma para todos los fallos .
- 15.- Utilizamos el modelo ideal de Produccion obtenido para predecir en las fechas del fallo con la red entrenada previamente y comparamos con el modelo Real.
- 16.- Verificamos que la condición de alarma se produce con el fallo seleccionado días previos a la detección de la alarma por el Scada.
- 17.- Activamos el Mantenimiento.
- 18.- Calculamos el ahorro económico por dicha anticipación.

Este es el algoritmo a utilizar y que vamos a aplicar en un caso concreto. Para ello los pasos que vamos a seguir son los siguientes:

- Definición de Condición e Alarma. Puesto que esta condición de alarma la vamos a utilizar en el algoritmo para cualquier tipo de fallo seleccionado, por tanto la definimos a priori para utilizarla posteriormente.
- Aplicación del algoritmo.
 - o Categorización de fallos en función de su Impacto.
 - o Selección de aquellos que afecten al menos a un Huerto Solar. Es decir, impacto 1.
 - o Aplicación de un caso concreto de fallo ya seleccionado.
 - o Comparación del modelo Ideal de Produccion con el Modelo Real.
 - o Conclusiones.

5.2.1 Priorización para la condición de Alarma.

El objetivo de este apartado es el de introducir una serie de conceptos que vamos a aplicar en nuestro sistema que serán la base para poder tener definida claramente la condición de alarma a implantar en nuestro sistema.

En cualquier sistema en el que existan posibles incidencias o averías es muy frecuente el tener definidos dichos conceptos de modo que se puedan asignar prioridades. Lo que pretendemos es poder parametrizar en nuestro sistema estos conceptos para que la detección posterior en el Scada se produzca de modo automático.

5.2.1.1 Definición de Severidad

Vamos a definir como severidad la importancia del daño causado, que en nuestro caso lo catalogaremos como aquella pérdida de producción de al menos el 40%. Este valor junto con el número de horas consideradas para catalogar como alta la severidad, no son elegidos al azar, sino que conlleva los siguientes matices:

- La red neuronal simula patrones de comportamiento, lo que implica cierta variación en la predicción de valores concretos. Tenemos que considerar por tanto comportamientos o producciones generales de ahí que sea un valor importante el considerado.
- Al simular el comportamiento en un rango horario de 8:00 a 17:00 implica importantes variaciones de producción a lo largo del día. La producción a primera hora es muy pequeña, a veces de 1 sólo Kw, por lo que una diferencia de 1 Kw implicaría duplicar la producción. Es por ello que es necesario definir al menos 4 horas consecutivas de pérdidas superiores al 40% para que la severidad sea alta, ya que ésta es la que será utilizada en la condición de alarma que definiremos.
- El activar un departamento de mantenimiento que repare implica considerables costes, por lo que para activarlo debe existir cierta seguridad en el fallo para no incurrir en gastos innecesarios.

A partir de estos conceptos definimos la severidad alta, media o baja según la siguiente tabla:

Severidad		
<i>Alta</i>	<i>Media</i>	<i>Baja</i>
Más de cuatro horas de pérdidas de producción de al menos el 40% de modo consecutivo	Más de 2 horas de pérdidas de producción de al menos el 40% de modo consecutivo	Más de 1 hora de pérdida de producción de al menos el 40%.

Tabla 2-23. Definición del Severidad.

5.2.1.2 Definición de Criticidad

El siguiente concepto a introducir es la criticidad, que como su propio nombre indica lo crítico del elemento afectado en el fallo o avería. En la siguiente tabla definimos las diferentes criticidades:

Criticidad	<i>Alta</i>	La caída de producción afecta a un inversor.
	<i>Media</i>	La caída de producción afecta a un seguidor
	<i>Baja</i>	La caída de producción afecta a un panel

Tabla 3-24. Definición de Criticidad.

5.2.1.3 Prioridad

Finalmente con el concepto de prioridad cruzamos los conceptos anteriores, puesto que lo que vamos buscando son los fallos que afecten a la producción de al menos un huerto (impacto 1) solar en un período mínimo de 4 horas consecutivas.

PRIORIDAD		Severidad		
		<i>Alta</i>	<i>Media</i>	<i>Baja</i>
Criticidad	<i>Alta</i>	1	2	3
	<i>Media</i>	2	3	4
	<i>Baja</i>	3	4	5

Tabla 4-25. Definición de Prioridad.

La prioridad como es lógico lleva implícita la importancia de la avería a resolver y por tanto tanto los tiempos de respuesta como los tiempos para la resolución de la avería.

Definimos tiempo de respuesta aquél tiempo que dispone el Departamento de Mantenimiento para acudir a la planta a reparar. Tiempo a contar a partir del aviso.

Por tiempo de resolución de la avería entendemos aquél tiempo máximo para reparar a contar desde la llegada a la instalación. Para cumplir con dicho tiempo como es lógico se necesita disponer de material en stockage.

Los tiempos definidos tanto de respuesta como de reparación en función de la prioridad son los que se indican en la siguiente tabla:

<u>PRIORIDAD</u>	Tiempo de Respuesta	Tiempo de Resolución
1	1	8
2	2	8
3	3	12
4	4	12
5		

Tabla 5-26. Prioridad – Tiempos de Respuesta y Resolución

5.2.2 Condición de Alarma

Para poder automatizar en algún software debemos definir parámetros que estén relacionados con los anteriores pero que sean fácilmente implementables pudiendo comparar valores. Para ello realizamos lo siguiente:

- A tiempo Real comparamos el valor de la producción horaria predecida con el modelo Ideal de Producción (en ese momento los valores de las variables de entrada son conocidos pues quedan recogidos en el Scada) con el valor de producción real de cada uno de los 61 Huertos Solares que posee la instalación objeto del estudio. Son muchos valores pero fácilmente automatizable e implementable en un software sin requerir apenas tiempo de computación.
- Si el valor de la producción real es un 40% inferior al valor de la predicción ponemos el valor 1 en la condición de alarma.
- El valor anterior se incrementa de modo consecutivo si la condición anterior se mantiene en la siguiente hora. Si la condición no se cumple el contador vuelve a 0.
- El valor de la condición de alarma va directamente relacionado con la prioridad, de modo que conocemos por tanto los tiempos de respuesta y resolución en función de la alarma.
- Si la condición de alarma pasa de 1, 2 o 3 a 0 entonces no actuamos por lo que desactivamos la prioridad dejándola de nuevo en ese caso en 0.
- Si la condición de alarma llega al valor 4 entonces se activa prioridad 1 y se actúa. Llegado a dicho valor lo más habitual es que el incremental de la alarma continúe sumando.

La siguiente tabla recoge los diferentes valores que recoge la condición de alarma:

Condición de Alarma		
1	2	<i>Xi horas</i>
La producción de un huerto de 100 kw es inferior un 40% de la predicción del modelo en el periodo de una hora	La producción de un huerto de 100 kw es inferior un 40% de la predicción del modelo en el periodo de dos horas consecutivas	La producción de un huerto de 100 kw es inferior un 40% de la predicción del modelo en el periodo de Xi horas consecutivas

Tabla 6-27. Definición Condición de Alarma

En cuanto a la relación entre condición de alarma y prioridad es la siguiente, que tal y como hemos indicado nos aporta los tiempos de respuesta y de reparación.

<u>PRIORIDAD</u>	Condición de Alarma	Tiempo para Actuar	Tiempo para Reparar
1	4	1	8
2	3	2	8
3	2	3	12
	1		
	0		

Tabla 7-28. Relación Prioridad – Condición de Alarma - Tiempos

La condición de alarma viene determinada por afección a un Huerto Solar, lo que implica que la criticidad es alta y ello conlleva que la prioridad solo puede tomar los valores, 1, 2 o 3.

La condición de alarma es la que se implementa en el sistema, de modo que es el motor que activa la prioridad. Activada la prioridad 1, implica que la condición de alarma ha llegado a 4 por lo que automáticamente se deducen tiempos de respuesta y tiempos de actuación. En esta situación es primordial la rapidez en la reparación.

5.3 Fallo Modelo para Estudio

Para seleccionar el Fallo que vamos a estudiar y al que le aplicaremos todos los conceptos explicados anteriormente, vamos a utilizar el algoritmo descrito en el apartado 5.2. Por tanto lo aplicamos tal cual:

- 1.- Selección de Fallos de la Instalación Fotovoltaica.
- 2.- Categorización de Fallos en función de la afección o impacto.

La siguiente tabla muestra los dos primeros pasos, centrados sólo en un centro de transformación pues es muy genérico del resto de los de la Instalación. En concreto el Centro de Transformación número 15.

FALLO	IMPACTO
<i>Alarma viento</i>	2
<i>Defecto de tierra2</i>	1
<i>E. Corto C</i>	1
<i>E.Compensacion CC</i>	1
<i>Emergencia</i>	3
<i>Fallo Alimentación Huerto</i>	1
<i>Fallo Aislamiento</i>	1
<i>Fallo Aislamiento</i>	1
<i>Fallo Aislamiento</i>	1
<i>Fallo Aislamiento</i>	1
<i>Fallo Aislamiento</i>	1
<i>Fallo Aislamiento</i>	1
<i>Fallo Aislamiento</i>	1
<i>Fallo Aislamiento</i>	1
<i>Fallo comunicación</i>	5
<i>Fallo final carrera Este</i>	3
<i>Fallo final carrera inferior</i>	3
<i>Fallo final carrera Oeste</i>	3
<i>Fallo final carrera Superior</i>	3
<i>Fallo maniobra altura</i>	3
<i>Fallo maniobra azimut</i>	3
<i>Fallo variador altura</i>	3
<i>Fallo variador azimut</i>	3
<i>Inv. Sobrecargado</i>	1
<i>Límite corriente línea</i>	1
<i>Sobrecorriente</i>	1
<i>Sobrefrecuencia</i>	1
<i>Sobrevoltaje</i>	1
<i>Temp. Disipador muy alta</i>	1
<i>Temp. Disipador muy baja</i>	1
<i>Temp. Interna alta</i>	1
<i>Temp. Interna baja</i>	1
<i>Voltaje electrónico 15v alto</i>	1
<i>Voltaje electrónico 15v bajo</i>	1

Tabla 8-29. Tipos de Fallo por Impacto

Una vez categorizados los fallos continuamos con el algoritmo. Puesto que dicho algoritmo es tedioso por todos los pasos y la repetición hasta la obtención de uno que sea susceptible de estudio por cumplir con todos los parámetros, se ha realizado este estudio con detalle hasta conseguir el fallo seleccionado. Es por ello que realizados todos los pasos, vamos directamente al paso 12 en el que mostramos el resultado tras haber pasado todos los filtros anteriores.

- 3.- Seleccionar aquellos fallos que afecten a la producción de al menos un huerto solar. Es decir, aquellos con impacto 1, tal y como hemos definido.
- 4.- De dichos fallos (con el apoyo de los operadores expertos en mantenimiento) seleccionar aquellos que pueden ser susceptibles del estudio. Es decir, aquéllos que pudieran permitir su detección temprana con la caída de la producción en días previos a que se produzca.
- 5.- De dichos fallos seleccionar inicialmente aquéllos de los que dispongamos de una muestra suficientemente representativa. Es decir, que se hayan producido en más de un huerto Solar de diferentes Centros de Transformación al que vierten y que se haya repetido en varias ocasiones.
- 6.- Seleccionar uno de los fallos anteriores y localizar el Huerto Solar en el que se ha producido.
- 7.- Seleccionar la fecha concreta del fallo.
- 8.- Ver la producción el día de fallo, los días anteriores y los días posteriores.
- 9.- Si la pérdida de producción sólo se produce el día del fallo volver al paso 6. En este caso no sería posible anticiparnos al fallo.
- 10.- Repetir los pasos 6 a 9 hasta localizar un fallo en el que exista pérdida de producción los días previos a la detección del fallo (por parte del Scada).
- 11.- De dicho fallo seleccionado volver a localizar todos los de dicha categoría. Obtener una muestra de al menos tres fallos en diferentes centros de transformación. Si el fallo es puntual y sólo tenemos un fallo no es representativo para el estudio, por lo que en este caso volver al paso 6.
- 12.- Verificar que en dicha muestra, el comportamiento de los fallos es el mismo. Es decir, en todos existe una caída de producción los días previos a la detección de la alarma por el Scada.

Una vez seleccionado el fallo (impacto 1), verificado el comportamiento análogo de todos ellos y centrados en un huerto solar concreto exponemos la muestra de los tres fallos. En dichos fallos cumplimos con todas las condiciones, de modo que existe pérdida de producción los días previos a la detección del fallo por parte del sistema de monitorización de la planta. A continuación exponemos varias tablas con el detalle de todo ello.

La siguiente tabla contiene el fallo seleccionado, así como la muestra de fallos en el Centro de Transformación 15. Dichos fallos son todos de impacto 1, y en todos ellos se ha verificado un comportamiento similar en la detección de las caídas de producción los días previos a la detección del fallo por el sistema de monitorización de la planta, por lo que es un fallo candidato a ser detectado por la Red Neuronal entrenada.

FECHA	HORA	ESTADO	AGRUPACIÓN	SCADA	SEGUIDOR	Fallo_Aislamiento	IMPACTO
31/10/2012	9:59:02	aparecido	CT15	Alarma	INV_1	Fallo_Aislamiento	1
26/10/2012	9:47:04	aparecido	CT15	Alarma	INV_3	Fallo_Aislamiento	1
02/10/2012	17:20:45	aparecido	CT15	Alarma	INV_4	Fallo_Aislamiento	1
04/12/2012	17:03:21	aparecido	CT15	Alarma	INV_4	Fallo_Aislamiento	1
27/09/2012	19:48:16	aparecido	CT15	Alarma	INV_5	Fallo_Aislamiento	1
20/10/2012	9:06:16	aparecido	CT15	Alarma	INV_5	Fallo_Aislamiento	1
26/10/2012	19:23:06	aparecido	CT15	Alarma	INV_5	Fallo_Aislamiento	1

Tabla 9-30. Fallos Seleccionados para el estudio.

Con el fallo seleccionado, es decir, el “Fallo de Aislamiento” y visto su comportamiento en todos ellos, nos centramos en el Inversor 5 al producirse en 3 ocasiones. Vamos por tanto a verificar la producción los días previos y posteriores al fallo, de modo que verifiquemos el paso 9 del algoritmo (9.- Si la pérdida de producción sólo se produce el día del fallo volver al paso 6. En este caso no sería posible anticiparnos al fallo). La siguiente tabla muestra la producción del inversor 5 del Centro de Transformación 15 en las fechas anteriores y posteriores a la detección del fallo por parte del Scada:

Fecha	AGRUPACION	Inversor	Produccion Diaria	Deteccion Fallo Scada	Detección Fallo RNA
24/09/2012	CT 15	5	757,325001		
25/09/2012	CT 15	5	397,3333338	NO	SI.- Caída Producción
26/09/2012	CT 15	5	249,2749997	NO	Anticipación al fallo.
27/09/2012	CT 15	5	61,05000027	SI.- Fallo Aislamiento	
28/09/2012	CT 15	5	61,19166672		
29/09/2012	CT 15	5	284,3750006		
30/09/2012	CT 15	5	925,4833326		
17/10/2012	CT 15	5	863,4749999		
18/10/2012	CT 15	5	481,9083322	NO	SI.- Caída Producción
19/10/2012	CT 15	5	264,4833316	NO	Anticipación al fallo
20/10/2012	CT 15	5	487,5499998	SI.- Fallo Aislamiento	
21/10/2012	CT 15	5	533,2750008		
22/10/2012	CT 15	5	307,399999		
23/10/2012	CT 15	5	769,4999989		
24/10/2012	CT 15	5	87,35000004	NO	SI.- Caída Producción
25/10/2012	CT 15	5	87,1833335	NO	Anticipación al fallo
26/10/2012	CT 15	5	182,7999991	SI.- Fallo Aislamiento	
27/10/2012	CT 15	5	367,5666672		
28/10/2012	CT 15	5	815,5749971		

Tabla 10-31. Producción en días previos al fallo por el Scada.

Del mismo modo que la tabla anterior en la que mostramos el detalle, adjuntamos gráficos de producción diaria de los inversores 1 y 3 para verificar que el comportamiento de este tipo de fallo es análogo, es decir, con la RNA nos anticipamos al fallo pues detecta la caída en la producción algo que no sucede con el Scada:

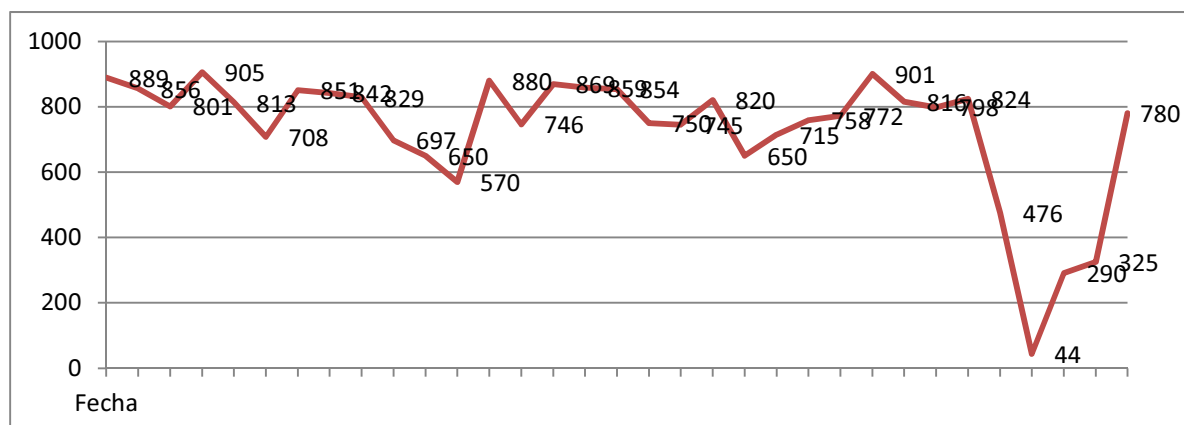


Figura 2-33. Gráfico de Producción diaria Inversor 1. Muestra de Fallo.

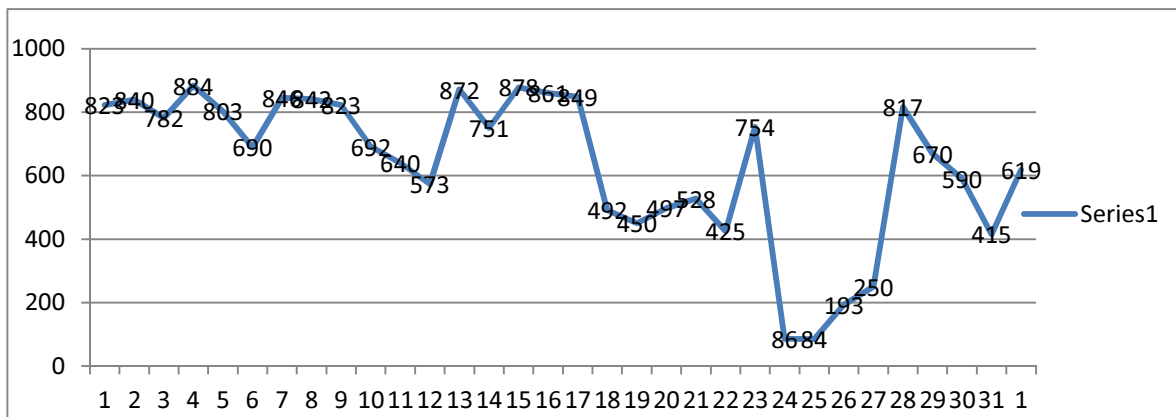


Figura 3-34. Gráfico de Producción diaria de Inversor 3. Muestra de Fallo.

- 13.- Categorizar el fallo (utilizaremos uno como modelo en el Proyecto).

Estamos por tanto en condiciones de categorizar el fallo, de modo que un Fallo de Aislamiento implicará una caída o degradación en la producción durante al menos 4 días, detectándose por el Sistema de Monitorización de la Planta a los dos días de producirse el inicio de la degradación y detectándolo la RNA a las 4 horas de producirse. Sin embargo una detección de caída de la producción por la RNA no implica que sea un fallo de aislamiento.

Es un fallo importante puesto que queda categorizado como de Impacto 1 y su duración es de al menos 4 días, lo que implica una importante pérdida de producción.

- 14.- Definimos una condición de alarma. Será genérica y la misma para todos los fallos.
- 15.- Utilizamos el modelo ideal de Produccion obtenido para predecir en las fechas del fallo con la red entrenada previamente y comparamos con el modelo Real.
- 16.- Verificamos que la condición de alarma se produce con el fallo seleccionado días previos a la detección de la alarma por el Scada.
- 17.- Activamos el Mantenimiento.
- 18.- Calculamos el ahorro económico por dicha anticipación.

Hasta ahora hemos aplicado un algoritmo que nos ha permitido detectar un fallo de impacto 1 en la planta, el cual produce una degradación en la producción que no es detectada por el sistema de monitorización hasta al menos dos días de su inicio en la degradación. No hemos utilizado hasta el momento sin embargo el modelo ideal de producción obtenido con la red Backpropagation, aunque el hallazgo es realmente importante porque nos permite aplicar el algoritmo y por tanto darle validez. En este sentido y haciendo referencia ya al paso 15 del Algoritmo es cuando cobra sentido la aplicación del modelo ideal de producción en la mejora de la eficiencia energética de la Instalación Fotovoltaica. Insistir en la importancia de este hecho, puesto que de no localizar o no existir ningún fallo en la Instalación Fotovoltaica cuya degradación fuese previa a la alarma que produce el Scada (de impacto 1) no tendría sentido la utilización del modelo ideal de producción obtenido en los apartados anteriores para su utilización en todos los objetivos específicos.

El localizar este fallo con un historial tan grande y en instalaciones complejas no es nada sencillo y conlleva una gran dedicación en tiempo, pero sin embargo gracias a la colaboración de expertos en la materia y a su localización es lo que ha hecho que esta aportación al conocimiento tenga sentido y aplicación práctica, la cual pretendo seguir avanzando con objeto de que en poco tiempo se ponga en funcionamiento en la instalación objeto de estudio.

En cuanto a los siguientes pasos del algoritmo para finalizar y obtener conclusiones, como por ejemplo la mejora económica lo vamos a ir viendo en diferentes apartados.

5.4 Comparación Modelo Ideal – Modelo Real en el fallo seleccionado.

Tal y como hemos indicado continuamos en este apartado con los diferentes pasos del algoritmo, siendo en este apartado en el que realmente vemos la aplicación práctica del modelo ideal de producción obtenido para poder generar la condición de alarma.

El primero de los pasos con el que continuamos la aplicación es el paso 14, en el que se define la condición de alarma, la cuál es genérica y ha quedado por tanto definida previamente en el apartado 5.2.2. Recordamos los principales conceptos definidos:

La siguiente tabla recoge los diferentes valores que recoge la condición de alarma:

Condición de Alarma		
1	2	<i>Xi horas</i>
La producción de un huerto de 100 kw es inferior al 40% de la predicción del modelo en el periodo de una hora	La producción de un huerto de 100 kw es inferior al 40% de la predicción del modelo en el periodo de dos horas consecutivas	La producción de un huerto de 100 kw es inferior al 40% de la predicción del modelo en el periodo de Xi horas consecutivas

Tabla 11-32. Condición de Alarma.

En cuanto a la relación entre condición de alarma y prioridad es la siguiente, que tal y como hemos indicado nos aporta los tiempos de respuesta y de reparación.

<u>PRIORIDAD</u>	Condición de Alarma	Tiempo para Actuar	Tiempo para Reparar
1	4	1	8
2	3	2	8
3	2	3	12
	1		
	0		

Tabla 12-33. Relación Prioridad – Condición de Alarma - Tiempos.

Una vez llegado al paso 15 del algoritmo (15.- Utilizamos el modelo ideal de Produccion obtenido para predecir en las fechas del fallo con la red entrenada previamente y comparamos con el modelo Real), conocemos el fallo que vamos a utilizar para verificar la validez del modelo presentado e igualmente ha quedado definida perfectamente la condición de alarma así como su correspondiente relación con la prioridad que nos indica los tiempos de respuesta para resolver el problema. Nos vamos a centrar en aplicar el algoritmo para detectar el primero de los fallos definidos anteriormente y que recordamos, siendo el siguiente:

Fecha	AGRUPACION	Inversor	Produccion Diaria	Deteccion Fallo Scada	Detección Fallo RNA
24/09/2012	CT 15	5	757,325001		
25/09/2012	CT 15	5	397,3333338	NO	<i>SI.- Caída Producción</i>
26/09/2012	CT 15	5	249,2749997	NO	<i>Anticipación al fallo.</i>
27/09/2012	CT 15	5	61,05000027	<i>SI.- Fallo Aislamiento</i>	
28/09/2012	CT 15	5	61,19166672		
29/09/2012	CT 15	5	284,3750006		

Tabla 13-34. Fallo concreto seleccionado para el estudio.

¿Cómo utilizamos ahora el modelo Ideal de Producción? Conocemos que hemos tenido en el inversor 5 del Centro de Transformación 15 3 fallos de Aislamiento entre Septiembre del 2012 y Octubre del 2012. Dicho fallos es lógico que se produzca en la fecha señalada puesto que es la época de inicio de lluvias, las cuáles provocan la aparición de dicho fallo pues aunque existiendo anteriormente solo muestran su existencia ante dichas condiciones meteorológicas adversas.

Por tanto, con el Modelo entrenado obtenido en el apartado 4.3.7 realizamos la predicción de producción de los meses de Septiembre y Octubre completos para poder comparar. Para realizar predicciones y dado que además introducimos una variable que controla las horas de funcionamiento con objeto de que agregue la degradación de los componentes con el tiempo, es conveniente que el modelo ideal esté entrenado en las fechas más próximas posibles a la realización de la predicción futura lo que aporta más exactitud a los resultados obtenidos. En nuestro caso predecimos dos meses después de haber obtenido el modelo y obtenemos unos resultados muy buenos.

El código utilizado para la predicción en R es el que se muestra y explica a continuación:

```
save(redselvc,file="Redfinal.RData")
```

Con esta función guardamos el resultado de la red entrenada, con el nombre de Redfinal.RData, de modo que podemos utilizarla a continuación para poder realizar predicciones.

```
load(file = "Redfinal.RData")
```

Con esta función cargamos la red guardada anteriormente.

```
xprueba<- read.xlsx(file = "Entradas.xlsx",1)
```

Cargamos en una variable (matriz de valores) a través de una hoja de cálculo los valores en este caso de las variables de entrada para poder predecir posteriormente. Por tanto cargamos sin encabezado alguno pero de modo ordenado las variables de entrada de la red con los datos de Septiembre y Octubre del 2012. La siguiente tabla muestra como ejemplo el formato de los datos cargados que corresponden por columnas con las variables: temperatura exterior, radiación, temperatura interior del inversor y horas de funcionamiento:

298	95	308,40	15590
306	120	309,60	15591
314	890	311,30	15592
313	910	314,30	15593
316	920	310,20	15594
320	750	311,30	15595
319	850	312,70	15596
321	870	310,60	15597
322	910	312,40	15598
321	950	310,80	15599
296	15	310,00	15600
299	75	309,80	15601
307	526	309,80	15602
318	878	309,20	15603
321	911	308,10	15604
321	918	309,80	15605
322	924	310,50	15606
323	924	309,70	15607
323	798	309,00	15608

Tabla 14-35. Datos ejemplos cargados en Software.

```
for (j in 1:ncol(x))
{
  xprueba[j]<- (xprueba[,j]-minimos[j])/(maximos[j]-minimos[j])
}
```

Con esta función normalizamos todos los valores del mismo modo que lo hicimos con los valores para el entrenamiento, detalle que no se nos puede olvidar pues podríamos echar al traste todo el trabajo realizado por errores de este tipo.

```
pred<-predict(redselvc,xprueba)
```

Una vez cargados los datos y normalizados, con la función pred realizamos la predicción de producción correspondiente a los valores introducidos. Del mismo modo recordar que el resultado obtenido estará normalizado.

```
cbind(xprueba,100*pred)
```

Finalmente con la función cbind mostramos el resultado de la predicción el cual lo multiplicamos por 100 para deshacer la normalización que se hizo a inicio exactamente del mismo modo.

Una vez obtenidos todos los valores de la predicción y conocida la producción realizamos una sencilla comparación entre ambos valores horarios, de modo que la condición de alarma tal y como se ha definido salta o se incrementa en 1 si el valor de la producción real es un 40% inferior al valor de la predicción (5.2.2 Condición de alarma). Esta comparación es muy simple de implementar en cualquier sistema de monitorización para que se pueda automatizar y avisar al departamento de mantenimiento ante cualquier contingencia de este tipo.

La siguiente tabla muestra la comparación de los resultados horarios de producción con la predicción realizada con el modelo añadiendo como columnas la condición de alarma que está automatizada y el contador de modo que si llega a 4 se activa la prioridad 1 avisando al correspondiente departamento para actuar de inmediato cumpliendo con los tiempos definidos anteriormente de respuesta e intervención.

Fecha	Hora	Inversor	Temp Ext	Rad.	Temp Int	H. Func.	Pred.	Prod.	Alarma	Contador
1-9-12	8	5- CT 15	298	95	308,40	15590	15	9,36	0	0
1-9-12	9	5- CT 15	306	120	309,60	15591	15,2	61,83	0	0
1-9-12	10	5- CT 15	314	890	311,30	15592	84,2	93,31	0	0
1-9-12	11	5- CT 15	313	910	314,30	15593	86	97,13	0	0
1-9-12	12	5- CT 15	316	920	310,20	15594	85,4	77,45	0	0
1-9-12	13	5- CT 15	320	750	311,30	15595	69,6	82,52	0	0
1-9-12	14	5- CT 15	319	850	312,70	15596	78,7	78,69	0	0
1-9-12	15	5- CT 15	321	870	310,60	15597	79,5	87,87	0	0
1-9-12	16	5- CT 15	322	910	312,40	15598	82,1	82,24	0	0
1-9-12	17	5- CT 15	321	950	310,80	15599	84,6	74,79	0	0

Tabla 15-36. Predicción frente a Producción. Inclusión condición de alarma.

La idea sería implementar en la Instalación objeto de estudio los cálculos mostrados en la tabla anterior, de modo automático y en modo continuo para que caso de producirse alguna degradación en la producción se pueda resolver el fallo lo antes posible sin necesidad de esperar a que el sistema de monitorización muestre una alarma. Sería del mismo modo interesante el tener en una gráfica la línea de producción comparada con la producción obtenida del modelo ideal, puesto que visualmente nos da idea de lo explicado en los cálculos realizados.

Continuando con la comparación y para demostrar la anticipación al fallo conocido que se va a producir por la alarma que produce el Scada, nos centramos en la comparación en los días previos. Para ello inicialmente y como hemos indicado mostramos un gráfico que nos muestra la desviación existente entre las líneas de producción real e ideal a finales del mes de Septiembre del 2012, debido justamente al fallo que vamos a estudiar.

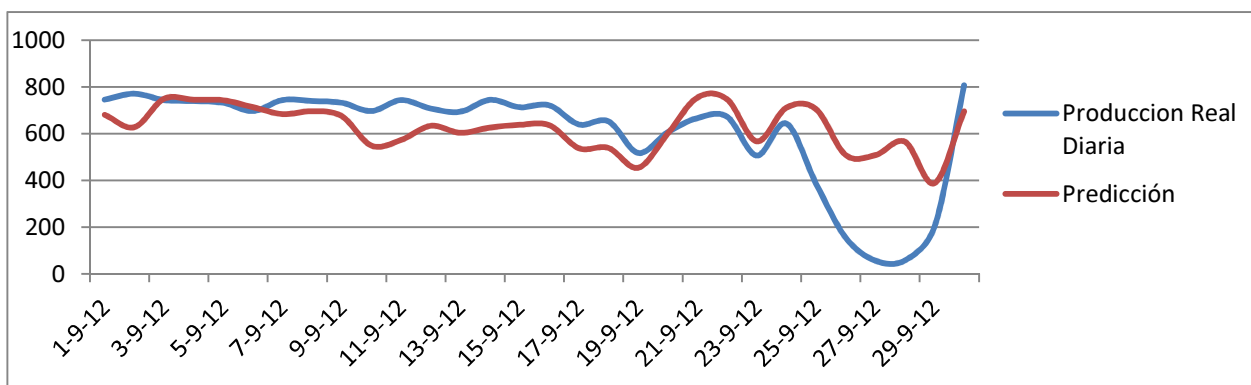


Figura 4-35. Gráfico Producción Real Frente a Predicción en mes de Fallo.

Realizando un “zoom” sobre el momento en el que se produce la desviación de ambas líneas obtenemos el siguiente gráfico detallado por hora:

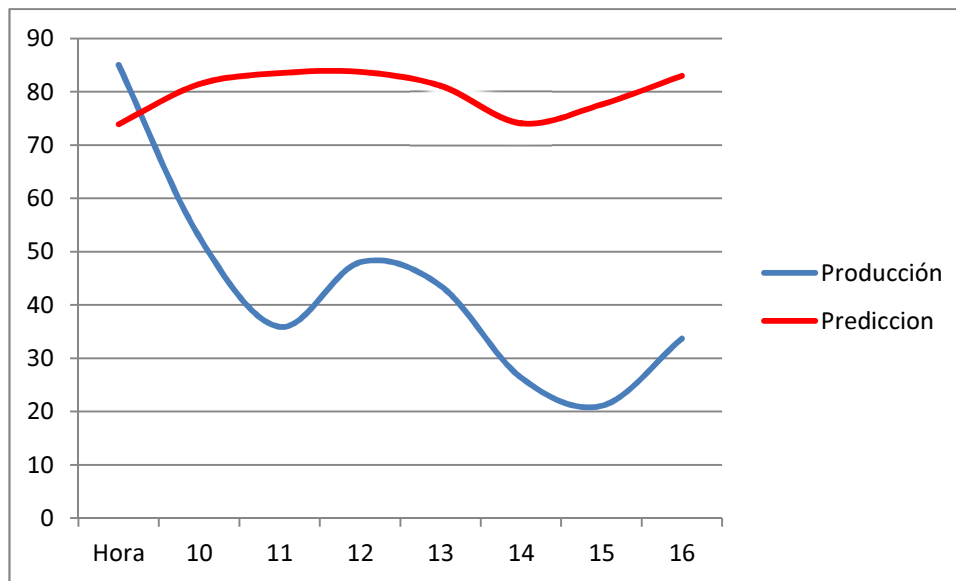


Figura 5-36. Gráfico Producción Real frente a Predicción en momento de detección del fallo por la RNA.

Las horas de dicho gráfico están referenciadas a la producción/predicción del 25 de Septiembre del 2012. Se observa claramente la desviación existente, la cuál e insistiendo hasta el día 27 de Septiembre no es detectada por el Scada de la planta. Entrando en el detalle a continuación exponemos los datos numéricos repitiendo la tabla anterior para observar como en esos momentos el contador de la alarma empieza a aumentar.

<u>Fecha</u>	<u>Hora</u>	<u>Producción</u>	<u>Predicción</u>	<u>Condicion de Alarma</u>	<u>Contador</u>
25-9-12	10	85,0333341	73,9	0	0
25-9-12	11	52,81666724	81,4	1	1
25-9-12	12	35,88333368	83,5	1	2
25-9-12	13	47,99999937	83,7	1	3
25-9-12	14	43,59166678	81,1	1	4
25-9-12	15	26,36666656	74,1	1	5
25-9-12	16	21,05833343	77,6	1	6
25-9-12	17	33,69999989	83	1	7

Tabla 16-37. Predicción frente a Producción. Activación en contador 4 de alarma.

Los datos anteriores implican por tanto que el día 25 de Septiembre a las 14 horas el contador de alarma llega a 4, activándose la prioridad 1, lo que implica actuación inmediata para reparar.

<u>PRIORIDAD</u>	<u>Condición de Alarma</u>	<u>Tiempo para Actuar</u>	<u>Tiempo para Reparar</u>
1	4	1	8

Tabla 17-38. Tiempos de resolución tras salto de alarma.

Todo ellos nos lleva a la conclusión de que ganamos desde las 14 horas del 25 de Septiembre hasta las 19 horas del 27 de Septiembre en la detección. Al estar definido el fallo se repara en 8 horas, por lo que ganamos igualmente dos días en la producción. Por tanto, según el modelo descrito perdemos sobre un 50% de la producción en un día, mientras que con la alarma clásica perdemos el 50% de 5 días, con lo que ganamos 4. Suponiendo que si un día se produce una media 700 Kw (depende de la época del año), con este método, por cada fallo de aislamiento en cada uno de los inversores de la planta obtenemos una ganancia de 1400 Kw de media en el mes de Septiembre. En función del precio de venta del Kw podríamos estar ganando por cada uno de esos fallos aproximadamente unos 600 euros. Extrapolando a una media de 3 fallos/año de este tipo en una planta con una vida media aún temprana de 4 años y a los 61 inversores del parque fotovoltaico estudiado supondría una ganancia de 256.200 Kw y un importe anual superior a los 110.000 €. O lo que es lo mismo un 1,8% más de producción y beneficio.

Indudablemente este resultado ratifica todo el esfuerzo realizado para la obtención del modelo por complejo que haya podido ser su desarrollo, lo interesante es que la implementación y automatización es relativamente simple en un software lo que puede suponer una mejora indudable en la Instalación objeto de estudio tal y como se ha indicado en la obtención de los objetivos específicos planteados.

6 RESULTADOS

6.1 Presentación de Resultados

En este apartado se presentan de modo general todos los resultados obtenidos en el presente trabajo, indicando la consecución tanto del objetivo general como de los objetivos específicos marcados a inicio del proyecto. Es decir, vamos a resumir los objetivos alcanzados y en la parte de las conclusiones incluimos la síntesis de dichos resultados referenciando datos concretos.

El Objetivo general del estudio es del predecir la producción de energía eléctrica de la Instalación Fotovoltaica, obteniendo un modelo ideal de producción. El modelo ideal de producción será obtenido utilizando como datos de entrada para el aprendizaje de la Red Neuronal Artificial seleccionada los datos históricos de producción de la planta.

Objetivo General: “*Obtención del Modelo Ideal de Producción*”.

Dicho objetivo se ha presentado en el apartado “4.4 Objetivo General: Modelo Ideal de Producción” y desarrollado en los anteriores apartados, cuyo resumen tanto de la metodología como del resultado obtenido con ella es el siguiente:

Técnica de Resolución: RNA

Red: Backpropatgation

Algoritmo de Minimización del ECM: Cuasi_Newton con Penalización.

Software: R

Iteraciones para el Entrenamiento: 1000

Valores para el Entrenamiento: 75% del conjunto

Valores para el Testeo: 25%

ECM: 75.21266

R2: 0.905068

Coefficiente de Penalización: 0

Variables de Entrada: 4

Variables de Salida: 1

Número de Células de la Capa Oculta: 10

Periodo de Estudio: Junio 2011 – Julio 2012

Horario de Produccion estudiado: desde las 8:00 hasta las 17:00

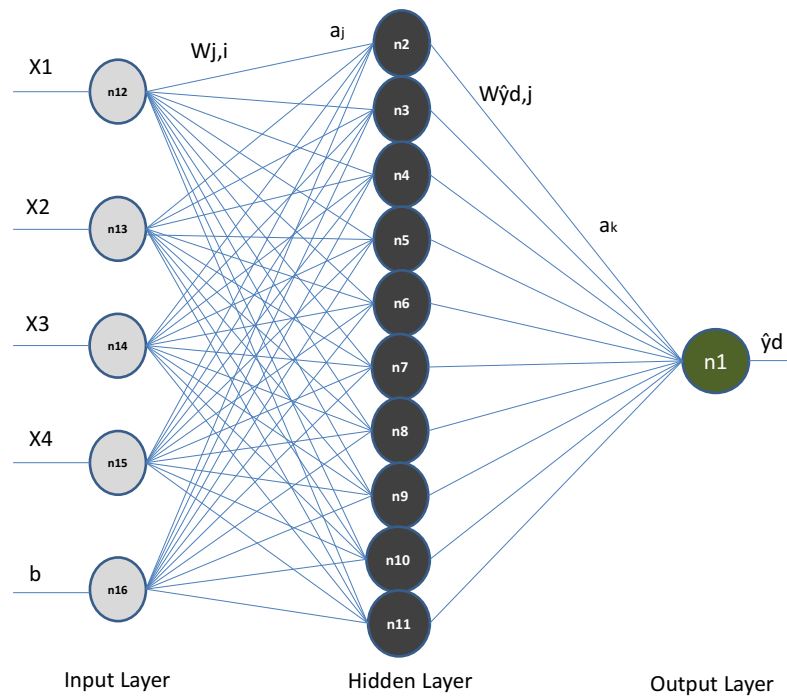


Figura 1-37. Arquitectura Seleccionada Red Óptima.

Se ha justificado la elección del modelo incluyendo como aspectos novedosos sobre los estudios existentes la inclusión de variables que aplican directamente sobre los activos como la temperatura interna del inversor y las horas de funcionamiento de la instalación. El resultado obtenido avala la elección al obtener buenos resultados lo que implica que la variación entre los valores deseados y los obtenidos queda explicada al 90,5% (R^2 : 0.905068).

A continuación especifico los resultados específicos que pretendíamos alcanzar justificando su consecución en cada uno de ellos. Los valores concretos de mejora se expondrán en el apartado 7.1

Comentamos cada uno de los objetivos específicos que hemos alcanzado según se expuso en el apartado 3.1 del presente trabajo.

Mejora de la Eficiencia Energética. La mejora de la eficiencia energética es el objetivo específico en el que nos hemos centrado en el Proyecto, pues de hecho ha sido la base para la consecución del resto. Para ello hemos utilizado el Modelo Ideal de Producción, el cuál nos ha servido como referencia de la producción ideal que deberíamos obtener en cada momento en función de las variables consideradas al ser un modelo en ausencia de fallo. Diferencias significativas o desviaciones entre ambos resultados (siempre la producción real por debajo de la ideal) en producción horaria de modo consecutivo alertan al departamento de mantenimiento pues es señal de fallo en algún componente, por lo que de este modo y previo al salto de alarma producido en el Scada nos anticipamos a dicho fallo obteniendo por ello la consiguiente mejora en la eficiencia energética de la planta. Para poder demostrar este hecho se ha definido un algoritmo (apartado 5.2), el cual nos ha permitido localizar un fallo (apartado 5.3) con degradación en la producción días previos a la detección del fallo por el Scada, el cual ha dado sentido a este apartado. De no encontrar fallos con estas características no hubiésemos obtenido ganancia alguna con el modelo ideal de producción, por lo que este hecho de por sí es el que hemos considerado más interesante siendo una nueva aportación al conocimiento.

Los dos siguientes objetivos específicos indicados, son los de Generalización a cualquier instalación fotovoltaica y generalización a multitud de procesos industriales, Para ello y aunque ha quedado definido, exponemos en el siguiente apartado (6.2) la metodología para su aplicación.

Para la consecución del objetivo inicial de Mejora de la Eficiencia Energética hemos tenido previamente que poder demostrar que efectivamente se puede conseguir con la detección de algún fallo cuya caída en la producción fuese previa (al menos en 24 horas) a la detección del fallo por el Scada. Por tanto, hemos conseguido un nuevo objetivo específico en este caso ya que de modo general hemos creado un “Algoritmo para la Detección de Fallos con degradación en la producción previa a la detección del fallo por el Scada”. Dicho algoritmo ha quedado definido en el apartado 5.2 y aplicado en el apartado 5.3 para su detección encontrando en este caso el Fallo de Aislamiento que es justamente el que hemos utilizado.

El siguiente objetivo específico indicado es el de la Aplicación en la Mejora del Mantenimiento. Puesto que se trata de mejorar la eficiencia energética de la Instalación no tendría sentido sin su aplicación práctica, por lo que hemos realizado un análisis de criticidad definiendo en los apartados 5.2.1 y 5.2.2 las condiciones de alarma y tiempos de respuesta para que sea aplicado en el correspondiente departamento de Mantenimiento suponiendo por ello la correspondiente mejora con la anticipación al fallo.

Otra aplicación práctica del Modelo Ideal de Producción por el hecho de poder predecir la producción futura es lo que nos aporta en el conocimiento sobre la ganancia económica que vamos a obtener. Nos permite por tanto una mejora nuestro modelo de negocio con el conocimiento previo del resultado económico ideal que debemos obtener.

Referente al mismo aspecto y a parte del modelo económico a futuro, el hecho de anticiparnos a ciertos fallos que provocan degradación sin ser detectadas por el Scada y si por nuestro modelo nos mejora la eficiencia energética de la instalación. Este hecho se traduce lógicamente en una mayor producción y por tanto en mayor beneficio económico tal y como se ha mostrado en el apartado 5.4.

Otro objetivo específico definido y que es implícito al modelo pero no por ello debemos obviarlo, es el hecho de la “anticipación de fallos” lo que supone una mejora en la planificación del departamento de mantenimiento de dichas instalaciones.

Como último objetivo específico definido el modelo ideal de producción sirve para verificar la eficacia del propio Scada, puesto que variaciones significativas (con antelación o no como es el caso) entre el modelo ideal de producción y la propia producción real denotan un fallo, su detección por nuestro modelo (si no existe degradación previa) debe coincidir con la alarma en el Scada. Por tanto es una doble validación y eficacia de ambos sistemas.

6.2 Generalización del Modelo.

Siguiendo el esquema de estudio realizado en el trabajo podemos aplicar la misma metodología a múltiples procesos industriales. Abstrayéndonos de la Instalación objeto de estudio la metodología de estudio es aplicable a cualquier otro proceso industrial que sea susceptible de aplicar modelos similares.

En el apartado 4.2 incluimos la siguiente metodología de trabajo:

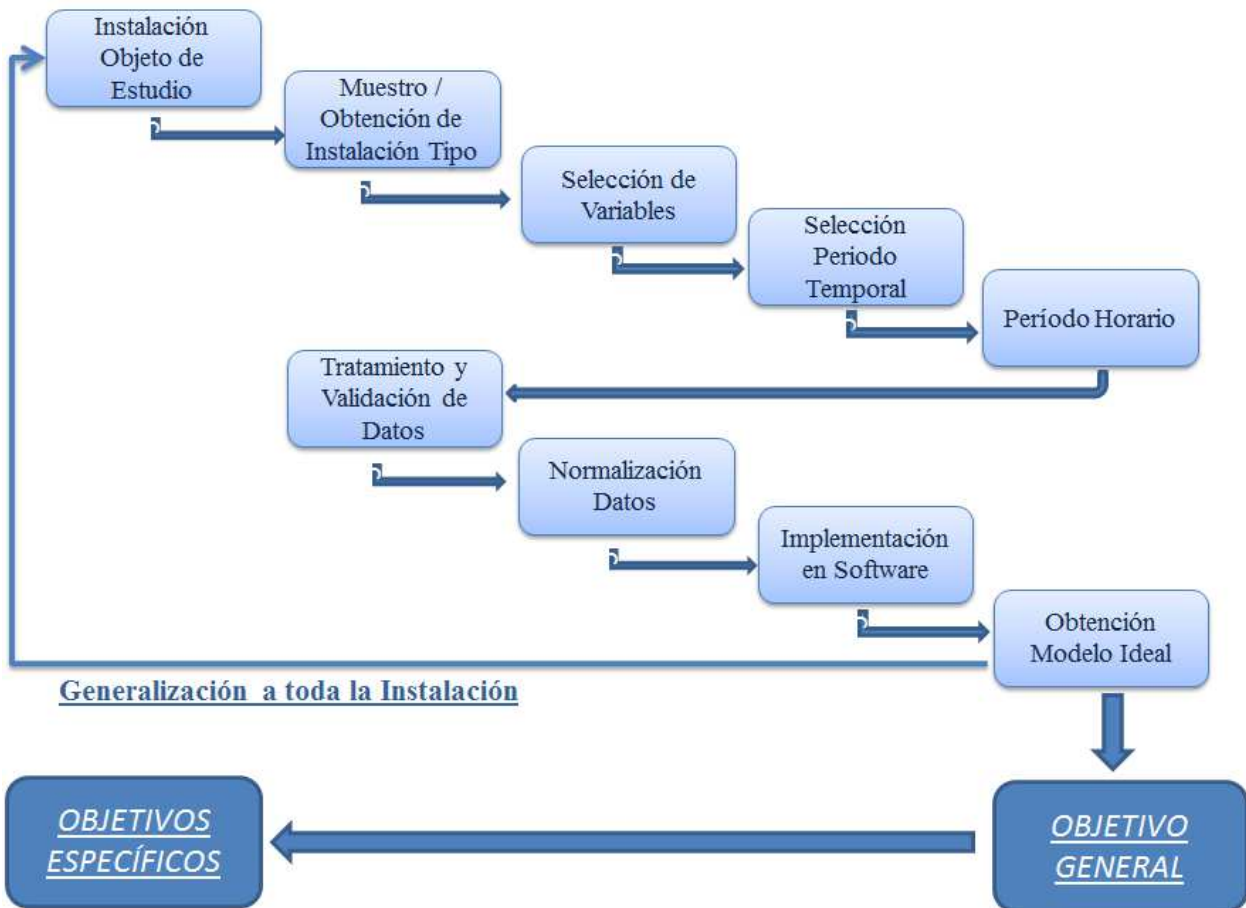


Figura 2-38. Metodología General.

En este momento si observamos el gráfico anterior que denota una metodología de trabajo y abstrayéndonos del trabajo objeto de estudio, en ningún caso se hace referencia a ningún tipo de instalación ni proceso industrial, Independientemente de ello lo hemos utilizado tal cual en el Proyecto para alcanzar los objetivos definidos, por lo que dicha metodología supone un método de trabajo genérico a aplicar en cualquier proceso industrial susceptible de aplicar. Para ello siempre es necesario contar con la colaboración de los expertos del proceso, con los que es preciso múltiples reuniones de seguimiento para definir cada uno de los pasos indicados en la figura anterior.

6.3 Discusión

Las Redes Neuronales Artificiales en general han provocado gran controversia en su utilización, lo que ha hecho dudar sobre la veracidad de los resultados obtenidos con ellas al no tener a veces fundamentación matemática que lo soportara. Bajo mi punto de vista la idea de simular el comportamiento del cerebro humano con algoritmos matemáticos es realmente interesante, y de hecho un mundo aún por descubrir, pues como hemos descrito a lo largo del documento problemas fácilmente resueltos por cualquier persona son tremendamente complicados de implementar o resolver mediante algoritmos matemáticos.

Bajo este prisma, me ha apasionado el estudio de estas técnicas que intentan simular ciertos patrones o comportamientos, los cuáles son capaces de aprender a través de la experiencia teniendo capacidad de generalización. Sin embargo, personalmente he de reconocer que también ha creado en mí ciertas susceptibilidades sobre su utilización y resultados, lo que me ha llevado a estudiar con más profundidad apasionado justamente por dicha incertidumbre.

Ahondando en la materia y al igual que el resurgir de las RNA, es en la técnica de propagación hacia atrás en la que he encontrado una potente herramienta de simulación de patrones la cual efectivamente mejoraba con el entrenamiento. No convencido incluso plenamente de ello he leído, ahondado en la materia y realizado numerosos ejercicios numéricos sin el apoyo de ordenador llegando en todos ellos a la misma conclusión: “realmente funciona”.

Bajo ciertas premisas y tras mi duda inicial, me ha apasionado su estudio reconociendo en las RNA un campo enorme de aplicación en la industria. Particularizando en el Proyecto y considerando que lo que pretendemos es predecir el comportamiento de algún proceso industrial hemos de considerar que con las RNA no podemos pretender conocer resultados exactos pues existen múltiples componentes y situaciones de contorno que en general imposibilitan dicho conocimiento. Esta situación se ha de conocer a priori por lo que quizás de por sí descartara su utilización para predicciones futuras cuya exactitud fuera necesaria. Estas situaciones se podrían dar en el menor de los casos posibles obteniendo siempre resultados por variables directamente relacionadas con la solución, por lo que dicha situación indica claramente la utilización de modelos paramétricos.

Como no es el caso de estudio y tras justificar su utilización me he decantado por la utilización de RNA para la resolución del problema planteado. No sólo la bibliografía ha refrendado el estudio realizado, sino que los resultados obtenidos han verificado su utilización siendo un método implantable para la mejora de la eficiencia (entre otros) de la producción en una instalación fotovoltaica.

Como puntos fundamentales a resaltar en el estudio y a los que se les debe prestar especial atención pues pueden llevarnos a conclusiones erróneas los siguientes:

- Dificultad general para la obtención de datos reales.
- Falta de calidad de los datos recibidos, los cuales sin un análisis detallado nos pueden llevar a resultados erróneos.
- Valoración esfuerzo coste-resultado.

Finalmente y aunque el modelo sea fácilmente implementable en cualquier sistema nos encontramos con la siguiente dificultad la cuál siempre me he planteado:

- ¿Cómo obtener la predicción de la producción en una determinada hora de cualquier fecha si no conozco las variables de entrada al depender de condiciones climatológicas?

En todo caso el modelo funciona y es válido, puesto que su implementación permite predecir recogiendo los datos de las variables de entrada con los sensores instalados, por lo que la comparación predicción-producción se puede realizar a tiempo real. Esto permite conocer las desviaciones y validar las condiciones de alarma definidas en el presente trabajo. Sin embargo a futuro ¿cuál de fiable es el dato de predicción concreto sin conocer las variables de entrada?.

Es una pregunta abierta que abre un nuevo campo de investigación.

7 CONCLUSIONES

7.1 Síntesis de las Conclusiones. Resultados

En la primera parte del trabajo nos hemos centrado (apartado 4) en la obtención del Objetivo General del Proyecto, cuyo resultado a modo resumen es el siguiente:

Técnica de Resolución: RNA

Red: Backpropatgation

Algoritmo de Minimización del ECM: Cuasi_Newton con Penalización.

Software: R

Iteraciones para el Entrenamiento: 1000

Valores para el Entrenamiento: 75% del conjunto

Valores para el Testeo: 25%

ECM: 75.21266

R2: 0.905068

Coefficiente de Penalización: 0

Variables de Entrada: 4

Variables de Salida: 1

Número de Células de la Capa Oculta: 10

Periodo de Estudio: Junio 2011 – Julio 2012

Horario de Produccion estudiado: desde las 8:00 hasta las 17:00

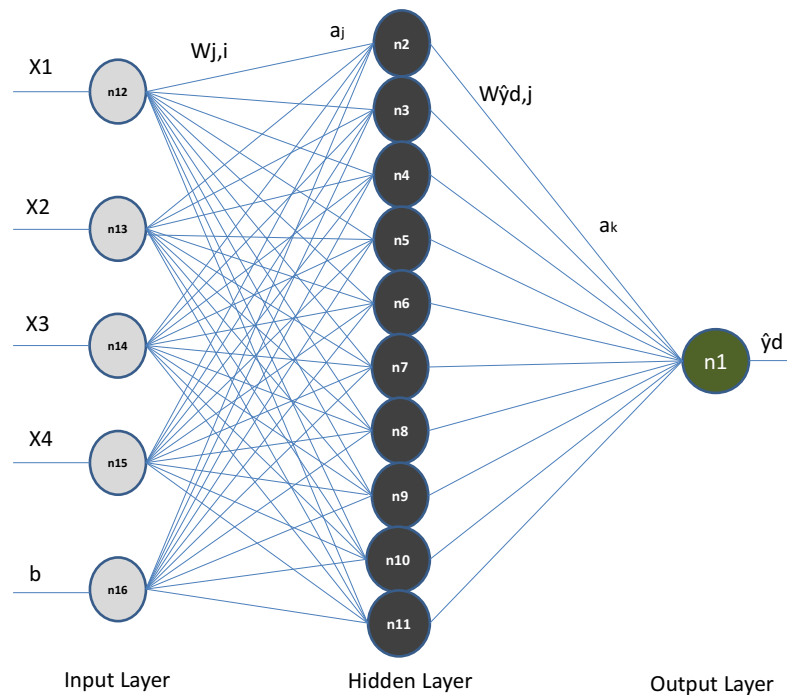


Figura 1-39. Arquitectura Seleccionada Red Óptima.

En la segunda parte del proyecto (apartado 5) nos hemos centrado en la obtención de los objetivos específicos, particularizando en la Mejora de la Eficiencia Energética de la Instalación Fotovoltaica pues de él se deducen todos los demás. Como resumen de ello lo siguiente:

- El fallo objeto de estudio por permitir alerta temprana es: fallo de aislamiento. Este tipo de fallos es de alerta temprana produciéndose al menos con 48 horas de antelación al salto de la alarma y una vez la alarma se ha activado entre su detección y reparación se mantiene el fallo al menos otras 36 horas.
- Dichos fallos igualmente vienen motivados porque una de las causas que lo motivan son la corrosión de ciertos materiales, por lo que con el tiempo al ser instalaciones a la intemperie dichos fallos tenderán a reproducirse. Implica que es un fallo muy representativo con el que poder obtener una mejora en la producción.
- Realizado un muestreo de los fallos ocurridos seleccionamos el Inversor 5 del Centro de Transformación 15 según la configuración de la instalación. Es análogo el comportamiento en cualquier otro inversor.
- El fallo seleccionado para el modelo ha sido un fallo que afecta a la producción de un inversor, o huerto de 100 Kw, de modo que las pérdidas de producción sean significativas y por tanto interesante implementar las medidas objeto de este trabajo. En el Scada son considerados fallos de prioridad o impacto 1.
- Condición de alarma: si la producción real en una determinada hora es inferior en un 40% a la producción del modelo ideal entonces se activa la condición de alarma con un contador inicialmente a 1.
- Si desaparece dicha condición en la siguiente hora el marcador se vuelve a 0. De lo contrario se incrementa en 1.
- Al existir posibles variaciones por la propia naturaleza del modelo obtenido con el entrenamiento de la red neuronal, no es necesario activar una intervención de mantenimiento con la alarma de señal 1, simplemente es un aviso o atención.

Mediante el Uso de Modelos Predictivos de Redes Neuronales Artificiales

- Si la condición se mantiene durante 4 horas consecutivas, la señal de alarma nos dará el aviso 4, lo que indica que programaremos una intervención de inmediato.
- Todo ello nos lleva a la conclusión de que ganamos (en un fallo modelo tipo) desde las 14 horas del 25 de Septiembre hasta las 19 horas del 27 de Septiembre en la detección. Al estar definido el fallo se repara en 8 horas, por lo que ganamos igualmente dos días en la producción. Por tanto, según el modelo descrito perdemos sobre un 50% de la producción en un día, mientras que con la alarma clásica perdemos el 50% de 5 días, con lo que ganamos 4. Suponiendo que un día se produce una media 700 Kw (depende de la época del año), con este método, por cada fallo de aislamiento en cada uno de los inversores de la planta obtenemos una ganancia de 1400 Kw de media. En función del precio de venta del Kw podríamos estar ganando por cada uno de esos fallos aproximadamente unos 600 euros. Extrapolando a una media de 3 fallos/año de este tipo por inversor en una planta con una vida media aún temprana de 4 años y a los 61 inversores del parque fotovoltaico estudiado supondría una ganancia de 256.200 Kw y un importe anual superior a los 110.000 €. O lo que es lo mismo un 1,8% más de producción y beneficio.
- Indudablemente este resultado ratifica todo el esfuerzo realizado para la obtención del modelo por complejo que haya podido ser su desarrollo, lo interesante es que la implementación y automatización es relativamente simple en un software lo que puede suponer una mejora indudable en la Instalación objeto de estudio tal y como se ha indicado en la obtención de los objetivos específicos planteados.

7.2 Experiencia en el Trabajo.

La experiencia personal en el trabajo y haciéndola extensiva al estudio de las diferentes asignaturas del Máster de Organización Industrial y Gestión de Empresas, ha sido por una parte muy gratificante y por otra parte complicada. Gratificante en el sentido de volver a la Universidad a estudiar aprendiendo mucho de diversas asignaturas a las cuáles además les veo la aplicación desde mi posición como trabajador.

Por otra parte ha sido complicada debido a la dificultad de compaginar trabajo, vida familiar y estudio debiendo en todo ello sacrificar sobre todo la vida familiar y muchas horas de sueño.

Centrándome en la realización del Proyecto la dificultad si cabe ha sido mayor por la falta de tiempo y la dificultad del tema elegido. Sin embargo y dada la libertad voluntaria de elección, el apoyo de los tutores y el gusto por el tema ha implicado en mi la dedicación de horas y horas con las que realmente he disfrutado por el hecho de poder investigar en un tema que realmente desconocía pero que me ha parecido apasionante. Dicho tema me ha apasionado sobre todo por la inmensa aplicación que tiene y en el caso particular de estudio por la fortuna de disponer de datos reales que me han permitido aplicar dicha técnica en ellos pudiendo implementar la solución de un modo real.

8 LÍNEAS DE INVESTIGACIÓN FUTURA

Se han esbozado a lo largo del Proyecto diversas líneas de investigación futura, las cuales junto con otras se resumen a continuación:

- Posibilidad de incluir en el estudio fechas, de modo que la predicción futura de producción se pueda realizar en base a dichas fechas sin necesidad de conocer las variables de entrada.
- Categorización de todos los fallos de las Instalaciones Fotovoltaicas susceptibles de detección con alerta temprana debido a que produzcan degradación en la producción de modo previo al fallo.
- Discriminación del modelo atendiendo a la tecnología implantada: estructura fija, 1 o 2 ejes.
- Discriminación del estudio en función de zona geográfica.

REFERENCIAS

- ASHRAF, I. & CHANDRA, A. (2004) Artificial neural network based models for forecasting electricity generation of grid connected solar PV power plant. *International journal of global energy issues*, 21, 119-130.
- BASHEER, I. A. & HAJMEER, M. (2000) Artificial neural networks: fundamentals, computing, design, and application. *Journal of Microbiological Methods*, 43, 3-31.
- BENGHANEM, M. & MELLIT, A. (2010) Radial Basis Function Network-based prediction of global solar radiation data: Application for sizing of a stand-alone photovoltaic system at Al-Madinah, Saudi Arabia. *Energy*, 35, 3751-3762.
- CAMPBELL, J. & JARDINE, A. *Maintenance Excellence: Optimising Equipment Life-cycle Decisions*, 2001. Marcel Dekker: New York, NY.
- CRESPO MÁRQUEZ, A. (2007) *The maintenance management framework: models and methods for complex systems maintenance*. Springer Verlag, London.
- CURRY, B., MORGAN, P. & BEYNON, M. (2000) Neural networks and flexible approximations. *IMA Journal of Management Mathematics*, 11, 19-35.
- GUASCH, D., SILVESTRE, S. & CALATAYUD, R. (2003) Automatic failure detection in photovoltaic systems. *Proceedings of 3rd World Conference on Photovoltaic Energy Conversion*, Vols a-C, 2269-2271.
- HASTIE, T., TIBSHIRANI, R., FRIEDMAN, J. (2001) *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Series: Springer Series in Statistics. XVI, 536 p. ISBN 978-0-387-21606-5.
- HIYAMA, T. (1997) Neural network based estimation of maximum power generation from PV module using environmental information - Discussion. *Ieee Transactions on Energy Conversion*, 12, 247-247.
- HORNIK, K., STINCHCOMBE, M. & WHITE, H. (1989) Multilayer feedforward networks are universal approximators. *Neural networks*, 2, 359-366.
- INTERNATIONAL ELECTROTECHNICAL COMMISSION (IEC) (2010) Area 191: Quality of service. www.electropedia.org
- KALOGIROU, S. A. (2001) Artificial neural networks in renewable energy systems applications: a review. *Renewable & Sustainable Energy Reviews*, 5, 373-401.
- KOSTYLEV, V. & PAVLOVSKI, A. (2011) Solar power forecasting performance—towards industry standards. *1st International Workshop on the Integration of Solar Power into Power Systems*, Aarhus, Denmark.
- KUO, C. (2011) Cost efficiency estimations and the equity returns for the US public solar energy firms in 1990–2008. *IMA Journal of Management Mathematics*, 22, 307-321.
- LAPEDES, A. & FARBER, R. (1987) Nonlinear signal processing using neural networks.
- MALCOLM, B., BRUCE, C. & MORGAN, P. (1999) Neural networks and finite-order approximations. *IMA Journal of Management Mathematics*, 10, 225-244.
- MELLIT, A., BENGHANEM, M., ARAB, A. H. & GUESSOUM, A. (2005a) An adaptive artificial neural network model for sizing stand-alone photovoltaic systems: application for isolated sites in Algeria. *Renewable Energy*, 30, 1501-1524.
- MELLIT, A., BENGHANEM, M., ARAB, A. H., GUESSOUM, A. & IEEE (2003) Modelling of sizing the photovoltaic system parameters using artificial neural network.
- MELLIT, A., BENGHANEM, M., BENDEKHIS, M. & IEEE (2005b) Artificial neural network model for prediction solar radiation data: Application for sizing, stand-alone photovoltaic power system. *2005 Ieee Power Engineering Society General Meeting*, Vols, 1-3.
- MELLIT, A., BENGHANME, M., HADJ ARAB, A., GUESSOUM, A. & IEEE (2004) An adaptive wavenet model for sizing of stand-alone photovoltaic systems.

- MELLIT, A. & KALOGIROU, S. A. (2008) Artificial intelligence techniques for photovoltaic applications: A review. *Progress in Energy and Combustion Science*, 34, 574-632.
- MELLIT, A. & SHAARI, S. (2009) Recurrent neural network-based forecasting of the daily electricity generation of a Photovoltaic power system. *Ecological Vehicle and Renewable Energy (EVER)*, Monaco, March, 26-29.
- MILLER III, W. T., GLANZ, F. H. & KRAFT III, L. G. (1990) Cmas: An associative neural network alternative to backpropagation. *Proceedings of the IEEE*, 78, 1561-1567.
- MOUBRAY, J. (1997) *Reliability-centered maintenance*. Industrial Press Inc.
- OLIVENCIA POLO, F., ALONSO DEL ROSARIO, J. & CERRUELA GARCÍA, G. (2010) Supervisory control and automatic failure detection in grid-connected photovoltaic systems. *Trends in Applied Intelligent Systems*, 458-467.
- ORIOLI, A. & DI GANGI, A. (2013) A procedure to calculate the five-parameter model of crystalline silicon photovoltaic modules on the basis of the tabular performance data. *Applied Energy*, 102, 1160-1177.
- PHAM, H. & WANG, H. (1996) Imperfect maintenance. *European journal of operational research*, 94, 425-438.
- RUMELHART, D. E., HINTON, G. E., & WILLIAMS, R. J. (1986) Learning internal representations by error propagation. In Rumelhart, D. E., McClelland, J. L., and the PDP Research Group, editors, *Paralleled Distributed Processing. Explorations in the Microstructure of Cognition. Volume 1: Foundations*, pages 318-362. The MIT Press, Cambridge, MA.
- RAUSAND, M. & HØYLAND, A. (2004) *System reliability theory: models, statistical methods, and applications*, vol. 396. John Wiley & Sons.
- RUTKOWSKI, LESZEK (2008) *Computational Intelligence: Methods and Techniques*. Original Polish edition published by Polish Scientific Publishers. ISBN 978-3-540-76288-1
- WERBOS, P. J. (1994) *The roots of backpropagation: from ordered derivatives to neural networks and political forecasting*, vol. 1. John Wiley & Sons.
- WILSON, R. L. (1986) Operations and support cost model for new product concept development. *Computers & Industrial Engineering*, 11, 128-131.
- YACEF, R., BENGHANEM, M. & MELLIT, A. (2012) Prediction of daily global solar irradiation data using Bayesian neural network: A comparative study. *Renewable Energy*, 48, 146-154.
- ZHANG, G. Q., PATUWO, B. E. & HU, M. Y. (1998) Forecasting with artificial neural networks: The state of the art. *International Journal of Forecasting*, 14, 35-62.

PARTICIPACIÓN EN PROYECTOS DE INVESTIGACION

Acontinuacion expongo las horas y proyectos de investigación en los que he participado.

LEDESY.- Sistema de detección de fugas de gas en plantas termosolares de tecnología cilindro-parabólica. (1381 horas)

Actividad 1 (Coordinación técnica), Actividad 2 (Requisitos y diseño funcional), Actividad 3 (Dispositivo mecánico, Diseño y Desarrollo), Actividad 4 (Red de Sensores, Diseño y Desarrollo), Actividad 5 (Plataforma, Diseño y Desarrollo) y Actividad 6 (T6.1 Dispositivo funcional y T6.2 Validación de la prueba)

MONICA.- Monitorización de la calidad del agua. (1030 horas)

Actividad 2 (Especificaciones técnicas), Actividad 3 (Diseño y desarrollo del prototipo de sensores), Actividad 4 (Sistema de comunicación), Actividad 5 (Modelos y sistemas de ayuda a la decisión) y Actividad 6 (Pruebas y validación).

AGERA.- Aplicación de Realidad Aumentada sobre dispositivo HMD, orientada a la optimización del trabajo de técnicos de campo en los sectores agrícola y medioambiental. (440 horas)

A3 (Análisis técnico sobre el desarrollo claves electrónicas interactivas funcionales), A7 (Estudio sobre la viabilidad de ofrecer un precio competitivo, orientado al mercado nacional e internacional), A8 (Estimación del ahorro que puede suponer, a organismos públicos y empresas privadas, la utilización de la nueva herramienta)

EDIN.- Sistema de toma de decisiones espaciales participativa para la dinamización de iniciativas productivas y la optimización del procedimiento de evaluación ambiental de proyectos. (292 horas)

Actividad 1 (Coordinación técnica y administrativa), Actividad 6 (Análisis de la viabilidad económica y comercial) y Actividad 7 (Conclusiones, riesgos y líneas futuras)

MÉRITOS ASOCIADOS AL PROYECTO

Artículo aceptado en revista JCR IMA Journal of Mathematics (<http://imamat.oxfordjournals.org>) con factor de impacto 1.194 cuyo resumen es el siguiente:

Maintenance optimization through dynamic failure mode reliability analysis

Reliability requirements are increasingly demanded in all economic sectors, practical applications of the reliability theory are pursued often as a tailor-made suit for each case, in order to manage the assets effectively according to specific operating and environmental conditions.

In sectors like renewable energy, these conditions can be changing importantly over time and reliability analysis is periodically required. At the same time, adapting a unique model to similar systems placed in different plants has proven to be troublesome.

In this paper we pursue to facilitate the adaptation of reliability models to controlled changes of the referred conditions and to encourage businesses in their “equipment reliability management”. This paper introduces a model which is based on artificial neural networks; allowing updating assets reliability analysis according to changes in operating and/or environmental conditions, and even more, this model could be easily automated within a SCADA system. Thus, by using the model, reference values and the corresponding warnings and alarms can be dynamically generated, serving as an on-line diagnostic or prediction of a potential degradation of the asset. The model estimates the energy production in a power inverter, and then uses this estimate to detect early degradation and to evaluate the associated economic risk to the system under existing conditions. This information can then trigger preventive maintenance activities.