Aplicación de técnicas estadísticas, matemáticas y de Inteligencia Artificial para la modelización ecológica

TRABAJO FIN DE GRADO

Doble Grado de Matemáticas y Estadística

Magdalena Jáñez Vaz

Sevilla, Junio de 2023

tutorizado por Luis Valencia Cabrera

Índice general

Resumen

La modelización ecológica trata de comprender y traducir las complejas interacciones presentes en un ecosistema y expresarlas como un modelo matemático de forma más esquematizada de cara a ayudarnos a comprender mejor la realidad que estamos estudiando, explicar mejor su comportamiento ante determinados escenarios de interés y tratar de predecir la respuesta del sistema ante nuevos casos. El presente proyecto trabaja con datos de campo recogidos en el parque natural de Doñana, Huelva, relativos al Cangrejo Rojo Americano, *Procambarus clarkii*.

En él se intentará procesar la información recolectada para presentarla de forma más clara y sencilla, descifrar el comportamiento de las variables implicadas, sus interacciones, vislumbrar patrones de respuesta temporal, así como utilizarla para modelizar distintos aspectos de esta especie y del ecosistema. Al tratarse de una especie invasora, es de interés analizar formas de controlar su población, y poder comprender mejor el desarrollo y comportamiento que tiene en las localizaciones estudiadas. Entre los modelos con los que se trabajará hay algunos dedicados a predecir las capturas en trampas de cangrejos según las condiciones climáticas, lo que nos sirve como una posible estimación del tamaño de la población, así como para poder observar cuál es la tendencia local, global o ante distintos posibles escenarios. Otros modelos se dedican a estudiar características de individuos en concreto como su tamaño, sexo y etapa de madurez.

Por último, simplemente matizar que este estudio trata de atacar un problema real, con todo lo que ello conlleva. Al tratarse de una situación auténtica, el interés es mayor, ya que puede ayudar a comprender mejor un problema existente que se está enfrentando. Por otro lado, implica un aumento en la dificultad debido a las limitaciones de los datos frente a los conjuntos de datos que suelen darse en problemas ficticios ya preparados para realizar el estudio.

Palabras clave: cangrejo rojo americano, *Procambarus clarkii*, Doñana, modelización ecológica.

Abstract

Ecological modeling is all about understanding and translating the complex interactions present in an ecosystem and express them as a mathematical model in a more schematic way in order to help us better understand the reality we are studying, better explain its behavior in certain scenarios of interest and try to predict the response of the system to new cases. The present project works with field data collected in the natural park of Doñana, Huelva, related to the Red Swamp Crayfish, *Procambarus clarkii*.

In this project we will try to process the information collected to present it in a clearer and simpler way, to decipher the behavior of the variables involved, their interactions, to glimpse patterns of temporal response, and to use it to model different aspects of this species and the ecosystem. Since it is an invasive species, it is of interest to analyze ways to control its population, and to better understand its development and behavior in the locations studied. Among the models we will work with, there are some dedicated to predict the catches in crab traps according to climatic conditions, which serves as a possible estimate of the population size, as well as to observe what the local and global trend is, as well as in the face of different possible scenarios. Other models are dedicated to study characteristics of specific individuals such as their size, sex and maturity stage.

Finally, it is important to note that this study attempts to address a real problem, with all that this entails. Since it is a real situation, the interest is greater, as it can help to better understand an existing problem that is being faced. On the other hand, it implies an increase in difficulty due to the limitations of the data as opposed to the data sets that usually occur in fictitious problems already prepared for the study.

Key words: Red Swamp Crayfish, *Procambarus clarkii*, Doñana, ecological modeling.

Índice de figuras

Índice de tablas

Parte I

Introducción

Capítulo 1

Introducción

Este primer capítulo sirve como presentación del documento que precede. En él se habla sobre las razones para la elección de la temática escogida y se detallan aquellos objetivos que se planteó alcanzar inicialmente. También se describe la estructura que siguen el resto de capítulos, así como la forma que tienen de organizarse en bloques, y la temática en que se centrará cada uno de ellos.

1.1. Motivación

Esta memoria refleja la labor llevada a cabo durante el desarrollo del Trabajo de Fin de Grado realizado durante el curso 2022-2023, involucrando tareas de estudio, investigación, análisis, desarrollo, experimentación, escritura y conclusión. Se centra en la modelización ecológica, concretamente en modelizar diferentes aspectos de la población del Cangrejo Rojo Americano asentada en el parque natural de Doñana. Esta especie es bien conocida por ser una especie invasora incluso en Sevilla, de modo que la cercanía de la situación y de la zona de estudio explican el interés en este asunto. Sin embargo, la razón prinicipal de la selección de este tema fue la colaboración con el Grupo de Investigación en Computación Natural del Departamento de Ciencias de la Computación e Inteligencia Artificial, que tenía previamente un proyecto en colaboración con los ecólogos que recogieron y publicaron los datos con los que se trabajó a lo largo de este estudio.

Junto con este Trabajo de Fin de Grado y aprovechando el conocimiento adquirido y las herramientas desarrolladas, se planteó como parte de una beca de colabotación la creación de un modelo PDP (*population dynamic P-system*). La idea era realizar algo similar a lo que se llevó a cabo con el Mejillón Cebra en [6] en conjunto con los investigadores involucrados en dicho proyecto. Sin embargo, dicho proyecto a día de hoy aún no recibió la financiación como para alargarlo lo suficiente para llevar esto a cabo, y la realización de un modelo de este estilo y magnitud era inviable sin tener ningún tipo de experiencia previa en este campo. Con todo, eso no influye en el trabajo aquí reflejado, pues en todo momento se pretendía que fuera llevado cabo de forma individual sin depender de dicha colaboración. Se mantuvo como objetivo la utilización de las técnicas de modelización y aprendizaje automático aprendidas a lo largo de los años de estudios universitarios para intentar comprender mejor al cangrejo rojo americano. Igualmente, las técnicas empleadas, lecciones aprendidas y el conocimiento extraído serán sin duda de utilidad en el futuro, además de las soluciones propuestas a las carencias detectadas.

1.2. Objetivos

El principal objetivo en el estudio de una especie invasora es aprender cómo controlar su población y cómo reducir el impacto que tiene sobre el ecosistema invadido. En el caso del Cangrejo rojo americano, el tiempo que lleva asentado en Doñana hace que su población sea bastante estable, hasta el punto de que su desaparición podría implicar nuevos cambios en el ecosistema cuyas consecuencias también habría que estudiar, y analizar no solo los beneficios de su supresión sino otros nuevos problemas que llevaría emparejados. Aun así, ser capaces de estimar la cantidad de cangrejos puede ser útil de cara a paliar los efectos negativos que pueda tener, o potenciar los positivos, por lo que este es uno de los objetivos del estudio. Además, el diseño de modelos nos ayudará a entender mejor los entresijos de las relaciones entre los fenómenos involucrados, los individuos que interactúan en el ecosistema, los elementos propios del mismo, los procesos derivados de la biología de la especie y los posibles factores bióticos y abióticos implicados.

Otro aspecto que sería de útilidad sería poder clasificar en función del tamaño de los cangrejos su sexo, de modo que no fuera necesario examinarlos uno a uno si se obtuviera un modelo lo suficientemente confiable. Lo mismo con la clasificación entre crías o individuos maduros, esta es otra de las variables que registraban los ecólogos, y puede ayudar a estimar el envejecimiento de la población. Para poder predecir la cantidad de huevos que se pondrán en cada puesta sería necesario poder estimar el número de hembras maduras que hubiera simultáneamente, para ello podrían utilizarse los dos modelos anteriormente planteados.

De cara a la cadena alimenticia de Doñana, ya se ha comentado que el cangrejo está establecido en este ecosistema, por tanto ha pasado a formar parte de la cadena y a ser utilizado como alimento por otras especies que también habitan en el parque natural. Por ende, es de interés predecir el tamaño que alcanzarán los cangrejos, ya que eso ayuda a predecir la cantidad de alimento que ellos pueden suponer para el resto de especies.

1.3. Estructura del documento

Este documento se compone de cuatro bloques distintos, cada uno compuesto a su vez de capítulos. Antes de empezar el primer bloque se encuentran el resumen, su traducción al inglés y los índices de tablas y figuras.

El bloque I, Introducción, se compone de dos capítulos. Este capítulo de introducción, el capítulo 1, forma parte de él junto con el capítulo 2. Dicho capítulo trata sobre los preliminares a este estudio, en él se habla sobre modelización ecológica en general y sobre el cangrejo rojo americano, comentando tanto de su llegada a España como de aspectos propios de la especie. Se explican también las herramientas utilizadas: las técnicas de análisis descriptivo, los métodos de inferencia utilizados y el software con el que se llevaron a cabo ambas cosas.

El bloque II, Estudio estadístico de los datos, contiene los capítulos 3 y 4. El capítulo 3 se dedica a la importación de los datos y su correspondiente tratamiento, se hace una breve introducción a cada conjunto tratado, así como una descripción de cada una de las variables que los componen. Una vez se han tratado los datos de forma adecuada y se han reducido a aquella información que se utilizará en el estudio, se muestra la forma en

que ha quedado dicho conjunto mediante un resumen de las variables. El capítulo 4 trata del análisis descriptivo, se hace hincapié en determinados aspectos sobre las variables, la distribución que siguen y las interacciones entre ellas. La forma de mostrarlo es principalmente mediante gráficas propiamente explicadas, aunque también se incluyen tablas numéricas para dar mayor firmeza a las conclusiones extraídas.

El bloque III, Modelos predictivos, también se compone de tres capítulos. El capítulo 5 es una introducción, en la que se explican las decisiones tomadas durante el estudio realizado, para así en los dos siguientes capítulos centrarse en los modelos que finalmente fueron seleccionados. Además de explicar por qué se descartaron algunos enfoques de modelización y hacer un análisis distribucional de una de las variables, en este capítulo se describen y analizan bloques de código que se utilizarán en los capítulos siguientes. El capítulo 6 trata sobre los modelos de clasificación; para cada una de las variables categóricas se realizan varios modelos distintos, se muestran y se comparan los resultados obtenidos. El capítulo 7, por su parte, trata sobre los modelos de regresión. En este caso no es tan sencillo como aplicar el modelo a los datos, sino que requiere de una toma de decisiones que se detalla en este capítulo, así como los resultados y conclusiones obtenidas.

El bloque IV, Conclusiones, se compone del capítulo 8 dedicado a las conclusiones, último del núcleo del trabajo, así como de dos apéndices. En dicho último capítulo, se resumen tanto las contribuciones realizadas como las lecciones aprendidas y principales conclusiones extraídas del estudio y desarrollo del trabajo. Adicionalmente, se analizan posibles vías de trabajo futuro. Por su parte en el primer apéndice se recopila el código utilizado a lo largo del estudio pero que no fue mostrado en el resto de la memoria, de modo que se ha destacado en el núcleo de la misma los aspectos esenciales pero se proporciona aquí en forma de apéndice por si se quiere consultar cualquier aspecto a nivel técnico. En el segundo apéndice se recopila el código que fue descartado por no dar resultados suficientemente buenos pero que merecía la pena ser mencionado. De nuevo, se proporciona como apéndice para permitir su consulta en caso de que resulte de interés de los caminos descartados o de recopilación de pruebas realizadas, pero queda fuera del foco principal del trabajo por no haber arrojado ninguna conclusión fuerte generalizable sobre caminos futuros a descartar, ya que un resultado en ese sentido también sería de interés para evitar explorar esa vía por parte de futuros trabajos por parte de investigadores y/o estudiantes.

Capítulo 2

Preliminares

El capítulo a continuación está dedicado a establecer las bases teóricas sobre las que se asentará el trabajo realizado. Entre otras cosas, se habla de una posible ruta a seguir para la modelización ecológica, se introduce la especie sobre la que se estará tratando, se describen las técnicas utilizadas y el software con el que se implementaron.

2.1. Modelización ecológica

La modelización ecológica es, como toda modelización, una simplificación de un sistema mucho más complejo. Los ecosistemas involucran una gran cantidad de procesos que tienen lugar, individuos que interactúan entre sí y con el entorno, y todo ello supone una gran complejidad y diversidad estructural. Además, su estudio suele abarcar grandes periodos de tiempo y espacios tan amplios que hacen difícil trabajar con el sistema sin tener que reducirlo a términos más simples. Modelizar las interacciones entre las especies que conviven en un mismo ecosistema requiere de un conocimiento sobre la conducta y comportamiento de cada una, así como sobre el clima y las condiciones del ecosistema.

Blanco [4] explica los pasos a seguir en la modelización ecológica, comenzando por crear un mapa conceptual en el que se describan los elementos que componen el sistema, se identifiquen las componentes y los procesos que las unen. Una vez se tiene un esquema de los elementos que se estudiarán, el siguiente paso es construir un diagrama de flujo en el que se refleje la influencia que tienen entre sí, así como las variables reguladoras que influencian estos flujos. Es importante que este paso evitar un exceso de complejidad. El tercer paso es crear un diagrama de bucles causales, pudiendo representar así la retroalimentación entre variables del modelo (que puede ser negativa o positiva). Esto es una herramienta de comunicación más que una herramienta analítica. A continuación se estiman los parámetros y condiciones iniciales, esto es, la calibración del sistema, que debe realizarse con datos de campo. En este paso el modelo pasa de ser conceptual a ser más numérico, y por tanto requiere de una gran atención al detalle a la hora de elegir los valores o el rango que se asignará a cada variable. En el siguiente paso el modelo se ejecuta varias veces, añadiendo variaciones a los valores antes escogidos para ver la influencia que un pequeño cambio puede tener. Esto es el análisis de sensibilidad del modelo. Por último se lleva a cabo la evaluación y validación del modelo, en el que se comprueba si las predicciones del modelo son lo suficientemente cercanas a datos reales independientes del modelo.

La magnitud de esta tarea sobrepasa al alcance de este trabajo, y no es el foco principal del mismo. Además, la información disponible no es suficiente para conseguir un modelo como el explicado anteriormente. No obstante, para llevar a cabo varios de los principales pasos de interés en ese proceso descrito o en un flujo similar, es conveniente profundizar en el conocimiento de todas las "entidades" involucradas, la información disponible, y poder incrementar el conocimiento que se tiene a partir de los datos en bruto recopilados por trabajadores de campo. Esta es la idea que se planteó en este trabajo, el estudio estadístico de la información disponible, con idea de extraer conocimiento a partir de ella, que sea útil para el diseño y la validación de unos primeros modelos explicativos y predictivos basados en machine learning y para caminar hacia esos modelos ecológicos de un mayor nivel de complejidad pero que se verían altamente beneficiados por las conclusiones de nuestros estudios. De modo que se llevará a cabo una versión reducida en la que se estudiará la población del cangrejo rojo americano asentada en el parque natural de Doñana.

2.2. Procambarus clarkii

El cangrejo rojo americano (*Procambarus clarkii*, según fue nombrado por Charles Girard en 1852) es una especie crustáceo decápodo originaria del sureste de Estados Unidos y el noreste de México. Tiene una gran capacidad de adaptación a distintos ecosistemas, tolerando incluso aguas algo salinas lo cual es inusual siendo un cangrejo de río, lo que le ha permitido extenderse por otros continentes en los que se le considera especie invasora. Junto con el cangrejo señal (*Pacifastacus leniusculus*), es una de las especies invasoras más representativas, habiendo llegado a prácticamente todos los continentes. España es uno de los mayores productores mundiales, siendo China el principal productor de esta especie a día de hoy [1]. Según [3] con datos hasta 2019 se importan más de 3000 toneladas al año, lo cual genera unos beneficios anuales de 3.300.000 € al año. Por otro lado, según [2], con datos hasta 2008, eran alrededor de 2 millones al año, y proporcionaba empleo para 400 personas en plantas de procesado de marisco y varios cientos de pescadores.

Su llegada a España se produjo de forma premeditada con fines comerciales. Las diversas fuentes coinciden en que ocurrió a principios de la década de los 70, y aunque no llegan a ponerse de acuerdo con el año exacto se estima que fue entre 1969 y 1973. Se propuso a raíz de los buenos resultados que la cría de esta especie había tenido en Estados Unidos, generando una industria millonaria. Los expertos que asesoraron sobre el tema contaban con poco conocimiento sobre el funcionamiento de las enfermedades de los cangrejos en esa época y se consideró que el impacto ecológico que tendría la introducción de esta especie sería mínimo. Los primeros ejemplares de esta especie que llegaron a la Península fueron importados legalmente desde Monroe (Louisiana, EE.UU.) hasta cerca de Puebla del Río, en las marismas del Guadalquivir. [23]

Los cangrejos no eran criados en granjas de astacicultura sino que se capturaban en su hábitat natural. Esto, junto con la sencillez para transportarlos, provocó que su expansión fuera rápida y poco controlada, ya que los pescadores o particulares que querían obtener beneficios por la venta de esta especie podían llevar a cabo introducciones en nuevas zonas. Por estos motivos el cangrejo rojo no tardó en ocupar la mayor parte del bajo Guadalquivir, de modo que hacia 1976 se comercializaban desde esta zona unos 9.000 kg de cangrejos, y en 1979 la cantidad era diez veces mayor [23]. A día de hoy los únicos lugares de la Península en las que parece escasear son Galicia y algunas otras zonas del

	A. pallipes	*P. clarkii*	*P. leniusculus*
Origen	Autóctono	Exótico	Exòtico
Longiutd máxima (mm)	120	130	180
Longevidad máxima (años)	12	6	20
Tamaño maduración (♀; mm)	50-60	50-70	60-90
Edad maduración (♀; meses)	36-48	3-5	24-36
Cópulas	Oct-Nov	May-Oct	Oct-Nov
Numero de huevos (rango)	20-140	200-500	200-400
Incubación (meses)	5-8	0.8-1	5-9
Número de puestas por año	Una	Hasta 2	Hasta 2
Eclosión	May - Jun	May - Jun	
Rango altitudinal (en Península Ibérica)	240-1550	0-900 (más frecuente < 500 m)	

Figura 2.1: Tabla comparativa características de los cangrejos de interés. Fuente: [23]

norte.

A pesar de lo que se creía en un principio, la introducción del cangrejo rojo americano en los ecosistemas acuáticos de la Península Ibérica ha provocado enormes cambios en sus características y funcionamiento. Algunos de estos efectos han sido la reducción o eliminación de vegetación acuática sumergida y transformar las aguas claras de los ecosistemas en que habitan en sistemas de aguas turbias, debido a la movilización de sedimentos del fondo que suelen llevar a cabo. Aparte de eso, el cangrejo rojo americano, al igual que el cangrejo señal, es portador de la afanomicosis o peste del cangrejo y, aunque se ve afectado por ella con baja frecuencia, los cangrejos de río europeos sí que son más susceptibles a esta enfermedad.

La alimentación del cangrejo es omnívora y oportunista, pudiendo alimentarse de plantas así como de otros macroinvertebrados acuáticos, y larvas o puestas de peces y anfibios, lo que suele ser propio de individuos juveniles. Al mismo tiempo, el cangrejo ha supuesto una fuente de alimento para muchas especies de animales como pueden ser anguilas, cigüeñas, gaviotas, milanos, zorros o nutrias. Esta última, la *Lutra lutra*, es una especie vulnerable según el Catálogo Español de Especies Amenazadas, de la cual el cangrejo puede suponer un 76 % de su alimentación si la población es suficientemente abundante. En general, las especies depredadoras del cangrejo han aumentado su riqueza en las marismas del Guadalquivir de manera considerable respecto a las especies herbívoras o que tienen un menor porcentaje de cangrejo en su dieta [25].

En cuanto a las condiciones necesarias para vivir, habitualmente los cangrejos de río son exigentes en cuanto a las características de las aguas en las que se establecen, pero en comparación el cangrejo rojo, que nos ocupa en este trabajo, tiene una gran tolerancia ecológica y flexibilidad a la hora de adaptarse a las condiciones del ecosistema. En general prefiere aguas templadas, y acostumbra a excavar refugios en los que enterrarse en periodos desfavorables como pueden ser sequías o épocas frías. Gracias a esto puede resistir en seco hasta 4 meses; también es resistente a la contaminación orgánica, las bajas concentraciones de oxígeno disuelto y a un amplio rango de salinidades. Por contra, destacar que parece ser delicado respecto a la altitud y otros factores que esta conlleva, siendo escaso por encima de 600 metros.

El aspecto reproductivo es una de las principales diferencias entre el cangrejo rojo

Figura 2.2: Foto comparativa aspecto de los cangrejos de interés. Fuente: [8]

americano y otros como el cangrejo señal o el cangrejo de río europeo, *Austropotamobius pallipes*. Este último es el cangrejo nativo que está siendo desplazado por el cangrejo rojo americano, y actualmente es una especie amenazada. Para comparar las principales diferencias entre estos tres cangrejos, se dispone de la figura 2.1. Como puede verse en ella, el cangrejo rojo americano, *P. clarkii*, es mucho menos longevo que la especie nativa, *A. pallipes*, o que otro de los principales cangrejos de río invasores, *P. leniusculus*, pero a cambio tiene una maduración prácticamente inmediata: mientras que el resto tardan más de dos años en alcanzar la madurez, el cangrejo rojo americano no tarda ni medio año. El número de huevos es bastante superior al del cangrejo local, estando igualado con el cangrejo señal, pero de nuevo destaca por el poco tiempo que necesitan incubarse. Aparte de su rápido crecimiento y maduración sexual, otro factor a tener en cuenta es su capacidad de adaptar sus ciclos a condiciones climáticas y ambientales muy diversas. Si las condiciones son favorables, prácticamente en cualquier época del año pueden existir hembras con huevos, y estas pueden reproducirse más de una vez en una temporada. Combinando esto con su alta resistencia a enfermedades, no es difícil entender el motivo de que tenga tango éxito a la hora de extenderse por nuevos ecosistemas.

A pesar de todos los cambios e inconvenientes que acarreó la introducción del cangrejo rojo americano en la Península Ibérica, a día de hoy lleva ya casi cinco décadas en la zona, y se ha establecido de forma bastante estable. Ha encontrado un lugar en la cadena alimenticia de los ecosistemas en los que habita, hasta el punto de que su desaparición podría acarrear problemas para las especies que se alimentan de él. Es por eso que algunos estudios han desistido de intentar disminuir su población para enfocarse en intentar mitigar los efectos que pueda tener su convivencia con aquellas especies con las que cohabitan e intentar sacar provecho de esta.

2.3. Técnicas de análisis descriptivo

Una vez se tienen los datos en el formato adecuado, es importante saber mirar qué estructura tienen, estudiar el rango de valores que toma cada variables, así como la interacción que puedan tener entre sí. Aunque el objetivo final sea obtener estimaciones y modelos que ajusten las variables de interés, un estudio previo permite que eso no tenga

que hacerse a ciegas. Es necesario por tanto saber qué mirar, y más aún qué es mejor ignorar, para tener una visión más clara de los datos con los que se trabaja.

En el capítulo de Análisis descriptivo se representan variables en distintos formatos. En esta sección se justifica la elección de cada formato en función de lo que quería resaltarse con las gráficas o tablas añadidas.

Los gráficos de dispersión, o *scatter plots*, son útiles para enfrentar dos variables numéricas, cada una colocada en uno de los ejes. Para estudiar la correlación entre dos variables es la mejor opción, pues cada punto tiene una coordenada de cada variable, como puede verse en la Figura 4.4. Si las variables son independientes la distribución es uniforme, mientras que en caso de tener una alta correlación se agruparían los puntos a lo largo de una línea (la pendiente dependería de si la correlación es positiva o negativa). Si a esto se suma un ajuste como el que proporciona `geom_smooth` de la librería **ggplot2**, se ve de forma todavía más clara dicha recta en caso de que las variables estén altamente correladas.

A los gráficos de dispresión se les pueden añadir estéticas, como es el color o la forma de los puntos en función de otra variable distinta a las que se usaron para los ejes. Generalmente se usan variables categóricas, ya que así cada color o forma corresponde a una de las categorías, y es posible ver si las que pertenecen a un mismo grupo se encuentran juntas o si están dispersas. El color proporciona un matiz más visual, es preferible sobre todo cuando se dispone de un conjunto de tamaño considerable. Sin embargo ante la posibilidad de que un documento sea consultado impreso en blanco y negro, conviene aplicar también la estética de la forma de los puntos. El tamaño de los puntos o su transparencia son también aspectos que se pueden modificar, aunque esto sirve más para cambiar la estética de todos los puntos a la vez más que para hacerlo en función de otra variable. En la Figura 4.14 se recurre a esto para poder diferenciar las múltiples líneas de la gráfica.

Un caso concreto en el que los gráficos de dispersión son altamente beneficiosos es para comparar los valores reales de una variable y las predicciones que se obtienen mediante un modelo predictivo. En este caso es imprescindible que para cada individuo se pueda visualizar al mismo tiempo el valor real y el estimado, así como lo cerca que están ambos valores, por lo que utilizar cada valor para una coordenada es apropiado. El ajuste perfecto ocurriría si todos los valores representados estuvieran sobre la diagonal (del primer cuadrante en caso de trabajar con valores positivos), por lo que es conveniente al mismo tiempo que se representan los pares de valores representar también la diagonal. Esto suele hacerse con una línea discontinua, por cuestiones estéticas. Puede verse un ejemplo de esto en la Figura 7.10.

Por último, si se dispone de las coordenadas de lugares de interés (como ocurre en este estudio con las localizaciones de capturas de cangrejos), al representarlas de esta forma enfrentando Latitud y Longitud puede verse la silueta del mapa sobre el que se está trabajando. Puede verse en la Figura 4.2.

Cuando solo quiere estudiarse una única variable numérica, si solo toma valores enteros los diagramas de barras ofrecen una buena estimación de su función densidad. En el eje Y se realiza el conteo de la frecuencia con que la variable aparece en cada intervalo representado. Si la variable no solo toma valores enteros puede usarse también esta representación, denominada en el caso de variables continuas histograma, teniendo en cuenta que es importante la elección del número de divisiones realizado. Un ejemplo de esto se ve en la Figura 4.9. Representar la función densidad es también una buena forma de visualizar la

distribución que siguen los valores de la variable.

Para comparar una variable numérica y una variable categórica se pueden usar diagramas de caja y bigote o diagramas de violín, que son bastante similares entre sí, o crear el diagrama de barras como se mencionó anteriormente, pero generando uno para cada valor de la variable categórica. En el primer tipo de gráfico se representan los cuartiles, así como los valores atípicos de la variable numérica para cada valor de la variable categórica. En el segundo tipo de gráfico la representación es más similar a la función de densidad, teniendo una forma simétrica y más redondeada que la caja. Se pueden añadir los cuantiles deseados al diagrama de violín, por lo que la información que puede proporcionar es más completa, aunque para casos simples es preferible el de caja y bigotes, que muestra solo la información básica. Al representarse lado a lado las funciones de densidad para cada categoría es sencillo compararlas a ojo, lo que permite sopesar si es conveniente aplicar un test estadístico que compare ambas distribuciones y confirmar si son distintas, o si al contrario el grupo al que se pertenece no influye en esta variable lo suficiente y por tanto es mejor trabajar con el conjunto al completo. Ejemplos de esto son la Figura 4.3, la segunda gráfica de la Figura 4.5 y la Figura 4.15.

Para enfrentar dos variables categóricas las gráficas no son de tanta utilidad, pero las tablas de contingencia permiten mostrar rápidamente el conteo de cada posible cruce entre las categorías de cada variable. También sirve por si las variables tienen valores perdidos, pues estos se pueden contar como una categoría más y se comprueba de qué forma se distribuyen los valores perdidos de cada una respecto a las categorías de la otra. Esto se utiliza en la Tabla 4.1.

Las tablas con datos de resumen presentando distintos estadísticos también pueden servir para mostrar valores como la media, otros cuantiles o la desviación típica de una variable en función de varias categorías, al igual que se hacía con los diagramas de caja y bigotes o de violín, pero pudiendo ver el valor numérico. En ocasiones es suficiente solo ver la gráfica, pero en otras la información numérica acompañada del gráfico puede ser aún más revelador. La Tabla 4.2 es un ejemplo de esto.

2.4. Métodos de inferencia

Tras haber realizado el análisis descriptivo y tener una mejor idea del aspecto de los datos que se están manejando, ya puede pasarse a la tarea de modelización teniendo una idea más precisa de lo que debe aplicarse. En este estudio fueron varios los métodos de inferencia que se aplicaron en los distintos campos de interés: el algoritmo de los k-vecinos más cercanos, árboles de decisión, bosques aleatorios, regresión lineal y regresión logística. Además, el ajuste de hiperparámetros se llevó cabo en aquellos métodos en los que era conveniente.

2.4.1. Algoritmo de los k-vecinos más cercanos

También conocido como algoritmo KNN (de sus siglas en inglés: *k-nearest neighbors*) es un método de aprendizaje supervisado no paramétrico que se utiliza tanto en problemas de clasificación como de regresión. Las predicciones se basan en la similaridad del individuo

que se pretende predecir con otros individuos para los que el valor de la variable objetivo es conocida.

El primer paso es seleccionar el número de vecinos que serán tenidos en cuenta (k) en base a la magnitud del conjunto con el que se esté trabajando y a su distribución. Generalmente este valor se obtiene comparando la eficiencia de distintos valores en una submuestra del conjunto total (conjunto de validación).

Debe tenerse en cuenta también la distancia que va a utilizarse para medir la proximidad entre individuos. Generalmente la distancia utilizada para medir la similitud entre los individuos es la distancia euclídea, que equivale a la distancia de Minkowski cuando p=2, aunque cualquier exponente válido podría servir:

$$d((x_1,..,x_m),(y_1,..,y_m)) = \left(\sum_{j=1}^{m} \mid x_j - y_j \mid^p\right)^{1/p}$$

Para cada uno de los individuos a predecir debe calcularse la distancia con todos los del conjunto de entrenamiento.

Una vez se han medido todas las distancias, se seleccionan los k individuos de la muestra entrenamiento más cercanos al que se quiere predecir, son sus k vecinos más cercanos. En función de si se trata de un problema de clasificación o de regresión el procedimiento es distinto para el paso siguiente:

- Clasificación: se asigna la clase que presente mayoritariamente su conjunto de k vecinos más cercanos. Para evitar empates en el caso de tener solo dos categorías es conveniente escoger un valor impar para k.
- Regresión: se asigna el promedio de los valores de sus k vecinos más cercanos.

Este proceso se repite para todos los individuos que se quieran predecir, no se construye un modelo realmente, por lo que la predicción de un nuevo caso puede suponer un coste computacional considerable para conjuntos de datos de gran tamaño. A cambio, se evita el coste asociado generalmente al proceso de aprendizaje, pero en muchos casos este coste puede ser asumible pero la aplicación del modelo requerirse en tiempo real por lo que se es más exigente con el tiempo de predicción.

Más información en [19].

2.4.2. Árboles de decisión

Los árboles de decisión son una técnica de aprendizaje supervisado no paramétrico que puede utilizarse tanto para problemas de clasificación como de regresión. Es una técnica bastante popular ya que son sencillos de interpretar.

Los árboles constan de una raíz desde la que se van generando de ramas y nodos. Los nodos internos presentan una división de los elementos del conjunto en función de las variables, y las ramas conectan los nodos entre sí. Si la variable utilizada para realizar la división es numérica, se realiza una división binaria; en caso de ser categórica puede hacerse también una división binaria o tener tantas alternativas como categorías haya en

la variable. Los nodos finales (hojas) proporcionan un resultado para la variable de interés que es la estimación asignada a todos aquellos elementos que fueron clasificados en ese nodo final en función de las divisiones que se fueron haciendo desde la raíz del árbol.

Los árboles se crean en base a una muestra de entrenamiento, procurando que las divisiones creadas separen en grupos lo más diferentes posibles al conjunto que llegó al nodo, esto es, obtener la división que proporcione una mayor ganancia de información. Debe establecerse un mínimo de elementos en cada nodo, que una vez alcanzado impida que se vuelva a dividir convirtiéndolo en un nodo final. Esto permite evitar el sobreajuste en cierta medida, y también aporta una condición de parada. Otro método de evitar el sobreajuste es realizar una poda del árbol una vez finalizado incluso si esto empeora la clasificación en el conjunto de entrenamiento. Aun así, la técnica de árboles de decisión carece de robustidad, y tiende a basarse en exceso en la muestra de entrenamiento. También puede ocurrir que al tomar la mejor decisión en cada momento no se llegue al mejor modelo final, pero no hay forma de poder saber si esto ocurre generando un único árbol. Es por eso que en base a esta técnica se desarrollan los Bosques Aleatorios, que intenta subsanar estas carencias.

Más información en [22].

2.4.3. Bosques aleatorios

También conocido como *Random Forest*, esta técnica de aprendizaje supervisado no paramétrico se basa en combinar los resultados obtenidos mediante árboles de decisión. Al contrario que estos, los bosques aleatorios dificultan en cierta medida la interpretación pero permiten construir modelos más robustos, sin las limitaciones inherentes al hecho de construir un árbol específico a partir de una muestra.

La técnica de bosques aleatorios es un método de ensamblaje, lo cual significa que combina varios modelos para formarse. Esto produce un aumento en la complejidad del modelo, pero también le aporta flexibilidad y permite reducir el sesgo y mejorar la capacidad de predicción. Para aplicar esta técnica se crea un número considerable de árboles de decisión (este número es un hiperparámetro fijado de antemano y al que se le puede aplicar el ajuste óptimo), cada uno de los árboles se entrena con una muestra aleatoria que se extrae del conjunto de entrenamiento mediante técnica *bootstrap* (muestreo aleatorio con reemplazamiento). Esto se conoce como *bagging*, que proviene de *bootstrap aggregation*. Además, antes de cada división que se crea en estos árboles, se selecciona aleatoriamente un subconjunto de entre las variables disponibles (el tamaño del subconjunto también está fijado, de nuevo es un parámetro ajustable) y la división que se crea es la mejor posible con ese subconjunto de variables. Esta aleatoriedad supone que cada árbol será distinto de los demás, al menos ligeramente, y aporta mayor variabilidad al bosque, para así reducir el sesgo una vez se tome como predicción el valor medio (regresión) o el más repetido (clasificación). El modelo final es más complejo que un árbol aleatorio y mucho menos intuitivo, pero también es considerablemente más robusto.

Gracias a que el número de variables que maneja cada nodo es un parámetro ajustable que suele ser bastante menor que el total de las variables, permite manejar conjuntos con una enorme cantidad de variables sin ser excesivamente costoso computacionalmente. Sin embargo sigue siendo una técnica que puede tardar bastante tiempo en entrentarse

de forma correcta, al multiplicar el proceso de entrenamiento (así como el de predicción, aunque este es mucho menos costoso) de un árbol por el número de árboles.

Otro detalle interesante de esta técnica es que gracias a aplicar *bagging* permite que el error cometido por cada árbol pueda estimarse sin necesidad de utilizar una muestra de entrenamiento y una muestra test. Las muestras *bootstrap*, al ser un muestreo con reemplazamiento y tener un tamaño limitado aunque significativo, tienen como consecuencia que algunos elementos no son seleccionados para la muestra de entrenamiento en cada árbol (generalmente cada muestra se compone de dos tercios de los elementos de la muestra original). Estos elementos se conocen como *out-of-bag* (OOB). De este modo, una vez creados los árboles, para cada elemento se puede obtener una predicción de la variable objetivo utilizando solo aquellos árboles que no utilizaron dicho elemento en su muestra de entrenamiento, lo que se conoce como error error fuera de la bolsa (del inglés *out-of-bag error* u *OOB error*).

Más información en [9] y [22].

2.4.4. Regresión lineal

La regresión lineal es una técnica estadística que se utiliza para predecir la variable objetivo, una variable continua (Y) llamada variable dependiente, en función de un conjunto de variables también continuas (X_1,.., X_m) llamadas variables independientes o predictoras. Se pretende obtener un ajuste de la forma:

$$Y = \beta_1 \cdot X_1 + \beta_2 \cdot X_2 + ... + \beta_m \cdot X_m$$

Los coeficientes de esta expresión, β_j, para $j = 1, \ldots, m$, se estiman de forma que minimicen el error y proporcionen el mejor ajuste posible. Uno de los métodos usados más habitualmente para seleccionarlos es el método de mínimos cuadrados ordinario:

$$min_{\beta_1,...,\beta_m} \sum_{i=1}^{n} (Y_i - \beta_1 \cdot X_{i1} - \beta_2 \cdot X_{i2} - \cdots - \beta_m \cdot X_{im})^2$$

donde n es el número de observaciones que hay en la muestra de entrenamiento sobre la que se está aplicando esté métodо, Y_i el valor de la variable objetivo del i-ésimo individuo y X_{ij} el valor de la j-ésima variable predictora del i-ésimo individuo de la muestra. En definitiva, se trata de minimizar la suma del error cuadrático (*sum of squared errors*, SSE). Los valores obtenidos para la muestra de entrenamiento no serán los β_j reales sino una estimación: $\hat{\beta}_j$. De hecho, si se tuvieran los coeficientes exactos, en caso de existir un ajuste perfecto de los mismos, el error obtenido sería 0.

Una vez se tienen estos estimadores, ya estaría creado el modelo. Para cualquier valor de la muestra test $X = (x_1, .., x_m)$ que se quisiera predecir, bastaría con obtener su estimación $\hat{Y} = x_1 \cdot \hat{\beta}_1 + x_2 \cdot \hat{\beta}_2 + ... + x_m \cdot \hat{\beta_m}$

Un elemento importante del modelo de regresión lineal es el coeficiente de determinación R^2. Este coeficiente mide la bondad de ajuste del modelo. Concretamente representa qué porcentaje de la varianza de la variable objetivo puede ser explicado por este modelo, tomando siempre valores entre 0 y 1 siendo más cercano a 1 cuanto mejor sea el modelo.

Su fórmula es

$$R^2 = \frac{\sum_{i=1}^{n}(\hat{Y}_i - \bar{Y})^2}{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}$$

donde $\hat{Y}_i$ es la estimación de la i-ésima observación de la muestra entrenamiento, e $\bar{Y}$ es la media de los valores de la variable objetivo en nuestra muestra de entrenamiento.

Este coeficiente no penaliza el número de variables explicativas utilizadas, sin embargo es conveniente que este número no sea demasiado elevado. Comparar modelos únicamente en base al coeficiente de determinación no tiene eso en cuenta, por lo que también se utiliza el coeficiente de determinación ajustado. Para una muestra de tamaño n y con k variables explicactivas, su fórmula sería:

$$R^2_{ajustado} = 1 - \frac{N-1}{N-k-1} \cdot \left[1 - R^2\right]$$

Cuanto más bajo sea el valor de k más cercano será a R^2 el coeficiente de determinación ajustado, pues tendrá menor penalización. Para modelos con un gran número de variables y muy mal ajuste este coeficiente puede llegar a tomar valores negativos.

Más información en [7] y [17].

2.4.5. Regresión logística

La regresión logística es un método estadístico similar a la regresión lineal, con la diferencia de que la variable objetivo (Y) es binaria, y por tanto es un modelo de clasificación. Mediante las variables predictoras $(X_1, \ldots, X_m)$ lo que se pretende modelar es la probabilidad de que ocurra uno de los valores de Y, y la predicción será aquella de las dos categorías que sea más probable.

La forma de expresar esto es mediante una función $\pi(X)$, que representa la esperanza condicionada de Y respecto de las variables predictoras $E(Y|X) = E(Y|X_1, \ldots, X_m)$. Se busca una expresión de la forma

$$\pi(X) = \frac{e^{\beta_0+\beta_1 X_1+\ldots+\beta_m X_m}}{1 + e^{\beta_0+\beta_1 X_1+\ldots+\beta_m X_m}}$$

De este modo, al aplicar la transformación *logit*, $g(X) = ln\left[\frac{\pi(X)}{1-\pi(X)}\right] = \beta_0 + \beta_1 X_1 + \ldots + \beta_m X_m$. Con esta g(x) se obtiene una transformación conveniente, ya que tiene las propiedades deseables del modelo de regresión lineal: depende linealmente de las variables predictoras, puede ser continua, y puede variar desde $-\infty$ hasta $+\infty$.

Si se intenta aplicar el método de mínimos cuadrados ordinarios para calcular los estimadores de los coeficientes en este modelo, no se cumplen las mismas propiedades que se conseguían en el modelo de regresión lineal. En este caso, lo que generalmente se usa es el estimador de máxima verosimilitud, para lo cual se necesita construir la función homónima. Esta función expresa la probabilidad de los resultados obtenidos en función de los parámetros desconocidos, por lo que estos se estimarán con los valores que mayor probabilidad o verosimilitud otorguen a la muestra de entrenamiento con la que se trabaja.

Dándole a Y la codificación 0 o 1 (siendo $\pi(X)$ la probabilidad de que dados los valores X de las variables predictoras la variable Y tome el valor 1), la verosimilitud de un elemento cualquiera de la muestra de entrenamiento (x_i,y_i) sería $\pi(x_i)^{y_i}\left[1-\pi(x_i)\right]^{1-y_i}$. Como por hipótesis las observaciones de la muestra de entrenamiento eran independientes, la verosimilitud de la muestra completa será el producto de la verosimilitud de todos sus elementos.

Por tanto la función de verosimilitud construida es:

$$l(\beta_1,..,\beta_m)=\prod_{i=1}^{n}\pi(x_i)^{y_i}\left[1-\pi(x_i)\right]^{1-y_i}$$

Aquellos valores de $\beta_1, \ldots, \beta_m$ que maximicen esta función para la muestra de entrenamiento serán sus estimadores $\hat{\beta_1}, \ldots, \hat{\beta_m}$, con los que se construiría $\hat{\pi}(X)$.

Más información en [11].

2.4.6. Ajuste de hiperparámetros

También conocido como *tuning* de hiperparámetros, es una técnica para mejorar las predicciones de los modelos que dependen de uno o más parámetros ajustables. Para realizar el ajuste se debe seleccionar un conjunto de combinaciones de valores de los parámetros que quieren ajustarse. A partir de este conjunto, se evaluará el modelo de forma iterativa con cada combinación de valores. Debe seleccionarse también una métrica para evaluar la calidad del modelo, de modo que cuando se haya evaluado dicha métrica para todos los valores deseados de los parámetros la decisión final será aquella combinación que proporcione un mejor valor de esta métrica. Para modelos de clasificación puede usarse la exactitud, especificidad o sensibilidad del modelo como métrica, también el valor del AUC (área bajo la curva ROC) es conveniente ya que oyede complementar muy bien a la exactitud, recogiendo implícitamente información sobre la especificidad y sensibilidad. También es una medida fiable para controlar que la buena exactitud provenga de una clasificación correcta de cada clase, y no por un buen ajuste de una clase mayoritaria en detrimento de una clase minoritaria. Para los modelos de regresión la métrica más utilizada es el error cuadrático medio (RMSE), aunque también podría utilizarse el error absoluto medio (MAE).

Este procedimiento forma parte del entrenamiento del modelo, por lo que la evaluación del rendimiento no puede comprobarse sobre la muestra test. En caso de tener un conjunto de datos suficientemente grande puede hacerse la división en tres grupos: muestra de entrenamiento, muestra de validación y muestra test. La muestra de entrenamiento es la que se usaría para crear los modelos, y el rendimiento de cada combinación de parámetros se evaluaría en la muestra de validación. Aquellos valores de los parámetros que dieran mejor resultado en la muestra de validación serían los usados en el modelo final, cuyo rendimiento se evaluaría en la muestra test.

Por desgracia, no siempre se tienen datos suficientes como para realizar estas tres divisiones, y en este estudio ese es un problema que se enfrenta a menudo. Una alternativa es utilizar validación cruzada. Gracias a esto solo se necesita dividir en muestra de entrenamiento y muestra test. La muestra de entrenamiento se divide aleatoriamente en pliegues del tamaño similar (habitualmente 10) de modo que cada elemento pertenezca a uno y

solo uno de los pliegues. Para cada valor de los parámetros que quiera evaluarse se crearán tantos modelos como pliegues haya, se reservará uno de los pliegues para la muestra de validación y todos los demás compondrán la muestra de entrenamiento. Una vez se ha hecho esto con todos los pliegues, el valor de la métrica que se asigna a estos valores de los parámetros es la media de la obtenida en todos los pliegues.

Existe un tipo de validación cruzada específico, en el que el número de pliegues es igual al número de registros. De este modo, en cada modelo se deja fuera de la muestra de entrenamiento uno solo de los individuos, de ahí su nombre en inglés *leave-one-out*. Aunque sea un método lo suficientemente utilizado como para tener su propio nombre, el número de pliegues habitual es 10 como ya se mencionó.

Esta técnica es computacionalmente más costosa que usar una única muestra de validación, ya que se debe crear un modelo por cada pliegue y por cada combinación de parámetros a evaluar, pero permite aprovechar al máximo los datos disponibles.

Para poder ver la influencia que el cambio de parámetros puede tener en el ajuste de un modelo, concretamente en dos modelos de este trabajo, se ha creado una aplicación shiny disponible en el contenido adicional. El objetivo de la pestaña de "Modelo reactivo" es aportar una visión rápida de la forma en que cambiar los valores de algunos parámetros puede influir en la exactitud y resultados. En dicho documento shiny interactivo se implementaron los modelos de clasificación mediante KNN que se llevarán a cabo en este trabajo, dejando los parámetros K y el exponente de la distancia de Minkowski a elección. Una vez se han especificado los valores deseados y la variable objetivo se puede pinchar en el botón de "Actualizar" para que se muestren la exactitud del modelo, su área bajo la curva roc y la representación gráfica de la matriz de confusión.

2.5. Software

El análisis realizado se llevó a cabo en el lenguaje de programación R, mediante el entorno de desarrollo integrado más ampliamente utilizado para él: Rstudio. Esta decisión fue principalmente motivada por la familiaridad que a lo largo de los estudios universitarios se fue adquiriendo con este lenguaje, además de por el potencial que su gran comunidad de usuarios le otorgan y su adecuación para el trabajo estadístico en particular y del ámbito de la ciencia de datos en general. Esto puede verse ilustrado por ejemplo por el hecho de contar con estructuras nativas como vectores o marcos de datos, con mecanismos convenientes que en otros lenguajes como Python requieren el empleo de paquetes adicionales como numpy o pandas, entre otros.

Existe una gran cantidad de paquetes, colecciones de funciones creadas por usuarios para optimizar procesos de uso habitual, que son de libre uso y que facilitan el trabajo enormemente para todos los usuarios. A continuación se listan aquellos paquetes y ecosistemas que tuvieron mayor relevancia en el trabajo realizado, con una pequeña explicación de aquellas funciones que fueron de más utilidad.

2.5.1. Ecosistema tidyverse

El ecosistema tidyverse ([28]) se compone de un conjunto de paquetes, cada uno con una finalidad y funcionalidad distintas, pero destinados a usarse concatenadamente para

facilitar la importación, tratamiento, ordenación y visualización de conjuntos de datos, así como el reconocimiento de patrones de texto y el manejo de variables categóricas. El libro *R for Data Science*, [29], fue escrito por Hadley Wickham, el creador de este ecosistema, y en él se explica cómo sacar máximo partido a los paquetes de tidyverse para el tratamiento de datos y modelización.

El paquete **readr** ([30]) está destinado a la correcta lectura de ficheros de datos; concretamente los ficheros csv que se mencionan en este documento fueron leídos con la función `read_csv` de este paquete. No tiene, sin embargo, una función que permita leer ficheros excel, ya que su finalidad es leer exclusivamente ficheros rectangulares, ya sea separados por comas (csv) o por tabuladores (tsv). El paquete **readxl** utilizado para la lectura de excel también forma parte de tidyverse, aunque no se carga de manera automática por no sobrecargar, sino que debe llamarse específicamente.

El paquete **dplyr** ([26]) tiene como finalidad facilitar los principales problemas que se enfrentan a la hora de manipular un conjunto de datos. Entre sus funciones más utilizadas se encuentran:

- `mutate`, que permite crear una nueva variable en función de las variables ya existentes.
- `rename`, para renombrar alguna de las variables del conjunto, o varias al mismo tiempo.
- `select`, para escoger un subconjunto de las variables del conjunto, permitiendo también excluir algunas en concreto sin tener que indicar una a una las que se mantienen.
- `filter`, para filtar los registros del conjunto en función de los que cumplan los criterios especificados.
- `group_by`, a juego con `summarise`, permiten agrupar los registros del conjunto en función de los valores de las variables por las que se agrupa. Una vez esta agrupación está establecida, pueden resumirse sus valores en nuevas variables que se definan y engloben a todo el subconjunto. Con `across` y `c_across` puede aplicarse un mismo patrón de resumen o transformación a diferentes variables de forma simultanea.
- `arrange`, ordena el conjunto en función de una o varias variables, de forma ascendente o descendente según se indique.
- Los joins (`left_join`, `rigth_join`, `inner_join`, `full_join`, `anti_join`) o uniones entre conjuntos, permiten combinar dos conjuntos que tengan en común una o varias variables, pudiendo hacer que permanezcan las de uno solo de los conjuntos, las de ambos, o incluso quedarse con todas las del primer conjunto que no aparezcan en el segundo.

El paquete **tidyr** ([31]) se utiliza para poder ordenar un conjunto de datos de acuerdo a lo que se considera un conjunto bien ordenado: una variable por columna, un individuo u observación por fila, un valor por celda. Por la estructura de los datos disponibles, una de las funciones más utilizadas de este paquete fue `separate`, ya que en varias ocasiones se incluía en una misma celda información sobre la fecha y la localización de eventos

registrados, y debía separarse en dos variables distintas. También `pivot_wider` fue útil para tratar con un conjunto en el que cuatro medidas distintas se registraban en una misma variable, habiendo una segunda variable para registrar de qué medida se trataba. Su complementario, `pivot_longer`, también se encuentra en este paquete. En caso de ser necesario también podría trabajarse con datos anidados gracias a `nest` y `unnest`, aunque no fue necesario para este proyecto en concreto.

El paquete **ggplot2** ([27]) es la herramienta por excelencia para crear gráficos a nivel profesional en este lenguaje. Trabaja con una estructura en capas, pudiendose añadir más y más detalles sobre un gráfico base, y pudiendo llegar a personalizar la estética con un enorme nivel de detalle de forma intuitiva gracias a su elegante gramática. Otras opciones para generar gráficos interactivos son **plotly** o **highchartr**, entre otros, aunque **ggplot2** sigue siendo el paquete más ampliamente utilizado para gráficos.

El tratamiento de datos de tipo fecha fue sencillo gracias a **lubridate**, en este caso ocurre al igual que con **readxl**, aunque forme parte de tidyverse no se carga automáticamente al cargar el ecosistema . Este paquete está dedicado a poder leer fechas sin importar los distintos formatos que puedan presentar para así poder trabajar correctamente con ellas. Permite así extraer el mes o el año de una fecha sin tener que recurrir a expresiones regulares constantemente.

Por último, el paquete **forcats** permite manejar las variables categóricas como factores, pudiendo reordenar sus niveles en función de otra variable (`fct_reorder`), en función de la frecuencia de cada categoría (`fct_infreq`), según una ordenación creada a mano (`fct_relevel`), o agrupar las cateogrías menos frecuentes en una única "otros" (`fct_lump`).

2.5.2. Ecosistema tidymodels y alternativas

El ecosistema tidymodels (la información de este ecosistema y todos sus paquetes se encuentra en [15]) sigue la idea de tidyverse para combinar distintos paquetes y hacer que funcionen en conjunto, aunque en este caso la finalidad es simplificar y sistematizar la tarea de modelización y el aprendizaje automático. Este ecosistema fue creado por Hadley Wickham y Max Kuhn, entre otras personas, siendo el primero el creador de **tidyverse**, y el segundo el creador del paquete **caret** ([12]). Este ecosistema es precisamente la evolución natural de **caret**, y en varias ocasiones a lo largo de este trabajo se alterna entre ambas opciones como se describe más adelante. En *Building predictive models in R using the caret package*, [13], Max Kuhn habla sobre el uso de este paquete para construir modelos predictivos en R. También puede recurrirse a los vídeos de Julia Silge, o a su libro *Tidy Modeling with R*, [14], escrito junto con Max Kuhn para saber más del tema.

Para comenzar, este ecosistema facilita la división en muestras de entenamiento y test con el paquete **rsample** y la función `initial_split`, de modo que ambas muestas queden almacenadas en un único objeto pero también puedan extrarse por separado con `training` y `testing`. Esta función además permite indicar qué proporción de la muestra destinar al conjunto de entrenamiento (por defecto es 0.75) y también estratificar en función de una variable. De ese modo puede garantizarse que el valor de la variable objetivo sea similar en las muestras de entrenamiento y test. En caso de que no se realizara la división de forma estratificada y por algún casual los casos en ambos grupos tuvieran valores significativamente distintos, el ajuste se vería perjudicado pues los valores a predecir no

son similares a aquellos utilizados para entrenar. Es por eso que es mejor estratificar de antemano, y al menos poder prevernir eso dentro del conjunto de datos disponibles. Sigue siendo posible, por supuesto, que en el futuro los datos que quieran predecirse sean muy distintos a aquellos con los que se entrenó y evaluó el modelo, lo cual no podría haberse previsto. En caso de empeorar mucho el rendimiento, tal vez obligase a generar un nuevo modelo para adaptarse a la nueva situación.

Otro de los paquetes esenciales de tidymodels, el paquete **parsnip**, sirve para crear la base de los modelos, indicando de cuál se trata, si es de regresión o de clasificación, y añadiéndole un motor que se trata de otro paquete de R que ya está implementado (como puede ser **rpart**, [24]).

El preprocesado de las variables se lleva a cabo con el paquete **recipes**, que incluye, entre otras muchas, funciones para:

- eliminar variables en caso de que tengan una correlación demasiado alta con `step_corr`.
- centrar y escalar las variables deseadas de forma simultánea, `step_normalize`, o bien solo centrar, `step_center`, o solo escalar, `step_scale`.
- imputar valores perdidos; hay una gran cantidad de funciones dedicadas a esto, pero en concreto la que fue usada para este trabajo fue `step_impute_knn`.
- eliminar automáticamente variables que tomen tomen solo un valor o tengan varianza muy cercana a cero mediante `step_nzv`.
- crear una categoría ficticia para las variables nominales, de modo que si se quiere predecir una observación que tenga un valor nunca visto en la muestra de entrenamiento no se produzca un error, `step_novel`.
- para las variables nominales con varias categorías, crear variables dummy que permitan traducir estas categorías en variables numéricas binarias, con `step_dummy`.

Tanto los modelos como las recetas de preprocesado pueden combinarse en un flujo de trabajo con **workflows**, que puede después modificarse y actualizarse de manera sencilla. En caso de que se quiera mejorar el ajuste de hiperparámetros para el modelo, con el paquete **tune** puede realizarse esta tarea de forma eficiente. Finalmente, una vez se tenga establecido el modelo final, su eficiencia puede evaluarse con la libería **yardstick**, así como obtener las predicciones.

Todo esto crea un flujo bastante bien definido para las tareas de modelización. Sin embargo, para el análisis que se llevó a cabo algunas de estas tareas fueron hechas de modo más manual debido a la familiaridad que se tenía con otras alternativas. En otro casos la eficacia de **tidymodels** fue lo suficientemente convincente como para optar por este ecosistema. Concretamente, gracias al paquete **caret** varias de estas tareas se pueden llevar a cabo sin un excesivo coste computacional. Para la división en dos muestras sí que se usó **rsample** aunque la función base `sample` también podría llevar a cabo esta división. Para los pasos posteriores, la función `train` de **caret** permite indicar el modelo a llevar a cabo con el argumento "method", y aplicar un centrado y escalado de las variables numéricas con "preProcess". También se puede hacer ajuste de hiperparámetros, aplicando por

ejemplo validación cruzada si se le indica, gracias a "tuneLength" si solo quiere indicarse el número de valores distintos para los parámetros que debe probar o "tuneGrid" si se quiere indicar expresamente los valores. Para los modelos mediante árboles de decisión y con bosques aleatorios se usaron respectivamente los paquetes **rpart** ([24]) o **C50** ([20]) y **randomForest** ([16]).

2.5.3. Otros paquetes

Aparte de los ya mencionados, algunos paquetes no englobados en **tidyverse** fueron de gran utilidad. A la hora de generar el documento en cuestión se utilizó el paquete **knitr** ([33]), que es un motor dedicado a la generación de informes dinámicos con R. Gracias a esto se puede integrar código R en documentos Markdown y el uso de LaTeX para la estructuración. La estructura de este documento es la proporcionada por Luque-Calvo [18], este libro utiliza la propia plantilla para explicar la forma en que puede ser utilizada para llevar a cabo trabajos de fin de estudios con Rmarkdown.

Para visualizar las tablas de forma más elegante se utilizó el paquete **kableExtra** ([34]), que está dedicado a simplificar la generación de tablas, permitiendo modificar su aspecto con sencillos argumentos.

En el estudio de medidas de tendencia central y de dispersión de los datos fue útil el paquete **skimr** ([10]), así como para hacer un conteo de los casos perdidos en el caso de haberlos, ya que proporciona de forma automática tanto el número como el porcentaje de completitud.

Para la creación de una aplicación interactiva que permite visualizar una gran cantidad de gráficas de forma sencilla y rápida se utilizó la librería **shiny** ([32]). Esta aplicación puede encontrarse en el apartado de contenido adicional, y gracias a ella es sencillo observar el comportamiento de algunas variables de este estudio sin tener que crear cada gráfica individualmente.

Los manuales de los paquetes mencionados pueden encontrarse en la bibliografía en caso de querer consultarse de forma más extensas las funcionalidades de cada uno.

Parte II

Estudio estadístico de los datos

Capítulo 3

Importación, tratamiento y descripción de los datos

Los datos utilizados para este estudio han sido recopilados por un equipo de ecólogos en el Parque Nacional de Doñana, Huelva. En dicha reserva hay una población considerable del animal de interés: el cangrejo rojo americano, *Procambarus Clarkii*. Este estudio se llevó a cabo entre 2004 y 2022, habiendo recopilado información sobre una gran cantidad de individuos de esta especie. Los datos del estudio, que fueron publicados el 11 de enero de 2023, constan de tres ficheros csv que a continuación serán analizados y ligeramente modificados para facilitar el tratamiento de la información. (Estos datos pueden consultarse en este enlace o en [5]).

Además de eso, gracias a la colaboración entre estos ecólogos y el departamento de Ciencias de la Computación e Inteligencia Artificial de la Universidad de Sevilla en un proyecto en común, se dispone de otro conjunto de datos que no fue publicado. Estos datos han sido refinados en menor medida, y por tanto no se publicaron, pero al ser de una naturaleza distinta de aquellos que sí fueron publicados aportan una información muy valiosa, y por tanto se tomó la decisión de incluirlos también en este estudio.

3.1. Archivo OCC: ocurrencias individuales

El archivo Don_ biom_red-swamp-crayfish _occ_20230111.csv contiene quince variables y 15483 registros. Sin embargo, la mayoría de las variables no aportan demasiada información al contener un mismo valor para todos los registros.

Cada registro de este documento se refiere a una captura de un individuo, del cuál la información que se registra es:

- **eventID**: tiene el formato {ID localización}_{fecha AAAA-MM-DD}; esta información se separará en dos variables: **IDL** y **fecha** .
- **occurrenceID**: similar al anterior, pero añadiendo al final un contador que identifica de manera única a cada registro. Esto sirve para tener un identificador para cada cangrejo que fue estudiado, de aquí se extrae la variable **Cangrejo**. En el archivo “mof” se utiliza esta misma identifiación, por tanto pueden combinarse los dos conjuntos de datos.

- **ocurrenceTime**: tiene el formato H:M:S con un "z" al final de la cadena; se leerá ignorando esto último para anotar solo con el momento en que se registró la captura de cada individuo.
- **individualCount**: ya que cada registro corresponde a un individuo, siempre toma el valor 1 y por tanto no es una variable de interés para el estudio posterior.
- **basicOfRecord**: informa del método de captura; en este archivo solo toma el valor "Capture/Human Observation"; ya que todas se realizaron de la misma forma, tampoco tiene un interés para el estudio posterior.
- **sex**: toma los valores "Male" o "Female" en función de si el cangrejo era macho o hembra, aunque también hay una proporción considerable de datos perdidos (aproximadamente 16 %)
- **lifestage**: toma los valores "Immature" o "Mature" en función de si se consideró que el cangrejo capturado era una cría o era ya adulto; en este caso, la cantidad de datos perdidos es mucho menor.
- Las siguientes variables se refieren a características de la especie, y al ser todos los registros sobre *Procambarus Clarkii* el valor que toman es el mismo en todo el conjunto; dicho valor se indica al lado de cada variable a continuación:
  - **kingdom** - "Animalia"
  - **phylum** - "Arthropoda"
  - **class** - "Malacostraca"
  - **family** - "Cambaridae"
  - **genus** - "Procambarus"
  - **specificEpithet** - "clarkii"
  - **scientificName** - "Procambarus clarkii"
  - **scientificNameAutorshup** - "Procambarus clarkii (Girard, 1852)"

Por tanto, las variables que tienen relevancia para el análisis que se llevará a cabo son:

```
##       IDL              fecha              Cangrejo     occurenceTime
##  Min.   :  1.00   Min.   :2004-06-09   Min.   :    1   Length:15483
##  1st Qu.: 17.00   1st Qu.:2008-04-22   1st Qu.: 3872   Class1:hms
##  Median : 34.00   Median :2010-06-23   Median : 7742   Class2:difftime
##  Mean   : 54.37   Mean   :2011-07-19   Mean   : 7742   Mode  :numeric
##  3rd Qu.: 55.00   3rd Qu.:2012-11-29   3rd Qu.:11612
##  Max.   :490.00   Max.   :2022-09-08   Max.   :15483
##      sex           lifestage
##  Female:6504   Immature:8968
##  Male  :6503   Mature  :6504
##  NA's  :2476   NA's    :  11
##
##
##
```

3.2. Archivo MOF: información biométrica

El archivo Don_ biom_red-swamp-crayfish _mof_20230111.csv contiene siete variables y 19599 registros. En este caso el formato es algo distinto al que se considera un conjunto de datos ordenados (cada fila un registro, cada columna una variable, cada celda un valor). Hay cuatro posibles medidas sobre las capturas individuales de cangrejos que en el archivo OCC se detallaban, pero para cada medida se ha generado una fila, en vez de tener una fila por cada cangrejo y cada medida en una variable.

A continuación se detalla la información que contiene cada variable del archivo original, así como la posterior forma de organizarlo en el conjunto de datos que se utilizarán para el posterior análisis.

- **occurrenceID**: tiene el formato {ID localización}_ {fecha AAAA-MM-DD}_{ID cangrejo} por lo que lo que se separará en tres variables: **IDL**, **fecha**, **Cangrejo**.
- **measurmentID**: es igual que la anterior variable, pero añadiendo al final también un identificador que numera cada registro. Esto hace que sea un identificador único para el registro, pero de cara al análisis posterior no tiene utilidad, con las tres variables antes explicadas es suficiente. **Nota**: debería ser, al igual que en las siguientes variables, "measurement", pero en el csv aparece con este nombre.
- **measurementValue**: variable numérica que indica la medida que se tomó, sin especificar unidades ya que eso se incluye en la siguiente variable.
- **measurementUnit**: para las variables de longitud el valor es "mm" pues se miden en milímetros y para la variable peso es "g" se mide en gramos.
- **measurementType**: toma cuatro valores distintos según la medida que se esté registrando: puede ser "total body length", "cephalothorax length", "cephalothorax width" o "weight" en función de si se está midiendo el largo del cuerpo, el largo del cefalotorax, el ancho del cefalotorax o el peso del cangrejo.
- **measurementAccuracy**: la precisión de la medida depende del tipo de medida: para el largo del cuerpo es 0.5 milímetros, para el largo y el ancho del cefalotorax es 0.1 milímetros, y para el peso es 0.2 gramos.
- **measurementMethod**: el método de medida depende del tipo de medida, toma solo un valor para cada una de las cuatro opciones y se especifica la forma en que se realizó la medición.

Se puede ver que tres de las variables toman el mismo valor para cada uno de los cuatro posibles tipos de medida que se está registrando, por tanto es conveniente quedarse solo con el tipo de medida; de esta forma se reduce el número de variables sin perder información. Este conjunto será reordenado para que cada fila corresponda a un cangrejo. Se tendrán por tanto las variables fecha, IDL, Cangrejo, y las cuatro medidas (en muchos casos serán NA por no haberse tomado todas las mediciones en todos los cangrejos).

Nota: Una vez realizada la ordenación, se puede observar que el cangrejo número 9, capturado el 4 de junio de 2010 en la localización con ID 9, tiene registrado el largo del cuerpo en 993 mm y ninguna otra medida. El siguiente valor más alto son dos cangrejos

que miden 149 mm, por tanto se ha considerado que esta medición fue un error de medida o de registro, y se ha eliminado.

El conjunto obtenido tras estas modificaciones tendría la siguiente forma:

```
##       IDL              fecha               Cangrejo      largo_cuerpo
##  Min.   :  1.00   Min.   :2004-06-09   Min.   :    1   Min.   :  2.00
##  1st Qu.: 17.00   1st Qu.:2008-04-22   1st Qu.: 3871   1st Qu.: 37.00
##  Median : 34.00   Median :2010-06-23   Median : 7742   Median : 63.00
##  Mean   : 54.37   Mean   :2011-07-19   Mean   : 7742   Mean   : 62.85
##  3rd Qu.: 55.00   3rd Qu.:2012-11-29   3rd Qu.:11613   3rd Qu.: 86.00
##  Max.   :490.00   Max.   :2022-09-08   Max.   :15483   Max.   :149.00
##
##  largo_cefalotorax ancho_cefalotorax      peso
##  Min.   : 1.30     Min.   : 1.0      Min.   :  1.00
##  1st Qu.:11.70     1st Qu.:12.0      1st Qu.:  4.00
##  Median :18.70     Median :19.0      Median : 12.00
##  Mean   :21.79     Mean   :23.2      Mean   : 16.93
##  3rd Qu.:29.00     3rd Qu.:32.0      3rd Qu.: 26.00
##  Max.   :78.00     Max.   :68.0      Max.   :117.00
##  NA's   :13552     NA's   :14434     NA's   :14344
```

Este conjunto de datos y el conjunto OCC pueden combinarse gracias al identificador de los cangrejos, de modo que en un único conjunto se tendrá la información sobre el sexo, estado de madurez, y las medidas de cada cangrejo, así como la fecha de captura, el lugar y su identificador. Como ambos conjuntos estaban bien relacionados, todos los registros que hay en uno los hay en el otro, así que a la hora de aplicar el *join* no hay que preocuparse sobre cuáles se mantienen, sino que todos están en el conjunto final llamado occ_mof.

3.3. Archivo EV: eventos

El archivo Don_biom_red-swamp-crayfish_ev_20230111.csv contiene 18 variables y 444 registros, cada uno correspondiente a una localización geográfica y un día concreto, y se registran en ellos las capturas que se produjeron mediante trampas, no de forma individual como en los anteriores archivos.

Hay muchas de estas variables que tienen un único valor, por tanto no se incluirán en el conjunto reducido que se usará en el análisis posterior. Concretamente, la información que se registra en las variables es:

- **eventID**: esta variable tiene el formato {ID localización}_{fecha en formato AAAA-MM-DD}, conviene que quede registrado en dos variables separadas, **IDL** y **fecha**, pero ya hay dos variables en este archivo que se corresponden con esa información, así que esta es redundante.
- **intitutionCode**: tiene un único valor: "ICTS-RBD". Por tanto no aporta ninguna información relevante.

- **institutionID**: también toma un único valor: "https://deims.org/bcbc866c-3f4f-47a8-bbbc-0a93df6de7b2".
- **datasetName**: valor único: "ES_DONANA_Biom_red-swamp-crayfish_2004_2022_v020230101"
- **eventDate**: registra la fecha en formato AAAA-MM-DD, coincide con el valor que aparecía en la variable eventID.
- **eventTime**: registra la hora a la que se produjo la recogida de la trampa, el formato que tiene es H:M:S con un "z" al final, se selecciona únicamente la hora.
- **continent**: toma solo el valor "Europe", las siguientes variables también tienen un único valor ya que todas las capturas de este archivo se corresponden con Doñana.
- **country**: solo toma el valor "ES".
- **stateProvince**: solo toma el valor "Huelva".
- **location**: solo toma el valor "Doñana".
- **localityID**: indica el identificador numérico de la localización en la que se produjeron las capturas, se corresponde con la primera parte de la variable eventID.
- **locality**: para cada localización aparte de un código numérico se asigna un nombre para identificarla. Es una información útil, pero para el análisis posterior esta variable no aporta nada que no se conozca con el identificador numérico, así que se eliminará.
- **decimalLatitude**: coordenada Latitud de la localización en la que se produjeron las capturas medida en grados decimales WGS 84.
- **decimalLongitude**: coordenada Longitud de la localización de las capturas medida en grados decimales WGS 84.
- **habitat**: indica si la localización se trata de una laguna o una marisma.
- **sampleSizeValue**: variable numérica que indica el número de capturas que se registraron.
- **sampleSizeUnit**: solo toma el valor "trap", todas las capturas se realizan de la misma forma.
- **sampleSizeEffort**: solo toma el valor "24 hours".

Una vez se eliminan las variables que toman un único valor o no aportan nada adicional, el conjunto anterior se reduce considerablemente. También se eliminan aquellos registros que tengan perdido el valor de SampleSizeValue, ya que no aportan nada. El aspecto final del conjunto de datos es el siguiente:

```
##    eventDate            localityID     eventTime        decimalLatitude
##  Min.   :2004-06-09   Min.   :  1.0   Length:391        Min.   :36.84
##  1st Qu.:2008-01-22   1st Qu.: 21.0   Class1:hms        1st Qu.:37.02
##  Median :2010-04-06   Median : 33.0   Class2:difftime   Median :37.10
##  Mean   :2010-02-06   Mean   : 45.5   Mode  :numeric    Mean   :37.06
##  3rd Qu.:2011-10-16   3rd Qu.: 53.0                     3rd Qu.:37.12
##  Max.   :2019-09-13   Max.   :490.0                     Max.   :37.17
##  decimalLongitude              habitat    sampleSizeValue
##  Min.   :-6.687   Lagunas temporales:202   Min.   : 1.000
##  1st Qu.:-6.490   Marismas          :189   1st Qu.: 5.000
##  Median :-6.464                           Median : 5.000
##  Mean   :-6.441                           Mean   : 6.302
##  3rd Qu.:-6.410                           3rd Qu.: 9.000
##  Max.   :-6.246                           Max.   :15.000
```

3.4. Archivo Databases

El archivo Doñana_Crayfish_Databases es un fichero excel que contiene cinco hojas con información respecto de la investigación realizada por los ecólogos, la cual no ha sido publicada y por tanto está bastante menos refinada que los anteriores archivos. La información con la que se ha trabajado en este documento se encuentra principalmente en la segunda hoja, pero también se incluye un pequeño resumen de la información que almacenaban las demás hojas.

3.4.1. Hoja 1: Captured Crayfish

La información de este conjunto es bastante similar en contenido y formato a la del fichero EV, descrito en la sección 3.3. Cuenta con un número mucho mayor de registros y también de datos perdidos, por lo que es probable que, tras hacer una limpieza exhaustiva de estos datos, lo que se obtuviesen fueran los datos que finalmente se publicaron en el archivo EV.

3.4.2. Hoja 2: Presence-Absence + FQvariables

Esta hoja sí contiene una enorme cantidad de información que no se englobaba en ninguno de los conjuntos publicados: información sobre las variables ambientales del entorno. Al formar parte de los datos que no se terminaron publicando, dispone de una gran cantidad de datos que pueden ser atribuidos a errores de registro, ya que no son coherentes con los demás, y también contiene bastantes datos perdidos en varias de las variables.

Estos datos son altamente útiles a la hora de realizar predicciones o estudiar la relación que pueden tener variables como la temperatura o el pH en la cantidad o el tamaño de la población del cangrejo rojo americano, pero requieren un pre-procesado considerable antes de poder ser estudiados. A continuación se detallan las variables de las que se dispone y cómo se llevó a cabo el procesado de la información.

- **ID**: es un identificador único para cada registro.
- **Año**: indica en qué año se realizó la medición. Se cambiará por "Anyo" para evitar el uso de la ñ en el código.
- **Mes**: indica en qué mes (del año antes indicado) se realizó la medición. En este caso no se asocian las mediciones a un día en concreto sino que se refieren a todo un mes.
- **IDL**: identificador numérico de la localización en que se realizó al medición.
- **Hábitat**: toma dos valores en función de si la localización se corresponde con una Laguna temporal o con una Marisma.
- **Tipo nasa**: detalla qué tipo de nasa fue la que se utilizó para las capturas, siendo las posibilidades las siguientes: Nasa Galapaguera, Nasa Camaronera, Nasa Cangrejera, Nasa Prototipo, Nasa Camaronera Doble y Nasa Camaronera Doble sin separar copos.
- **Código nasa**: consta de una letra y un número, sirve para identificar las nasas anteriores de forma distintiva.
- **Procambarus clarkii**: toma valores entre 0 y 4, no queda claro lo que indica.
- **Superficie encharcada**: indica la cantidad de superficie que había encharcada en la localización, dividiendo las posibilidades en cinco grupos: $< 10\ m^2$, entre 10 y 100 m^2, entre 100 y 1000 m^2, entre 1000 y 10000 m^2 o más de 10000 m^2. También hay una cantidad considerable de datos que están perdidos.
- **Profundidad máxima**: mide en centímetros la produndidad máxima de la localización en la que se tomaron las medidas del agua.
- **Vegetación**: puede tomar valores Sí o No, pero en casi la mitad de los casos este dato está perdido.
- **Profundidad de medida...12**: esta variable toma el valor "Superficie" si se realizó medición, o también puede tomar el valor NA, en cuyo caso ninguna de las variables numéricas posteriores se registran tampoco.
- **Profundidad de medida...13**: esta variable solo toma el valor NA, no tiene ningún otro tipo de dato.
- **Temperatura del agua**: mide la temperatura del agua en grados centígrados.
- **Conductividad**: mide la conductividad del agua en $miliSiemens/cm^3$.
- **Salinidad**: mide la salinidad del agua en g/L.
- **Oxígeno disuelto en porcentaje**: mide el oxígeno que se encontraba disuelto en el agua según el porcentaje de saturación.
- **Oxígeno disuelto en valor absoluto**: mide el oxígeno que se encontraba disuelto en el agua por mg/L.
- **pH**: mide el pH del agua.

- **Turbidez**: mide la turbidez en Unidades Nefelométricas de Turbidez (UNT).
- **Clorofila a**: mide la clorofila en mg/m^3.
- **Amonio**: mide el amonio en mg/L.
- **Nitrato**: mide el nitrato en mg/L.

Para el procesamiento de los datos, se filtran aquellos que no tienen ninguna información numérica gracias a la variable **Profundidad de medida...12**. Además de eso se eliminan las variables que no aportan información útil como lo son **ID**, **Tipo nasa**, **Código nasa**, **Procambarus clarkii** o **Profundidad de medida...13**. Además, al resto se les modifican los nombres para evitar tildes, espacios y caracteres problemáticos. También se crea una variable nueva, Fecha, que agrupa el Año y el Mes, haciendo corresponder a cada mes 1/12 de año y sumando ambas, esto es de utilidad a la hora de representar gráficamente las variables numéricas ambientales respecto al tiempo. No obstante, de cara a futuros modelos que usen estas variables como predictores debe tenerse en cuenta la alta correlación que habrá entre las variables Año y Fecha.

Nota: una vez realizada esta limpieza, mirando con detalle podría observarse que algunas localizaciones tienen más de un registro para una misma fecha. Examinándolo detenidamente, se puede apreciar que en los casos que ocurre esto teniendo 3 o más valores para un mismo mes, lo que ocurre es que una de las variables numéricas está aumentando en una unidad su valor para cada registro, permaneciendo iguales todas las demás variables. Se ha considerado que esto se debe a un error a la hora de introducir los datos, por tanto todos aquellos pares Fecha e IDL que tengan más de 2 registros distintos de las variables numéricas han sido eliminados.

Tabla 3.1: Variables ambientales numéricas

	Valores perdidos	% completos	Media	Desviación típica
Anyo	0	1.000	2009.644	2.705
Mes	0	1.000	5.103	3.182
IDL	0	1.000	62.633	75.212
Prof_max	163	0.792	55.739	33.169
Temperatura	6	0.992	16.435	6.568
Conductividad	11	0.986	3.354	5.902
Salinidad	237	0.698	2.086	4.120
O2_porc	135	0.828	77.001	54.525
O2_abs	121	0.846	7.431	5.589
pH	25	0.968	8.224	0.797
Turbidez	196	0.750	87.034	148.327
Clorofila	114	0.855	66.070	124.512
Amonio	173	0.779	9.030	13.756
Nitrato	234	0.702	70.668	149.765
Fecha	0	1.000	2009.986	2.687

En la Tabla 3.1 se muestra la información sobre las variables numéricas. Puede verse que algunas de las variables están completas, pero la mayoría tienen valores perdidos,

pudiendo llegar incluso al 30 % del total. Esto dificultará el estudio, y requiere plantearse si conviene eliminar registros, variables o imputar datos..

Para las variables categóricas puede verse que en todos los registros se indica el hábitat, habiendo dos posibilidades y estando ambos grupos bastante equilibrados Las otras dos variables, en cambio, tienen una cantidad considerable de valores perdidos, como puede observarse en el resumen incluido a continuación.

```
##                Habitat                    Sup_encharcada Vegetacion
##  Lagunas temporales:400   < 10 m2                 :  5    No  :141
##  Marisma           :384   > 10 000 m2             :256    Si  :427
##                           entre 1 000 y 10 000 m2: 60    NA's:216
##                           entre 10 y 100 m2       : 69
##                           entre 100 y  1 000 m2   : 57
##                           NA's                    :337
```

Cabe destacar que lo que en un principio eran 6710 registros, ha quedado reducido a 784 tras esta limpieza. Esto es debido a dos factores: primero, que 1619 de los registros no contenían valores de las variables numéricas sino que eran todo NA, fueron por tanto eliminados. Segundo, que para cada fecha y lugar (en las que sí se tomaron valores) se generaban tantos registros como nasas distintas hubiera colocadas. La información de las variables ambientales numéricas era la misma para todos los registros ya que no es algo que dependiese de esto. Como el código de nasa no aporta información relevante para la modelización posterior, sino que era de interés para aquellos que realizaban el trabajo de campo, en este estudio se han eliminado las repeticiones. Sin embargo, esto sigue siendo una cantidad suficiente para intentar realizar estudios y analizar la influencia de estas variables sobre la población del Cangrejo Rojo Americano en Doñana.

3.4.3. Hoja 3: Metadata FQvariables

Contiene información sobre las variables antes explicadas y sobre otras muchas que no aparecen en este documento. Esta información es relativa a los nombres originales, los que se utilizaron finalmente y una pequeña descripción sobre su contenido. Sirve para entender mejor el resto de hojas del fichero, así como saber las unidades en que se midieron las variables numéricas.

3.4.4. Hoja 4: Average FQvariables per IDL

Para cada IDL se especifica un periodo de tiempo durante el que se realizó un muestreo, un seguimiento de las variables ambientales. Para cada uno de esos periodos y para cada variable de las numéricas de la hoja Presence-Absence, se registra el Promedio, el valor mínimo y máximo alcanzados.

El número de mediciones realizadas en cada IDL es diferente, y no se hacen con una periodicidad fija. Además la longitud del periodo es muy variable, pudiendo abarcar desde menos de una semana hasta varios años. Por tanto, es útil para conocer los valores aportados, pero de cara a obtener resultados o predicciones no es tan sencillo utilizarlo.

3.4.5. Hoja 5: FQ variables

Esta hoja tiene una estructura algo desorganizada, los primeros 40 registros no corresponden a nada y los nombres que hay en la cabecera no se corresponden con los valores de las variables que hay una vez comienzan a darse los registros reales.

Se especifican detalles muy técnicos de las mediciones, como qué personas fueron las encargadas de realizarlas, color/olor del agua, presencia o no de algas y aspecto de estas, y otras observaciones cuyo interés ecológico es significativo, pero que matemáticamente no servirán para hacer ningún estudio posterior, porque tampoco se pueden asociar adecuadamente con las mediciones proporcionadas en otras hojas descritas anteriormente.

Capítulo 4

Análisis descriptivo

Una vez los datos han sido correctamente importados, hay distintos aspectos de estos que pueden analizarse y que pueden ser de interés, o arrojar luz sobre las relaciones entre las variables. Algunas de las preguntas planteadas al inicio de la investigación pueden ser respondidas visualizando correctamente las variables. En este capítulo se explica el procedimiento seguido para obtener una idea general de la estructura de los datos, así como los puntos destacables que se encontraron en las relaciones entre variables y la evolución de estas mismas.

4.1. Fase exploratoria

Gran parte del trabajo previo de este estudio consistió en adquirir familiaridad con los datos disponibles, tanto respecto a las distribuciones de las gráficas como a la cantidad de valores disponibles o las interacciones entre variables. Muchas de las gráficas, tablas y demás representaciones obtenidas no aportaron información relevante por tanto no han sido incluidas Eso no significa que fueran inútiles, simplemente era necesario ver que tal vez una representación no aportaba nada ya que las variables representadas no interactuaban entre sí de la forma esperada.

La forma de reflejar el tiempo que se dedicó a esta fase es mediante una aplicación interactiva construida con la librería **shiny**. Esta aplicación se compone de distintas pestañas, cada una enfocada a permitir la visualización de unas variables o conjuntos diferentes.

- La primera pestaña, Cangrejos individuales, tiene un selector para indicar si se quiere visualizar un resumen de información del conjunto mof o de occ. Este resumen, que está en la primera subpestaña de esta pestaña, consiste en el uso de las funciones `skim` y `summary` sobre el conjunto indicado. Hay una segunda subpestaña que muestra una reducida tabla de cangrejos, ordenados de mayor a menor longitud de cuerpo. El largo de esta tabla también puede seleccionarse gracias a un selector numérico, y con un selector de rango puede elegirse el conjunto de cangrejos del que se quiere obtener los de mayor tamaño. Hay un checkbox para aplicar un filtro extra y ver los cangrejos Inmaduros del rango seleccionado, ya que de lo contrario lo habitual es que solo se abarquen los individuos maduros en la tabla.

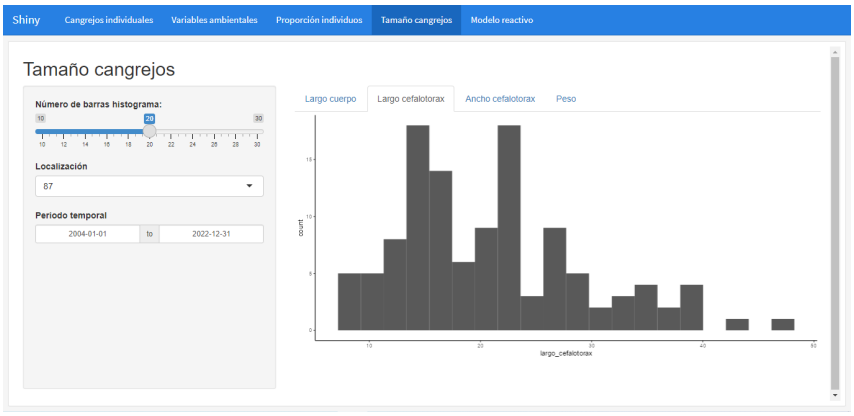


Figura 4.1: Captura documento shiny

- La segunda pestaña, Variables ambientales, tiene un selector que filtra los datos del conjunto pres_abs para reducirse a los de una localización en concreta. Hay también un selector para cada variable ambiental, cada uno añade su correspondiente variable a la gráfica en caso de ser seleccionado, y la elimina en caso de deseleccionarse.

- La tercera pestaña, Proporción de individuos, permite ver información sobre los cangrejos capturados en una localización concreta y para un rango de fecha personalilzable con un selector. Hay dos subpestañas, en función de si quiere verse la proporción de Machos/Hembras o de individuos Maduros/Inmaduros. Se muestra el número de capturas por cada fecha con algún registro, así como la proporción total y en cada fecha de la variable de interés.

- La cuarta pestaña, Tamaño cangrejos, muestra un histograma para cada una de las cuatro variables relacionadas con el tamaño que se tomaron en los cangrejos. El número de barras se puede elegir con un selector, así como la localización y el rango de fechas sobre los que quiere visualizarse la información.

- La quinta pestaña, Modelo reactivo, sirve para visualizar los resultados de los modelos KNN de clasificación en función de determinados parámetros. Los selectores de esta pestaña son: la variable objetivo (Sexo o Madurez), el número de vecinos utilizado y el exponente de la distancia de Minkowski utilizado. El modelo se genera al pulsar el botón de “Actualizar”, y se muestran la precisión, el área bajo la curva roc, y una gráfica mostrando la matriz de clasificación. Como se comentó brevemente en el apartado de ajuste de hiperparámetros, la rapidez que permite esta aplicación para variar los parámetros y ver cómo eso afecta a los resultados sirve para resaltar la importancia de un buen ajuste.

La aplicación permite generar un gran número de gráficas disintas en un tiempo reducido, pero estas no siempre aportan información relevante sobre el conjunto de datos. Es por eso que algunos de los puntos más destacables que se descubrieron a lo largo de la fase exploratoria se detallan en los apartados siguientes de este capítulo. Puede verse en la Figura 4.1 una captura de la aplicación generada.

4.2. Diferencias entre hábitats

Se pueden representar las coordenadas de las distintas localizaciones, obteniendo así una visión orientativa a modo de mapa del terreno sobre el que se realizó el estudio. Incluso puede identificarse para cada localización si se trata de una Laguna o de una Marisma.

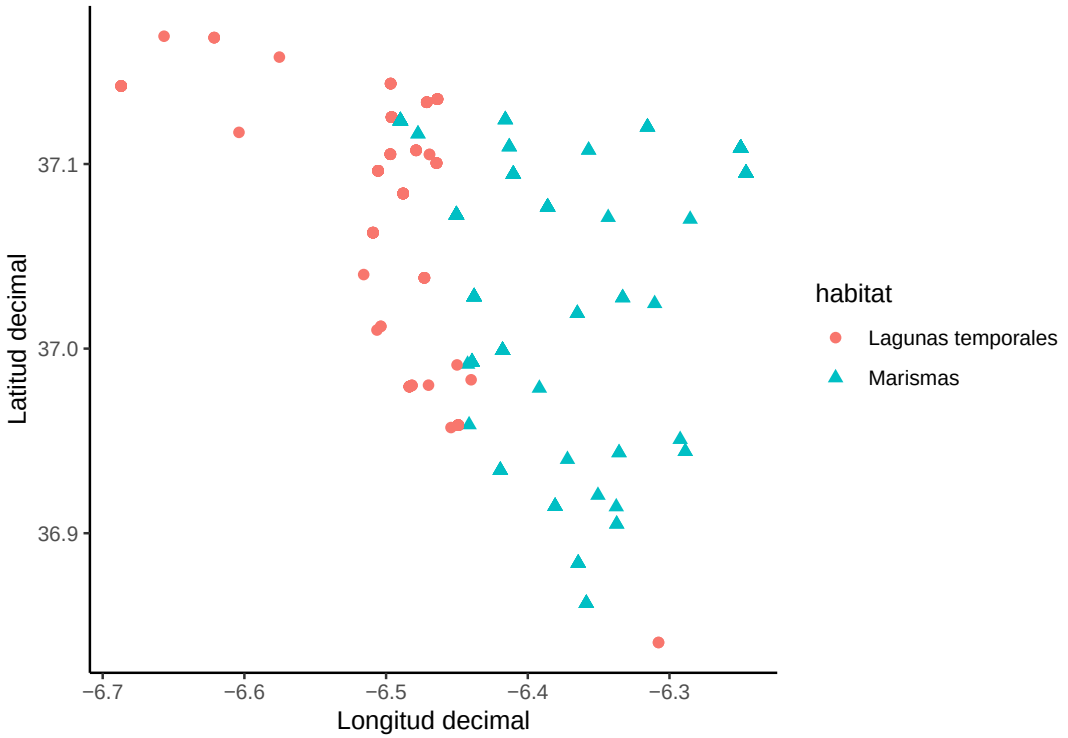


Figura 4.2: Localizaciones geográficas

Una vez visto que ambos tipo de hábitat están en zonas bastante diferenciadas, podría surgir la duda de si en cada una de estas zonas el tamaño de los cangrejos tiene una diferencia significativa o si esta diferencia en la localización no es un factor relevante.

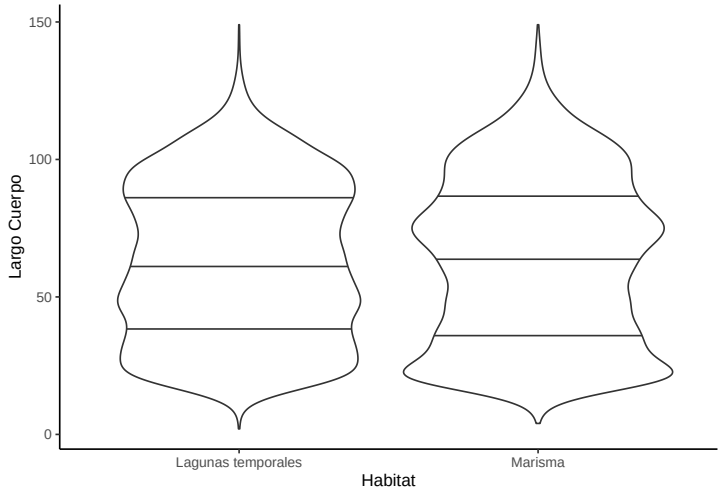


Figura 4.3: Longitud de los cangrejos según el hábitat

A simple vista parece que la distribución que sigue la variable "Largo cuerpo" en ambos hábitat es bastante similar, los cuartiles tienen valores muy similares en ambos grupos. Esto permitirá que los modelos que utilicen las variables relacionadas con el tamaño no tengan que ser tratados de forma distinta, sino que pueda englobarse la población total.

4.3. Variables ambientales

Es importante observar las correlaciones entre las variables, ya que de cara a algunos modelos podría suponer un problema en caso de haber multicolinealidad. Dado que en el conjunto de datos ambientales hay quince variables numéricas, se podría representar su matriz de correlaciones, pero el número de pares posibles es excesivamente elevado, y la mayoría de valores son tan bajos que no merece la pena. En vez de hacer la representación total, se han extraído los dos pares de variables que tienen una correlación en valor absoluto mayor de 0.8: Salinidad-Conductividad y Oxígeno disuelto en porcentaje-Oxígeno disuelto en valor absoluto. Para ambos pares se hace la representación de los pares de valores colocando cada variable en un eje, así como el ajuste de la variable en el eje Y respecto a la del eje X mediante `geom_smooth()`.

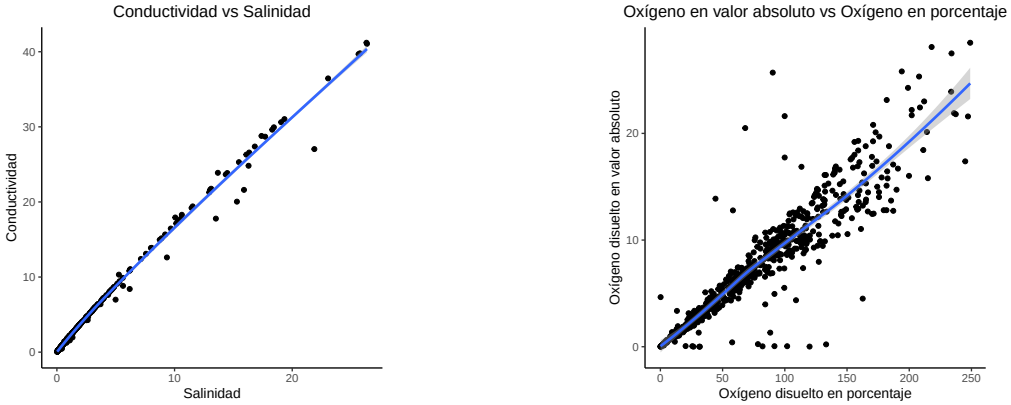


Figura 4.4: Variables ambientales más correlacionadas

La relación que se observa entre el primer par de variables se debe a propiedades químicas que eran esperables, la salinidad del agua favorece su conductividad. Viendo el ajuste se puede observar que ambas siguen una tendencia casi lineal con una dispersión considerablemente pequeña, apenas hay valores que se alejen de la diagonal.

La relación entre el segundo par de variables se debe a que ambas tratan sobre el mismo tipo de dato, solo que en distintas escalas. Incluso así, el ajuste que se puede realizar entre ambas no es tan bueno como en el primer caso, conforme aumenta el valor de las variables también aumenta la dispersión.

La correlación que tienen ambos pares de variables es muy alto, lo cuál significa que de cara a futuros modelos se debe tener en cuenta que incluir ambas puede producir problemas derivados de la multicolinealidad. Concretamente, del primer par se eliminará la Salinidad, ya que tiene 237 valores perdidos, mientras que la Conductividad tiene 11. De ese modo se evita inutilizar un gran número de registros, pero sin perder apenas información.

Aparte de estos dos pares de variables, las únicas con una correlación considerable eran Fecha-Año, ya que Fecha fue obtenida de la combinación entre Año y Mes, pero su finalidad es únicamente para que en las gráficas pueda apreciarse mejor la evolución de las variables. Por otro lado, la Temperatura está algo relacionada (aunque bastante menos que en los casos anteriores) con la Fecha y con el Año, en cambio no con el Mes. La relación que tienen estas variables no es tan clara como las dos anteriores, pero también es conveniente ver cómo evoluciona. La temperatura tiene un comportamiento que parece ser cíclico a lo largo de los años del estudio; si se ignora el año y se considera solo la variable Mes, puede verse que esta evolución es la que cabía esperar: mayores valores en los meses centrales (verano) y valores más reducidos en el invierno.

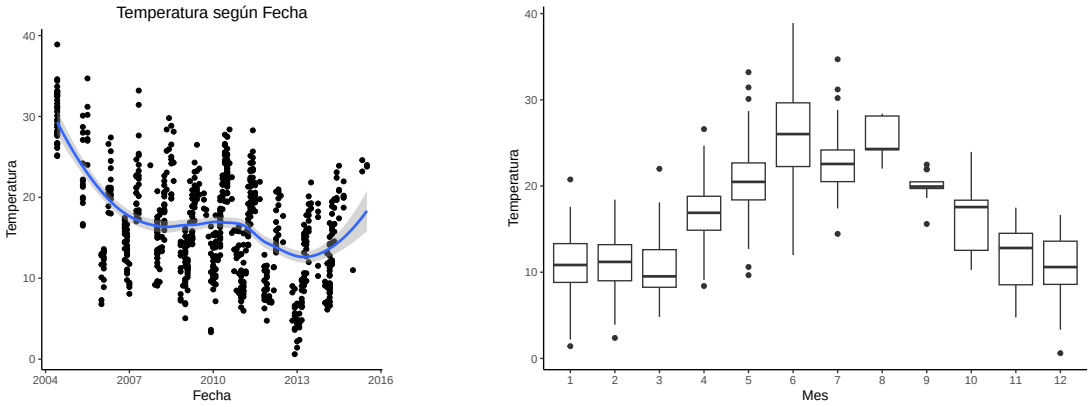


Figura 4.5: Evolución de la temperatura

Por otro lado, es relevante observar cómo varían los valores de Conductividad (y por extensión los de Salinidad) en función de si se trata de una Marisma o una Laguna.

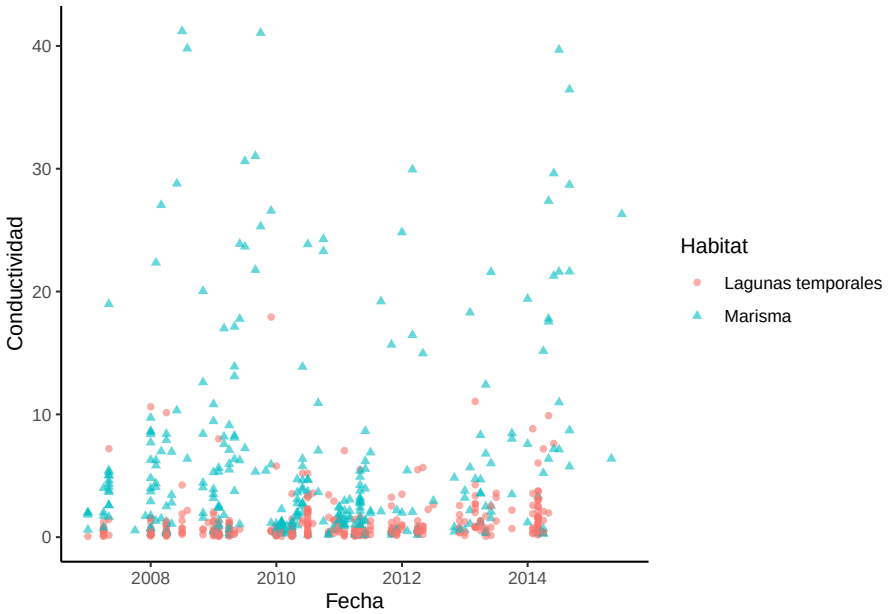


Figura 4.6: Conductividad según hábitat

Puede observarse que en las Marismas se dan los valores más altos de esta variable, aunque también se dan algunos valores reducidos, mientras que en las Lagunas los valores

son bajos en la mayoría de los casos. Esto indica que, de cara a los análisis que engloben esta información en distintas localizaciones, se debe hacer una separación entre las Lagunas y las Marismas.

4.4. Tamaño según sexo y madurez

Es interesante plantearse también si el tamaño de los cangrejos varía mucho en función de si son macho o hembra. Esta información podría ser útil para poder clasificar el sexo de los cangrejos en función de su tamaño.

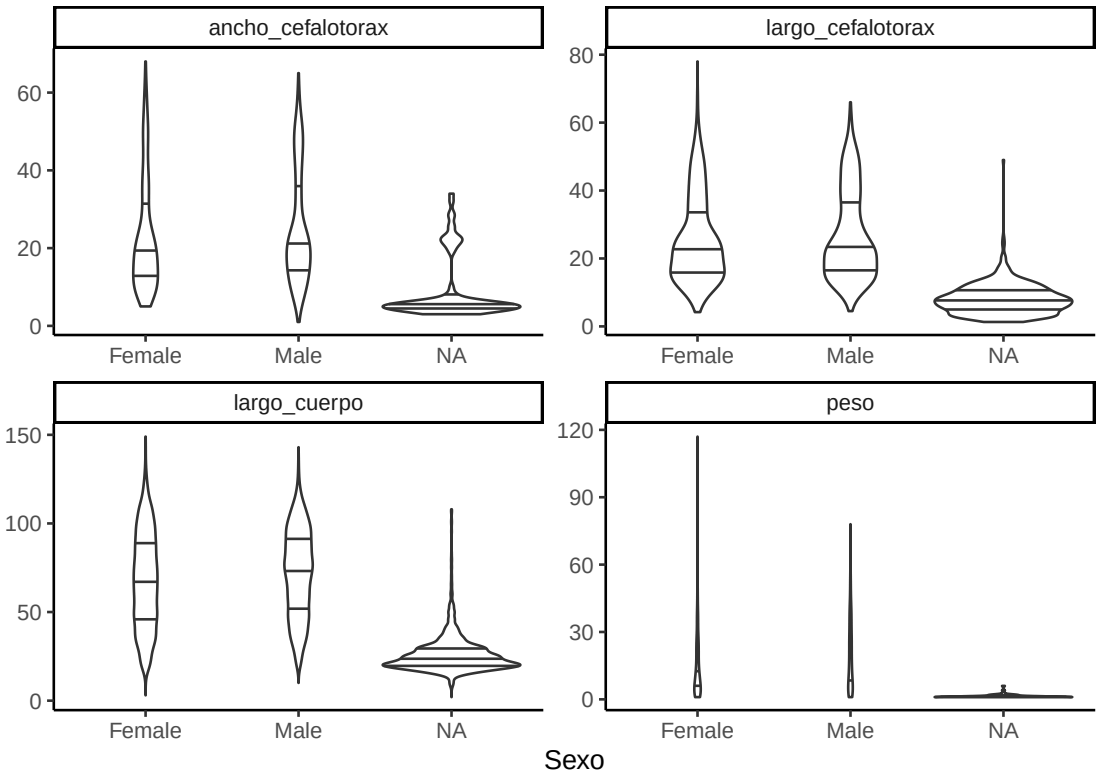


Figura 4.7: Distribución de las medidas de tamaño según sexo del cangrejo

La distribución entre machos y hembras parece ser similar en las cuatro variables, pero en cambio los valores perdidos actúan de manera bastante diferenciada. Esto puede dificultar la clasificación de los valores perdidos en una de las dos categorías si se intentase realizar únicamente en base a estas cuatro variables.

Para entender el motivo de que la distribución de los valores perdidos tenga este comportamiento, puede ser útil realizar esta misma representación pero respecto a la madurez de los cangrejos.

Tabla 4.1: Tabla cruzada Madurez y Sexo

	Female	Male	NA
Immature	3438	3079	2451
Mature	3059	3419	25
NA	6	5	0

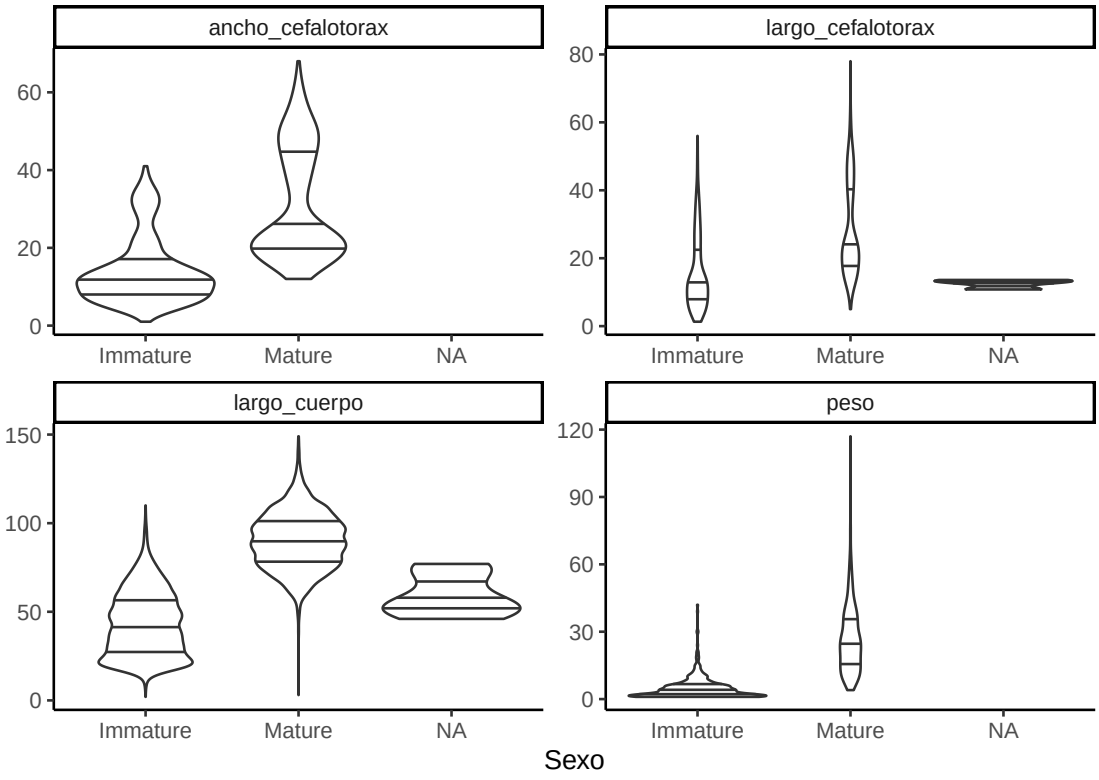


Figura 4.8: Distribución de las medidas de tamaño según madurez del cangrejo

En este caso las variables sí tienen comportamientos significativamente distintos entre los individuos Maduros e Inmaduros, lo cuál es bastante razonable. Significa también que una clasificación sobre la etapa de Madurez del individuo probablemente fuese más adecuada que una sobre el sexo del individuo. La clave está en observar qué relación siguen los valores perdidos de cada una de las variables como puede verse en la Tabla 4.1.

Puede apreciarse que casi todos los datos perdidos del sexo son en individuos inmaduros, seguramente porque esto dificulta su clasificación, y al tener un tamaño menor que los individuos maduros eso hace que los valores perdidos del sexo tengan un tamaño considerablemente menor que los que fueron catalogados como machos o como hembras.

4.5. Capturas en trampas

En un principio esta variable aparentaba tener una gran cantidad de información disponible, lo cual es cierto, pero la forma en que fueron recogidos dificulta la interpretación del tamaño de la población de cangrejos en base a estos datos. No se trata de una variable que sea constante en el tiempo, sino que en cada fecha el número de localizaciones en que se

realizaron capturas es disinto. Tampoco los registros son periódicos, sino que se reparten a lo largo de los años del estudio sin seguir un patrón y siendo escasos en algunos períodos extendidos de tiempo. El número de registros en una localización concreta tampoco son suficientemente numerosos como para poder estudiar la evolución de los cangrejos en ninguna de las localizaciones disponibles. Todo esto hace que interpretar esta variable para sacar conclusiones sobre el tamaño de la población en Doñana a lo largo del estudio sea complicado.

Por tanto, de cara a tratar de estimar la evolución que ha seguido la población de cangrejos, es difícil representarlo gráficamente. Tal vez un mes hubo menos capturas, pero porque solo se visitaron un pequeño número de localizaciones. En cambio, si en otro mes que se visitaron muchas más localizaciones o a lo largo de más días se hubiera alcanzado el mismo número de capturas. Es necesario poder ver reflejado que, en esa situación, en cada lugar se habrían recogido menos cangrejos (aunque de forma acumulada formasen un número considerable). La forma de corregir esto será mediante una transformación de los datos, pero antes de representarlo conviene mostrar los valores obtenidos sin importar fecha ni localización, para tener una idea del rango que abarcan y la frecuencia de aparición.

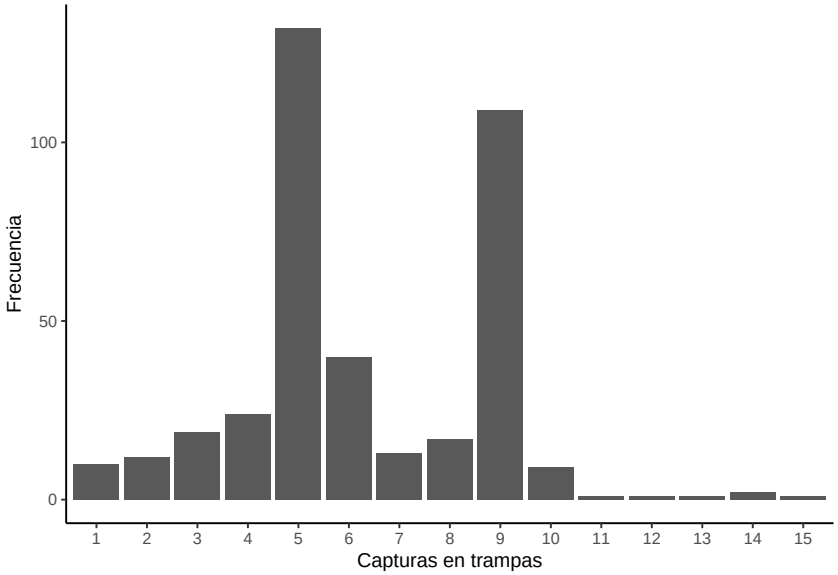


Figura 4.9: Distribución capturas en trampas

Parece ser que la distribución seguida es bimodal, siendo los dos valores con mayores frecuencias 5 y 9 individuos por trampa y día. Cabe destacar que la gráfica anterior engloba los dos tipos de hábitat, pero cada uno de ellos tienen una distribución considerablemente distinta.

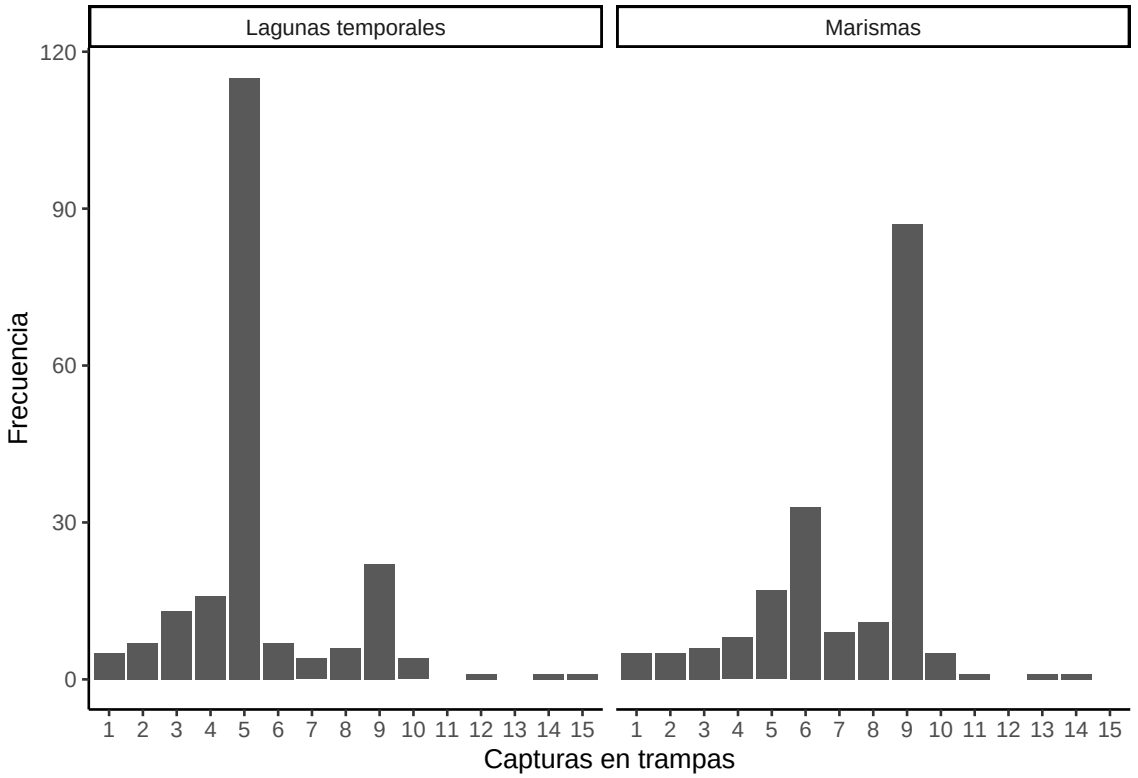


Figura 4.10: Distribución capturas en trampas según hábitat

En las lagunas temporales la frecuencia más habitual es 5, mientras que en las marismas es 9 (aunque siga habiendo ligeros picos en los valores contrarios). Esto indica que en cada tipo de hábitat, o tal vez incluso en cada localización distinta, los valores de cangrejos capturados pueden ser considerablemente distintos. Por tanto, se confirma que el detalle antes mencionado de separar los datos según el hábitat debe ser tenido en cuenta si se quiere analizar la evolución de las capturas a lo largo del tiempo.

Para poder hacer este análisis y tener en cuenta al mismo tiempo la localización en que se registraron las capturas, se realizará una poderación de los valores antes mostrados. La forma de hacerlo será evaluando la media de capturas en cada una de las localizaciones, y respecto a esa media se ponderarán las capturas obtenidas cada día por cada localización. De ese modo, un valor cercano a 1 representaría un número de capturas estándar, cuanto mayor fuese el valor este mayor sería el número de capturas en ese día respecto a lo que se suele encontrar, y viceversa.

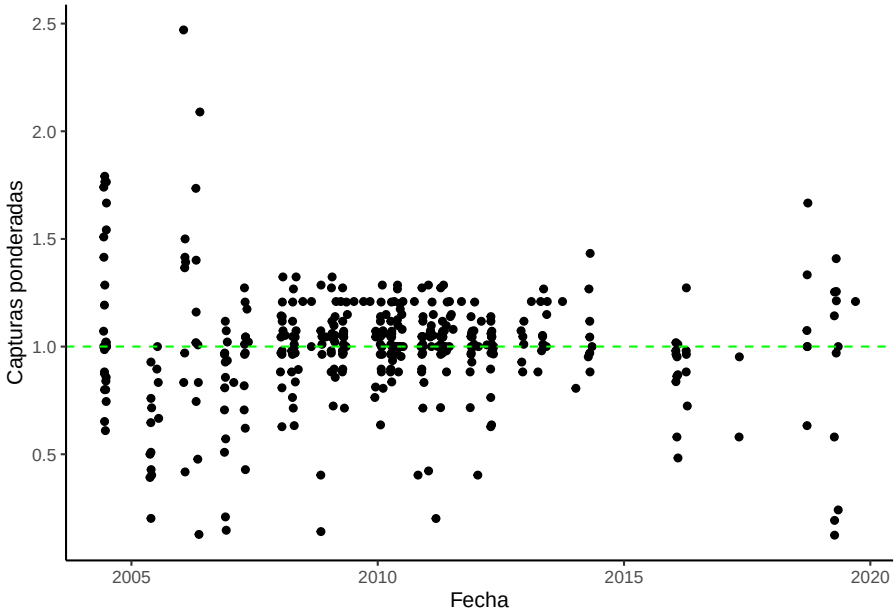


Figura 4.11: Capturas ponderadas

Puede verse que en general los puntos están distribuidos de forma bastante equilibrada. En las fechas que en algunas localizaciones se realizaron más capturas hubo otras en las que se realizaron menos, lo cual parece indicar que se trataba de una fluctuación local, pero no de una evolución por parte de la población local de cangrejos.

Por ver de forma más concreta el comportamiento que seguían las capturas, pueden visualizarse los seis localizaciones que tuvieron mayor número de capturas:

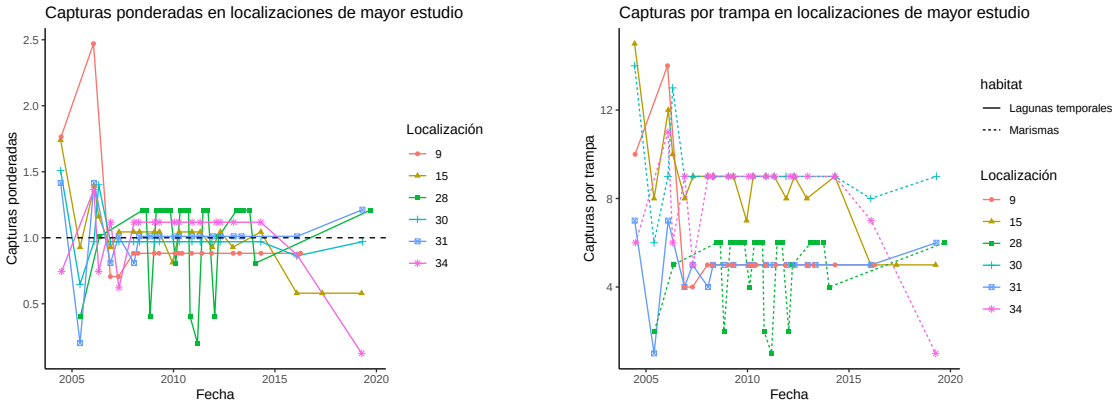


Figura 4.12: Capturas en localizaciones más estudiadas

En la gráfica de la izquierda se pueden ver las capturas ponderadas, todas están alrededor del valor 1 aunque pueden observarse fluctuaciones a lo largo del tiempo. En la gráfica de la derecha se representan los valores reales, además de diferenciar con el tipo de línea aquellas localizaciones que son Marismas de las que son Lagunas. Puede verse que los valores se agrupan sobre todo a alturas 5 y 9, que eran los que se vio que eran más comunes, aunque a diferencia de lo que cabría esperar los dos hábitats se encuentran mezclados, no hay una división clara. Una de las marismas tiene valores más cercanos a

5, que era habitual en las lagunas, y una de las lagunas tiene valores más cercanos a 9 la mayoría del tiempo.

Algo que llama la atención de ambas gráficas es que en la localización 34 los dos últimos registros reflejan una disminución drástica de la cantidad de cangrejos capturados, a pesar de que en los anteriores registros casi siempre tenía valores altos. Esto podría tener múltiples explicaciones, ya sea por un desplazamiento de los cangrejos de esta localización a alguna cercana, o debido a las condiciones de dicho lugar. Sin embargo responder a esta cuestión quedaría en mano de aquellos ecólogos que realizaron el trabajo de campo.

Otra pregunta que puede hacerse es si la cantidad de cangrejos se ve influenciada por la estación. Para tratar de responder a la misma, se agrupan las capturas en intervalos de tres meses y se suman todas las que se realizaron por cada localización. Después de eso vuelve a aplicarse la ponderación, pero con los nuevos valores de capturas.

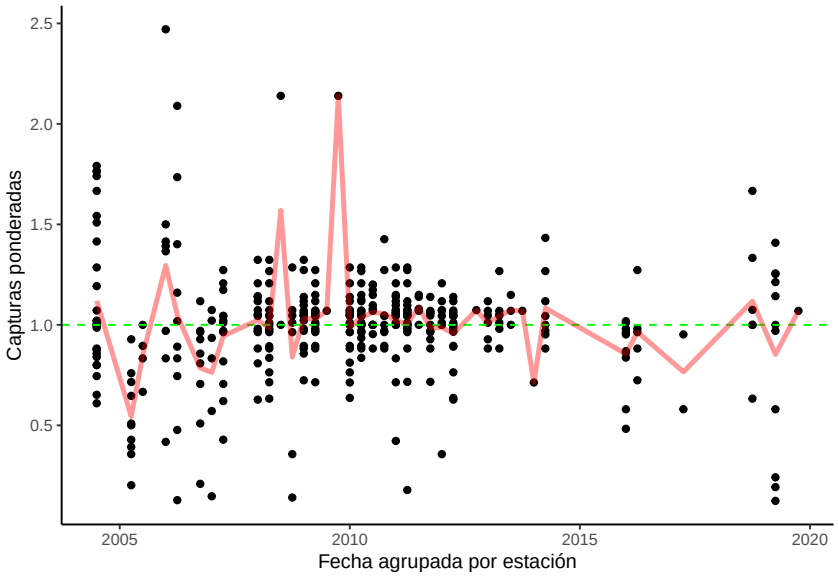


Figura 4.13: Capturas ponderadas agrupadas por estación

La distribución agrupando las capturas por trimestre no varía demasiado, pero al reducir el número total de puntos en la gráfica ayuda a que se vea más claramente y puede representarse la media de cada fecha sin que sea demasiado confisa la gráfica. De nuevo, no parece que haya ido evolucionando por épocas, sino que en cada estación hubo valores por encima y por debajo de la media habitual en distintas localizaciones. Los mayores picos se corresponden a fechas con pocos registros (uno o dos en los casos más altos) aunque el segundo trimestre en el que se realizó el muestreo sí que presenta sus ocho registros con valores por debajo de la media.

Lo mismo ocurre al visualizar las seis localizaciones con mayor número de registros. En algunas las capturas se mantienen de forma más estable mientras que en otras está más disperso, pero en general el comportamiento no dista demasiado del que se vio en la Figura 4.12 con las capturas diarias. El IDL 28 es el único que muestra algunos valores elevados que no tenía con la anterior agrupación y ponderación.

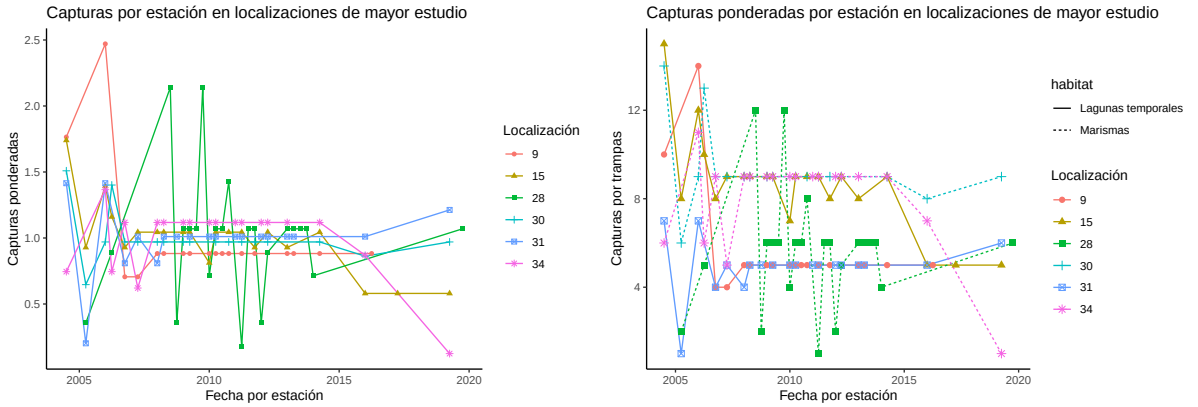


Figura 4.14: Capturas por estación en localizaciones más estudiadas

Aparte de ver la evolución a lo largo de los años del estudio, cabría preguntarse si el número de capturas de cangrejos se ve influenciado por la estación. Una forma de hacer la comparación es agrupar para cada estación todas las capturas realizadas y visualizar la distribución de valores registrados en cada una. Además de ello, en las Tablas 4.2 y 4.3 pueden verse respectivamente las medias y las desviaciones típicas de las capturas en cada uno de los subgrupos, según hábitat y según estación.

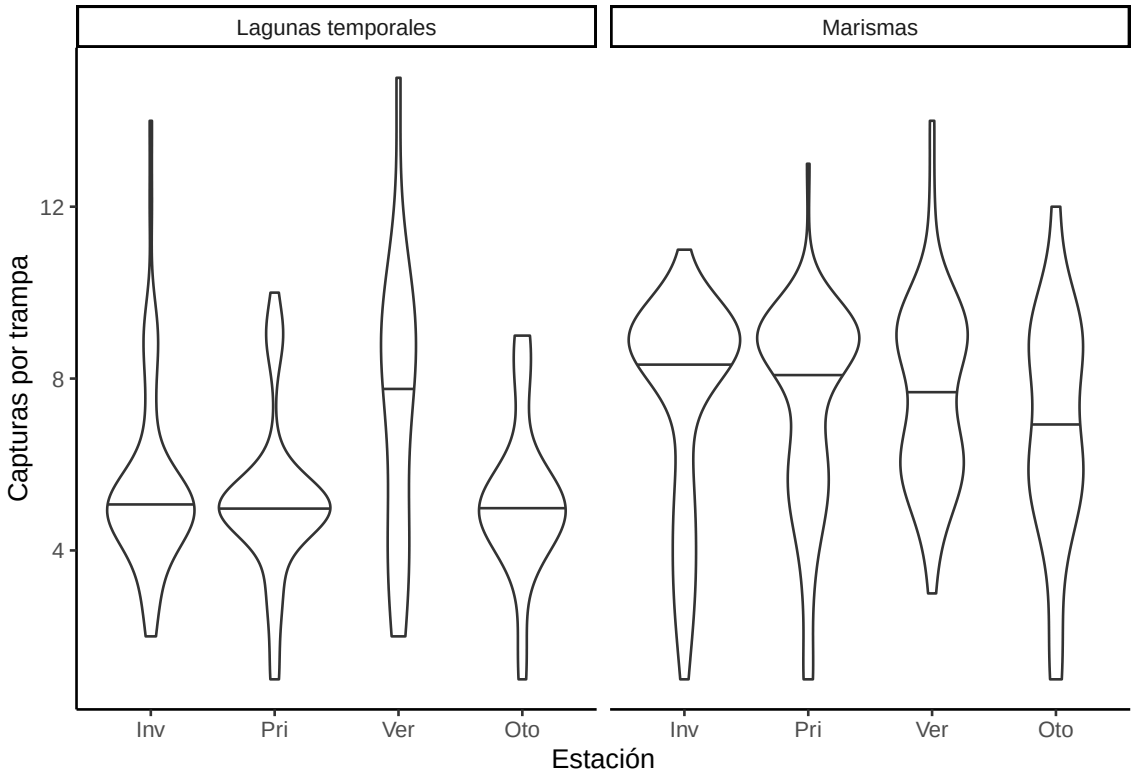


Figura 4.15: Capturas según estación y hábitat

La media de capturas es bastante constante en cada uno de los hábitats, quitando el verano en las lagunas, que tiene un valor más alto que el resto de estaciones. La diferencia entre hábitats en cambio sí que es significativa, siendo en todas las estaciones mayor la media de capturas en las marismas que en las lagunas. Primavera e invierno tienen una

Tabla 4.2: Capturas en trampas, media por hábitat y estación

habitat	Inv	Pri	Ver	Oto
Lagunas temporales	5.48	5.1	7.35	5.19
Marismas	7.44	7.3	7.65	6.80

Tabla 4.3: Capturas en trampas, desviación típica por hábitat y estación

habitat	Inv	Pri	Ver	Oto
Lagunas temporales	2.14	1.85	3.30	1.79
Marismas	2.48	2.42	2.19	2.69

moda mucho más marcada en ambos casos, igual que el otoño en las lagunas. En cambio en verano parece que los datos se distribuyen de forma menos marcada, al igual que en otoño en las marismas.

La mayor desviación típica ocurre en las lagunas en verano, que es justo el caso anómalo en cuanto al valor medio. Esto podría estar justificado ya que, como se ve en la Tabla 4.4, es el grupo que menor número de registros tiene con diferencia, con solo 17. Otoño en las marismas es el siguiente con menos registros, 20, y es también la estación cuya media se aleja más de las otras en las marismas, y la que tiene la segunda mayor desviación típica.

El otoño y el verano son las estaciones en las que un menor número de registros se realizaron en ambos hábitats, en comparación con primavera e invierno. Esta diferencia es bastante más drástica en las Lagunas temporales, aunque esta división se cumple en ambos casos. En general, parece que la desviación típica viene determinada inversamente por el número de capturas. De igual forma, la media de capturas no depende tanto de la estación como del hábitat, y los casos en que una estación se aleja del resto coinciden con aquellas en las que menos registros se tomaron.

Comparando los datos ponderados (al dividir el número de capturas por la media de la localización correspondiente) con los datos originales, puede verse que en todos los casos la moda es bastante más acentuada. El único periodo que sigue una distribución casi uniforme es el verano en las Lagunas temporales, tal vez debido al disminuido número de registros que se tomaron, lo cual hace que tenga mayor variabilidad que los demás casos.

Tabla 4.4: Número de registros por estación y hábitat

Estación	Lagunas	Marismas
Inv	61	50
Pri	98	73
Ver	17	43
Oto	26	20

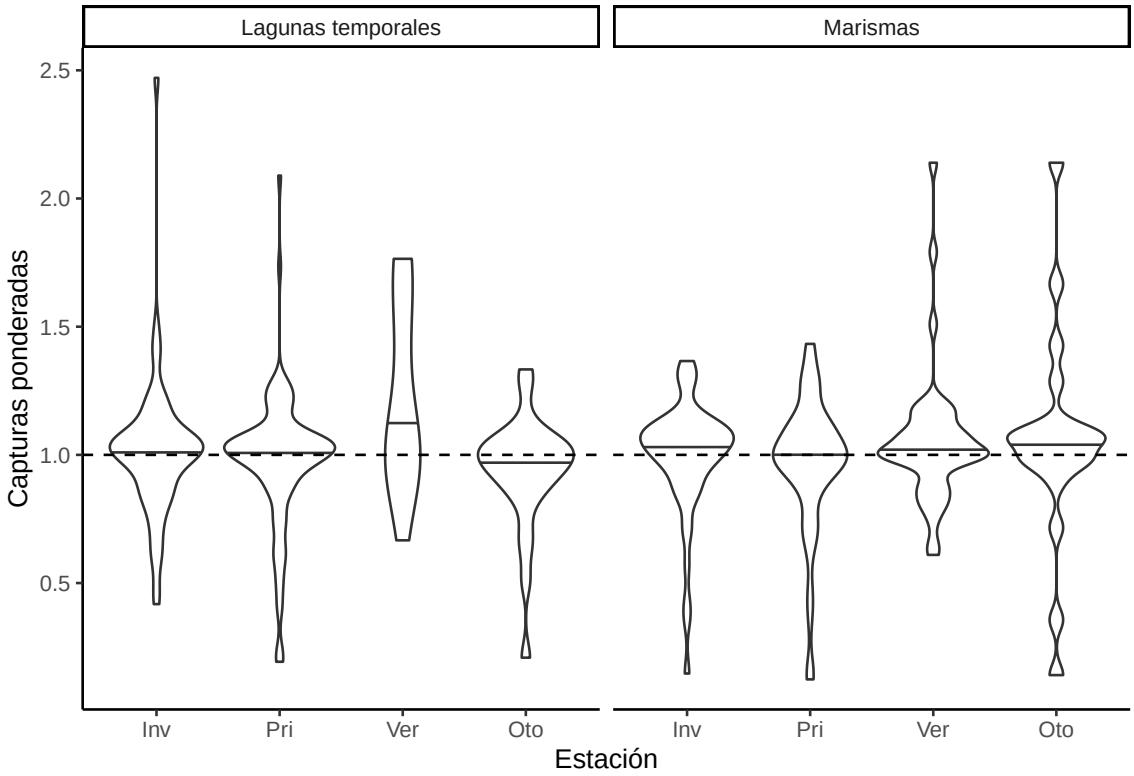


Figura 4.16: Capturas según estación y hábitat

En general se puede concluir que el comportamiento seguido en cada estación es bastante similar (quitando aquel con menor número de registros, el verano en las lagunas temporales) teniendo valores centrados en 1 por la naturaleza de la transformación que se llevó a cabo en los datos. Destacan ligeramente en ambos hábitats el invierno, estando centrados en valores ligeramente superiores a 1, que puede indicar que en este periodo sea más abundante la población de cangrejos en Doñana.

4.6. Proporción de individuos inmaduros

Las representaciones anteriores eran referidas a capturas realizadas en trampas. Para cada registro se indica el número de cangrejos capturados, pero ninguna información más allá. Hay un segundo tipo de datos recogidos, en el cual las capturas se hacían de forma individual, y para cada cangrejo se registraba el sexo y el estado de madurez. Gracias a estos datos puede estuciarse la evolución en la cantidad de individuos maduros e inmaduros a lo largo del año, lo cual puede ayudar a comprender el ciclo reproductivo del cangrejo.

Para las siguientes gráficas se trabajará con este conjunto de datos. Haciendo la agrupación por meses de las capturas registradas, se observa lo siguiente:

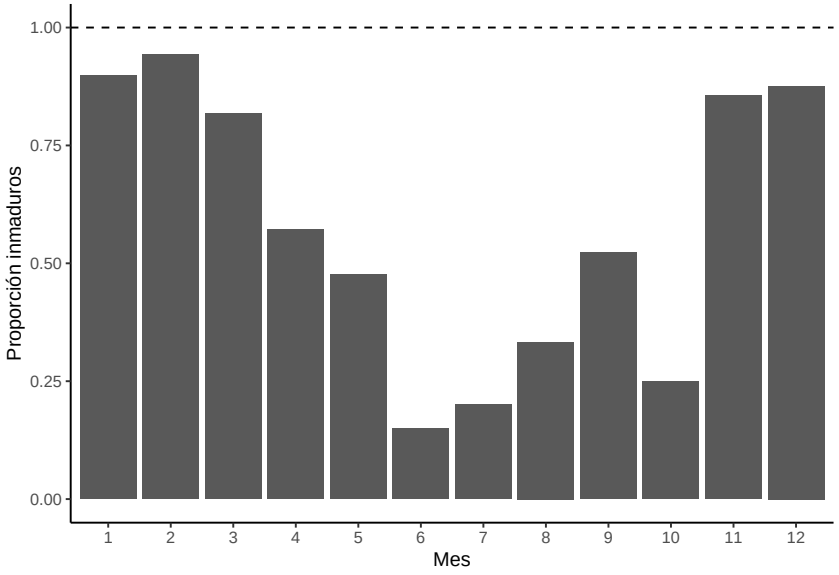


Figura 4.17: Proporción de inmaduros capturados por mes

En la gráfica se muestra cómo evoluciona la proporción de individuos Inmaduros a lo largo del año, en los primeros meses prácticamente todos los individuos capturados son crías. El menor porcentaje de individuos inmaduros se da en Junio, y en general durante los meses de verano es menor la proporción, volviendo a aumentar considerablemente en Noviembre. Hay que tener en cuenta también que esta gráfica que solo muestra la proporción no aporta información sobre el número de individuos capturados en cada mes. Para ello se mira la siguiente gráfica:

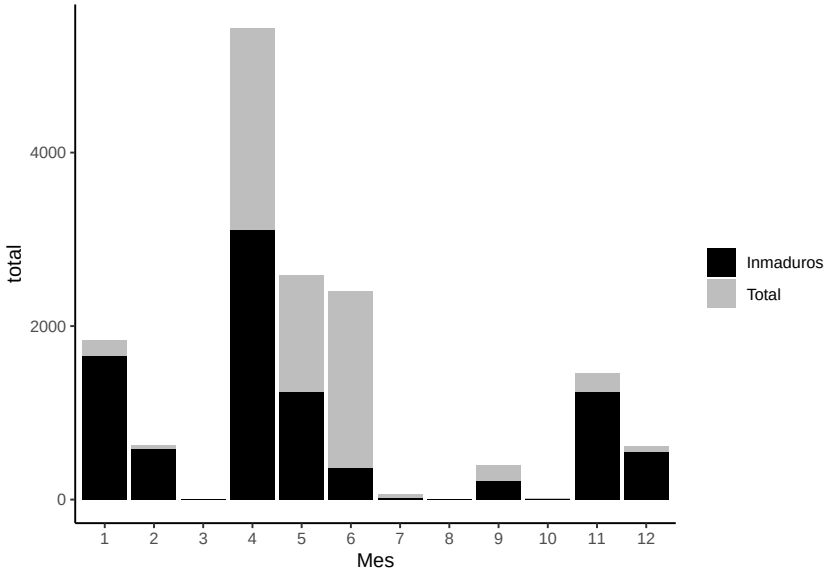


Figura 4.18: Individuos inmaduros y totales capturados por mes

Puede verse que entre Abril y Junio fue cuando mayor número de individuos se capturaron. En cambio, entre Julio y Octubre el número de capturas fue bastante inferior a los de otros meses, así como en Marzo tampoco se registraron apenas individuos. Eso

significa que en estos meses el porcentaje puede estar algo sesgado a causa del tamaño muestral recogido en cada mes.

Para solucionar esto, se agrupan los meses de tres en tres en el orden natural, esto coincide con agrupar los tres meses que mayor número de individuos registrados acumulan.

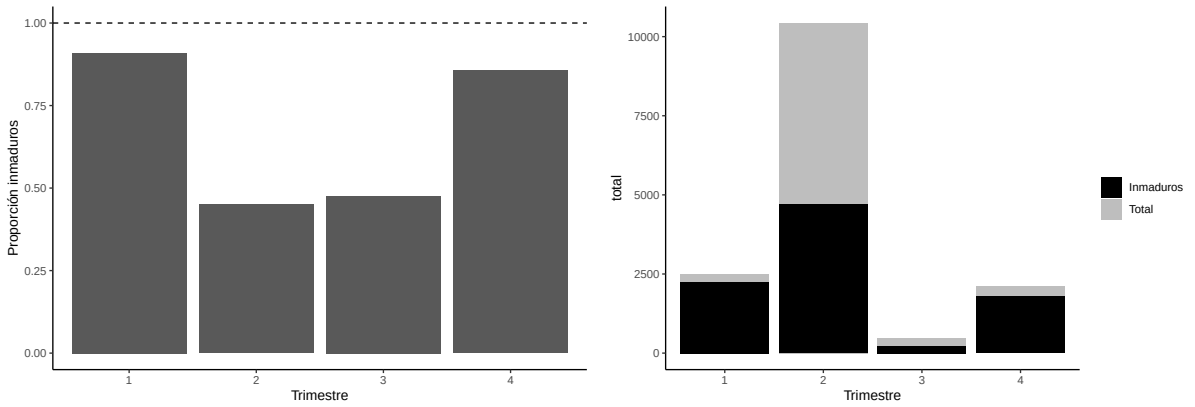


Figura 4.19: Individuos inmaduros por estación

Al agrupar en trimestres, la información que antes se tenía en 12 columnas ahora está en solo 4. Se puede ver en la primera gráfica que la proporción de individuos inmaduros es mucho mayor al inicio y al final del año, mientras que en los trimestres centrales el número de individuos maduros e inmaduros que se registraron está prácticamente igualado. De nuevo se representa también esta información pero poniendo en escala el número de capturas de cada periodo.

En la segunda gráfica se puede apreciar el total de individuos que fueron registrados en cada trimestre. Era de esperar dada la gráfica con la representación por meses que el trimestre con mayor número de registros fuera el segundo, teniendo más que los otros tres juntos. Eso significa que, aunque el valor de la proporción fuera similar al del tercer trimestre, en el segundo es mucho más fiable, pues el número de registros es veinte veces superior (más de 10000 frente a casi 500). En el primer y último trimestre tanto el número de capturas como la proporción de individuos inmaduros son bastante similares.

Parte III

Modelos predictivos

Capítulo 5

Introducción a los modelos predictivos

Tras haber realizado el estudio descriptivo de los datos, ya se tiene una idea general de cómo se comportan y cómo interaccionan entre ellas las variables, por lo que ha llegado el momento de intentar crear modelos que las ajusten de la mejor forma posible. Los modelos predictivos pueden ser de dos tipos: de clasificación, si tratan de predecir una variable categórica, o de regresión, si tienen como objetivo la predicción de una variable numérica. En este capítulo se hará un esbozo del guion que se seguirá para crear los modelos, con algunos trozos del código para explicar cómo fueron llevados a cabo. Los otros dos capítulos que componen este bloque se dedican a explicar dichos modelos, reflejando los resultados que se obtuvieron para cada una de las variables, y las conclusiones que se alcanzan.

En cada uno de los capítulos se estudian varias variables, aplicando distintos métodos a algunas de ellas para poder compararlos entre ellos, y también hacer comparaciones entre las variables. En el de clasificación son la variable Sexo y la variable Madurez las que se intentan predecir, utilizando como variables predictoras las variables de tamaño (largo del cuerpo, largo del cefalotórax, ancho del cefalotórax y peso), aunque en ocasiones se usan todas y en otras solo el largo del cuerpo. En el de regresión se estudian las variables de tamaño, intentando predecirlas en función de las demás, y la variable de las capturas ponderadas, en función de las variables ambientales.

5.1. Ideas iniciales

Antes de explicar los modelos llevados a cabo, merece la pena mencionar aquellas ideas que no llegaron a incorporarse en este estudio, y el motivo de que no fueran incluidas en el producto final. En aquellas variables que trataban sobre individuos concretos (tamaño, sexo, madurez) la línea a seguir estaba bastante clara, lo lógico era intentar modelizar cada una de ellas en función del resto de las variables. No es coherente intentar predecir lo que medirá un hipotético cangrejo futuro aleatorio, sin tener ni idea de alguna otra variable de tamaño. Tampoco clasificar si será maduro o inmaduro sin saber nada de lo que mide. Por tanto, aunque como se verá más adelante los modelos no siempre sean tan buenos como se pretendía, con los datos disponibles era lo único que podía hacerse, o al

menos lo único que humildemente pudimos concebir. La cuestión a ir desarrollando fue la elección de los modelos, y el ajuste de hiperparámetros en cada uno de ellos.

En cuanto a las variables ambientales, el asunto se complicaba de forma considerable. Por la forma en que se toman las mediciones (de forma mensual) una primera idea podría ser tratar esos datos como series temporales, ver su evolución a lo largo del tiempo e intentar modelizar eso para alguna variable como la temperatura o el pH. Sin embargo, la forma en que están tomados los datos hizo que eso fuera impracticable. Las mediciones se tomaban en localizaciones distintas, en los meses que había al menos un registro, generalmente había registros sobre varias localizaciones distintas. Pero en un gran porcentaje de los meses no se visitaba ninguna de las localizaciones. Intentar estudiar una localización en concreto era imposible, pues los registros estaban muy espaciados en el tiempo, y la cantidad de valores perdidos en medio era superior a la cantidad de registros tomados. Las técnicas habituales para tratar con series temporales requieren que no haya valores perdidos, y en caso de haberlos lo apropiado sería intentar imputarlos, pero no es coherente imputar un número de datos mayor que la cantidad disponible, sobre todo siendo esta tan pequeña.

Por tanto, predecir localizaciones individuales quedó descartado, así que habría que agruparlas de alguna forma para intentar tener suficientes registros como para crear un buen modelo. Intentar agrupar aquellas que fueron medidas en una misma fecha tenía como inconveniente que la mayoría de variables tenían un amplio rango y generalmente en un mismo mes podían alcanzar un rango prácticamente igual al rango total de la variable. Para verlo de forma más concluyente, se calcularon los valores máximos y mínimos alcanzados por cada variable en una fecha, además el rango de dicha fecha y la desviación típica de las observaciones. Se muestran dichos datos para las variables Temperatura y Nitrato, que tienen escalas bastante diferentes y en ambos hábitats son similares, pero en ninguno de los casos parece que agrupar los datos medidos en el mismo mes vaya a ser adecuado.

```
pres_abs_Temp <- pres_abs %>%
  filter(!is.na(Temperatura)) %>%
  group_by(Fecha) %>% summarise(
    across(
      c(Temperatura),
      list(max = ~max(., na.rm = TRUE),
           min = ~min(., na.rm=TRUE),
           rango = ~(max(., na.rm = TRUE)-min(., na.rm = TRUE)),
           sd = ~sd(., na.rm=TRUE)),
      .names = "{.col}_{.fn}"
      )
  ) %>% ungroup() %>% select(-Fecha)

summary(pres_abs_Temp)
```

```
##  Temperatura_max Temperatura_min Temperatura_rango Temperatura_sd
##  Min.   : 5.63   Min.   : 0.61   Min.   : 0.000    Min.   :0.1131
##  1st Qu.:15.06   1st Qu.: 8.40   1st Qu.: 3.250    1st Qu.:1.6683
##  Median :19.46   Median :13.17   Median : 6.630    Median :2.2641
```

```
##  Mean   :19.23   Mean   :13.26   Mean   : 5.972    Mean   :2.2612
##  3rd Qu.:23.91   3rd Qu.:18.53   3rd Qu.: 8.140    3rd Qu.:2.8410
##  Max.   :38.90   Max.   :25.95   Max.   :15.900    Max.   :5.0727
##                                                    NA's   :6
```

```
pres_abs_nit <- pres_abs %>%
  filter(!is.na(Nitrato)) %>%
  group_by(Fecha) %>%
  summarise(
    across(
      c(Nitrato),
      list(max = ~max(., na.rm = TRUE),
           min = ~min(., na.rm=TRUE),
           rango = ~(max(., na.rm = TRUE)-min(., na.rm = TRUE)),
           sd = ~sd(., na.rm=TRUE)),
      .names = "{.col}_{.fn}"
      )
  ) %>% ungroup() %>% select(-Fecha)

summary(pres_abs_nit)
```

```
##   Nitrato_max       Nitrato_min      Nitrato_rango       Nitrato_sd
##  Min.   :  0.104   Min.   :  0.048   Min.   :  0.000   Min.   :  0.1773
##  1st Qu.:  5.204   1st Qu.:  1.102   1st Qu.:  3.483   1st Qu.:  2.8190
##  Median : 67.688   Median :  1.994   Median : 40.128   Median : 17.6047
##  Mean   :177.884   Mean   : 23.641   Mean   :154.243   Mean   : 60.7002
##  3rd Qu.:188.350   3rd Qu.:  9.052   3rd Qu.:149.202   3rd Qu.: 76.3773
##  Max.   :998.500   Max.   :463.700   Max.   :993.293   Max.   :324.3659
##                                                        NA's   :6
```

Puede verse en el rango de la temperatura que en un mismo mes puede haber una diferencia de casi 16 grados en las mediciones de dos localizaciones distintas, por lo que intentar agruparlos con el valor medio daría lugar a un error de unos 8 grados, lo cual es demasiado. Para el nitrato, el rango máximo alcanzado es de 993, cuando el rango total de la variable no llega a 1000, por lo que en un mismo mes se han registrado valores cercanos a lo más bajo y a lo más alto que llega a alcanzar la variable en todo el estudio. Se descarta también la posibilidad de utilizar el valor medio de un mes para así tener valores en un número de meses mayor. Por tanto, con los datos disponibles no se puede llevar a cabo un estudio con el enfoque de series temporales.

Se probaron también técnicas de data augmentation como muestreo con reemplazamiento para así tener conjuntos de mayor tamaño, pero no se obtuvo ninguna mejora en los modelos. Finalmente, lo que pudo remediarse fueron los datos perdidos en algunos registros, aplicando imputación mediante knn, ya que de esa forma aquellos registros a los que les faltasen una o dos variables no tenían que ser descartados automáticamente. Esto se hizo mediante la función `step_impute_knn` de **tidymodels** tal como se puede ver en la sección 5.4.

También se hizo el intento de utilizar redes neuronales para intentar crear un modelo que mejorase los que se habían creado. Se partió de una red sencilla, utilizando la librería **neuralnet**, para realizar la clasificación de la variable Sexo. Tanto con la variable "Largo cuerpo" como con las cuatro variables del tamaño se obtenían resultados similares a los que se verán en la sección 6.1. Estos modelos están en auge dada su versatilidad y capacidad de predicción, pero es cierto que requieren de un gran entrenamiento y una dedicación considerable para obtener un modelo que realmente aproveche todo lo que esta técnica pueda ofrecer. Finalmente, terminó descartándose abarcar ese ámbito en este trabajo, ya que no era trivial conseguir un modelo que mejorase los que ya se habían desarrollado.

5.2. Análisis distribucional

Según el manual sobre Tendencias e índices de seguimiento de datos, conocido como TRIM (*TRends & Indices for Monitoring data*) [21], dedicado precisamente a analizar datos de esta naturaleza, el conteo de poblaciones debería seguir una distribución Poisson. Por tanto es lo que utilizan para intentar estimar poblaciones en dicho manual. Sin embargo, también inciden en que hay que tener en cuenta que en los casos reales la varianza suele ser mayor a lo que se esperaría de una Poisson. Aunque el software del que hablan en su manual, TRIM, no es el que se utilizará para este estudio, sigue siendo de interés comprobar si la información sobre el conteo de los cangrejos puede aproximarse con esta distribución.

En las gráficas antes representadas pudo apreciarse que la distribución que siguen las capturas con trampas en cada hábitat tenían una distribución algo distinta. Para saber si esta diferencia es realmente significativa, se realizará el análisis de la varianza ANOVA.

```
##              Df Sum Sq Mean Sq F value   Pr(>F)
## ev$habitat    1    328   328.0   63.72 1.63e-14 ***
## Residuals   389   2002     5.1
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

La hipótesis de igualdad de medias de la variable entre ambos conjuntos es rechazada (p-valor del orden de 10^{-14}), por tanto se tratará con la información de ambos hábitats por separado. De modo que el análisis sobre bondad de ajuste de la distribución Poisson se realizará por separado para ambos subconjuntos.

```
##
##   Goodness-of-fit test for poisson distribution
##
##              X^2 df     P(> X^2)
## Pearson 275.1126 13 3.802528e-51
```

En el caso de las marismas se rechaza la hipótesis de que los datos sigan una distribución Poisson (p-valor del orden de 10^{-51}), lo cual significa que un ajuste realizado con el software TRIM no sería apropiado, y tampoco permite hacer estimaciones con este tipo de ajuste.

```
##
##   Goodness-of-fit test for poisson distribution
##
##               X^2 df     P(> X^2)
## Pearson 271.5248 14 9.953403e-50
```

También para el caso de las lagunas temporales la hipótesis nula es rechazada (p-valor del orden de 10^{-50}) por lo que se concluye que los datos no siguen una distribución Poisson.

Ya que no cumplen lo que se esperaba de una variable de conteo poblacional, se puede intentar estimar con una distribución normal. Sin embargo, esto tampoco tiene éxito, ya que aplicando ambos conjuntos el test de normalidad de Shapiro-Wilk por separado se obtienen p-valores del orden de 10^{-12} (marismas) y 10^{-15} (lagunas).

```
##
##  Shapiro-Wilk normality test
##
## data:  .
## W = 0.86173, p-value = 4.213e-12
```

```
##
##  Shapiro-Wilk normality test
##
## data:  .
## W = 0.81464, p-value = 9.156e-15
```

Por tanto se rechaza que la distribución del número de capturas de cangrejos realizadas en trampas siga una distribución normal ni en las Marismas ni en las Lagunas. El enfoque paramétrico no es apropiado para los datos disponibles, por lo que se intentará recurrir a las variables ambientales para ver si esa información adicional ayuda a garantizar un mejor ajuste de la variable.

5.3. Modelos de clasificación

Para los modelos de clasificación se trató con las variables Sexo y Madurez. La primera de ellas tiene un porcentaje mucho mayor de valores perdidos, lo cual supone que la muestra disponible es menor. En ambos casos se utilizó un 70 % de los registros completos para la muestra de entrenamiento y un 30 % para la muestra test, de forma estratificada, mediante el comando `initial_split`:

```
occ_mof_comp_sexo_split <- occ_mof_completo_sexo %>%
  initial_split(prop=0.7,strata = sex)

occ_mof_comp_sexo_ent <- occ_mof_comp_sexo_split %>% training()
occ_mof_comp_sexo_test <- occ_mof_comp_sexo_split %>% testing()
```

Se aplicaron tres métodos de clasificación a cada una de las variables: k-vecinos más cercanos, bosques aleatorios y regresión logística.

Para el caso de los k-vecinos más cercanos, los modelos se construyeron mediante la librería **caret**, y se realizó el ajuste del parámetro k mediante validación cruzada. El número de valores comprobados fueron 11, como puede verse en el código, y antes de realizar el modelo se aplicó un centrado y escalado de las variables predictoras, de modo que todas las variables mantuvieran a priori un peso similar a la hora de realizar el cálculo de distancias.

```
ctrl <- trainControl(method="cv",classProbs=TRUE,
                     summaryFunction = defaultSummary )

KNN_todas_sexo <-
  train(sex ~ largo_cuerpo + largo_cefalotorax +
          ancho_cefalotorax + peso,
        data = occ_mof_comp_sexo_ent,
        method = "knn",
        trControl = ctrl,
        preProcess =c("center","scale"),
        tuneLength=11 )
```

Este mismo procedimiento se repitió utilizando como variable predictora únicamente el largo del cuerpo, y lo mismo para modelar la variable Madurez.

Como en bosques aleatorios no tiene sentido plantearse el modelo solo según el largo del cuerpo, esto se reservó para crear un árbol de decisión mediante la librería **C50**:

```
arbol_largo_sexo <- C5.0(sex ~ largo_cuerpo,
                         data = occ_mof_largo_sexo_ent)
```

El bosque aleatorio se hizo con la librería **randomForest**. El parámetro mtry se fijó en 3, sin aplicar ajuste, ya que se consideró que la mejora que se obtenía no era suficiente como para justificar el coste computacional.

```
bosque_todas_sexo <-
  randomForest(sex ~ largo_cuerpo + largo_cefalotorax +
                 ancho_cefalotorax + peso,
               data = occ_mof_comp_sexo_ent,
               mportance = TRUE, replace = TRUE,
               ntree=100, mtry=3)
```

La regresión logística se realizó mediante el comando `glm`, en este caso no fue necesario aplicar ingeniería de características.

```
logi_todas_sexo <- glm(sex ~ largo_cuerpo + ancho_cefalotorax +
                         largo_cefalotorax + peso,
                       family = binomial,
                       data = occ_mof_comp_sexo_ent)
```

El código reflejado en esta sección tiene como objetivo dejar claros los procedimientos seguidos, para así en el capítulo siguiente poder mostrar únicamente los resultados de forma más resumida. El código utilizado en cada uno de los modelos se encuentra en el Apéndice A.

5.4. Modelos de regresión

Las variables que fueron modelizadas mediante modelos de regresión fueron el largo de cefalotórax, ancho del cefalotórax, peso y las capturas ponderadas. En este caso, el porcentaje dedicado a la muestra de entrenamiento fue el 80 %, pues para la última variable el número de registros es reducido y era necesario un porcentaje algo superior para entrenar los modelos.

La división se generó de nuevo con `initial_split` de forma estatificada. Por cuestiones que se explican de forma más extensa en el capítulo de Modelos de regresión, para el ancho y el largo de cefalotórax se realiza también una división del conjunto inicial en función de la fecha de captura de los cangrejos. Una vez creada esa división, se crearon modelos de regresión lineal mediante `lm` para cada subconjunto y cada variable. El error se midió mediante validación cruzada, aplicando 10 pliegues, mediante la función `cv.lm` tal como se ve en el código a continuación:

```
reg_mof_larcf_f1 = lm(largo_cefalotorax ~ largo_cuerpo,
                      data=training(mof_larcf_f1_split))

VC_mof_larcf_f1 <- cv.lm(largo_cefalotorax ~ largo_cuerpo, m=10,
                         data = training(mof_larcf_f1_split),
                  seed = 1909, plotit = FALSE, printit = FALSE)
```

Para la variable peso los resultados obtenidos mediante regresión lineal no eran suficientemente buenos, ni realizando la división que se hizo para las otras dos variables. Por la forma de las predicciones del modelo, se realizó la transformación logarítmica, y gracias a eso mejoraron los resultados de forma notable. De nuevo el error cometido se midió mediante validación cruzada.

```
reg_log_peso = lm(log(peso) ~ largo_cuerpo,
                    data=training(mof_pes_split))
```

Como esta variable no había obtenido buenos resultados para la regresión lineal sin aplicar la transformación, se realizó también el ajuste mediante bosque aleatorio utilizando las tres otras variables de tamaño como predictoras. En este caso se llevó a cabo con la librería **caret**, y se aplicó el ajuste del parámetro mtry pudiendo tomar valores 1, 2 o 3. Con esto pudo obtenerse también un buen ajuste.

```
bosque_peso_tuning <-
  train(peso ~ ancho_cefalotorax + largo_cefalotorax +
          largo_cuerpo, data = training(mof_comp_split),
        method = 'rf', tuneGrid  = expand.grid(.mtry = 1:3),
```

```
        trControl = control <- trainControl(method='repeatedcv',
                                            number=10,
                                            search = 'grid'))
```

Para la variable capturas se realizó una agrupación por la variable Fecha que engloba Mes y Año, se calculó el número de capturas por trampas ocurridas en cada localización y mes. Posteriormente se halló la media por cada localización, y se realizó la ponderación para quedarse con la variable "capt" que lo que expresa es si el número de capturas fue superior o inferior a la media y en qué medida. Aquellas localizaciones con menos de tres registros a lo largo de todo el estudio fueron descartadas. Por último, se unieron con el conjunto de los registros de variables ambientales, en función de los que compartían fecha y localización.

```
ev_pres_abs <- ev %>%
  mutate(Fecha=year(eventDate)+(month(eventDate)-1)/12,
         IDL=localityID) %>%
  group_by(IDL, Fecha) %>%
  summarise(tot_capt = sum(sampleSizeValue,na.rm=TRUE)) %>%
  ungroup() %>% group_by(IDL) %>%
  mutate(med = mean(tot_capt, na.rm=TRUE),
         capt = tot_capt/med,
         cont= n()) %>% filter(cont > 3) %>%
  select(-tot_capt, -med, -cont) %>% ungroup() %>%
  inner_join(pres_abs, by=c("IDL", "Fecha"))
```

Los datos anteriores tienen un tamaño considerable, aunque en el caso de las variables ambientales el porcentaje de registros incompletos era superior a la mitad. Para solucionar esto, al aplicar el preprocesado de los datos mediante `recipe` se incluyó la imputación de las variables predictoras en función del resto de predictores numéricos. Aparte de eso, se eliminaron aquellas variables con una correlación superior a 0.8 en valor absoluto y se realizó un centrado y escalado de los predictores numéricos.

```
lagunas_imputado <- ev_pres_abs %>%
  filter(Habitat == "Lagunas temporales") %>%
  select(-Habitat, -Salinidad, -Fecha, -IDL)%>%
  recipe(capt ~ .) %>%
  step_corr(all_numeric(), threshold = 0.8) %>%
  step_normalize(all_numeric_predictors()) %>%
  step_impute_knn(all_predictors(),
                  impute_with = imp_vars(all_numeric_predictors(),
                                         -capt)) %>%
  prep() %>%   bake(new_data = NULL)
```

Se realizó este mismo procedimiento para las marismas. De igual forma, para ambos hábitats se aplicó el preprocesado a los datos pero sin realizar la imputación, de modo que pudiera compararse el rendimiento de los modelos creados en base a cada uno de los conjuntos.

La división en entrenamiento y test se llevó a cabo igual que en los casos anteriores, utilizando la división 80/20 también en estos casos. En este caso se trabajó con el paquete **workflows** para generar los modelos de bosques aleatorios y de k-vecinos más cercanos, ambos con su correspondiente ajuste de hiperparámetros.

Se comenzó creando los modelos base, indicando los parámetros que serán ajustados:

```
rf_lag_capt_tune_model <-  rand_forest(mtry = tune(),
              trees = tune(), min_n = tune()) %>%
  set_engine("randomForest") %>%   set_mode("regression")


knn_tune_model_reg <-  nearest_neighbor(neighbors = tune(),
                    weight_func = tune(),
                    dist_power = tune()) %>%
  set_engine("kknn") %>% set_mode("regression")
```

Este modelo se añade a un workflow en blanco, así como la fórmula indicando cuál es la variable de interés. Para el ajuste de hiperparámetros se utilizan valores aleatorios gracias a `grid_random`, y se aplica validación cruzada con `tune_grid` usando los pliegues proporcionados por `vfold_cv`. Por último con `select_best` se indica la medida que se quiere utilizar para decidir cuál es la mejor combinación de parámetros, se aplica `finalize_workflow` y con `last_fit` estaría listo para obtener las métricas y las predicciones.

El código ha sido explicado de forma reducida, pero puede encontrarse por completo el código de creación de todos los modelos en el Apéndice A.

Capítulo 6

Modelos de clasificación

Uno de los principales aspectos de interés a la vista de los datos anteriores sería saber clasificar los cangrejos entre macho y hembra o entre maduro e inmaduro utilizando alguna de las otras variables. En este caso, la información adicional de la que se dispone son las medidas del tamaño de los cangrejos. Para todos ellos se ha tomado al menos el largo del cuerpo, y para algunos se han tomado también las medidas ancho del cefalotórax, largo del cefalotórax y peso.

Visualmente se pudo comprobar en el capítulo anterior que el largo del cuerpo de los cangrejos no parece estar demasiado influenciado por el sexo, mientras que la etapa de madurez sí que tiene un gran impacto. Eso significa que, si se quiere utilizar esta variable para clasificar los cangrejos, se tendrá un resultado considerablemente mejor a la hora de determinar la etapa de madurez que el sexo. Sin embargo, el número de datos perdidos de la variable etapa de madurez en el conjunto disponible es de 11 de los 15482 individuos, mientras que hay 2476 valores perdidos de la variable sexo. Sería de mayor interés poder clasificar esta variable sin cometer un error significativo.

Por tanto, en este capítulo va a analizarse el rendimiento de cada una de las predicciones aplicando tres técnicas distintas: el algoritmo de los k vecinos más cercanos, bosques aleatorios y regresión logística. Para poder verificar la tasa de aciertos se realiza una división del conjunto de los datos que están completos en dos grupos: entrenamiento y test. La proporción en los tamaños de ambos grupos es aproximadamente un 70 % y 30 % del total, respectivamente.

Si se pretende utilizar las cuatro variables que tienen información del tamaño de los cangrejos, el número de casos con toda la información disponible es 989. Si solo se necesitase el largo del cuerpo se dispondría de los 15482 registros, lo cual es una muestra considerablemente más grande, por tanto también es de interés comprobar cómo difieren los resultados obtenidos usando cada uno de estos conjuntos.

6.1. Predicción del Sexo

El primer paso es realizar la división de los datos en dos subconjuntos disjuntos, para lo que se van a utilizar las mismas muestras de entrenamiento y prueba para las tres técnicas, de forma que se pueda comparar su eficacia evitando sesgos. Es importante crear las muestras de forma estratificada, para que ambas mantengan la misma proporción

Tabla 6.1: Matriz de confusión KNN, sexo vs variables de tamaño

	Female	Male
Female	77	50
Male	50	102

Tabla 6.2: Matriz de confusión KNN, sexo vs variables de tamaño. Proporción

	Female	Male
Female	60.63	39.37
Male	32.89	67.11

entre Machos y Hembras que el conjunto original, de modo que los grupos no estén desbalanceados.

En el conjunto original había 15482 individuos, 6503 de cada grupo, y 2476 perdidos, pero una vez se filtran aquellos que tenían algún valor perdido entre sus medidas de tamaño la proporción entre ambas categorías no se mantiene exactamente idéntica: hay 423 Hembras y 505 Machos. Por tanto, la muestra de entrenamiento se compone de 296 Hembras y 354 Machos, y la muestra test de 127 Hembras y 152 Machos. Para los modelos que se realicen solo en base al Largo del cuerpo no hay valores perdidos, por tanto las proporciones se mantienen idénticas entre ambos grupos, habiendo 4552 individuos de cada categoría en la muestra de entrenamiento y 1951 en la muestra test. En cualquier caso, podemos concluir que el desequilibrio entre las clases una vez eliminados los registros con valores perdidos no es tan drástico como para que se deba producir un sesgo excesivo que comprometa los potenciales resultados.

6.1.1. K vecinos más cercanos

Para aplicar esta técnica, lo primero que se debe determinar es cuál es el k óptimo para el conjunto de datos, teniendo en cuenta que, al ser un problema de clasificación binaria, es conveniente que sea un número impar, para así evitar empates.

Se hace la comprobación mediante la librería **caret** para todos los números impares entre 5 y 25, por defecto no considera un número de vecinos menor que 5, podría indicarse manualmente pero se consideró que dado el número de casos disponibles es una restricción adecuada. El resultado óptimo obtenido es:

```
## [1] "El mejor k para el conjunto de entrenamiento es 15"
```

Una vez se ha determinado la k del modelo que se utiliza, lo siguiente es realizar las predicciones sobre el conjunto test. Se calcula la matriz de confusión en términos absolutos (Tabla 6.1) y en proporción de aciertos (Tabla 6.2).

Se pueden comparar estos resultados con los que se obtendrían si solo se utilizara la variable largo cuerpo, pero a cambio se dispusiese de un conjunto mucho mayor de datos (Tablas 6.3 y 6.4)

```
## [1] "El mejor k para el conjunto de entrenamiento es 25"
```

Tabla 6.3: Matriz de confusión KNN, sexo vs largo cuerpo

	Female	Male
Female	938	1013
Male	837	1114

Tabla 6.4: Matriz de confusión KNN, sexo vs largo cuerpo. Proporción

	Female	Male
Female	48.08	51.92
Male	42.90	57.10

Los aciertos en el modelo con todas las variables son de un 58 % en las hembras y un 66 % en los machos. En el segundo modelo se tiene casi un 50 % para hembras y 56 % para los machos. Este segundo modelo es peor que el primero, siendo prácticamente inservible, aunque en este caso ninguno de los dos tiene fiabilidad suficiente como para ser utilizados en estimaciones futuras.

Para la variable sexo, las predicciones obtenidas con el modelo de k-vecinos más cercanos no son válidas. Hay que probar con otros modelos antes de afirmar que no sea posible llevar a cabo esta tarea con los datos disponibles.

6.1.2. Bosques aleatorios y árboles de clasificación

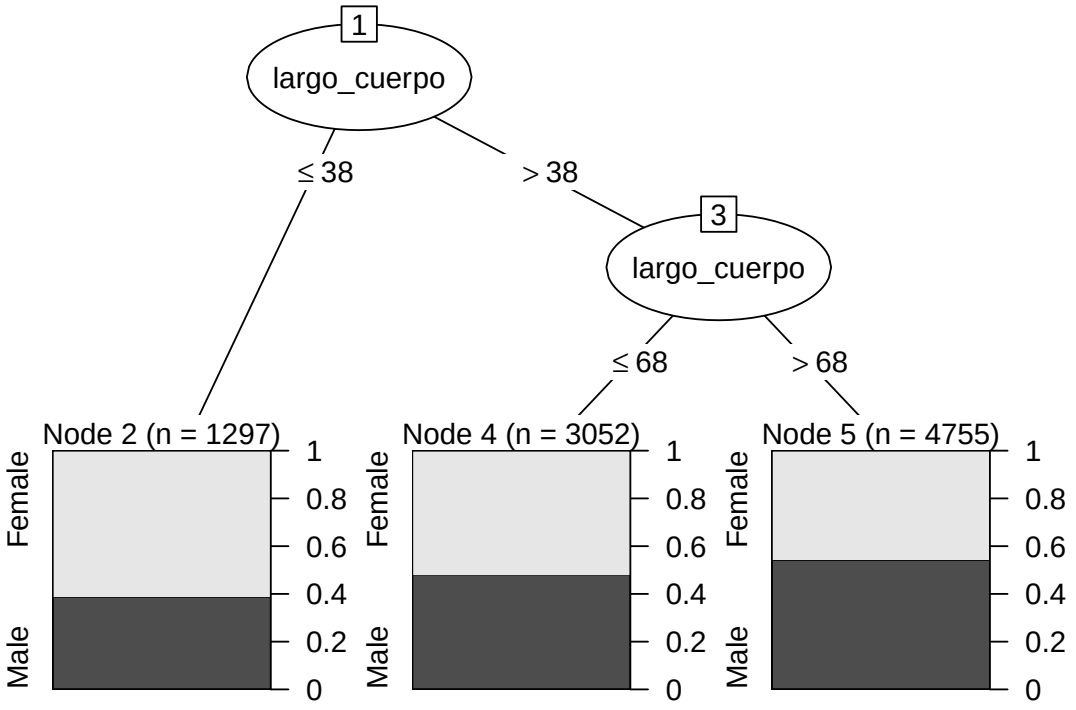
Para el modelo mediante Random Forest no tiene sentido intentar la predicción solo con una variable, pues parte de la aleatoriedad de este método está en elegir solo un número reducido de las variables totales en cada nodo y tomar la mejor decisión gracias a ellas. Para el modelo con una única variable predictora lo que se crea es un árbol de clasificación, concretamente se obtuvo el siguiente árbol:

Tabla 6.5: Matriz de confusión árbol clasificación, sexo vs largo, entrenamiento

	Female	Male
Female	52.20	47.80
Male	43.34	56.66

Tabla 6.6: Matriz de confusión árbol clasificación, sexo vs largo, test

	Female	Male
Female	50.49	49.51
Male	43.16	56.84



El árbol generado es bastante sencillo, teniendo solo dos nodos ya que, con una única variable, si se generasen más subdivisiones se podría sospechar que se está realizando un sobreajuste sobre la muestra de entrenamiento. Cada uno de los tres nodos asigna una clasificación, pero puede verse a simple vista que incluso en la muestra de entrenamiento parecen estar bastante equilibrados. Lo ideal sería que para cada nodo el porcentaje de una de las clases fuera muy superior. Se puede ver la matriz de confusión que se obtiene al aplicar esta clasificación en la muestra test (Tabla 6.5) y en la muestra entrenamiento (Tabla 6.6).

En ambos casos se tiene un nivel de acierto similar, lo cuál indica que para datos futuros este sería el nivel de precisión esperable. Sin embargo, dado que la sensibilidad y especificidad que pueden observarse rondan el 50-55 %, este modelo no podría ser considerado fiable en absoluto.

Para el uso del resto de variables de tamaño sí es conveniente aplicar la técnica de árboles aleatorios, para obtener mayor robustez en el modelo. Se calcula la matriz de

Tabla 6.7: Bosque clasificación, matriz de confusión

	Female	Male
Female	86	41
Male	48	104

Tabla 6.8: Bosque clasificación, matriz de confusión, proporción

	Female	Male
Female	67.72	32.28
Male	31.58	68.42

confusión en valores absolutos y en proporción. Ver Tablas 6.7 y 6.8.

Este modelo tiene un acierto algo superior en la muestra test, alrededor del 60 %. Sigue sin ser un nivel fiable pero, dada la visualización que se realizó en el capítulo anterior, era de esperar que la capacidad de clasificar el sexo según las cuatro variables del tamaño no fuera demasiado precisa.

6.1.3. Regresión lógística

El último modelo que se intentará para esta clasificación es la regresión logística. El objetivo es ver si se puede mejorar la tasa de acierto antes alcanzada, manteniendo las mismas muestras de entrenamiento y test que con los dos métodos anteriores para evitar diferencias debidas a la aleatorización.

Al igual que en el método de los k vecinos más próximos, se estudia la eficacia de un modelo que utilice únicamente la variable Largo del cuerpo y otro que utilice las cuatro variables disponibles.

Por lo que se puede observar, la tasa de aciertos sobre la muestra de entrenamiento del modelo de regresión logística utilizando únicamente la variable Largo del cuerpo es similar a la que se obtuvo con el modelo KNN, entre un 50 y 55 %, no es suficientemente preciso.

El modelo utilizando las cuatro variables de tamaño obtiene los resultados que se muestran en las Tablas 6.11 y 6.12.

Este modelo tiene un sesgo que hace que tienda a clasificar los individuos como Machos, lo cual hace que en la categoria de Machos el acierto sea de un 75 %, pero en cambio para las Hembras es menor al 50 %. Este modelo es erróneo, ya que no sirve de nada saber clasificar correctamente a una categoría a costa de no saber clasificar la otra.

En conclusión, se puede ver que con ninguno de los tres métodos se consigue una clasificación adecuada, esto parece indicar que las diferencias fisiológicas entre los Machos

Tabla 6.9: Matriz de confusión regresión logística, sexo vs largo cuerpo

	Female	Male
Female	994	957
Male	861	1090

Tabla 6.10: Matriz de confusión regresión logística, sexo vs largo cuerpo, proporción

	Female	Male
Female	50.95	49.05
Male	44.13	55.87

Tabla 6.11: Matriz de confusión regresión logística, sexo vs variables tamaño

	Female	Male
Female	64	63
Male	48	104

y Hembras en esta especie no radican en el tamaño de los mismos. En caso de que sea necesario poder clasificar los individuos sin tener que examinar especificamente el sexo de cada uno, este estudio nos indica que debería barajarse qué otras variables podrían servir para este cometido, ya que las disponibles en este conjunto no son suficientes.

Se pueden reflejar las medidas de rendimiento de cada uno de los modelos para compararlos. Concretamente en la Tabla 6.13 se muestra el modelo con todas las variables de tamaño de cada tipo. En las tres medidas el modelos de bosques aleatorios tiene mejores resultados. Aparte, la sensibilidad del modelo de Regresión Logística es muy mala, por lo que podría decirse que este modelo es el peor aunque tenga mejor especificidad que el de K vecinos más cercanos. Ninguno de los tres modelos obtiene medidas como para poder sacar conclusiones fiables de ellos.

6.2. Predicción de la Madurez

El proceso con esta variable es el mismo que se siguió con la variable Sexo, primero se divide de aquellos individuos que se conoce su variable Madurez y que no tenían ningún valor perdido en las variables de tamaño en dos muestras: entrenamiento y test. Lo mismo se realiza sobre el conjunto de aquellos para los que se conoce la variable Madurez, pues de todos ellos se dispone de la variable Largo del cuerpo, y por tanto se puede intentar realizar la predicción solo con esa variable y teniendo un tamaño muestral mucho mayor.

En este caso se dispone de información sobre 8968 individuos inmaduros, 6503 maduros y 11 sin determinar. Por tanto, al restringir aquellos de los que sí se dispone información y dividir en dos muestras se obtendrían: 6277 individuos inmaduros y 4552 individuos maduros en la muestra de entrenamiento, 2691 individuos inmaduros y 1951 maduros en la muestra test.

Por otro lado, una vez se obtienen solo los que tienen registradas sus cuatro variables de tamaño, se reduce a: 447 individuos inmaduros y 542 maduros. Se han invertido los tamaños entre ambas categorías. Por tanto, para la muestra de entrenamiento la estructura

Tabla 6.12: Matriz de confusión regresión logística, sexo vs variables tamaño, proporción

	Female	Male
Female	50.39	49.61
Male	31.58	68.42

Tabla 6.13: Comparativa modelos determinación Sexo

Modelo	Exactitud	Especificidad	Sensibilidad
K vecinos más cercanos	0.645	0.678	0.614
Bosques aleatorios	0.685	0.704	0.677
Regresión logística	0.602	0.684	0.504

Tabla 6.14: Matriz de confusión KNN, Madurez vs variables de tamaño

	Immature	Mature
Immature	120	15
Mature	24	139

es 312 inmaduros y 379 maduros, y para la muestra de test es de 135 inmaduros y 163 maduros.

6.2.1. K vecinos más cercanos

Se debe determinar el k óptimo para cada uno de los dos modelos, para ello se utiliza la muestra de entrenamiento para comprobar los valores impares que van de 5 a 25 ambos inclusive. Se obtiene el k óptimo para el modelo con las cuatro variables.

```
## [1] "El mejor k para el conjunto de entrenamiento es 17"
```

Una vez se tiene ese parámetro se pueden realizar las predicciones para la muestra test y comparar con los valores reales para ver su precisión.

La precisión de este modelo está alrededor del 90 % en la muestra test, así que cabe esperar resultados similares si se utilizase para clasificar nuevos individuos en el futuro. Es un modelo mucho más fiable que el que se creó para clasificar Machos y Hembras, pero antes de sacar conclusiones se comprobarán también los demás métodos antes utilizados.

Para el modelo utilizando solo la variable Largo del cuerpo también se obtiene el número óptimo de vecinos a utilizar.

```
## [1] "El mejor k para el conjunto de entrenamiento es 21"
```

Utilizando este parámetro se predice la muestra test, y los niveles de acierto resultantes son los que se indican en las Tablas 6.16 y 6.17.

Incluso usando solo una variable para predecir el estado de Madurez de los cangrejos, los resultados tienen una fiabilidad del 90 % aproximadamente. Este modelo y el anterior pueden ser considerados fiables y podrían utilizarse en un futuro en caso de tener nuevos datos que se quieran registrar, o para clasificar aquellos valores perdidos que había en el conjunto disponible.

Tabla 6.15: Matriz de confusión KNN, Madurez vs variables de tamaño. Proporción

	Immature	Mature
Immature	88.89	11.11
Mature	14.72	85.28

Tabla 6.16: Matriz de confusión KNN, Madurez vs largo cuerpo

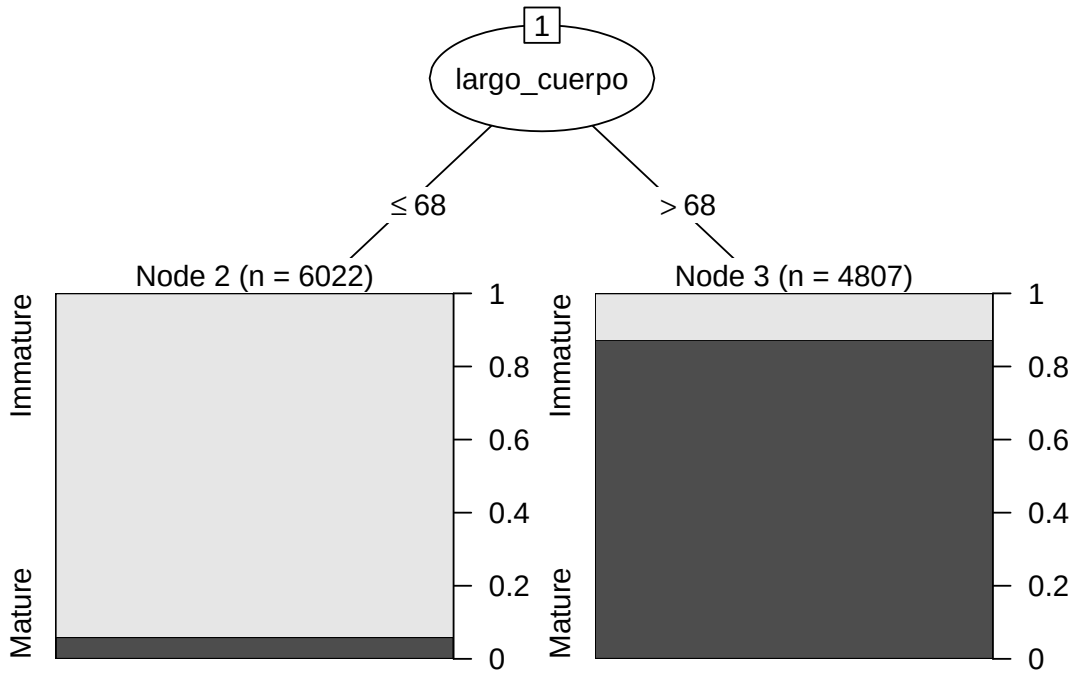
	Immature	Mature
Immature	2467	224
Mature	192	1759

Tabla 6.17: Matriz de confusión KNN, Madurez vs largo cuerpo. Proporción

	Immature	Mature
Immature	91.68	8.32
Mature	9.84	90.16

6.2.2. Bosques aleatorios y árboles de clasificación

En este caso, al igual que para clasificar Hembras y Machos, se presenta un ejemplo de un árbol de clasificación utilizando únicamente la variable Largo del cuerpo, y además se crea un modelo mediante bosques aleatorios utilizando las cuatro variables.



Hay un único nodo en este caso, esto se debe a que la distribución era mucho más clara para esta variable: los individuos inmaduros presentan generalmente un menor tamaño y los maduros un tamaño mayor. Los nodos finales presentan una distribución mucho menos balanceada que en el caso de la variable sexo, por lo que al menos para la muestra de entrenamiento se obtiene una clasificación aparentemente buena. El punto clave estaba en fijar un punto crítico que permita hacer una buena división de los grupos. Al aplicar esto al conjunto entrenamiento y al conjunto test se obtienen los resultados mostrados en las Tablas 6.18 y 6.19.

Tabla 6.18: Matriz de confusión árbol clasificación, Madurez vs largo, entrenamiento

	Immature	Mature
Immature	90.25	9.75
Mature	7.84	92.16

Tabla 6.19: Matriz de confusión árbol clasificación, Madurez vs largo, test

	Immature	Mature
Immature	90.52	9.48
Mature	8.20	91.80

En ambos casos es muy similar la tasa de aciertos, lo cuál confirma que no se trata de un modelo sobreajustado, sino que su fiabilidad es verídica.

Tras haber visto esto, para el conjunto de todas las variables de tamaño se utiliza el método de bosques aleatorios. Los resultados obtenidos son los de las Tablas 6.20 y 6.21.

El nivel de fiabilidad que tiene este modelo es similar al del árbol antes creado, ambos están en torno al 85-90 % de acierto. En caso de tener todas las variables es conveniente utilizar este modelo ya que tiene mayor información, pero si solo se dispone del largo del cuerpo podría utilizarse como norma de clasificación la establecida en el árbol.

6.2.3. Regresión lógística

Finalmente, se plantea diseñar un modelo de regresión logística, pudiendo comparar la eficacia que tenga el modelo que utilice solo la variable Largo del cuerpo respecto al que las usa todas, así como comparar la eficacia de estos con sus equivalentes en las técnicas anteriores.

La predicción utilizando solo la variable Largo del cuerpo da mejores resultados para los individuos inmaduros, aunque en ambas categorías está alrededor del 90 %, como era de esperar viendo los modelos anteriores.

Para el modelo de regresión logística con todas las variables tamaño como predictoras, los resultados obtenidos se muestran en las Tablas 6.24 y 6.25.

Este modelo tiene un rendimiento ligeramente peor que el de una sola variable, pero puede deberse a la selección de la muestra. Lo que no es coincidencia es que los modelos que predicen la etapa de Madurez de los cangrejos sean mucho más fiables que los que predicen el Sexo. Al menos los basados en las variables de tamaño, que son los que se han podido estudiar con la información proporcionada.

Por tanto, se concluye que en caso de quererse predecir en un futuro la etapa de Madurez de los cangrejos, bastaría con medir las variables de tamaño, o incluso solo el Largo

Tabla 6.20: Bosque clasificación, matriz de confusión

	Immature	Mature
Immature	119	16
Mature	24	139

Tabla 6.21: Bosque clasificación, matriz de confusión proporción

	Immature	Mature
Immature	88.15	11.85
Mature	14.72	85.28

Tabla 6.22: Matriz de confusión regresión logística, Madurez vs largo cuerpo

	Immature	Mature
Immature	2492	199
Mature	223	1728

Tabla 6.23: Matriz de confusión regresión logística, Madurez vs largo cuerpo, proporción

	Immature	Mature
Immature	92.60	7.40
Mature	11.43	88.57

Tabla 6.24: Matriz de confusión regresión logística, Madurez vs variables tamaño

	Immature	Mature
Immature	122	13
Mature	24	139

Tabla 6.25: Matriz de confusión regresión logística, Madurez vs variables tamaño, proporción

	Immature	Mature
Immature	90.37	9.63
Mature	14.72	85.28

Tabla 6.26: Comparativa modelos determinación Madurez

Modelo	Exactitud	Especificidad	Sensibilidad
K vecinos más cercanos	0.909	0.9	0.916
Árbol de clasificación	0.911	0.918	0.905
Regresión logística	0.909	0.886	0.926

del cuerpo, para poder conseguir una estimación fiable. En cambio, si el objetivo del estudio fuera poder predecir si el cangrejo en cuestión es Macho o Hembra, se necesitaría información adicional que estuviera relacionada con otras características de la especie.

Esta cuestión podría trasladarse a los ecólogos en caso de que se diera otra colaboración en el futuro, y tal vez replantear las mediciones tomadas o añadir alguna más para facilitar la predicción de la variable Sexo. También sería interesante saber si el hecho de que la mayoría de valores perdidos para esta variable se encuentren en individuos Inmaduros es por la dificultad de identificar a estos cuando aún son demasiado pequeños. Aunque el modelo matemático reduzca estas cuestiones a números y porcentajes, no hay que perder de vista que se trata de una situación real y que involucra distintas disciplinas, de modo que intentar obtener resultados desde un punto de vista puramente matemático puede provocar que el enfoque no sea el más eficaz para la situación.

Por último, para resumir esta sección se refleja el mejor modelo de cada tipo en una tabla comparativa (Tabla 6.26). En este caso, el mejor ajuste se da utilizando únicamente la variable “Largo cuerpo”. El árbol de clasificación tiene la mejor Exactitud y Especificidad, pero la peor Sensibilidad. En conjunto parece que los tres modelos tienen resultados similares, por lo que cualquiera de ellos sería válido.

Capítulo 7

Modelos de regresión

Tratar con las variables numéricas supone una gran diferencia con respecto a las variables categóricas. Los métodos antes utilizados podrían servir, pudiendo cambiar la regresión logística por regresión lineal. Sin embargo, la principal diferencia se encuentra en la información disponible y la forma de tratarla y combinar los diferentes conjuntos. En el caso anterior se intentaba predecir información sobre un cangrejo concreto dados unos valores de ese mismo cangrejo (longitud, peso, ancho). El primer apartado de este capítulo trata de eso mismo: intentar predecir una de las variables de tamaño de un cangrejo en función del resto. Pero no siempre ocurre esto, especialmente cuando se trabaja con el conjunto de datos ambientales. Es importante recordar que este conjunto de datos no ha sido publicado al mismo tiempo que los otros tres que se manejan en este estudio, y por tanto es un fichero con una estructura diferente y un trabajo de pre-procesado bastante menor por parte de las personas que recopilaron los datos. Gran parte del esfuerzo dedicado a este capítulo radicó en intentar pulir el conjunto de datos y aprovechar al máximo la información disponible, aun cuando se encontraron las dificultades que a continuación se explican.

En primer lugar, los datos publicados trataban con fechas en días concretos, ya fuera el día de la captura de un cangrejo concreto o el día en que se revisaron las trampas; todos estos valores estaban asignados a una fecha. En cambio, en el fichero con los datos ambientales los datos se registran por meses: se indica una localización, un año y un mes, y se registran los valores de las variables, pero no queda registrado en qué día en concreto se realizó la medición. La forma de solucionar esto fue agrupar las otras variables por meses también, de modo que ambos conjuntos pudieran combinarse en los casos que coincida la fecha y la localización.

En segundo lugar, aunque en cada mes y localización se tomase un único dato de cada variable (en caso de haberse tomado alguno), examinando el fichero puede verse que hay un gran número de registros que contienen la misma información. La única diferencia entre estos registros, tal como se explicó en el apartado de lectura de datos, era el código de nasa que se asignaba, pero eso no influye en las variables ambientales y por tanto es una información que fue descartada. Había también casos en los que, para un mismo mes y localización, se registraban datos prácticamente iguales, pero que diferían en una única variable. Estos datos fueron considerados erróneos y eliminados por completo, ya que daban lugar a incongruencias, pudiendo llegar a tomar valores simplemente imposibles a nivel científico como era el caso del pH. En general, esta limpieza convirtió el conjunto original de 6710 registros en uno de 784, pero ya sin repeticiones ni ruido.

En tercer lugar, de estos registros solo 197 eran completos. Todos los demás tenían algún valor perdido, en la mayoría de casos una cantidad considerable de ellos. En el caso de la variable Nitrato, si se eliminase se conseguiría llegar a 221 registros completos, pero eso solo ocurre ya que es la que mayor porcentaje de valores perdidos tenía, y no funcionaría intentar seguir aplicando una estrategia similar. El problema que esto supone es que a la hora de hacer modelos se deben separar los datos de cada tipo de hábitat, ya que tienen comportamientos muy distintos tanto en valores de estas variables como en las capturas en trampas. Haciendo esto los conjuntos que quedan se componen de 111 registros completos para las Lagunas y 86 para las Marismas. Si a eso se le añade que debe hacerse una división en conjunto de entrenamiento y conjunto de prueba (o test) para poder evaluar el rendimiento de los modelos, esto hace que la cantidad de datos con los que se podía trabajar fuera insuficiente. Una forma de solucionarlo podría ser aplicar validación cruzada a los modelos, pero los valores perdidos siguen suponiendo un enorme desperdicio de información ya que en modelos como árboles aleatorios aquellas observaciones que no están completas son descartadas directamente.

Este último problema no tiene una solución como tal, pero la forma de intentar remediarlo que se adoptó en este estudio fue imputar los datos perdidos mediante el método de los k-vecinos más cercanos como se explicó en el capítulo introductorio de este bloque. Gracias al paquete **recipes** del ecosistema tidymodels de R, esta tarea se llevó a cabo de forma automatizada, y los conjuntos finales tienen 143 registros en el caso de las Lagunas y 140 registros en el caso de las Marismas, lo cual sigue siendo bastante menos de lo que sería ideal, pero al menos es un tamaño considerable.

A pesar de todo, a la hora de establecer una relación entre las variables ambientales y las capturas realizadas no está claro que vaya a darse una buena modelización limitándose a relacionar aquellos datos coetáneos. Tal vez el número de cangrejos se vea más influenciado por un acumulado de estas variables en los meses (o semanas) anteriores, y con los datos disponibles no hay forma de llevar a cabo ese estudio. En la mayoría de localizaciones los datos no se corresponden con meses consecutivos, sino que se trata de registros dispersos a lo largo del tiempo en que se llevó a cabo la recogida de datos, habiendo a lo sumo dos meses consecutivos en aquellas localizaciones en las que se registraron mayor número de registros. Por tanto, para poder cuadrar los datos de la forma que se consideró más adecuada, se combinaron los datos que coincidían en localización y fecha, y se intentó obtener modelos apropiados en base a esto.

7.1. Tamaño de los cangrejos

De las cuatro variables de tamaño sobre los cangrejos, una de ellas (largo del cuerpo) se midió para todos los individuos; en cambio, las otras tres apenas se midieron en el 10 %. Concretamente, el largo del cefalotórax se midió en el 12.5 % de los cangrejos que se estudiaron, y el ancho de cefalotórax y el peso en el 7 %. Sería de interés por tanto ver si es posible estimar alguna de estas variables respecto al largo del cuerpo.

Para cada una de las tres variables se trabaja sobre el subconjunto de cangrejos en los que esta variable fue medida, aplicando un modelo de regresión lineal solo con una sola variable predictora: el largo del cuerpo. El error cometido se calcula mediante Validación Cruzada (fijando una semilla) de modo que se mida sobre la muestra de entrenamiento, y la muestra test se utilice para visualizar las predicciones y compararlas con los valores

reales. Ya que cada una de las tres variables tiene rangos distintos, comparar el error en términos absolutos podría ser una forma engañosa de comparar los modelos. El método elegido para evaluar el error cometido por los modelos es calcular el RMSE por Validación Cruzada, y calcular el porcentaje que este error supone con respecto al valor medio de la variable en el conjunto de entrenamiento.

Antes de empezar a hacer ningún tipo de modelización, es conveniente analizar las correlaciones que tienen las variables. Utilizando para cada par de variables todos los registros completos de ambas, la matriz de correlaciones obtenida se ve en la Figura 7.1.

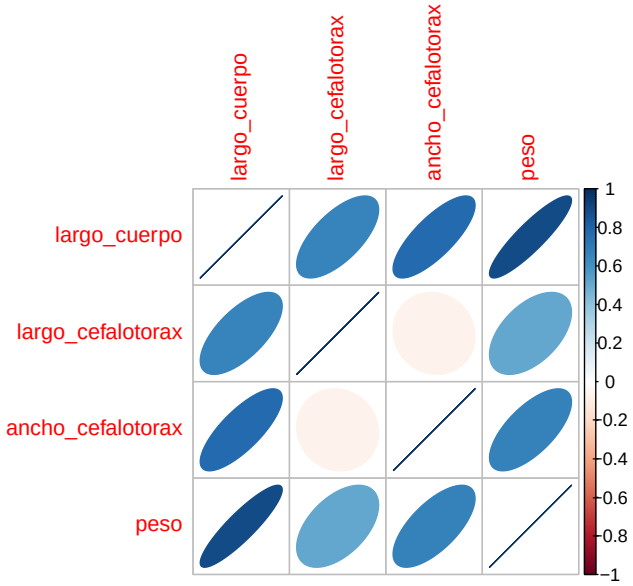


Figura 7.1: Correlaciones variables tamaño

A primera vista llama la atención que el largo del cefalotórax y el ancho del cefalotórax sean el par que menos correlación tienen entre sí. Concretamente, su correlación es -0.0698 mientras que la siguiente menor (en valor absoluto) es 0.515 entre largo del cefalotórax y peso. El motivo de que esto ocurra no tiene explicación biológica, pero mirando a fondo los datos puede observarse un fenómeno que se refleja en la Figura 7.2.

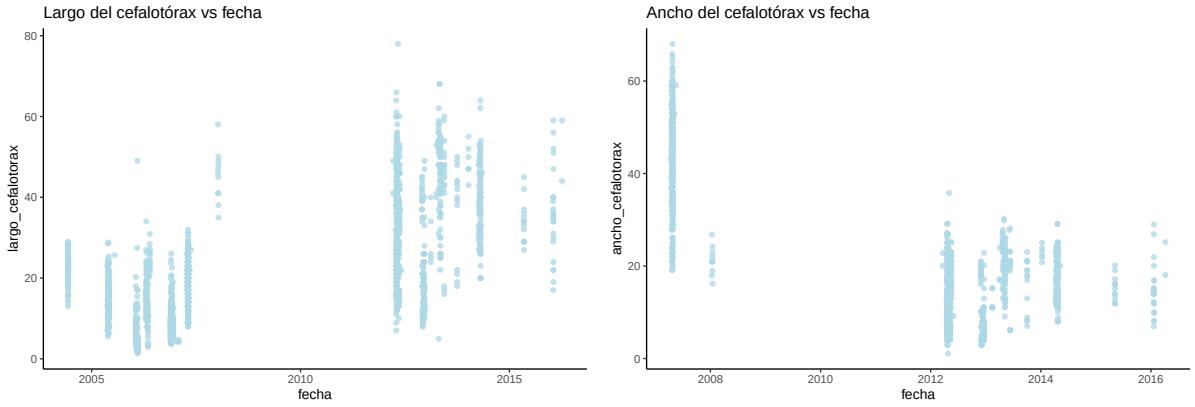


Figura 7.2: Tamaños cefalotórax según fecha

Los datos parecen dividirse en dos periodos. Por poner un punto de separación, podría distinguirse el grupo de cangrejos cuyas medidas se tomaron antes de 2010 y el grupo de

aquellos capturados después de 2010. Ambos grupos están separados por un periodo de tiempo considerable en el que no se tomaron medidas, y los valores que alcanzaron estas variables en cada uno de los dos tramos son significativamente distintos. En las variables peso y largo del cuerpo esta división no es tan notable, como puede apreciarse en la Figura 7.3; en la primera, el peso, sí que se produce el mismo parón en los registros, pero los valores antes y después de esto no parecen diferir demasiado. En cambio, para el largo del cuerpo durante ese periodo se registraron valores como en cualquier otro, tal vez incluso más. Esto indica que los parones observados en las otras tres variables no se debe a que dejasen de capturarse cangrejos para observar, sino que debe ser por una causa diferente que es desconocida.

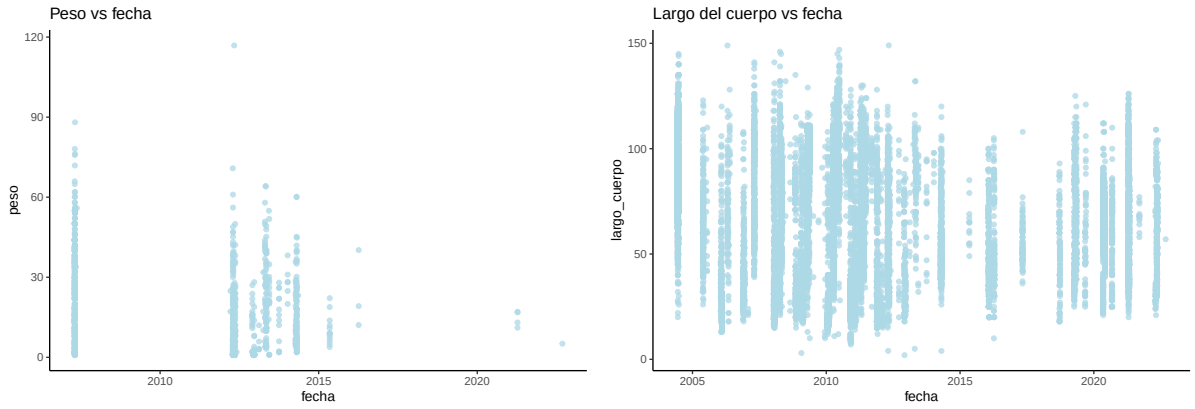


Figura 7.3: Largo cuerpo y peso según fecha

Aunque no se conozca el motivo de que los cangrejos antes de 2010 pareciersen tener un largo del cefalotórax de menor tamaño que aquellos capturados después de 2010 y al mismo tiempo que presentasen un ancho de cefalotórax mayor, lo que sí se ve es que esto podría explicar la mala correlación de las dos variables en el conjunto total. Para confirmar esto, se divide el conjunto "mof", descrito en la sección 3.2, en dos (tomando el 1 de enero de 2010 como punto de corte) y se vuelven a calcular las correlaciones entre las variables de tamaño.

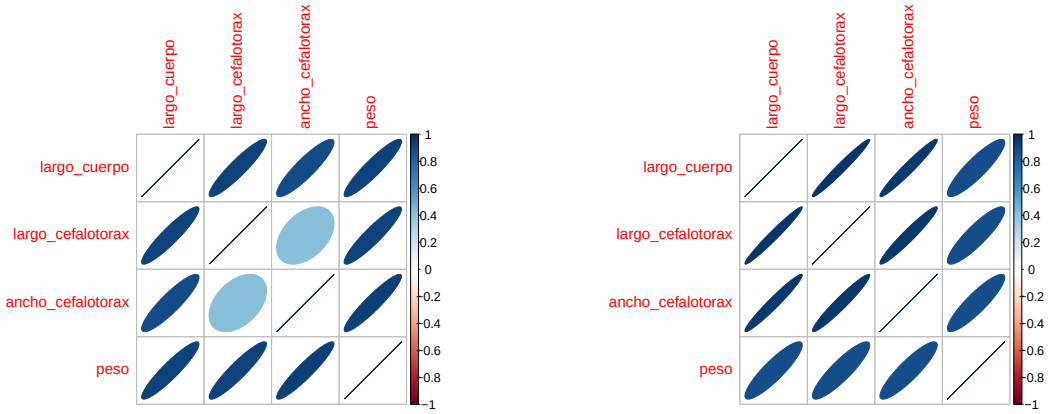


Figura 7.4: Correlaciones variables tamaño según periodos temporales

Las correlaciones en estos dos subconjuntos son considerablemente mayores como puede verse en la Figura 7.4, incluso en las variables que no parecen verse afectadas por la división

Tabla 7.1: Regresión: largo cefalotórax según largo cuerpo

R cuadrado	R cuadrado ajustado
0.44442	0.44406

en dos periodos. Esto parece indicar que la modelización es mejor si se aplica por separado a cada uno de los grupos, en vez de intentarlo en el conjunto de datos total.

7.1.1. Regresión lineal

Antes de comentar los resultados de realizar estos modelos por separado, es interesante plantearse qué hubiera ocurrido si no se hubiera realizado la división. Para tener referencia, se realiza el modelo de regresión lineal con variable predictora largo del cuerpo y variable objetivo largo del cefalotórax, tratando con el conjunto sin realizar división.

El ajuste que tiene el modelo de regresión no es demasiado bueno (ver Tabla 7.1), teniendo un R^2 y un R^2 ajustado con valores bastante más bajos de lo que se esperaría de un buen modelo. Es comprensible por tanto que el error cometido también sea considerablemente grande en comparación con el valor medio de la variable.

```
## [1] "El error cometido es del 47.62 %"
```

Por último, si se dibujan las predicciones frente a los valores reales, puede verse que hay dos grupos diferenciados (ver Figura 7.5). En el grupo de cangrejos capturados antes de 2010 las predicciones son menores que el valor real, en cambio en aquellos de después de 2010 ocurre lo contrario (quitando algunos casos sueltos en los que se da lo contrario). Antes de dar con esta explicación para el comportamiento extraño del modelo de predicción, algunas de las otras explicaciones que se intentaron buscar fueron: dividir entre individuos maduros e inmaduros, dividir machos y hembras, dividir individuos de las lagunas y de las marismas. Sin embargo, para todo estos casos las dos categorías se encontraban mezcladas en el grupo que se estimaba a la alza y el que se estimaba a la baja. Tras examinar con detenimiento los dos subgrupos se determinó que el factor decisivo era la fecha de captura de los cangrejos.

Tabla 7.2: Regresión largo cefalotórax según largo cuerpo según periodo temporal

Periodo	R cuadrado	R cuadrado ajustado
Antes de 2010	0.8653	0.86514
Después de 2010	0.9483	0.94821

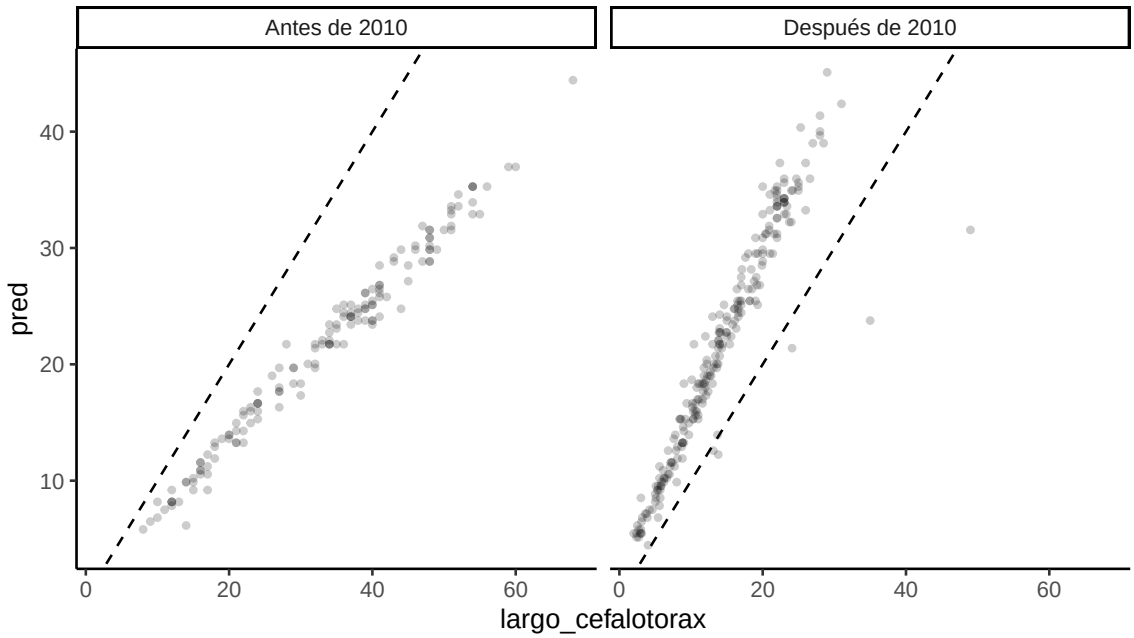


Figura 7.5: Predicciones según periodo temporal regresión: largo cefalotórax según largo cuerpo

Una vez se ha visto el modelo con el conjunto total, es aún más perceptible la mejora en la calidad del modelo que se obtiene cuando se trabaja con ambos conjuntos por separado (ver Tabla 7.2).

```
## [1] "El error cometido antes de 2010 es del 19.22 %"
```

```
## [1] "El error cometido después de 2010 es del 8.97 %"
```

Tabla 7.3: Regresión ancho cefalotórax según largo cuerpo según periodo temporal

Periodo	R cuadrado	R cuadrado ajustado
Antes de 2010	0.785	0.78419
Después de 2010	0.937	0.93693

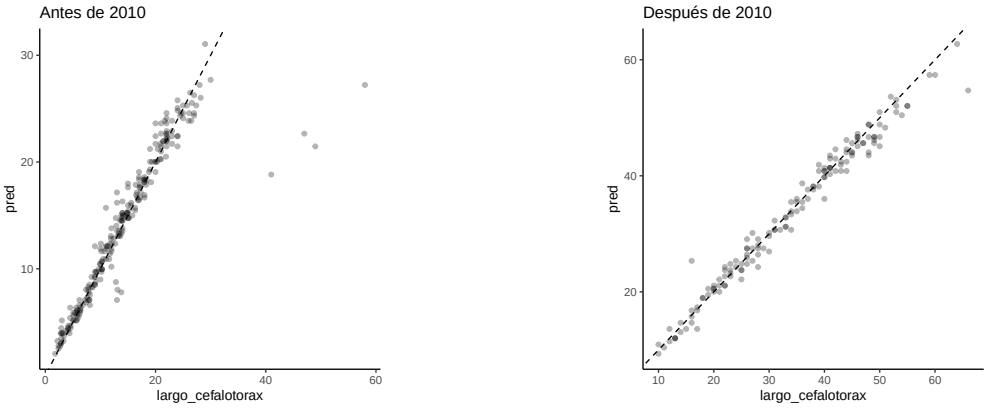


Figura 7.6: Predicciones regresión según periodo temporal: largo cefalotórax según largo cuerpo

Los resultados obtenidos con modelos separados son mucho mejores. El ajuste es adecuado y las predicciones se agrupan alrededor de la línea diagonal que indica que coinciden con los datos reales (ver Figura 7.6). Hay algunos datos en el conjunto anterior a 2010 que hacen que el ajuste sea peor. Además, como el valor medio en este subconjunto es menor, eso hace que el porcentaje de error sea mayor, pero incluso así el modelo sigue siendo mucho mejor una vez se hizo la separación.

Este mismo proceso puede repetirse sobre la variable ancho cefalotórax, siendo también necesaria la división en dos subconjuntos. En este caso se realizarán directamente los modelos correctos.

```
## [1] "El error cometido antes de 2010 es del 12.31 %"
```

```
## [1] "El error cometido después de 2010 es del 10.58 %"
```

En este caso el ajuste antes de 2010 tiene coeficientes de determinación ligeramente peores (ver Tabla 7.3), pero en cambio el porcentaje de error es considerablemente menor que en el caso del largo del cefalotórax. Esto se debe a que en el caso de esta variable, era antes de 2010 cuando presentaba una media mayor, por lo que aunque el RMSE sea mayor, en porcentaje disminuye el error cometido.

Tabla 7.4: Regresión: peso según largo cuerpo

R cuadrado	R cuadrado ajustado
0.79612	0.7959

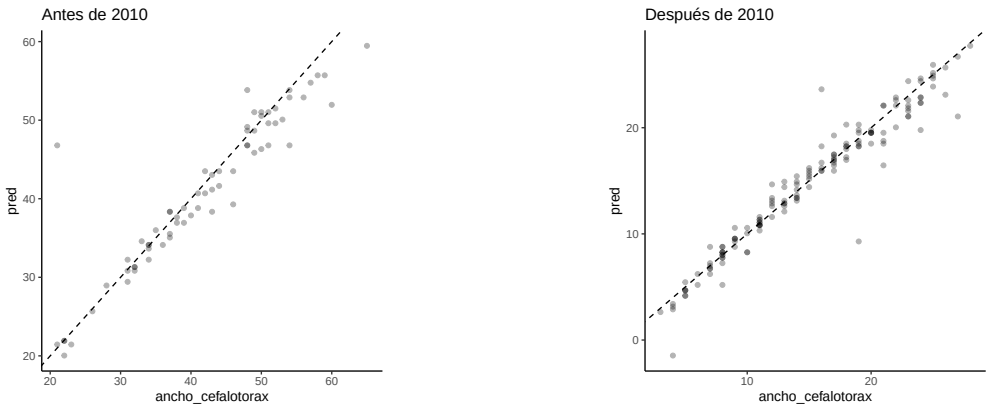


Figura 7.7: Predicciones regresión según periodo temporal: ancho cefalotórax según largo cuerpo

Finalmente, se realiza este mismo estudio para la variable peso. En la gráfica inicial (ver 7.3) no se apreciaba la división que tenían las otras dos variables, pero es de interés comparar tanto el modelo sin realizar la división como los dos modelos por separado.

```
## [1] "El error cometido es del 42.42 %"
```

El error cometido es considerable. El coeficiente de determinación (ver Tabla 7.4) tiene un valor muy superior al que se dio en el primer modelo realizado, aunque es menor que el de los modelos que se crearon con los conjuntos divididos. En la gráfica de las predicciones (ver Figura 7.8) se pueden visualizar ambos grupos por separado para ver si la división es tan clara como en el caso del largo del cefalotórax.

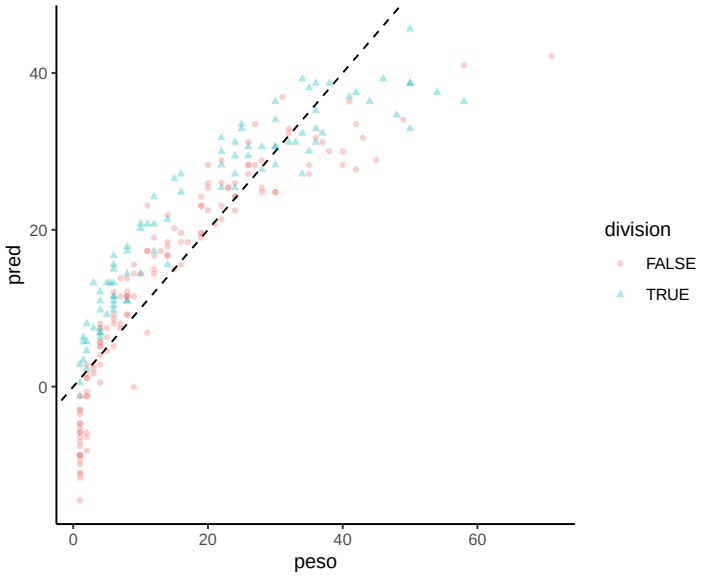


Figura 7.8: Predicciones regresión: peso según largo cuerpo

Tabla 7.5: Regresión peso según largo cuerpo según periodo temporal

Periodo	R cuadrado	R cuadrado ajustado
Antes de 2010	0.8469	0.84652
Después de 2010	0.7808	0.78044

La división no es tan clara como lo era en las medidas del cefalotórax. Además, se producen predicciones negativas que no tienen sentido teniendo en cuenta que se trabaja con la variable peso. Es posible que aplicar la división en dos conjuntos permita realizar un ajuste mejor, por lo que se comparan estos resultados con de los modelos haciendo la división de los datos en función de la fecha.

```
## [1] "El error cometido antes de 2010 es del 31.69 %"
```

El error cometido ha disminuído, pero no de forma tan drástica como en los dos casos anteriores. Esto puede deberse a que la división entre ambos periodos temporales no era significativa para la variable peso.

```
## [1] "El error cometido después de 2010 es del 47.34 %"
```

En este caso se mantiene el error cometido. Esto parece indicar que la variable largo del cuerpo no es un buen predictor para el peso (al menos no a solas), y que la división realizada en este caso no permite mejorar el ajuste como para que sea conveniente llevarla a cabo

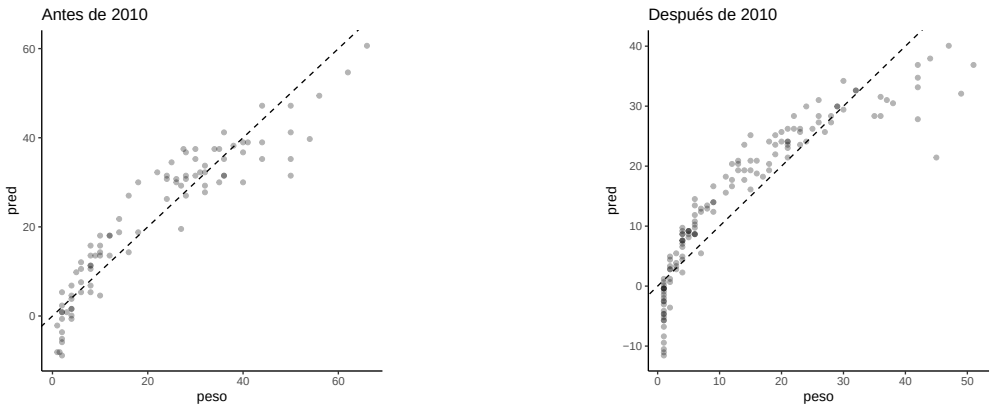


Figura 7.9: Predicciones regresión según periodo temporal: peso según largo cuerpo

Puede apreciarse en las gráficas que la estimación en este caso es peor que con las otras variables. Para intentar mejorar esta predicción podrían añadirse las otras dos variables de tamaño para ver si de esta forma el peso queda mejor explicado, pero dada la alta correlación en este conjunto de variables no sería conveniente crear un modelo de regresión lineal de esa forma.

Por tanto, la regresión lineal solo con la variable largo del cuerpo no da un buen ajuste, pero llama la atención que el aspecto de la Figura 7.8 es similar al de una curva logaritmica. Debe tenerse en cuenta que lo que se ha representado es en el eje x la variable real y en

el eje y la predicción; eso no significa que la variable objetivo tenga esa distribución. Aun así, es razonable plantearse si al hacer alguna transformación logarítimica en el modelo anterior los resultados obtenidos serían mejores; por tanto, se procede a estudiar dicho modelo.

```
## [1] "El error cometido con la transformación logarítmica es del 16.19 %"
```

El error cometido en la predicción del logaritmo del peso (respecto a la media del logaritmo del peso en la muestra test) es mucho menor que el que se cometía en la predicción del peso normal. Eso es buena señal, ya que los resultados concuerdan con las gráficas.

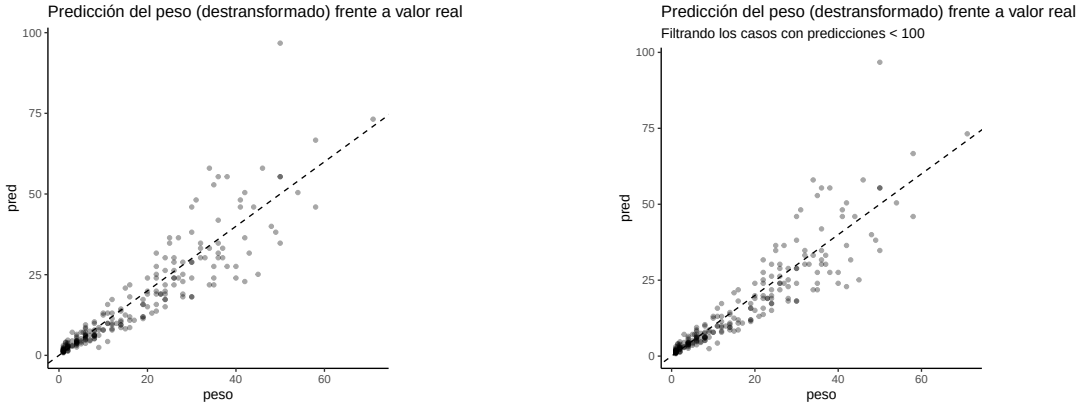


Figura 7.10: Predicciones regresión con transformación logarítmica peso

Para la visualización se enfrenta la predicción del peso (la exponencial de la estimación del logaritmo que se obtuvo mediante la regresión lineal) respecto al valor real del peso. Al haber tenido que aplicar la exponencial, hay algunos casos cuya predicción se dispara, y hacen que la gráfica quede distorsionada. Para poder visualizar mejor la parte de la gráfica que concentra la mayor parte de los casos, en la segunda gráfica se observan solo aquellos que presentaron una predicción inferior a 100 gr. En ambas puede verse que, aunque algunos casos se alejen bastante del valor real, en la mayoría de los individuos la predicción mejora de forma significativa.

7.1.2. Bosques aleatorios

Otra alternativa al modelo inicial es intentar utilizar un modelo de regresión distinto, como el método de bosques aleatorios, para poder así utilizar las tres variables predictoras en vez de solo una. Para este modelo se realiza el ajuste sobre el parámetro "mtry", que indica cuántas variables deben estudiarse en cada nodo para decidir la mejor partición, obteniendo en este caso como resultado que el valor óptimo es 1. Por tanto, en cada nodo se elige aleatoriamente una de las tres variables predictoras, y se divide de la mejor forma posible usando esta variable. Para el número de árboles generados en el bosque se deja el valor por defecto, ya que con el tamaño del conjunto de datos disponible tampoco es necesario generar un enorme número de árboles, y la mejora que podría conseguirse sería a conllevaría un aumento del coste computacional excesivo.

Con estos parámetros, se calcula el porcentaje de error cometido sobre la muestra entrenamiento, al igual que se hizo con los modelos anteriores, y se muestra la gráfica de las predicciones de la muestra test frente a los valores reales.

```
## [1] "El error cometido es del 14.34 %"
```

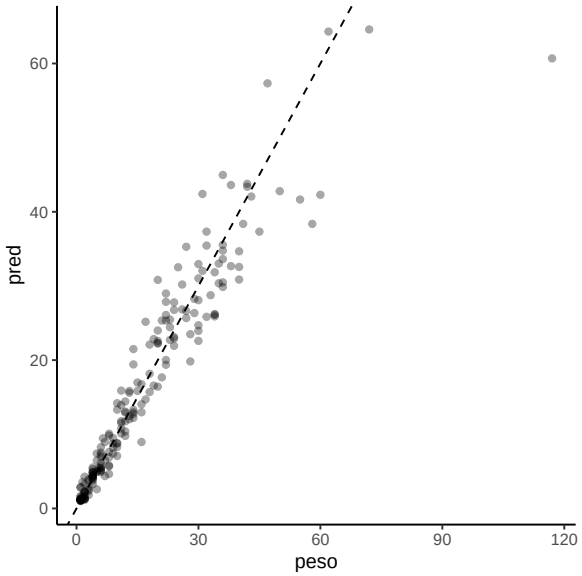


Figura 7.11: Predicciones bosque aleatorio: peso según variables tamaño

Puede verse que el error cometido con el modelo de bosques aleatorios es mucho menor que con el modelo de regresión, en parte porque han podido aprovecharse las tres variables disponibles, y también porque este método tiene un proceso más complejo de entrenamiento que permite hacer un ajuste más preciso. También se puede ver gráficamente cómo las predicciones se acercan más a la línea diagonal que representa el ajuste perfecto.

De este apartado pueden sacarse varias conclusiones:

- Alrededor de la pausa que hubo en las mediciones de largo del cefalotórax y en ancho del cefalotórax de los cangrejos en 2010 los comportamientos de estas variables se vieron modificados de manera considerable. En cambio esto no ocurrió para la variable peso, que también tuvo un parón en las mediciones tomadas, ni para la variable largo del cuerpo, que fue tomada de forma continuada durante los años del estudio. El motivo de este suceso es desconocido.

- Es conveniente, para realizar estimaciones de las variables relacionadas con el cefalotórax, tener en cuenta en cuál de los dos periodos se encuentra el individuo de interés. Para predicciones futuras deberían utilizarse únicamente los valores posteriores a 2010 como conjunto de entrenamiento, pues los datos anteriores producen una perturbación que empeora la calidad de las predicciones.

- Al estimar la variable peso mediante un modelo de regresión con la variable largo del cuerpo se comete un error muy superior a lo que sería adecuado. En caso de

Tabla 7.6: Comparativa modelos regresión lineal según periodo

Modelo	R^2 ajustado	% error	RMSE
Largo Cefalotórax, antes de 2010	0.86514	19.22	2.768
Largo cefalotórax, después de 2010	0.94821	8.97	3.051
Ancho Cefalotórax, antes de 2010	0.78419	12.31	5.11
Ancho cefalotórax, después de 2010	0.93693	10.58	1.586

Tabla 7.7: Comparativa modelos predicción peso

Modelo	% error	RMSE
Regresión logarítmica, Peso	16.19	1.444
Bosque aleatorio, Peso	14.34	2.403

querer obtener predicciones de esta variable, es mejor aplicar una transformación logarítmica antes de aplicar el modelo de regresión lineal. Por otro lado, también se obtienen buenas predicciones utilizando el método de bosques aleatorios, que genera un nivel de error similar. En el caso de la regresión lineal, pueden darse valores puntuales más alejados de lo deseado, pero solo requiere de la variable largo del cuerpo como predictora. En cambio, los bosques aleatorios proporcionan una estimación menos propensa a generar predicciones alejadas en extremo, pero requiere de las tres variables de tamaño para obtener la predicción.

A modo de resumen se muestran los resultados de los mejores modelos para las variables Largo de Cefalotórax y Ancho del Cefalotorax (ver Tabla 7.6). Y lo mismo para los modelos del peso en Tabla 7.7. Ya que el porcentaje de error depende de la media de la variable para el conjunto en que fue estudiada (entrenamiento), es una medición del error disinta al RMSE, por tanto se muestran ambos para cada modelo realizado.

7.2. Número de capturas en trampas

Para poder utilizar las variables ambientales, como se explicaba al inicio del capítulo, lo que debe hacerse es agrupar los registros diarios en registros mensuales, de forma que puedan coincidir con los datos de las variables ambientales que fueron tomados. También se filtraron aquellas localizaciones en las que a lo largo de todo el estudio tuvieron 3 o menos registros, ya que una localización en la que se haya puesto una trampa o dos no aporta información suficiente como para ser tenida en cuenta. La variable objetivo son las capturas ponderadas, obtenidas dividiendo el número real de capturas entre la media de la localización correspondiente. Para estudiar esta variable respecto a las variables climáticas, se obtienen los valores ponderados de forma mensual por cada localización.

Una vez realizadas estas transformaciones, se combinan ambos conjuntos y se realiza la imputación. Al mismo tiempo, se normalizan las variables numéricas predictoras y se eliminan aquellas que tienen demasiada correlación. En el caso de la Salinidad y la Conductividad, dada la enorme cantidad de datos perdidos de la primera se elimina esta de forma manual para evitar perder información innecesariamente.

Tras obtener el conjunto de datos ambientales sin valores perdidos, puede combinarse con el fichero de capturas en trampas y proceder a la modelización.

7.2.1. Bosques aleatorios

El número de variables predictoras en este caso es muy superior al del apartado anterior. Incluso eliminando aquellas con correlaciones demasiado altas, siguen quedando 13 variables, por lo cual el método de bosques aleatorios es bastante conveniente, ya que solo utiliza un subconjunto de estas en cada división. Los parámetros que pueden optimizarse son: mtry (número de variables elegidas para cada nodo), trees (número de árboles en el bosque) y min_n (tamaño mínimo para que un nodo vuelva a dividirse). El tuning de estos parámetros se realiza mediante **tidymodels**.

Se evalúa tanto para los datos de las lagunas temporales como para las marismas el porcentaje de error cometido sobre la variable de capturas ponderadas en la muestra test.

```
## [1] "El valor óptimo de mtry en las lagunas es  1"
```

```
## [1] "El error cometido en las lagunas es del 25.84 %"
```

```
## [1] "El valor óptimo de mtry en las marismas es  1"
```

```
## [1] "El error cometido en las marismas es del 27.55 %"
```

En ambos casos el error está alrededor del 20 %. Es bastante grande, pero hay que considerar que el valor medio de esta variable es 1, de modo que cualquier pequeña interferencia supone un porcentaje de error considerable.

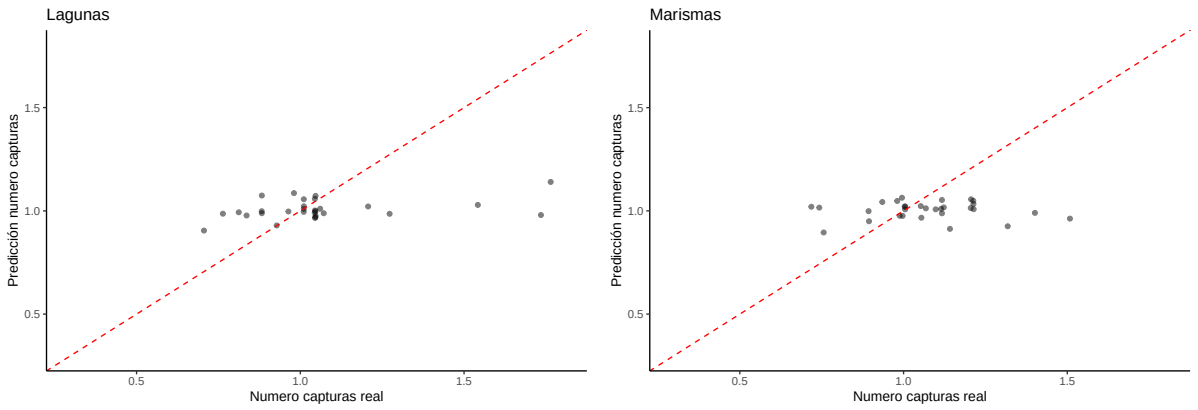


Figura 7.12: Predicciones capturas bosques aleatorios según hábitat

En las gráficas se puede ver la comparación entre los valores reales de la variable y los valores de sus predicciones. Para poder hacer una comparación adecuada, ambas se han representado con una misma escala. En ninguna de las dos gráficas hay demasiados individuos sobre la diagonal, y no se ve que los comportamientos sean muy diferentes entre ambos modelos. El caso de las lagunas tiene casos más dispersos pero el resto están más concentrados cerca de la diagonal, lo cual compensa el error cometido y por ello ambos modelos tenían un porcentaje de error similar.

Estos dos modelos fueron realizados sobre datos de las variables predictoras que en algunos casos fueron imputados, esto favorece una muestra de mayor tamaño pero tal vez

provoque una estimación incorrecta. En caso de haber tenido suficientes datos al descartar aquellos con valores perdidos, hubiera sido preferible limitarse a usar estos.

Para confirmar que la decisión de imputar los valores perdidos de las variables predictoras fue correcta, se generan también los correspondientes modelos utilizando únicamente los registros completos. El error cometido en ese caso es:

```
## [1] "El valor óptimo de mtry en las lagunas es  1"
```

```
## [1] "El error cometido en las lagunas es del 12.81 %"
```

```
## [1] "El valor óptimo de mtry en las marismas es  1"
```

```
## [1] "El error cometido en las marismas es del 11.41 %"
```

El error cometido parece ser menor, aunque cabe destacar que en el proceso de creación y evaluación del modelo fueron numerosas las alertas (*warnings*) que saltaron, advirtiendo de que se requerían muestras de mayor tamaño que las disponibles. Esto significa que el modelo generado no cumple los mínimos con los que fueron ideados los comandos de R utilizados, y por tanto no tienen garantías de ser correctos.

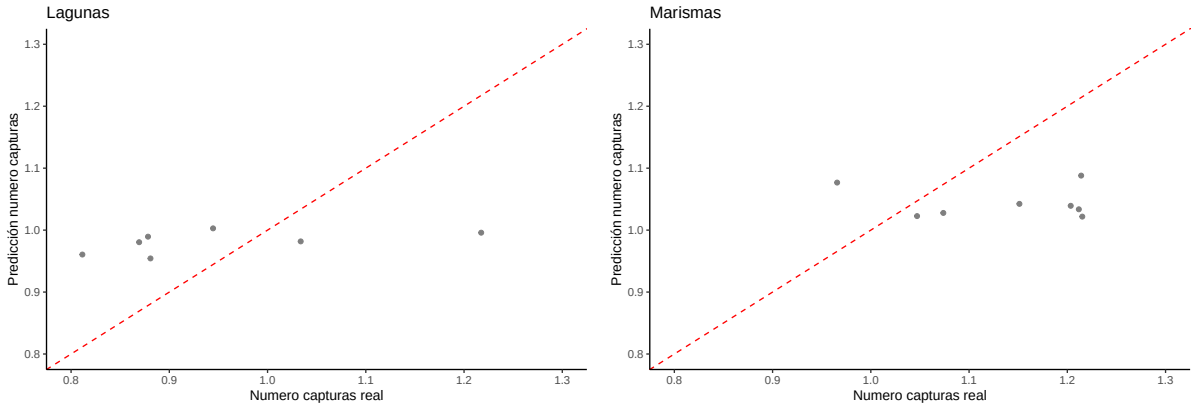


Figura 7.13: Predicciones capturas bosques aleatorios según hábitat

Visualmente lo primero que se aprecia es la cantidad de puntos en cada una de las gráficas, mucho menor que en el anterior caso. Además, en el caso de las lagunas casi todas las predicciones fueron iguales, incluso cuando el rango de valores reales era amplio para la variable con la que se está tratando. En el caso de las marismas esto no es tan llamativo, pero también se nota que los datos están más dispersos que en el modelo creado en base a los datos imputados.

Por tanto, el uso de datos imputados para las variables predictoras que tenían valores perdidos está justificado en este caso, y no haber recurrido a este método podría haber llevado a error con modelos que engañosamente parecen tener un error pequeño pero que no proporcionan estimaciones de calidad.

7.2.2. K vecinos más cercanos

Por último, puede implementarse el método de los K vecinos más cercanos. De nuevo se realizarán los modelos de forma separada en cada uno de los hábitats y partiendo del conjunto de datos con los valores perdidos de las variables predictoras imputados. No se crea en este caso el modelo con los datos sin imputar, ya que se vio que no es correcto utilizar ese conjunto. Se realiza tuning para los parámetros, de modo que se escojan los mejores para la muestra de entrenamiento generada, y el porcentaje de error se estudia sobre la muestra test.

Para el conjunto de datos de las lagunas los resultados obtenidos son:

```
## [1] "El mejor k para el conjunto de entrenamiento de las lagunas es 9"
```

```
## [1] "El error cometido en las lagunas es del 25.51 %"
```

Y lo mismo para el conjunto de las marismas:

```
## [1] "El mejor k para el conjunto de entrenamiento de las marismas es 14"
```

```
## [1] "El error cometido en las marismas es del 26.48 %"
```

El error cometido en porcentaje es cercano al que se cometía con los modelos mediante bosques aleatorios, no hay una diferencia significativa como para afirmar que uno de los métodos sea mejor que el otro.

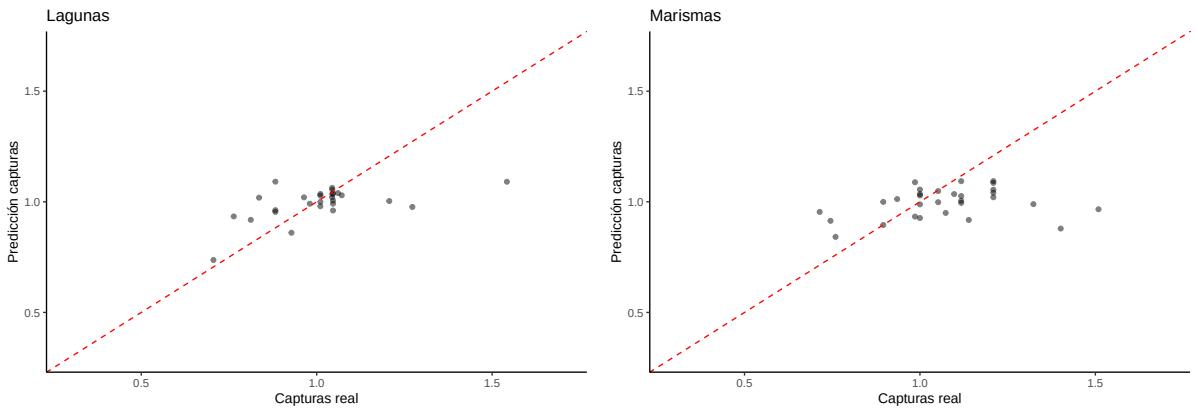


Figura 7.14: Predicciones capturas KNN según hábitats

Las predicciones parecen tener una distribución similar a las obtenidas en los modelos de bosques aleatorios en ambos casos. Para las lagunas están más agrupadas en la zona central de la diagonal pero con algún caso dispreso. Para las marismas están más repartidas en general, aunque el error medio cometido es casi igual en ambos conjuntos.

De todos modos, como ya se comentó al inicio del capítulo, han sido muchos los problemas afrontados para conseguir obtener modelos coherentes dados los conjuntos de datos disponibles. Probablemente si se tuvieran datos de forma más continuada podrían haberse

Tabla 7.8: Comparativa modelos sobre capturas

Modelo	% error	RMSE
Bosques aleatorios, lagunas	25.843	0.267
Bosques aleatorios, marismas	27.553	0.278
K vecinos más cercanos, lagunas	25.506	0.263
K vecinos más cercanos, marismas	26.482	0.268

implementado otros métodos que permitieran relacionar la cantidad de capturas de un mes en concreto no solo con las variables registradas ese mes sino con los meses anteriores. No se tienen garantías de que ese modelo fuera a cometer un error menor que los aquí presentados, pero de cara al significado biológico de los datos tendría una mayor sentido.

Al igual que en la sección anterior, los mejores modelos se representan en la Tabla 7.8. Hay dos modelos por cada hábitat, solo se han mostrado aquellos que utilizaron los datos imputados climáticos. Los resultados son muy similares para cada par de modelos, de modo que podrían utilizarse indistintamente para obtener estimaciones según el hábitat de interés.

Parte IV

Conclusiones

Capítulo 8

Conclusiones

En este último capítulo del documento se pretende reflejar cómo fueron abordados los objetivos iniciales presentados en la Introducción, los pasos que se dieron a lo largo del estudio y las conclusiones que fueron extraídas. Antes de entrar en detalle sobre los modelos obtenidos, se hará un repaso de aquellos puntos de interés aportados en este trabajo como medios para alcanzar dichos objetivos. También se mencionarán las vías de trabajo que no pudieron llevarse a cabo, y aquellas que en un trabajo posterior podría plantearse emprender.

8.1. Aportaciones iniciales

Para el estudio inicial, se disponía de cuatro conjuntos de datos. Tres de ellos con un formato más depurado que fueron publicados simultáneamente, y otro menos preparado que requería de un tratamiento previo para poder ser estudiado. La información disponible trataba sobre capturas individuales de cangrejos, así como de capturas en trampas e información sobre el clima y las variables ambientales en las disitntas localizaciones de Doñana que fueron estudiadas. Estos conjuntos fueron examinados en detalle, eliminando aquella información redundante o errónea, y se realizaron múltiples transformaciones para poder trabajar mejor con ellos. Esta limpieza y ordenación de los datos permitió tener conjuntos más reducidos pero ya listos para ser tratados, en caso de querer realizar un estudio con un enfoque distinto. Es importante destacar la creación de una nueva variable: capturas ponderadas. Esta variable fue creada para permitir tratar con la variable de capturas en trampas de una forma distinta, prestando mayor atención a si hubo más o menos capturas de lo habitual en cada sitio y no solo al número de capturas, ya que esta variable no es en sí un fin sino un medio para estimar la población total de cangrejos y su evolución, cosa que difícilmente se puede estimar a partir de los datos disponibles.

Al haber enfrentado un problema real, esto conllevó tratar con problemas en la metodología y continuidad de la recopilación de datos, además de la considerable cantidad de datos perdidos. Es debido a esto que el estudio a través de Series Temporales, entre otros, tuvo que ser descartado. Fue necesario revisar en detalle, importar y tratar cuidadosamente los conjuntos, relacionándolos entre sí cuando fuera posible, o reestructurándolos cuando fuera necesario. Se estudiaron a fondo las variables, para discernir cuáles aportaban información y cuales no eran de provecho para el estudio llevado a cabo. Para

aquellas que fueron seleccionadas se estudiaron a fondo sus posibles interacciones y se hizo un análisis de correlaciones.

El exhaustivo análisis exploratorio realizado permitió extraer conocimiento de gran utilidad, tanto de cara a los datos obtenidos como sobre la especie y el entorno. En el capítulo 4 se destacan en distintas gráficas y tablas aquellos puntos que se consideraron de mayor interés, y se plasmaron las conclusiones fundamentales extraídas a lo largo del proceso previo. Algunas de estas aportaciones que se vieron gráficamente fueron posteriormente contrastadas con métodos estadísticos en la sección de análisis distribucional, esto permitió confirmar lo que se había intuido gráficamente y de esa forma realizar unos modelos que ajustasen mejor los datos. Entre las gráficas destacables están aquellas relativas a la cantidad de capturas, a la proporción de individuos inmaduros y a la evolución en el número de individuos. Todas ellas que pueden ser útiles para comprender mejor la población establecida en Doñana.

Además de eso, se preparó una aplicación basada en R shiny que permite visualizar la información disponible limitándose a una localización concreta o un periodo. Esta aplicación visual sirvió tanto para poder explorar los datos de forma más cómoda durante este estudio, como para facilitar a los gestores del ecosistema visualizar determinados aspectos de sus datos. La ventaja de esta aplicación es la capacidad de mostrar un enorme número de gráficas abarcando subconjuntos reducidos de la información, en vez de tener que visualizarla en su totalidad sin poder prestar atención al detalle.

En general, se puede considerar que una contribución útil de este trabajo fueron las lecciones aprendidas respecto a la metodología de recopilación de este tipo de información. De cara al futuro, podría utilizarse para aconsejar a los gestores, de modo que pueda dotarse de datos más útiles para el desarrollo de modelos. También se propusieron soluciones para la aparente imposibilidad inicial de vislumbrar la evolución de la población en base a muestreos que no seguían una periodicidad adecuada a lo largo del tiempo.

Desde un punto de vista más personal, este estudio y el esfuerzo realizado para comprender los datos y poder trabajarlos han servido para desarrollar la capacidad de resolución de problemas de cara al mundo real. Haber ido trazando las líneas de estudio en base a lo que se consideró de interés, pero al mismo tiempo con las limitaciones encontradas, ha aportado una visión más amplia que la que se tenía tras los estudios universitarios, esto seguro facilitará el futuro acercamiento a problemas de esta magnitud.

Las aportaciones aquí mencionadas fueron útiles para poder comprender los conjuntos de datos con los que se trabajó, y gracias a esto pudieron llevarse a cabo los modelos del bloque III. Estos modelos son los que responden a los objetivos planteados inicialmente, y se evaluará si cumplieron con su propósito en la sección a continuación.

8.2. Conclusiones

El objetivo principal de comprender la evolución de la población fue llevado a cabo de forma separada en cada tipo de hábitat, Lagunas y Marismas, pues, según se vio en el análisis previo, el comportamiento de las variables ambientales y del número de capturas realizadas era significativamente distinto en ambos grupos. Para cada hábitat se realizaron modelos mediante Bosques aleatorios y mediante el método de K vecinos más próximos. Una de las carencias encontradas al intentar realizar este tipo de modelización fueron los

datos perdidos en las variables ambientales, y la dificultad de combinar la información de las capturas diarias con las variables ambientales mensuales. Para resolver lo primero, se realizó una imputación de dichos valores perdidos, de modo que pudiera contarse con los registros completos. Para lo segundo, las capturas se estudiaron agrupándolas de forma mensual. Aun así, era reducida la información disponible, pero los modelos finales ajustaban moderadamente bien la variable objetivo, estando aún abiertos a posibles mejoras.

La variable objetivo no fue el número de capturas en bruto como se registraba en los datos originales, sino la variable de capturas ponderadas creada en este trabajo. Esta variable, obtenida al dividir el número de capturas de cada registro entre la media de la localización correspondiente, permite estudiar si las capturas estarán por encima o por debajo de la media. Una vez obtenida la estimación de las capturas ponderadas, bastaría con multiplicar por la media de capturas para obtener una estimación del valor real. Los modelos obtenidos en esta línea tienen todos un rendimiento similar, pero puede destacarse que para ambos tipos de modelos en las lagunas el error fue menor que en las marismas, tanto en porcentaje como el RMSE. Para la comparación entre los dos tipos de modelos, en ambos hábitats puede observarse una ligera disminución del error al usar los K vecinos más cercanos frente al modelo de bosques aleatorios. Sin embargo, la variación es mínima, en las lagunas el RMSE era 0.263 frente a 0.267 y en las marismas 0.268 frente a 0.278. Las medidas exactas del rendimiento de los cuatro modelos pueden verse en mayor detalle en la Tabla 7.8.

Respecto a las cuestiones de clasificación, se intentó predecir la variable Sexo y la etapa de Madurez de un cangrejo en base a las cuatro variables que se registraban sobre su tamaño: largo del cuerpo, largo del cefalotórax, ancho del cefalotórax y peso. Para ambas clasificaciones se utilizaron los mismos métodos para crear los modelos: K vecinos más próximos, Árboles de clasificación, Bosques aleatorios y Regresión Logística. Debido al considerable número de valores perdidos en todas las variables de tamaño salvo en el Largo del cuerpo, se crearon dos tipos de modelos: solo utilizando el largo del cuerpo, o utilizando las cuatro variables como predictoras.

En el caso de la variable Sexo, los mejores modelos se dieron utilizando las cuatro variables del tamaño como predictoras. El modelo mediante Bosques aleatorios parece tener un mejor rendimiento que los otros dos, teniendo una exactitud de 0.685, pero aun así las predicciones obtenidas no son suficientemente buenas como para que pueda considerarse fiable. El siguiente mejor modelo, tomando la Exactitud en consideración, fue el de K vecinos más cercanos con un valor de 0.645, y por último el de Bosques aleatorios con 0.602. Pueden consultarse las demás medidas del rendimiento obtenidas por cada modelo en 6.13. Se concluye por tanto que en caso de que se tuviera interés en realizar esta clasificación debería hablarse con los ecólogos expertos en el tema para replantear el modelo. Se podría estudiar otro tipo de variables que se relacionasen más con el sexo de los cangrejos, ya que el tamaño no es una característica distintiva entre Machos y Hembras de esta especie.

Para la variable Madurez, en cambio, los resultados fueron mucho más favorables. En este caso, los mejores modelos fueron aquellos que utilizaban únicamente el Largo del cuerpo como variable predictora. En comparación, los tres modelos estudiados parecen ser bastante similares en cuanto a su capacidad de predicción, teniendo el modelo de Bosques aleatorios una exactitud de 0.911 y los otros dos modelos de 0.909. Todos ellos tienen un ajuste suficientemente bueno como para poder confiar en ellos para predicciones

futuras, y se pueden ver en más detalles sus medidas de rendimiento en la Tabla 6.26.

Por último, de cara a las variables de tamaño de los cangrejos, se intentó modelizar el largo del cefalotórax, el ancho del cefalotórax y el peso en función del largo del cuerpo. Para ello, inicialmente lo que se aplicó fue Regresión Lineal, pero los resultados obtenidos no eran suficientemente buenos. Tras un análisis en detalle, se observó una diferencia en el comportamiento que seguían el largo del cefalotórax y el ancho del cefalotórax en distintos periodos de tiempo. Este extraño comportamiento, aunque de explicación desconocida, permitió obtener modelos mucho mejores simplemente al dividir los datos en dos grupos (antes de 2010 y después de 2010) y modelizarlos por separado. Para el largo del cefalotórax en ambos periodos el RMSE fue similar, 2.77 para datos antes de 2010 y 3.05 después de 2010. En cambio, en el ancho del cefalotórax la diferencia es mayor, teniendo un RMSE de 5.11 antes de 2010 y de 1.59 después, aunque el porcentaje de error no tiene tanta diferencia, así que puede deberse a que los valores de esta variable en el primer periodo eran mayores. Para ver en más detalle las medidas del rendimiento de los cuatro modelos puede consultarse la Tabla 7.6. Es importante tener en cuenta la diferencia en los dos periodos de cara a futuras predicciones que pudieran quererse llevar a cabo, ya que, si quisiera utilizarse por completo el conjunto disponible como entrenamiento, aquellos datos anteriores a 2010 provocarían un grave empeoramiento del ajuste. Sería de interés poder consultar a aquellos ecólogos que trataron con los individuos en cada uno de los periodos temporales si la diferencia fue considerablemente notoria para ellos y qué explicación puede tener esa diferencia en la tendencia observada.

Respecto al peso de los cangrejos, su comportamiento no se veía afectado por la división temporal, pero aun así tampoco era bueno el ajuste que se obtenía con la Regresión Lineal simple. La solución fue aplicar una transformación logarítmica, tras la cual el ajuste mejoró enormemente, obteniendo un porcentaje de error del 16.2 % frente al 42.42 % del modelo lineal simple. Ya que esta variable no parecía comportarse como las demás, también se realizó un modelo mediante Bosques Aleatorios utilizando las otras tres variables de tamaño como predictoras, en este caso el porcentaje de error fue 14.3 %. Puede verse también el RMSE de ambos modelos en la Tabla 7.7.

8.3. Caminos descartados

Al principio del bloque de Modelos Predictivos se mencionaron algunas ideas con las que se intentó trabajar para las variables ambientales pero que no fue posible llevar a cabo. El estudio como Serie Temporal sería posible si las mediciones se hubieran tomado de forma más consecutiva, y no tan aisladas. En caso de haber realizado mediciones de las variables ambientales todos los meses durante un periodo de algunos años, podría haberse realizado un estudio mediante modelos SARIMA (*Seasonal Auto Regresive Integrated Moving Average* o Modelo Estacional Auto Regresivo Integrado de Medias Móviles). Esto permitiría ajustar las variaciones en función de los meses, así como la tendencia general y el aumento progresivo de la temperatura que ocurre en el planeta. Este modelo no pudo llevarse a cabo ya que la toma de datos que se realizó no tenía la regularidad necesaria en un único lugar, sino que abarcaba un amplio número de localizaciones de forma más esporádica.

De cara a la falta de registros completos se intentó mejorar la situación aplicando muestreo con reemplazamiento o *bootstrap*, pero el resultado obtenido no mejoró lo suficiente

los resultados, y por tanto se terminó descartando. La alternativa que se llevó a cabo no aumentó el número real de registros, sino que intentó completar los valores perdidos de los ya disponibles mediante imputación knn, para de esa forma poder aprovechar al máximo aquellos valores que se tomaron de forma correcta.

La modelización que se llevó a cabo sobre las capturas se hizo a escala de hábitat, agrupando todas las localizaciones en dos grupos según eran Marismas o Lagunas. Esto fue debido a que algunas de las localizaciones tenían un número muy reducido de registros tomados, que aunque eran de utilidad no podían servir por sí solos para construir ningún modelo válido. Algunas de las localizaciones sí tenían un número considerable de registros tomados, de modo que se intentó en esas en concreto llevar a cabo el estudio más personalizado centrándose en ellas. El procedimiento fue el mismo que el llevado a cabo en los hábitats y por tanto no se consideró que incluirlo en esta memoria fuera a aportar nada nuevo, pero cabe destacar que, debido al reducido tamaño muestral, el error cometido era excesivo y por tanto se deshechó la idea. No obstante, en caso de tener suficiente información sobre una única localización, sería de interés ver si las predicciones mejoran respecto a los modelos que engloban varias. En caso de ser una zona de gran interés este modelo sería útil, pero no podría extrapolarse a zonas distintas, en cambio los modelos generados podrían aplicarse a nuevas localizaciones que fueran del mismo tipo de hábitat y estudiar la eficacia.

Por último, mencionar los modelos que se encuentran en el Apéndice B, que engloba el código cuyos resultados no se incluyeron en el bloque de Modelos Predictivos, pero que fueron considerados merecedores de aparecer reflejados en este documento. El error cometido por los modelos de regresión lineal no era excesivamente alto, en algunos casos se asemejaba al cometido por los modelos sí incluídos en esta memoria, sin embargo en muchos casos la hipótesis de nulidad de los coeficientes era aceptada, e incluso cuando se rechazaba el coeficiente de determinación ajustado no era suficientemente alto como para considerar fiables los modelos.

8.4. Líneas futuras de trabajo

Como ya fue mencionado en la Introducción, la temática de este trabajo nacía de la colaboración con el grupo de investigación en Computación Natural, y de un proyecto anterior que estos investigadores tenían en conjunto con los ecólogos que recolectaron los datos. Gracias al profundo análisis que se ha llevado a cabo en este tema, sería muy interesante intensificar el contacto con los expertos ecólogos que colaboraron en el proyecto inicial. A lo largo de este estudio se han planteado algunas dudas y cuestiones que no han podido ser resueltas, ya que escapan al ámbito matemático, pero que en un futuro podrían ser consultadas. Aunque, a día de hoy, no pudieron expandir el proyecto con el que pretendían abordar complejos modelos ecológicos sobre esta especie y ecosistema, es posible acudir a futuras convocatorías, y en caso de aprobarse el proyecto se podría aprovechar el conocimiento adquirido y las herramientas proporcionadas por este Trabajo de Fin de Grado.

También se podría plantear el uso de redes neuronales para crear modelos que tuvieran un mejor rendimiento. En este estudio se hizo un intento superficial de aplicar este método, pero no se obtuvo ninguna mejora y por tanto no se incluyó el modelo. Es probable que si se hubiera tenido un conjunto mayor de datos disponibles, se hubiera podido conseguir

un modelo considerablemente mejor. En todo caso, no se consideró conveniente intentar abarcar demasiado sino que se priorizó que los modelos llevados a cabo fueran realizados a conciencia. De cara a trabajos futuros, sería una vía interesante que explorar en mayor profundidad y que tal vez podría dar buenos resultados.

En cualquier caso, tras este estudio quedan abiertas algunas dudas que sería conveniente consultar con expertos en la materia a ser posible. Tal vez de esta forma las ideas planteadas se pudieran madurar y abrir nuevas cuestiones que mejorasen el análisis realizado. La profundidad que puede alcanzarse en el estudio de este tema es prácticamente ilimitada, y el trabajo que aquí se presenta no pretendía dar respuesta absolutas, sino mostrar los resultados del esfuerzo realizado. Del mismo modo, se aspiraba a establecer una base sobre la que partir en caso de que se continuase el estudio de la población de *Procambarus Clarkii* en Doñana, y aligerar la carga que puedan tener en el futuro estudios similares puedan llevarse a cabo.

Apéndice A

Apéndice: Código omitido

A.1. Importación

En primer lugar se refleja el código utilizado para importar cada uno de los conjuntos que fueron tratados en este trabajo. El proceso fue detallado en el apartado de importación y preprocesado, aunque de nuevo se hará un pequeño resumen de los detalles a tener en cuenta.

Solo cuatro de las quince variables del conjunto OCC eran útiles, por tanto se hace una selección de ellas. La primera englobaba información correspondiente a dos variables distintas, así que se separan y por último cada variable se transforma en el formato correcto.

```
occ_bruto <-
  read_delim("data/Don_ biom_red-swamp-crayfish _occ_20230111.csv",
    delim = ";", escape_double = FALSE,
    col_types = cols(occurenceTime = col_time(format = "%H:%M:%Sz")),
    trim_ws = TRUE) %>%
  select(!matches("\\.\\.\\.")) %>%
  filter(!is.na(eventID))

occ <- occ_bruto %>%
  select(occurrenceID, occurenceTime, sex, lifestage) %>%
  separate(occurrenceID, sep="_", into=c("IDL","fecha","Cangrejo")) %>%
  mutate(IDL = as.numeric(IDL),
         fecha = as.Date(fecha, format="%Y-%m-%d"),
         Cangrejo = as.numeric(Cangrejo),
         sex = as.factor(sex),
         lifestage = as.factor(lifestage))
```

En el fichero original de MOF había siete variables, de las cuales solo tres eran de interés. De nuevo una de ellas debía separarse en dos distintas. Además está el detalle de que debía realizarse una reordenación para que los datos tuvieran la forma adecuada, y al hacerlo los nombres de las variables quedaban con espacios, así que se renombraron y de paso se pusieron los nombres en español. Por último, se filtra un dato que era erróneo y se corrige el tipo de algunas de las variables.

```
mof_bruto <-
  read_delim("data/Don_ biom_red-swamp-crayfish _mof_20230111.csv",
    delim = ";", escape_double = FALSE, trim_ws = TRUE)


mof <- mof_bruto %>%
  select(occurrenceID, measurementValue, measurementType) %>%
  separate(occurrenceID, sep="_",
           into=c("IDL","fecha","Cangrejo")) %>%
  mutate(fecha = as.Date(fecha, format="%Y-%m-%d")) %>%
  pivot_wider(names_from = measurementType,
              values_from = measurementValue) %>%
  rename(largo_cuerpo = `total body length`,
         largo_cefalotorax = `cephalothorax length`,
         ancho_cefalotorax = `cephalothorax width`,
         peso = weight) %>%
  filter(largo_cuerpo < 900) %>%
  mutate(IDL = as.numeric(IDL),
         Cangrejo = as.numeric(Cangrejo))
```

Una vez se han leído y depurado estos dos conjuntos, se puede hacer la unión de ambos. Este nuevo conjunto es con el que se trata para los métodos de clasificación.

```
occ_mof <- occ %>%
  inner_join(mof, by=c("Cangrejo", "IDL", "fecha"))
```

El fichero original de EV contenía 18 variables, sin embargo solo siete de ellas contenían información útil. El formato que tenían las coordenadas latitud y longitud no era numérico sino una cadena de texto con puntos tanto en los miles como para la coma decimal. Se corrige el formato de la cadena de texto y se transforma en numérico. La variable hábitat se transforma en factor y se cambian los nombres de sus categorías para que sean más cortos y estén en español. Por último, aquellos registros que tuvieran un valor perdido fueron descartados, ya que la única variable con valores perdidos era la que registra el número de capturas, y tener ese valor es la finalidad de este conjunto.

```
ev_bruto <-
  read_delim("data/Don_biom_red-swamp-crayfish_ev_20230111.csv",
    delim = ";", escape_double = FALSE,
    col_types = cols(eventDate = col_date(format = "%d/%m/%Y"),
        eventTime = col_time(format = "%H:%M:%Sz"),
        decimalLatitude = col_character()),
    locale = locale(encoding = "ISO-8859-1"),
    trim_ws = TRUE)

ev <- ev_bruto %>%
  select(eventDate, localityID, eventTime,decimalLatitude,
         decimalLongitude, habitat, sampleSizeValue) %>%
```

```
   mutate(
     decimalLatitude = as.numeric(str_remove(decimalLatitude,"\\."))/100,
     decimalLongitude = as.numeric(decimalLongitude)/100) %>%
  mutate(habitat = as.factor(habitat) %>%
           `levels<-`(c("Lagunas temporales","Marismas"))) %>%
  drop_na()
```

El último conjunto es el que registra las variables ambientales, como ya se comentó este proviene de un excel, a diferencia del resto que provenian de ficheros csv. Además, estos datos no fueron publicados al mismo tiempo que los otros, y por tanto requirieron de un trabajo de depuración más meticuloso.

La variable originalmente nombrada `Profundidad de medida...12` tenía solo dos posibles valores, o bien "Superficie" o bien NA, en cuyo caso ninguna de las variables numéricas tomaba un valor, eran todos perdidos también. Por tanto, lo primero que se hizo fue filtrar que esta variable no tuviera un valor perdido. Después de esto se eliminaron aquellas variables que no tenían información útil para los análisis posteriores. En este caso el número de variables de utilidad era mayor, por lo que se realizó la selección a la inversa.

En varios de los casos las variables tenían espacios o caracteres extraños, por lo que se renombraron para evitar tildes, ñ y espacios. Se creó una nueva variable que agrupase Mes y Año, para poder hacer visualizaciones de las variables ambientales a lo largo del tiempo. También se corrigió el tipo de algunas que no fue registrado correctamente. Finalmente tuvo que aplicarse la función `unique()` pues la variable `Código nasa` generaba gran cantidad de registros por cada localización y mes sin ninguna variación en las variables ambientales, pero al eliminarla ya no era necesario tener tantas repeticiones.

Además de esto, tras eliminar los valores repetidos se advirtió que algunos de los datos tenían un comportamiento extraño. Para una misma fecha y localización había muchos registros que solo diferían en una única variable, y esta variable iba aumentando 1 unidad su valor en cada registro. Se asumió que se trataba de un error al introducir los datos, ya que esto producía valores imposibles en el pH entre otras cosas. Por tanto fueron eliminados todos los pares Fecha y localización con 3 o más registros diferentes.

```
pres_abs_bruto <- read_excel("data/Doñana_Crayfish_Databases.xlsx",
    sheet = "Presence-Absence + FQvariables",
    col_types = c("numeric", "numeric", "numeric",
        "numeric", "text", "text", "text",
        "numeric", "text", "text", "text",
        "text", "text", "numeric", "numeric",
        "numeric", "numeric", "numeric",
        "numeric", "numeric", "numeric",
        "numeric", "numeric"),
    na = "N/D")

pres_abs_depurado <- pres_abs_bruto %>%
  filter(!is.na(`Profundidad de medida...12`)) %>%
  select(-ID, -`Tipo nasa`, -`Código nasa`, -`Procambarus clarkii`,
         -`Profundidad de medida...12`, -`Profundidad de medida...13`) %>%
```

```
  rename(Ano = Año,
         Habitat = Hábitat,
         Sup_encharcada = `Superficie encharcada`,
         Prof_max = `Profundidad máxima`,
         Vegetacion = Vegetación,
         Temperatura = `Temperatura del agua`,
         O2_porc = `Oxígeno disuelto en porcentaje`,
         O2_abs = `Oxígeno disuelto en valor absoluto`,
         Clorofila = `Clorofila a`) %>%
  mutate(Fecha = Ano + (Mes-1)/12,
          Habitat = as.factor(Habitat),
          Sup_encharcada = as.factor(Sup_encharcada),
          Prof_max = as.numeric(Prof_max),
          Vegetacion = as.factor(Vegetacion)) %>%
  unique()

pres_abs <- pres_abs_depurado %>%
  anti_join(pres_abs_depurado  %>%
              group_by(Ano, Mes, IDL) %>%
              summarise(registros = n()) %>%
              filter(registros > 2),
            by=c("Ano","Mes","IDL"))
```

La visualización de estos datos no se hizo con `summary()` como en los conjuntos anteriores, sino que se recurrió a la librería **skimr** para obtener algunas medidas de las variables numéricas y estas se reflejaron en una tabla.

```
pa_skim <- pres_abs %>% select_if(is.numeric) %>% skim()

pa_tab <- cbind(pa_skim$n_missing, pa_skim$complete_rate,
                pa_skim$numeric.mean, pa_skim$numeric.sd)

rownames(pa_tab) <- pres_abs %>% select_if(is.numeric) %>% names()

pa_tab %>% round(3) %>%
  kable(col.names = c("Valores perdidos", "% completos",
                      "Media", "Desviación típica"),
        caption = "\\label{PresAbsNum}Variables ambientales numéricas")
```

Para la visualización de las variables categóricas sí que se usó `summary()` tras seleccionar solo esas tres variables.

A.2. Análisis descriptivo

Una vez realizada la importación y limpieza de los datos, lo siguiente fue el análisis descriptivo. En la mayoría de los casos no hubo que hacer modificaciones a los conjuntos para

crear las gráficas, y el código de estas no tiene especial interés por lo que no será incluido en este apéndice. Sin embargo, para el conjunto EV sí que se realizó una transformación para hacer la ponderación y la posterior agrupación por trimestres.

La ponderación consistió en hacer la agrupación por localización, hallar la media de las capturas en cada una y luego crear una variable que almacenase el cociente entre las capturas reales registradas y la media habitual en esa localización. La información se guardó en un nuevo objeto `ev_graf` ya que inicialmente se utilizó para el apartado de Análisis gráfico.

```
ev %>% select(eventDate, localityID, sampleSizeValue, habitat) %>%
  group_by(localityID) %>%
  mutate(media = mean(sampleSizeValue, na.rm=TRUE),
         capt = sampleSizeValue/media) -> ev_graf
```

Para la agrupación por trimestres se partió del conjunto `ev_graf` y se creó una variable para almacenar el trimestre al que correspondía el registro.

```
ev_graf %>%
  mutate(fecha_trim = floor(month(eventDate)/3) /4 + year(eventDate)) %>%
  group_by(fecha_trim, localityID, habitat) %>%
  summarise(sampleSizeValue = sum(sampleSizeValue, na.rm = TRUE)) %>%
  ungroup() %>% group_by(localityID) %>%
  mutate(media = mean(sampleSizeValue, na.rm=TRUE),
         capt = sampleSizeValue/media,
         estacion = fecha_trim%%1*4+1,
         est_nombre = as.factor(c("Inv","Pri","Ver","Oto")[estacion])
         ) -> ev_graf_trimestre
```

Para las tablas que reflejaban la media y desviación típica de las capturas en trampas en función del hábitat y la estación, se aplicó primero la función `skim()` de la librería **skimr**. Posteriormente se seleccionó la medida de interés, habiendo filtrado previamente solo aquellos datos que se correspondían a sampleSizeValue. Una vez seleccionados los datos, se reordenaron para ser visualizados de forma más sencilla y gracias a la función `kable()` de **kableExtra** se creó la tabla ya vista anteriormente.

```
ev_graf_skim <- ev_graf_trimestre %>% ungroup() %>%
  group_by(est_nombre, habitat) %>%
  skim() %>% filter(skim_variable == "sampleSizeValue")

ev_graf_t1 <- ev_graf_skim %>%
  select(est_nombre, habitat, numeric.mean) %>%
  rename(media = numeric.mean) %>% mutate(media = round(media,2)) %>%
  pivot_wider(names_from = c("est_nombre"),
              values_from = c("media"))

ev_graf_t2 <- ev_graf_skim %>%
  select(est_nombre, habitat, numeric.sd) %>%
```

```
  rename(desviacion_tipica = numeric.sd) %>%
  mutate(desviacion_tipica = round(desviacion_tipica,2)) %>%
  pivot_wider(names_from = c("est_nombre"),
              values_from = c("desviacion_tipica"))
```

A.3. Modelos clasificación

Para predecir las variables Sexo y Madurez se utilizaron como variables predictoras las variables sobre el tamaño de los cangrejos: largo del cuerpo, largo del cefalotorax, ancho del cefalotorax y peso. Se crearon modelos por el método de los K-vecinos más cercanos y regresión logística tanto usando solo el largo del cuerpo como con las demás. Para árboles de decisión solo se usó el largo del cuerpo y para bosques aleatorios se utilizaron todas. Las muestras de entrenamiento y test se generaron de forma estratificada respecto de la variable objetivo, y se mantuvieron las mismas para todos los métodos realizados.

Hay dos subconjuntos por cada variable: el primero es el que no tiene ningún valor NA, ni en la variable objetivo ni en las variables predictoras. El segundo solo se necesita que sea conocida la variable objetivo, ya que el largo del cuerpo es conocido en todos los cangrejos registrados. Las muestras generadas utilizan un 70 % de los datos para la muestra de entrenamiento y un 30 % para muestra test.

```
occ_mof_completo_sexo <- occ_mof %>%
  filter(!is.na(sex),!is.na(largo_cefalotorax),
         !is.na(ancho_cefalotorax),!is.na(peso))

occ_mof_comp_sexo_split <- occ_mof_completo_sexo %>%
  initial_split(prop=0.7,strata = sex)

occ_mof_comp_sexo_ent <- occ_mof_comp_sexo_split %>% training()
occ_mof_comp_sexo_test <- occ_mof_comp_sexo_split %>% testing()


occ_mof_largo_sexo <- occ_mof %>% filter(!is.na(sex))

occ_mof_largo_sexo_split <- occ_mof_largo_sexo %>%
  initial_split(prop=0.7,strata = sex)

occ_mof_largo_sexo_ent <- occ_mof_largo_sexo_split %>% training()
occ_mof_largo_sexo_test <- occ_mof_largo_sexo_split %>% testing()
```

```
occ_mof_completo_lfst <- occ_mof %>%
  filter(!is.na(lifestage),!is.na(largo_cefalotorax),
         !is.na(ancho_cefalotorax),!is.na(peso))

occ_mof_comp_lfst_split <- occ_mof_completo_lfst %>%
  initial_split(prop=0.7,strata = lifestage)
```

```
occ_mof_comp_lfst_ent <- occ_mof_comp_lfst_split %>% training()
occ_mof_comp_lfst_test <- occ_mof_comp_lfst_split %>% testing()


occ_mof_largo_lfst <- occ_mof %>% filter(!is.na(lifestage))

occ_mof_largo_lfst_split <- occ_mof_largo_lfst %>%
  initial_split(prop=0.7,strata = lifestage)

occ_mof_largo_lfst_ent <- occ_mof_largo_lfst_split %>% training()
occ_mof_largo_lfst_test <- occ_mof_largo_lfst_split %>% testing()
```

A continuación se muestra el código utilizado para crear los modelos en cada caso. El de las gráficas o las matrices de confusión es omitido ya que es repetitivo y tiene menor interés.

Para el modelo KNN con todas las variables predictoras se utiliza la librería **caret**, se aplica un centrado y escalado de las variables predictoras y se generan 11 valores para el número de vecinos, quedando en el modelo final el que mejor rendimiento tuviera en la muestra entrenamiento.

```
ctrl <- trainControl(method="cv",classProbs=TRUE,
                     summaryFunction = defaultSummary )

KNN_todas_sexo <-
  train(sex ~ largo_cuerpo + largo_cefalotorax + ancho_cefalotorax + peso,
        data = occ_mof_comp_sexo_ent,
        method = "knn",
        trControl = ctrl,
        preProcess =c("center","scale"),
        tuneLength=11 )

KNN_todas_lfst <-
  train(lifestage ~ largo_cuerpo + largo_cefalotorax +
          ancho_cefalotorax + peso,
        data = occ_mof_comp_lfst_ent,
        method = "knn",
        trControl = ctrl,
        preProcess =c("center","scale"),
        tuneLength=11)
```

Lo mismo se aplica para los modelos solo con el largo del cuerpo como variable predictora.

```
KNN_largo_sexo <-
  train(sex ~ largo_cuerpo,
        data = occ_mof_largo_sexo_ent,
```

```
        method = "knn",
        trControl = ctrl,
        preProcess =c("center","scale"),
        tuneLength=11)

KNN_largo_lfst <-
  train(lifestage ~ largo_cuerpo,
        data = occ_mof_largo_lfst_ent,
        method = "knn",
        trControl = ctrl,
        preProcess =c("center","scale"),
        tuneLength=11)
```

Los árboles de clasificación se generan con la librería **C50** que con una única instrucción genera el árbol, al que luego puede aplicarse un `plot()` para visualizarlo tal como se mostró en el apartado correspondiente.

```
arbol_largo_sexo <- C5.0(sex ~ largo_cuerpo,
                         data = occ_mof_largo_sexo_ent)

arbol_largo_lfst <- C5.0(lifestage ~ largo_cuerpo,
                        data = occ_mof_largo_lfst_ent)
```

Para el método de bosques aleatorios se utilizó la librería **randomForest**. Se indicó que el parámetros mtry fuera igual a tres, para así en cada nodo usar tres variables predictoras de las cuatro disponibles. De este modo hay variabilidad en los árboles por no usar siempre todas las variables pero permite optimizar en función de tres de ellas para obtener los mejores resultados. El número de árboles generados se fijó en 100.

```
bosque_todas_sexo <- randomForest(sex ~ largo_cuerpo + largo_cefalotorax +
                                    ancho_cefalotorax + peso,
                                  data = occ_mof_comp_sexo_ent,
                                  importance = TRUE,
                                  replace = TRUE, ntree=100, mtry=3)


bosque_todas_lfst <- randomForest(lifestage ~ largo_cuerpo +
                                    largo_cefalotorax +
                                    ancho_cefalotorax + peso,
                                  data = occ_mof_comp_lfst_ent,
                                  importance = TRUE,
                                  replace = TRUE, ntree=100, mtry=3)
```

Los modelos generados de regresión logística se hicieron con la función `glm()`, y se hicieron para una única variable predictora y para las cuatro.

```
logi_largo_sexo <- glm(sex ~ largo_cuerpo,family = binomial,
                       data = occ_mof_largo_sexo_ent)

logi_todas_sexo <- glm(sex ~ largo_cuerpo + ancho_cefalotorax +
                         largo_cefalotorax + peso,
                       family = binomial,
                       data = occ_mof_comp_sexo_ent)


logi_largo_lfst <- glm(lifestage ~ largo_cuerpo,family = binomial,
                       data = occ_mof_largo_lfst_ent)

logi_todas_lfst <- glm(lifestage ~ largo_cuerpo + ancho_cefalotorax +
                         largo_cefalotorax + peso,
                       family = binomial,
                       data = occ_mof_comp_lfst_ent)
```

A.4. Modelos regresión

Al inicio del capítulo sobre métodos de regresión se generaron algunas gráficas y se representaron las matrices de correlación de las variables de tamaño, ya que era relevante para justificar las posteriores decisiones. La decisión tomada fue separar la información de `mof` en dos conjuntos en función de si los registros se tomaron antes o después del 1 de enero de 2010. El conjunto anterior a esta fecha se nombró `mof_f1` y el posterior `mof_f2`, el código a continuación genera los dos conjuntos y muestra la gráfica de la matriz de correlación de cada uno.

```
mof_f1 <- mof %>% filter(fecha < "2010-01-01")

mof_f1 %>% select(largo_cuerpo, largo_cefalotorax,
                  ancho_cefalotorax, peso) %>%
  cor(use="pairwise.complete.obs") %>% corrplot(method="ellipse")


mof_f2 <- mof %>% filter(fecha > "2010-01-01")

mof_f2 %>% select(largo_cuerpo, largo_cefalotorax,
                  ancho_cefalotorax, peso) %>%
  cor(use="pairwise.complete.obs") %>% corrplot(method="ellipse")
```

El primer modelo generado era de regresión lineal, queriendo predecir el largo del cefalotorax en función del largo del cuerpo, y tomando el conjunto de datos sin dividir. Primero se filtraron los casos completos, luego se creó la división en muestra de entrenamiento (80 % de los casos) y muestra test (20 % de los casos). El error cometido se calculó mediante validación cruzada sobre la muestra de entrenamiento. La muestra test se utilizó para visualizar las predicccciones en comparación con los valores reales de la variable.

```
mof_larcf <- mof %>% filter(!is.na(largo_cefalotorax))

mof_larcf_split <- initial_split(mof_larcf, prop=0.8,
                                 strata=largo_cefalotorax)

reg_mof_larcf = lm(largo_cefalotorax ~ largo_cuerpo,
                   data=training(mof_larcf_split))

VC_mof_1_lcf <- cv.lm(largo_cefalotorax ~ largo_cuerpo, m=10,
                      data = training(mof_larcf_split),
                  seed = 1909, plotit = FALSE, printit = FALSE)
```

Este modelo tenía como finalidad recalcar la necesidad de separar los dos conjuntos, ya que las estimaciones obtenidas no eran correctas. Una vez se visualizó esto, se repitió el proceso con cada uno de los subconjuntos antes generados.

```
mof_larcf_f1 <- mof_f1 %>% filter(!is.na(largo_cefalotorax))

mof_larcf_f1_split <- initial_split(mof_larcf_f1, prop=0.8,
                                    strata=largo_cefalotorax)

reg_mof_larcf_f1 = lm(largo_cefalotorax ~ largo_cuerpo,
                      data=training(mof_larcf_f1_split))

VC_mof_larcf_f1 <- cv.lm(largo_cefalotorax ~ largo_cuerpo, m=10,
                         data = training(mof_larcf_f1_split),
                  seed = 1909, plotit = FALSE, printit = FALSE)


mof_larcf_f2 <- mof_f2 %>% filter(!is.na(largo_cefalotorax))

mof_larcf_f2_split <- initial_split(mof_larcf_f2, prop=0.8,
                                    strata=largo_cefalotorax)

reg_mof_larcf_f2 = lm(largo_cefalotorax ~ largo_cuerpo,
                      data=training(mof_larcf_f2_split))

VC_mof_larcf_f2 <- cv.lm(largo_cefalotorax ~ largo_cuerpo, m=10,
                         data = training(mof_larcf_f2_split),
                  seed = 1909, plotit = FALSE, printit = FALSE)
```

El error cometido por estos modelos era considerablemente menor que el del modelo sin división temporal de los datos. También las predicciones eran notablemente más acertadas que en el otro modelo.

Para la variable ancho del cefalotorax ocurría lo mismo, por tanto se generaron únicamente los modelos con los datos separados, sin hacer primero el modelo incorrecto.

```
mof_anccf_f1 <- mof_f1 %>% filter(!is.na(ancho_cefalotorax))

mof_anccf_f1_split <- initial_split(mof_anccf_f1, prop=0.8,
                                    strata=ancho_cefalotorax)

reg_mof_anccf_f1 = lm(ancho_cefalotorax ~ largo_cuerpo,
                      data=training(mof_anccf_f1_split))

VC_mof_anccf_f1 <- cv.lm(ancho_cefalotorax ~ largo_cuerpo, m=10,
                         data = training(mof_anccf_f1_split),
                  seed = 1909, plotit = FALSE, printit = FALSE)


mof_anccf_f2 <- mof_f2 %>% filter(!is.na(ancho_cefalotorax))

mof_anccf_f2_split <- initial_split(mof_anccf_f2, prop=0.8,
                                    strata=ancho_cefalotorax)

reg_mof_anccf_f2 = lm(ancho_cefalotorax ~ largo_cuerpo,
                      data=training(mof_anccf_f2_split))

VC_mof_anccf_f2 <- cv.lm(ancho_cefalotorax ~ largo_cuerpo, m=10,
                         data = training(mof_anccf_f2_split),
                  seed = 1909, plotit = FALSE, printit = FALSE)
```

Para la variable peso en la gráfica inicial no se veía que tuviera un comportamiento demasiado distinto entre antes y después de 2010, de modo que se creó primero el modelo usando el conjunto total.

```
mof_pes <- mof %>% filter(!is.na(peso))

mof_pes_split <- initial_split(mof_pes, prop=0.8, strata=peso)

reg_mof_peso = lm(peso ~ largo_cuerpo,
                    data=training(mof_pes_split))
VC_mof_1_peso <- cv.lm(peso ~ largo_cuerpo, m=10,
                       data = training(mof_pes_split),
                  seed = 1909, plotit = FALSE, printit = FALSE)
```

Este modelo era mejor que en el caso del largo del cefalotorax usando todos los datos, pero era peor que aquellos modelos creados con los datos divididos. Igualmente el ajuste no daba la impresión de ser malo por el mismo motivo que ocurría con los tamaños del cefalotorax, pero por confirmarlo se crearon los modelos haciendo la separación de los datos.

```
mof_peso_f1 <- mof_f1 %>% filter(!is.na(peso))

mof_peso_f1_split <- initial_split(mof_peso_f1, prop=0.8,
                                   strata=peso)

reg_mof_peso_f1 = lm(peso ~ largo_cuerpo,
                     data=training(mof_peso_f1_split))

VC_mof_peso_f1 <- cv.lm(peso ~ largo_cuerpo, m=10,
                        data = training(mof_peso_f1_split),
                  seed = 1909, plotit = FALSE, printit = FALSE)

mof_peso_f2 <- mof_f2 %>% filter(!is.na(peso))

mof_peso_f2_split <- initial_split(mof_peso_f2, prop=0.8,
                                   strata=peso)

reg_mof_peso_f2 = lm(peso ~ largo_cuerpo,
                     data=training(mof_peso_f2_split))

VC_mof_peso_f2 <- cv.lm(peso ~ largo_cuerpo, m=10,
                        data = training(mof_peso_f2_split),
                  seed = 1909, plotit = FALSE, printit = FALSE)
```

Estos modelos no disminuían el error ni mejoraban las predicciones de una forma notable, por otro lado, el modelo con la transformación logarítmica sí que dio mejores resultados. En este caso sí se incluye el código de la gráfica que compara los valores reales y las predicciones, ya que tiene algunos matices que no se daban en los casos anteriores.

```
reg_log_peso = lm(log(peso) ~ largo_cuerpo,
                  data=training(mof_pes_split))


VC_log_peso <- cv.lm(log(peso) ~ largo_cuerpo, m=10,
                        data = training(mof_pes_split),
                  seed = 1909, plotit = FALSE, printit = FALSE)

testing(mof_pes_split) %>%
  bind_cols(pred = exp(predict(reg_log_peso,
                               testing(mof_pes_split)))) %>%
  filter(pred <100) %>%
  ggplot(aes(peso, pred)) +
  geom_point(alpha=0.35, size=1.3) +
  theme_classic() + theme(aspect.ratio = 1) +
  geom_abline() +
  ggtitle("Predicción del peso (destransformado) frente a valor real",
          subtitle = "Filtrando los casos con predicciones < 100")
```

Otra forma de intentar remediar el error cometido en el modelo lineal fue generando un modelo de bosque aleatorio para predecir la variable peso en función de las otras tres variables de tamaño. Se necesita filtrar aquellos datos que no tengan ninguna variable tamaño perdida, aunque el largo del cuerpo no hay que comprobarlo porque no tiene valores perdidos. Se aplica validación cruzada para comprobar el mejor valor del parámetro mtry, en este caso el valor óptimo se da para 1, por tanto para cada nodo de los árboles se selecciona aleatoriamente uno de los tres predictores y se hace la mejor división en función de este. El bosque se crea mediante el paquete **caret**, y el resultado obtenido es mucho mejor que el que se consiguió con la regresión lineal.

```
mof_comp <- mof %>% filter(!is.na(largo_cefalotorax),
                           !is.na(ancho_cefalotorax),
                           !is.na(peso))

mof_comp_split <- initial_split(mof_comp, prop=0.8, strata=peso)

bosque_peso_tuning <-
  train(peso ~ ancho_cefalotorax + largo_cefalotorax + largo_cuerpo,
        data = training(mof_comp_split), method = 'rf',
        tuneGrid  = expand.grid(.mtry = 1:3),
        trControl = control <- trainControl(method='repeatedcv',
                                            number=10,
                                            search = 'grid'))
```

Con ese bosque se finaliza el apartado de regresión sobre las variables tamaño.

A continuación se estuvo trabajando sobre el conjunto EV, o más bien sobre el conjunto que se obtiene después de filtrar aquellas localizaciones con más de tres registros y hacer su ponderación. El motivo de exigir un mínimo de 4 observaciones a lo largo de todo el estudio es evitar que hubiera valores aislados muy dispares que produjeran valores extraños para la variable ponderada de manera errónea. Al realizar este filtrado y realizar la unión con el conjunto de la información sobre variables ambientales, el número de registros disponibles pasó de ser 391 a 295.

```
ev_pres_abs <- ev %>%
  mutate(Fecha=year(eventDate)+(month(eventDate)-1)/12,
         IDL=localityID) %>%
  group_by(IDL, Fecha) %>%
  summarise(tot_capt = sum(sampleSizeValue,na.rm=TRUE)) %>%
  ungroup() %>% group_by(IDL) %>%
  mutate(med = mean(tot_capt, na.rm=TRUE),
         capt = tot_capt/med,
         cont= n()) %>% filter(cont > 3) %>%
  select(-tot_capt, -med, -cont) %>% ungroup() %>%
  inner_join(pres_abs, by=c("IDL", "Fecha"))
```

De estos 295 registros, solo 69 eran registros completos. Si a eso se le añade que los registros de cada tipo de hábitat deben estudiarse por separado, el número de datos para

intentar generar un modelo era increíblemente bajo. Por ello se aprovechó el pre-procesado que se aplicaba a los datos antes de aplicar ningún modelo (normalización de variables predictoras numéricas y eliminar aquellas con excesiva correlación) para también hacer una imputación mediante `step_impute_knn()` y así completar los valores perdidos de las variables predictoras. Esto se aplicó por separado en cada uno de los hábitats, para así tener ya listos los datos para modelizar cada uno.

```
lagunas_imputado <- ev_pres_abs %>%
  filter(Habitat == "Lagunas temporales") %>%
  select(-Habitat, -Salinidad, -Fecha, -IDL)%>%
  recipe(capt ~ .) %>%
  step_corr(all_numeric(), threshold = 0.8, -Mes, -Ano) %>%
  step_normalize(all_numeric_predictors(), -Mes, -Ano) %>%
  step_impute_knn(all_predictors(),
                  impute_with = imp_vars(all_numeric_predictors(),
                                         -capt)) %>%
  prep() %>%   bake(new_data = NULL)

marismas_imputado <- ev_pres_abs %>%
  filter(Habitat == "Marisma") %>%
  select(-Habitat, -Salinidad, -Fecha, -IDL) %>%
  recipe(capt ~ .) %>%
  step_corr(all_numeric(), threshold = 0.8, -Mes, -Ano) %>%
  step_normalize(all_numeric_predictors(), -Mes,-Ano) %>%
  step_impute_knn(all_predictors(),
                  impute_with = imp_vars(all_numeric_predictors(),
                                         -capt)) %>%
  prep() %>% bake(new_data = NULL)
```

Por tener referencia con respecto a los datos originales (sin imputar) también se crearon modelos de bosques aleatorios con estos datos. El pre-procesado que se aplicó fue el mismo, pero quitando el último paso.

```
lagunas_sin_imp <- ev_pres_abs %>% drop_na() %>%
  filter(Habitat == "Lagunas temporales") %>%
  select(-Habitat, -Salinidad, -Fecha, -IDL)%>%
  recipe(capt ~ .) %>%
  step_corr(all_numeric(), threshold = 0.8, -Mes, -Ano) %>%
  step_normalize(all_numeric_predictors(), -Mes, -Ano) %>%
  prep() %>%   bake(new_data = NULL)

marismas_sin_imp <- ev_pres_abs %>% drop_na() %>%
  filter(Habitat == "Marisma") %>%
  select(-Habitat, -Salinidad, -Fecha, -IDL)%>%
  recipe(capt ~ .) %>%
  step_corr(all_numeric(), threshold = 0.8, -Mes, -Ano) %>%
  step_normalize(all_numeric_predictors(), -Mes, -Ano) %>%
  prep() %>%   bake(new_data = NULL)
```

Para los datos imputados se aplicaron los métodos de bosques aleatorios y de los k vecinos más cercanos, mientras que para los datos sin imputar solo se aplicó bosques aleatorios. En este caso los modelos no daban resultados suficientemente buenos, por lo que se crearon con **tidymodels** ya que permite aplicar tuning a más parámetros que **caret**. En el caso de los datos imputados la división en muestra de entrenamiento y muestra test fue la misma para ambos modelos. Dado el reducido tamaño de los conjuntos, el porcentaje destinado a la muestra de entrenamiento fue un 80 %.

```
lag_imp_split <- lagunas_imputado %>%
  initial_split(0.8, strata=capt)

lag_imp_ent <- lag_imp_split %>% training()
lag_imp_test <- lag_imp_split %>% testing()


mar_imp_split <- marismas_imputado %>%
  initial_split(0.8, strata=capt)

mar_imp_ent <- mar_imp_split %>% training()
mar_imp_test <- mar_imp_split %>% testing()



lag_sin_imp_split <- lagunas_sin_imp %>%
  initial_split(0.8, strata=capt)

lag_sin_imp_ent <- lag_sin_imp_split %>% training()
lag_sin_imp_test <- lag_sin_imp_split %>% testing()


mar_sin_imp_split <- marismas_sin_imp %>%
  initial_split(0.8, strata=capt)

mar_sin_imp_ent <- mar_sin_imp_split %>% training()
mar_sin_imp_test <- mar_sin_imp_split %>% testing()
```

En el caso de las lagunas en datos imputados, el tuning de mtry dio como mejor valor 2, por lo que de las trece variables en cada nodo se elegían aleatoriametne 2 para buscar la mejor división. En el caso de las marismas fue 6 el mejor valor.

```
rf_lag_capt_tune_model <-
  rand_forest(mtry = tune(),
              trees = tune(),
              min_n = tune()) %>%
  set_engine("randomForest") %>%
  set_mode("regression")

lag_capt_rf_tune_wkfl <- workflow() %>%
```

```
  add_model(rf_lag_capt_tune_model)  %>%
  add_formula(capt ~ .)

lag_capt_folds <- vfold_cv(lag_imp_ent, v=10, strata = capt)

rf_capt_grid <- grid_random(mtry(c(1,ncol(lag_imp_ent)-1)),
                            trees(),min_n(),size=10)

lag_capt_rf_tuning <- lag_capt_rf_tune_wkfl %>%
  tune_grid(resamples= lag_capt_folds,
            grid=rf_capt_grid,
            metrics = metric_set(rmse))

lag_capt_rf_best_model <- lag_capt_rf_tuning %>%
  select_best(metric = "rmse")

lag_capt_rf_tune_wkfl_final <- lag_capt_rf_tune_wkfl %>%
  finalize_workflow(lag_capt_rf_best_model)
```

En este caso el error cometido en las estimaciones en porcentaje con respecto al valor medio de la variable (todo evaluado en la muestra test) fue de un 20 %. Esto es un error considerable, aunque teniendo en cuenta que la variable toma valores muy reducidos (alrededor de 1) esto supone 0.2 en valor absoluto del error.

```
rf_mar_capt_tune_model <-
  rand_forest(mtry = tune(),
              trees = tune(),
              min_n = tune()) %>%
  set_engine("randomForest") %>%
  set_mode("regression")

mar_capt_rf_tune_wkfl <- workflow() %>%
  add_model(rf_mar_capt_tune_model)  %>%
  add_formula(capt ~ .)

mar_capt_folds <- vfold_cv(mar_imp_ent, v=10, strata = capt)

mar_capt_rf_tuning <- mar_capt_rf_tune_wkfl %>%
  tune_grid(resamples= mar_capt_folds,
            grid=rf_capt_grid,
            metrics = metric_set(rmse))

mar_capt_rf_best_model <- mar_capt_rf_tuning %>%
  select_best(metric = "rmse")

mar_capt_rf_tune_wkfl_final <- mar_capt_rf_tune_wkfl %>%
  finalize_workflow(mar_capt_rf_best_model)
```

En el caso de las marismas el error fue similar.

Para comparar se aplicó el mismo modelo a los casos sin imputar. En el caso de las lagunas en vez de tener 114 registros se tienen solo 24 para la muestra de entrenamiento, para las marismas se pasa de tener 119 a 30 registros completos. En las muestras test los tamaños son 8 y 7 respectivamente. De nuevo se aplicó tuning para obtener los mejores parámetros para estos conjuntos, en el caso de las lagunas el mtry óptimo fue 2 de nuevo, y para las marismas 7.

```
lag_sin_capt_folds <- vfold_cv(lag_sin_imp_ent, v=10, strata = capt)

lag_sin_capt_rf_tuning <- lag_capt_rf_tune_wkfl %>%
  tune_grid(resamples= lag_sin_capt_folds,
            grid=rf_capt_grid,
            metrics = metric_set(rmse))

lag_sin_capt_rf_best_model <- lag_sin_capt_rf_tuning %>%
  select_best(metric = "rmse")

lag_sin_capt_rf_tune_wkfl_final <- lag_capt_rf_tune_wkfl %>%
  finalize_workflow(lag_sin_capt_rf_best_model)
```

En las lagunas el error en porcentaje en la muestra test fue menor que en el modelo anterior, pero dado que solo había 7 registros en la muestra test y 24 en la muestra de entrenamiento no se consideró que fuera un valor fiable. Al ejecutar el código anterior saltaban numerosos warnings, advirtiendo del reducido tamaño de muestra disponible.

```
mar_sin_capt_folds <- vfold_cv(mar_sin_imp_ent, v=10, strata = capt)

mar_sin_capt_rf_tuning <- mar_capt_rf_tune_wkfl %>%
  tune_grid(resamples= mar_sin_capt_folds,
            grid=rf_capt_grid,
            metrics = metric_set(rmse))

mar_sin_capt_rf_best_model <- mar_sin_capt_rf_tuning %>%
  select_best(metric = "rmse")

mar_sin_capt_rf_tune_wkfl_final <- mar_capt_rf_tune_wkfl %>%
  finalize_workflow(mar_sin_capt_rf_best_model)
```

En las marismas ocurrió lo mismo, el error en este modelo fue menor que en el modelo con los datos imputados. Igualmente al provenir de una muestra de entrenamiento de tamaño 30 y muestra test de tamaño 8 no se puede considerar un modelo fiable.

Para los modelos de k vecinos más cercanos solo se utilizó **tidymodels**, se realizó tuning sobre el valor de k, sobre los pesos de los individuos y sobre la función utilizada para medir distancias. Se aplicó únicamente a los conjuntos con datos imputados para las variables predictoras.

```
knn_tune_model_reg <- nearest_neighbor(neighbors = tune(),
                                       weight_func = tune(),
                                       dist_power = tune()) %>%
  set_engine("kknn") %>%
  set_mode("regression")

knn_tune_wkfl <- workflow() %>%
  add_model(knn_tune_model_reg)  %>%
  add_formula(capt ~ .)

knn_grid <- grid_random(parameters(knn_tune_model_reg),
                        size = 10)


lag_folds <- vfold_cv(training(lag_imp_split), v = 10,
                      strata = capt)

lag_knn_tuning <- knn_tune_wkfl %>%
  tune_grid(resamples = lag_folds,
            grid = knn_grid)

lag_knn_best_model <- lag_knn_tuning %>%
  select_best(metric = "rmse")

lag_knn_tune_wkfl_final <- knn_tune_wkfl %>%
  finalize_workflow(lag_knn_best_model)


mar_folds <- vfold_cv(training(mar_imp_split), v = 10,
                      strata = capt)

mar_knn_tuning <- knn_tune_wkfl %>%
  tune_grid(resamples = mar_folds,
            grid = knn_grid)

mar_knn_best_model <- mar_knn_tuning %>%
  select_best(metric = "rmse")

mar_knn_tune_wkfl_final <- knn_tune_wkfl %>%
  finalize_workflow(mar_knn_best_model)
```

Para las lagunas el valor óptimo de k era 12 y para las marismas 13. En ambos casos el error cometido fue similar al de sus bosques aleatorios equivalentes, por lo que se concluyó que era el mejor resultado que podía obtenerse con los datos disponibles.

Apéndice B

Apéndice: Código descartado

A lo largo de la realización de este trabajo, se plantearon múltiples ideas que no obtuvieron buenos resultados. Algunas de ellas quedan reflejadas en este apéndice, para justificar que no fueran llevadas a cabo en la parte principal del documento.

B.1. Regresión para capturas en trampas

B.1.1. Datos sin imputar, por hábitat

Hay 24 registros en la muestra test y 11 predictores numéricos. Si se usan todas en un único modelo, el resultado obtenido sería:

```
summary(lm(capt ~ ., data = lag_sin_imp_ent %>% select_if(is.numeric)))
```

```
##
## Call:
## lm(formula = capt ~ ., data = lag_sin_imp_ent %>% select_if(is.numeric))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.32614 -0.03342  0.01071  0.02713  0.22858
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -1.132e+02  1.183e+02  -0.957    0.357
## Anyo           5.681e-02  5.883e-02   0.966    0.353
## Mes           -1.654e-03  1.376e-02  -0.120    0.906
## Prof_max      -1.744e-02  3.366e-02  -0.518    0.614
## Temperatura    4.137e-02  3.794e-02   1.090    0.297
## Conductividad -6.410e-02  5.215e-02  -1.229    0.243
## O2_porc       -1.177e-02  5.084e-02  -0.232    0.821
## pH            -1.413e-02  5.303e-02  -0.267    0.794
## Turbidez      -1.439e-03  6.148e-02  -0.023    0.982
```

Tabla B.1: Correlación capturas lagunas respecto al clima sin imputar

Conductividad	Temperatura	Nitrato	pH	Mes	O2_porc
0.262	0.221	0.176	0.16	0.137	0.117

Tabla B.2: Regresiones capturas lagunas, clima sin imputar

Variables	R^2 ajustado	p-valor
Conductividad + Temperatura + Nitrato + pH + Mes + O2_porc	-0.01725	0.4955
Conductividad + Temperatura + Nitrato + pH + Mes	0.03892	0.35454
Conductividad + Temperatura + Nitrato + pH	0.07191	0.25777
Conductividad + Temperatura + Nitrato	0.09353	0.1813
Conductividad + Temperatura	0.03879	0.25396

```
## Clorofila     -4.855e-03  5.702e-02  -0.085    0.934
## Amonio        -1.645e-02  8.469e-02  -0.194    0.849
## Nitrato        4.658e-02  1.264e-01   0.368    0.719
##
## Residual standard error: 0.1373 on 12 degrees of freedom
## Multiple R-squared:  0.3101, Adjusted R-squared:  -0.3224
## F-statistic: 0.4902 on 11 and 12 DF,  p-value: 0.876
```

El p-valor es muy alto, y el R^2 ajustado llega a ser negativo. Hay demasiadas variables predictoras, e irlas quitando una a una sería excesivamente lento. Una opción es partir del modelo con aquellas variables que tengan mayor correlación con la variable objetivo, y luego irlo refinando.

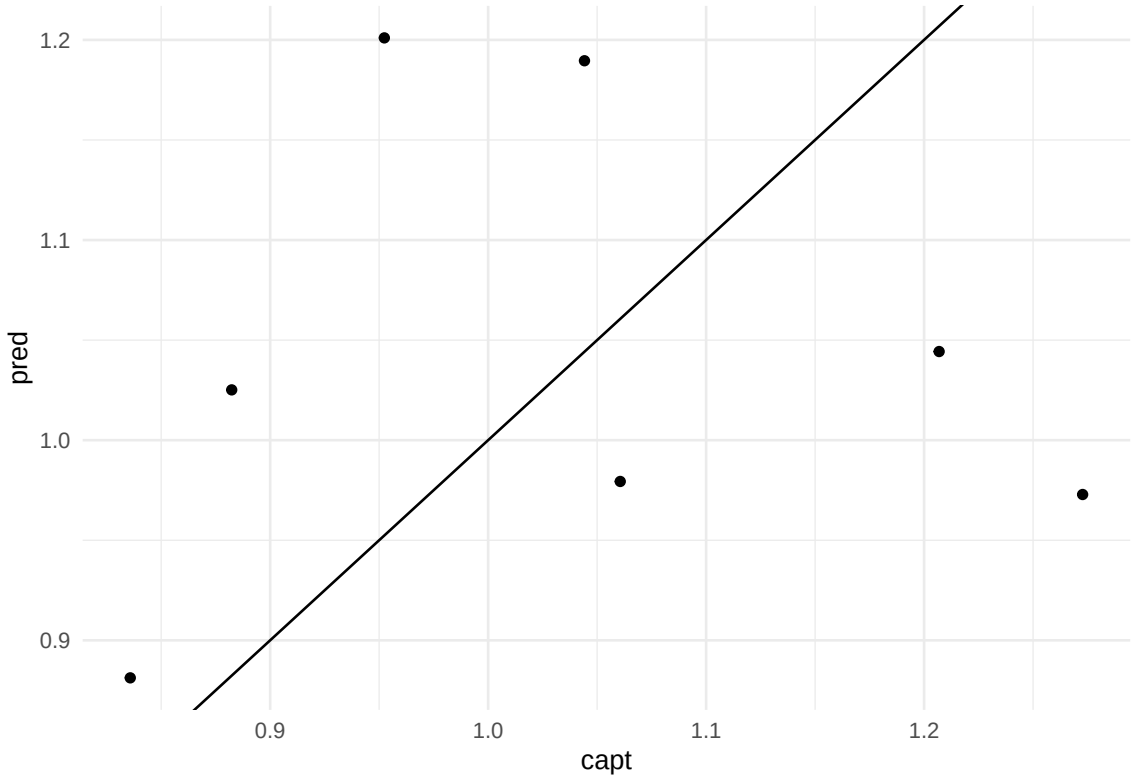
```
capt_lag_sin_cor <- lag_sin_imp_ent %>% select_if(is.numeric) %>%
  cor(use = "pairwise.complete.obs")

capt_lag_sin_cor2 <- capt_lag_sin_cor["capt",] %>% abs() %>%
  sort(decreasing = TRUE)

capt_lag_sin_cor2[2:7] %>% t() %>%
  kable(digits=3,
        caption=
          "Correlación capturas lagunas respecto al clima sin imputar")
```

Partiendo de las variables con mayor correlación, se va reduciendo el número de variables del modelo y se muestra en la Tabla B.2 los resultados.

En todos estos modelos se acepta que los coeficientes sean 0 para el modelo total, y el coeficiente de determinación apenas llega a 0.1. Se calcula el porcentaje de error en el modelo con mejor coeficiente de determinación:

```
err_reg_ev_lag_sin <- lag_sin_imp_test %>% select(capt) %>%
  cbind(pred = predict(lm(capt ~ Conductividad + Temperatura + Nitrato,
                          data = lag_sin_imp_ent),
                       lag_sin_imp_test)) %>%
```

```
  rmse(truth = capt, estimate = pred) %>%
  pluck(3)

lag_sin_imp_test %>% select(capt) %>%
  cbind(pred = predict(lm(capt ~ Conductividad + Temperatura + Nitrato,
                          data = lag_sin_imp_ent),
                       lag_sin_imp_test)) %>%
  ggplot(aes(x=capt, y = pred)) + geom_point() +
  geom_abline() + theme_minimal()
```




```
## [1] "El error cometido es del 17.43 %"
```

Esta misma estructura se seguirá en las marismas, también en ambos hábitats para los datos del clima imputados.

En las marismas el número de registros para la muestra entrenamiento es 30.

```
summary(lm(capt ~ ., data = mar_sin_imp_ent %>% select_if(is.numeric)))
```

```
##
## Call:
## lm(formula = capt ~ ., data = mar_sin_imp_ent %>% select_if(is.numeric))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```

Tabla B.3: Correlación capturas marismas respecto al clima sin imputar

Amonio	Prof_max	O2_porc	Mes	Anyo	pH
0.449	0.289	0.223	0.198	0.169	0.149

```
## -0.19855 -0.08196  0.01032  0.07676  0.19414
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -62.359527  69.759444  -0.894  0.38316
## Anyo            0.031500   0.034693   0.908  0.37590
## Mes             0.015526   0.011053   1.405  0.17712
## Prof_max       -0.063786   0.036717  -1.737  0.09943 .
## Temperatura     0.026075   0.041001   0.636  0.53280
## Conductividad  -0.007796   0.028223  -0.276  0.78551
## O2_porc         0.086723   0.038843   2.233  0.03851 *
## pH             -0.007216   0.043264  -0.167  0.86939
## Turbidez        0.009176   0.040316   0.228  0.82252
## Clorofila      -0.030165   0.039769  -0.758  0.45797
## Amonio         -0.103100   0.029998  -3.437  0.00294 **
## Nitrato        -0.052303   0.034547  -1.514  0.14740
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1505 on 18 degrees of freedom
## Multiple R-squared:  0.5764, Adjusted R-squared:  0.3175
## F-statistic: 2.227 on 11 and 18 DF,  p-value: 0.06358
```

El modelo total tiene mejor ajuste que en las lagunas, y aunque a un 5 % de confianza siga sin rechazarse la nulidad de coeficientes, a un 10 % sí se rechazaría.

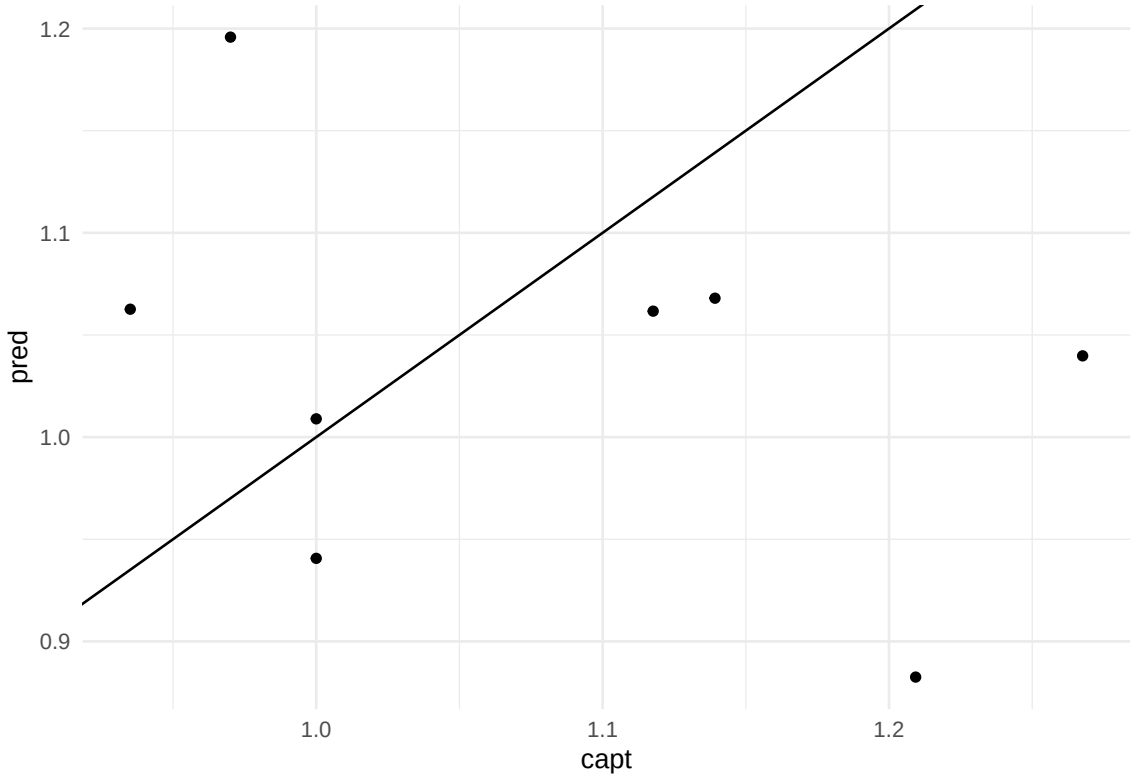
```
capt_mar_sin_cor <- mar_sin_imp_ent %>% select_if(is.numeric) %>%
  cor(use = "pairwise.complete.obs")

capt_mar_sin_cor2 <- capt_mar_sin_cor["capt",] %>% abs() %>%
  sort(decreasing = TRUE)

capt_mar_sin_cor2[2:7] %>% t() %>%
  kable(digits=3,
        caption=
          "Correlación capturas marismas respecto al clima sin imputar")
```

En este caso al menos se rechaza que los coeficientes del modelo sean nulos, aunque el coeficiente de determinación dista mucho de ser suficientemente grande. Se mide el RMSE de la muestra test en el modelo con mejor coeficiente de determinación.

Tabla B.4: Regresiones capturas marismas, clima sin imputar

Variables	R^2 ajustado	p-valor
Amonio + Prof_max + O2_porc + Mes + Anyo + pH	0.33997	0.01334
Amonio + Prof_max + O2_porc + Mes + Anyo	0.35558	0.00696
Amonio + Prof_max + O2_porc + Mes	0.35904	0.00397
Amonio + Prof_max + O2_porc	0.36206	0.00201
Amonio + Prof_max	0.21966	0.01339

```
err_reg_ev_mar_sin <- mar_sin_imp_test %>% select(capt) %>%
  cbind(pred = predict(lm(capt ~ Amonio + Prof_max + O2_porc,
                          data = mar_sin_imp_ent),
                       mar_sin_imp_test)) %>%
  rmse(truth = capt, estimate = pred) %>%
  pluck(3)

mar_sin_imp_test %>% select(capt) %>%
  cbind(pred = predict(lm(capt ~ Amonio + Prof_max + O2_porc,
                          data = mar_sin_imp_ent),
                       mar_sin_imp_test)) %>%
  ggplot(aes(x=capt, y = pred)) + geom_point() +
  geom_abline() + theme_minimal()
```




```
## [1] "El error cometido es del 15.96 %"
```

B.1.2. Datos imputados, por hábitat

```
summary(lm(capt ~ ., data = lag_imp_ent %>% select_if(is.numeric)))
```

```
##
## Call:
## lm(formula = capt ~ ., data = lag_imp_ent %>% select_if(is.numeric))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.81339 -0.08796  0.00142  0.09074  1.24595
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -4.266531  26.264679  -0.162   0.8713
## Anyo           0.002642   0.013070   0.202   0.8402
## Mes           -0.007997   0.008416  -0.950   0.3443
## Prof_max       0.012443   0.027814   0.447   0.6556
## Temperatura    0.053084   0.035598   1.491   0.1390
## Conductividad  0.007738   0.026787   0.289   0.7733
## O2_abs         0.028750   0.031360   0.917   0.3614
## pH            -0.075226   0.027179  -2.768   0.0067 **
## Turbidez      -0.069947   0.046932  -1.490   0.1392
## Clorofila      0.029030   0.026653   1.089   0.2787
## Amonio         0.053832   0.034474   1.561   0.1215
## Nitrato        0.004219   0.031265   0.135   0.8929
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2721 on 102 degrees of freedom
## Multiple R-squared:  0.1162, Adjusted R-squared:  0.02085
## F-statistic: 1.219 on 11 and 102 DF,  p-value: 0.2843
```

```
capt_lag_imp_cor <- lag_imp_ent %>% select_if(is.numeric) %>%
  cor(use = "pairwise.complete.obs")

capt_lag_imp_cor2 <- capt_lag_imp_cor["capt",] %>% abs() %>%
  sort(decreasing = TRUE)

capt_lag_imp_cor2[2:7] %>% t() %>%
  kable(digits=3,
        caption=
          "Correlación capturas lagunas respecto al clima imputado")
```

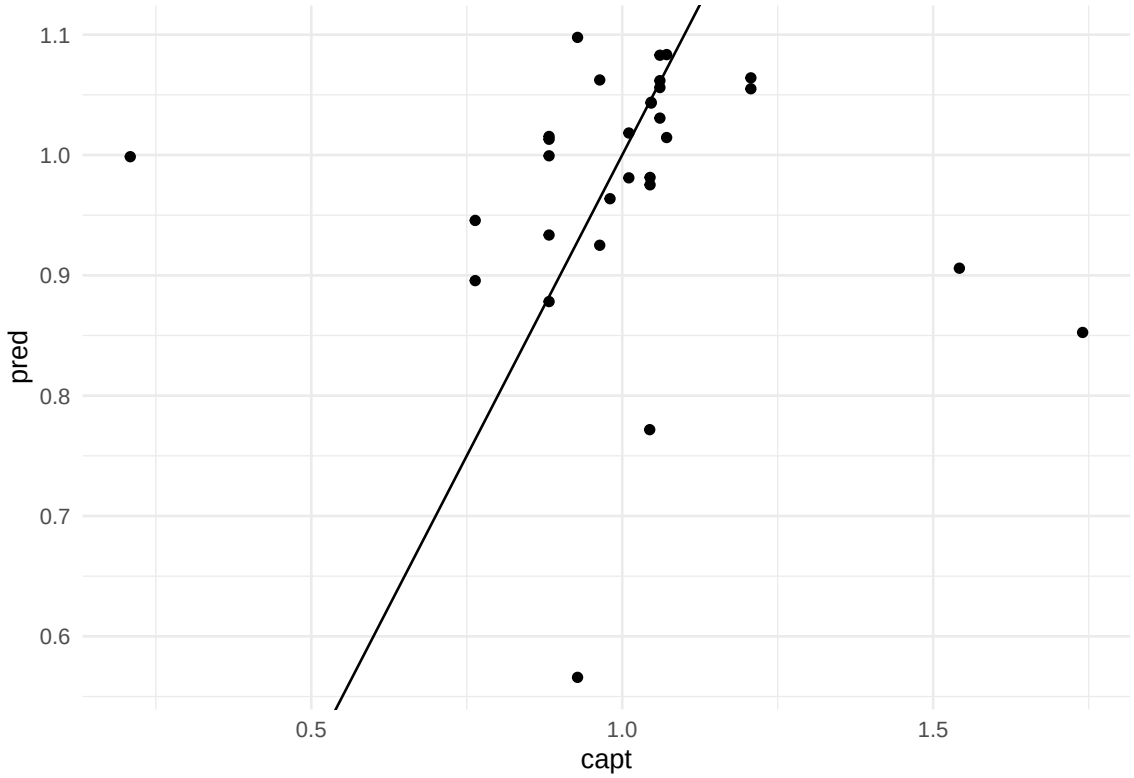
Tabla B.5: Correlación capturas lagunas respecto al clima imputado

pH	Turbidez	Mes	Temperatura	O2_abs	Clorofila
0.205	0.113	0.091	0.078	0.068	0.058

Tabla B.6: Regresiones capturas lagunas, clima imputado

Variables	R^2 ajustado	p-valor
pH + Turbidez + Mes + Temperatura + O2_abs + Clorofila	0.04121	0.10399
pH + Turbidez + Mes + Temperatura + O2_abs	0.04415	0.07817
pH + Turbidez + Mes + Temperatura	0.03725	0.0866
pH + Turbidez + Mes	0.03692	0.06789
pH + Turbidez	0.0416	0.03511

```
err_reg_ev_lag_imp <- lag_imp_test %>% select(capt) %>%
  cbind(pred = predict(lm(capt ~ pH + Turbidez,
                          data = lag_imp_ent),
                       lag_imp_test)) %>%
  rmse(truth = capt, estimate = pred) %>%
  pluck(3)

lag_imp_test %>% select(capt) %>%
  cbind(pred = predict(lm(capt ~ pH + Turbidez,
                          data = lag_imp_ent),
                       lag_imp_test)) %>%
  ggplot(aes(x=capt, y = pred)) + geom_point() +
  geom_abline() + theme_minimal()
```



```
## [1] "El error cometido es del 27 %"
```

```
summary(lm(capt ~ ., data = mar_imp_ent %>% select_if(is.numeric)))
```

```
##
## Call:
## lm(formula = capt ~ ., data = mar_imp_ent %>% select_if(is.numeric))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.78389 -0.09273  0.01632  0.14610  0.86428
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -54.202606  23.994444  -2.259   0.0259 *
## Anyo            0.027480   0.011939   2.302   0.0233 *
## Mes            -0.003914   0.008006  -0.489   0.6259
## Prof_max       -0.040595   0.031627  -1.284   0.2021
## Temperatura     0.048820   0.028982   1.684   0.0950 .
## Conductividad  -0.011699   0.024619  -0.475   0.6356
## O2_porc        -0.008273   0.026213  -0.316   0.7529
## pH              0.032059   0.029882   1.073   0.2857
## Turbidez       -0.055311   0.032258  -1.715   0.0893 .
## Clorofila       0.047156   0.030944   1.524   0.1305
## Amonio         -0.021830   0.027188  -0.803   0.4238
## Nitrato         0.023537   0.030246   0.778   0.4382
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2628 on 107 degrees of freedom
## Multiple R-squared:  0.1353, Adjusted R-squared:  0.04641
## F-statistic: 1.522 on 11 and 107 DF,  p-value: 0.134
```

```
capt_mar_imp_cor <- mar_imp_ent %>% select_if(is.numeric) %>%
  cor(use = "pairwise.complete.obs")

capt_mar_imp_cor2 <- capt_mar_imp_cor["capt",] %>% abs() %>%
  sort(decreasing = TRUE)

capt_mar_imp_cor2[2:7] %>% t() %>%
  kable(digits=3,
        caption=
          "Correlación capturas marismas respecto al clima imputado")
```

```
err_reg_ev_mar_imp <- mar_imp_test %>% select(capt) %>%
  cbind(pred = predict(lm(capt ~ Turbidez + Anyo + Nitrato,
```

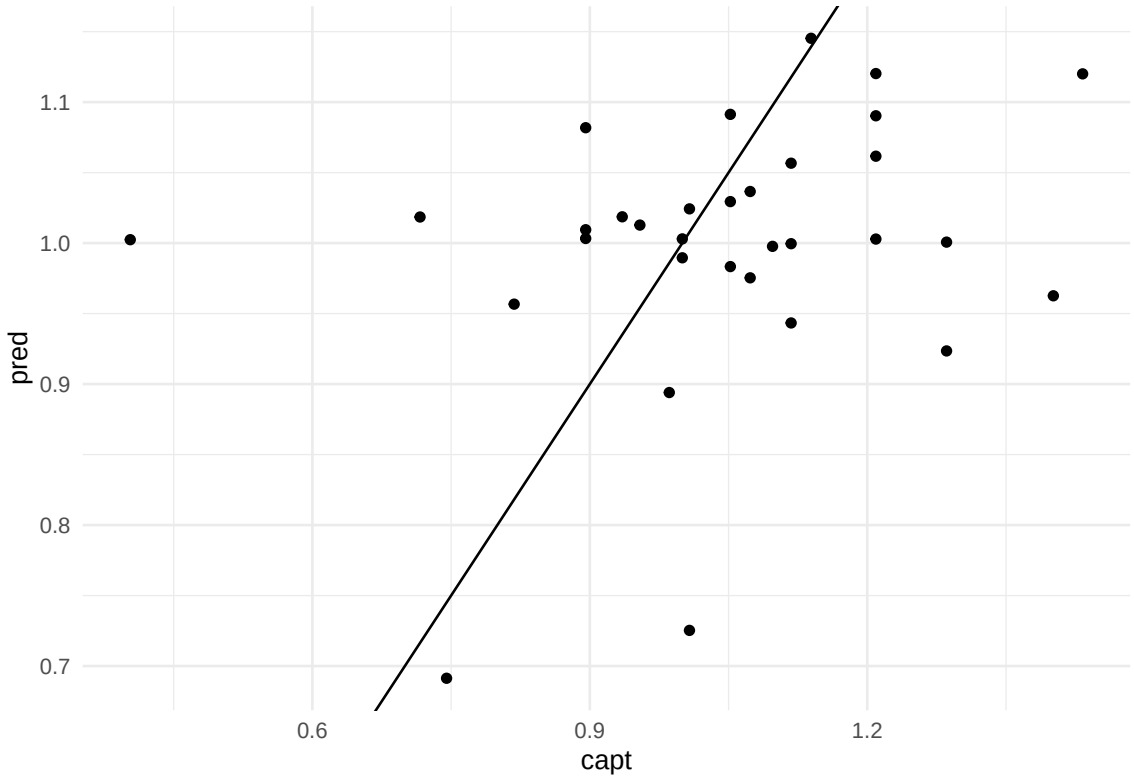
Tabla B.7: Correlación capturas marismas respecto al clima imputado

Turbidez	Anyo	Nitrato	Clorofila	pH	Mes
0.189	0.185	0.124	0.099	0.079	0.073

Tabla B.8: Regresiones capturas marismas, clima imputado

Variables	R^2 ajustado	p-valor
Turbidez + Anyo + Nitrato + Clorofila + pH + Mes	0.04323	0.08886
Turbidez + Anyo + Nitrato + Clorofila + pH	0.05052	0.05357
Turbidez + Anyo + Nitrato + Clorofila	0.05224	0.03821
Turbidez + Anyo + Nitrato	0.05214	0.02725
Turbidez + Anyo	0.04322	0.0286

```
                          data = mar_imp_ent),
                        mar_imp_test)) %>%
  rmse(truth = capt, estimate = pred) %>%
  pluck(3)

mar_imp_test %>% select(capt) %>%
  cbind(pred = predict(lm(capt ~ Turbidez + Anyo + Nitrato,
                          data = mar_imp_ent),
                        mar_imp_test)) %>%
  ggplot(aes(x=capt, y = pred)) + geom_point() +
  geom_abline() + theme_minimal()
```




```
## [1] "El error cometido es del 19.28 %"
```

B.2. Proporción de machos y hembras

En el apartado de análisis gráfico se representó de varias formas la proporción entre individuos maduros e inmaduros, esto era de interés para estudiar cómo evolucionaba la población del cangrejo rojo americano a lo largo del año. Otro de los factores de interés sobre la evolución de la población era la proporción entre machos y hembras, sin embargo la proporción solía estar alrededor del 50 % por lo que es una variable de poco interés.

Para poder estudiar esta variable, primero se unen los conjuntos occ y pres_abs, aunque de pres_abs solo se necesitan los pares IDL y Hábitat. Tras aplicar esta unión se filtran aquellas que pertenecen al hábitat buscado, y de esos se eliminan aquellos individuos de los cuales no se registró el sexo.

A partir de ese subconjunto de cangrejos se agrupan aquellos capturados en el mismo mes, se calcula el número de machos y hembras capturados en cada mes, se reordena el conjunto para tener una fila por mes y una columna para cada sexo, además se suman ambos para obtener el total. La proporción es de Machos entre el total, se mide en porcentaje. Además, después de esto se filtran aquellos días en los que se tuvieron 10 o menos cangrejos, ya que pueden dar lugar a valores muy extremos de la proporción que no son realmente significativos.

Después de tener calculada la proporción por cada mes, se vuelve a combinar con el conjunto pres_abs, se hace en base a la Fecha.

```
occ_prop_lag <- inner_join(occ, pres_abs %>%
                                 select(IDL, Habitat) %>%
                               unique(), by=c("IDL")) %>%
  filter(Habitat == "Lagunas temporales") %>%
  filter(!is.na(sex)) %>%
  mutate(Fecha = year(fecha) + (month(fecha)-1)/12) %>%
  group_by(Fecha, sex, IDL) %>%
  summarise(total = n()) %>%
  pivot_wider(names_from="sex",values_from = "total") %>%
  mutate(
    total = case_when(is.na(Female) ~ Male,
                      is.na(Male) ~ Female,
                      TRUE ~ Male + Female),
          proporcion = case_when(is.na(Male) ~ 100-Female/total*100,
                                 TRUE ~ Male/total*100)) %>%
  filter(total > 10)

occ_clima_lag <-
  occ_prop_lag %>% select(Fecha, IDL, proporcion) %>%
  inner_join(pres_abs,  by=c("Fecha", "IDL")) %>%
  filter(Habitat == "Lagunas temporales") %>% ungroup() %>%
  select(-IDL, -Habitat) %>%
  unique()

occ_lag_sin_imp <- occ_clima_lag %>% drop_na() %>%
  select(-Salinidad, -Fecha) %>%
```

```
  recipe(proporcion ~ .) %>%
  step_corr(all_numeric(),threshold = 0.8) %>%
  step_normalize(all_numeric_predictors()) %>%
  prep() %>% bake(new_data = NULL)

occ_lag_sin_imp_split <- occ_lag_sin_imp %>%
  initial_split(0.8)
occ_lag_sin_imp_ent <- occ_lag_sin_imp_split %>% training()
occ_lag_sin_imp_test <- occ_lag_sin_imp_split %>% testing()

occ_lag_imp <- occ_clima_lag %>%
  select(-Salinidad, -Fecha) %>%
  recipe(proporcion ~ .) %>%
  step_corr(all_numeric(),threshold = 0.9) %>%
  step_normalize(all_numeric_predictors()) %>%
  step_impute_knn(all_predictors(),
                  impute_with = imp_vars(all_numeric_predictors(),
                                         -proporcion))%>%
  prep() %>% bake(new_data = NULL)

occ_lag_imp_split <- occ_lag_imp %>%
  initial_split(0.8, strata = proporcion)
occ_lag_imp_ent <- occ_lag_imp_split %>% training()
occ_lag_imp_test <- occ_lag_imp_split %>% testing()
```

```
occ_prop_mar <- inner_join(occ, pres_abs %>%
                             select(IDL, Habitat) %>%
                           unique(), by=c("IDL")) %>%
  filter(Habitat == "Marisma") %>%
  filter(!is.na(sex)) %>%
  mutate(Fecha = year(fecha) + (month(fecha)-1)/12) %>%
  group_by(Fecha, IDL, sex) %>%
  summarise(total = n()) %>%
  pivot_wider(names_from="sex",values_from = "total") %>%
  mutate(
    total = case_when(is.na(Female) ~ Male,
                      is.na(Male) ~ Female,
                      TRUE ~ Male + Female),
         proporcion = case_when(is.na(Male) ~ 100-Female/total*100,
                                TRUE ~ Male/total*100)) %>%
  filter(total > 10)


occ_clima_mar <-
  occ_prop_mar %>% select(Fecha, IDL, proporcion) %>%
  inner_join(pres_abs,  by=c("Fecha", "IDL")) %>%
  filter(Habitat == "Marisma")  %>% ungroup() %>%
```

```
  select(-IDL, -Habitat) %>%
  unique()

occ_mar_sin_imp <- occ_clima_mar %>% drop_na() %>%
  select(-Salinidad, -Fecha) %>%
  recipe(proporcion ~ .) %>%
  step_corr(all_numeric(),threshold = 0.8) %>%
  step_normalize(all_numeric_predictors()) %>%
  prep() %>% bake(new_data = NULL)

occ_mar_sin_imp_split <- occ_mar_sin_imp %>%
  initial_split(0.8)
occ_mar_sin_imp_ent <- occ_mar_sin_imp_split %>% training()
occ_mar_sin_imp_test <- occ_mar_sin_imp_split %>% testing()


occ_mar_imp <- occ_clima_mar %>% ungroup() %>%
  select(-Salinidad, -Fecha) %>%
  recipe(proporcion ~ .) %>%
  step_corr(all_numeric(),threshold = 0.8) %>%
  step_normalize(all_numeric_predictors()) %>%
  step_impute_knn(all_predictors(),
                  impute_with = imp_vars(all_numeric_predictors(),
                                         -proporcion))%>%
  prep() %>% bake(new_data = NULL)


occ_mar_imp_split <- occ_mar_imp %>%
  initial_split(0.8, strata = proporcion)
occ_mar_imp_ent <- occ_mar_imp_split %>% training()
occ_mar_imp_test <- occ_mar_imp_split %>% testing()
```

Después de hacer estas transformaciones, para las lagunas hay 79 registros de los cuales 12 están completos. Para las marismas hay 78 registros, de los cuales 25 están completos.

Nota: en este apartado la variable Año en vez de Anyo es Ano, ya que este código es previo al renombramiento que se hizo en el documento general.

B.2.1. Datos sin imputar, por hábitat

La estructura de este apartado es igual que la que se siguió con la variable capturas en la sección anterior.

Lo primero es que el modelo total no puede aplicarse a la muestra test, porque no tiene suficientes observaciones como para estimar los coeficientes.

```
summary(lm(proporcion ~ ., data = occ_lag_sin_imp %>%
             select_if(is.numeric)))
```

```
##
## Call:
## lm(formula = proporcion ~ ., data = occ_lag_sin_imp %>% select_if(is.numeric))
##
## Residuals:
##       1       2       3       4       5       6       7       8       9      10
## -7.3673  5.9097 -4.4976  0.4473  1.9576  5.9348 -0.2236  3.3414 -3.9892 -2.0176
##      11      12
##  1.7860 -1.2816
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    51.9254     2.7743  18.717  0.00284 **
## Ano             4.4503    10.7056   0.416  0.71799
## Mes            -5.1301    13.2972  -0.386  0.73682
## Prof_max       -0.8288    18.0491  -0.046  0.96755
## Temperatura     8.6416    10.0607   0.859  0.48088
## Conductividad -10.4586    10.5926  -0.987  0.42755
## pH             -2.5250    12.2631  -0.206  0.85592
## Turbidez       -1.8013    15.2195  -0.118  0.91660
## Clorofila       0.4008     7.8022   0.051  0.96370
## Amonio         12.4080    15.0732   0.823  0.49694
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.61 on 2 degrees of freedom
## Multiple R-squared:  0.5998, Adjusted R-squared:  -1.201
## F-statistic: 0.333 on 9 and 2 DF,  p-value: 0.8998
```

No se rechaza que los coeficientes sean nulos, y el R^2 ajustado es muy negativo. Hay que reducir el número de variables.

```
prop_lag_sin_cor <- occ_lag_sin_imp_ent %>% select_if(is.numeric) %>%
  cor(use = "pairwise.complete.obs")

prop_lag_sin_cor2 <- prop_lag_sin_cor["proporcion",] %>% abs() %>%
  sort(decreasing = TRUE)

prop_lag_sin_cor2[2:7] %>% t() %>%
  kable(digits=3,
        caption=
          "Correlación proporción lagunas respecto al clima sin imputar")
```

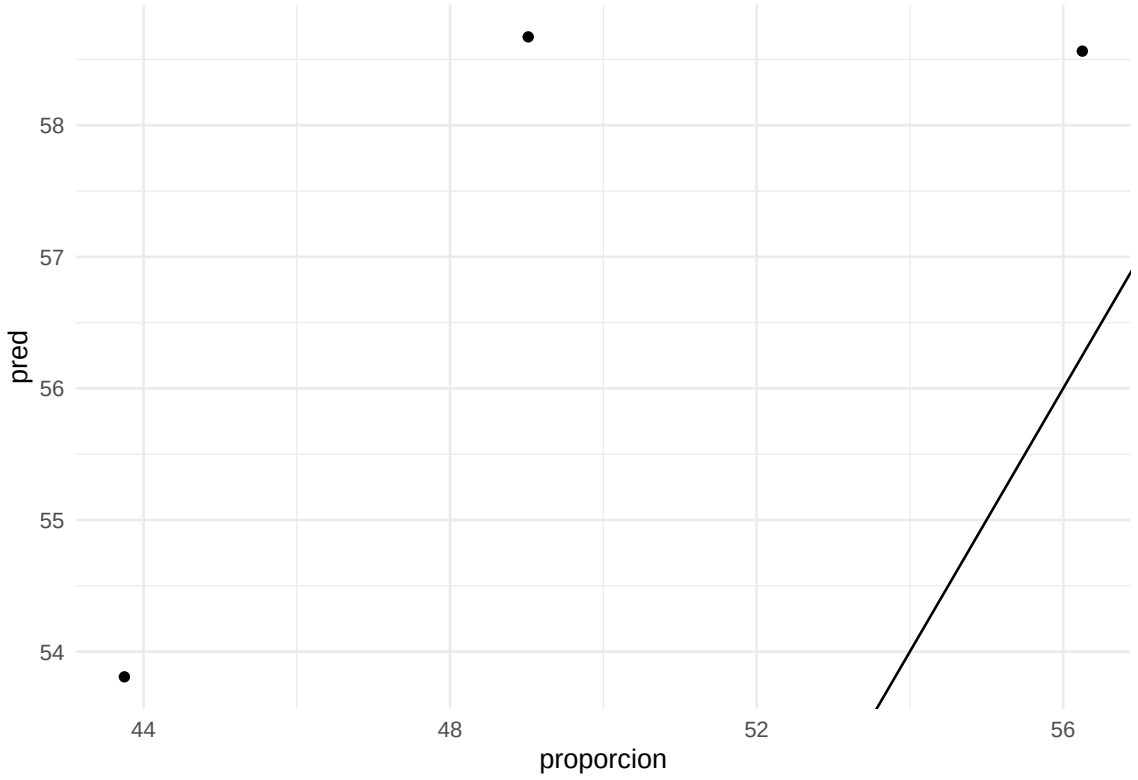
Tabla B.9: Correlación proporción lagunas respecto al clima sin imputar

Conductividad	Prof_max	Clorofila	Temperatura	Mes	pH
0.411	0.27	0.233	0.186	0.173	0.112

Tabla B.10: Regresiones proporción lagunas, clima sin imputar

Variables	R^2 ajustado	p-valor
Conductividad + Prof_max + Clorofila + Temperatura + Mes + pH	-0.52347	0.76267
Conductividad + Prof_max + Clorofila + Temperatura + Mes	-0.49227	0.78341
Conductividad + Prof_max + Clorofila + Temperatura	-0.47014	0.82662
Conductividad + Prof_max + Clorofila	-0.1967	0.66326
Conductividad + Prof_max	-0.01643	0.44302

```
err_reg_occ_lag_sin <- occ_lag_sin_imp_ent %>% select(proporcion) %>%
  cbind(pred = predict(lm(proporcion ~ Conductividad + Prof_max,
                          data = occ_lag_sin_imp_ent))) %>%
  rmse(truth = proporcion, estimate = pred) %>%
  pluck(3)

occ_lag_sin_imp_test %>% select(proporcion) %>%
  cbind(pred = predict(lm(proporcion ~ Conductividad + Prof_max,
                          data = occ_lag_sin_imp_ent),
                       occ_lag_sin_imp_test)) %>%
  ggplot(aes(x=proporcion, y = pred)) + geom_point() +
  geom_abline() + theme_minimal()
```



```
## [1] "El error cometido es del 11.16 %"
```

Para las marismas sí hay registros suficientes para hacer el modelo completo con la muestra test.

```
summary(lm(proporcion ~ ., data = occ_mar_sin_imp_ent %>%
             select_if(is.numeric)))
```

```
##
## Call:
## lm(formula = proporcion ~ ., data = occ_mar_sin_imp_ent %>% select_if(is.numeric))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -20.3985  -4.5523  -0.7434   4.3888  29.8086
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    52.9490    10.6283   4.982  0.00108 **
## Ano            44.0642    30.6749   1.436  0.18879
## Mes            -9.4513     9.3324  -1.013  0.34083
## Prof_max        0.2127     4.7288   0.045  0.96523
## Temperatura    11.5700     9.7143   1.191  0.26778
## Conductividad   1.2780     8.6718   0.147  0.88648
## O2_porc       -11.3822    15.1366  -0.752  0.47361
## pH             15.4701    11.9196   1.298  0.23050
## Turbidez        1.5167     4.2417   0.358  0.72992
## Clorofila      18.7454    11.5146   1.628  0.14218
## Amonio         16.3305    33.0448   0.494  0.63445
## Nitrato       -35.6750    33.2952  -1.071  0.31521
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.95 on 8 degrees of freedom
## Multiple R-squared:  0.5471, Adjusted R-squared:  -0.07564
## F-statistic: 0.8785 on 11 and 8 DF,  p-value: 0.59
```

No se rechaza la nulidad de coeficientes, y el R^2 es negativo.

```
prop_mar_sin_cor <- occ_mar_sin_imp_ent %>% select_if(is.numeric) %>%
  cor(use = "pairwise.complete.obs")

prop_mar_sin_cor2 <- prop_mar_sin_cor["proporcion",] %>% abs() %>%
  sort(decreasing = TRUE)

prop_mar_sin_cor2[2:7] %>% t() %>%
  kable(digits=3,
```

Tabla B.11: Correlación proporción marismas respecto al clima sin imputar

Temperatura	Mes	Conductividad	Prof_max	O2_porc	Amonio
0.375	0.214	0.133	0.11	0.074	0.051

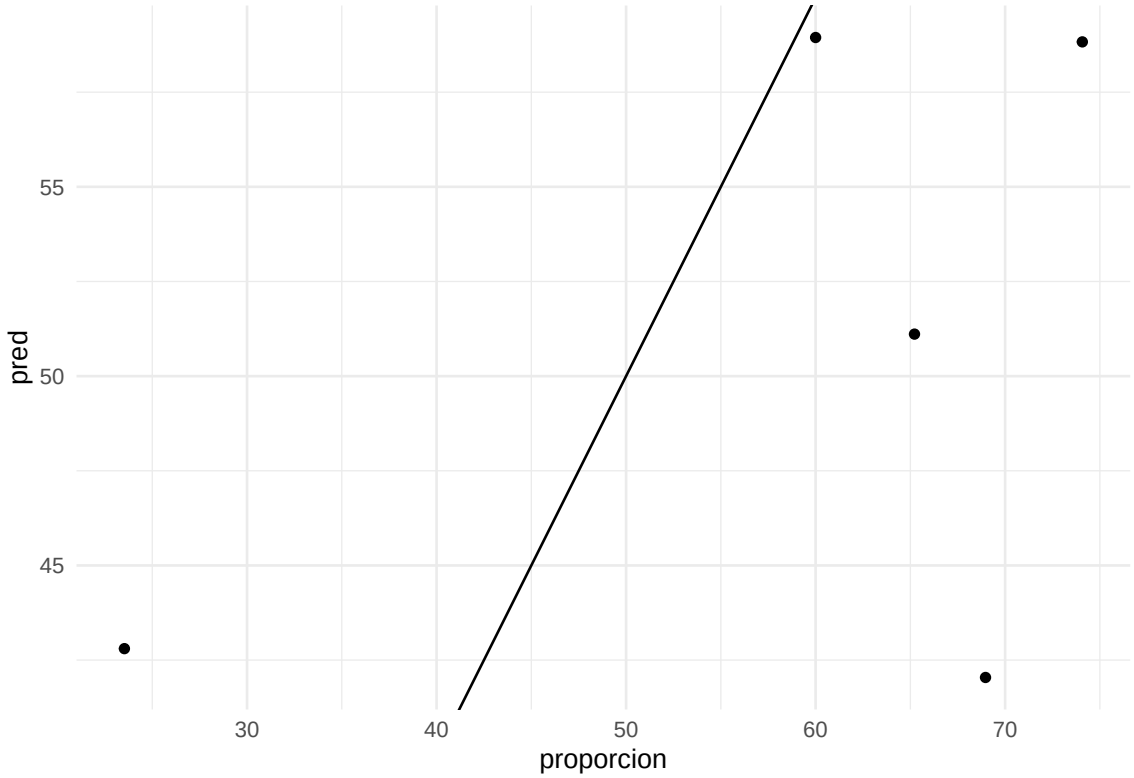
Tabla B.12: Regresiones proporción marismas, clima sin imputar

Variables	R^2 ajustado	p-valor
Temperatura + Mes + Conductividad + Prof_max + O2_porc + Amonio	0.07437	0.34205
Temperatura + Mes + Conductividad + Prof_max + O2_porc	-0.15243	0.77321
Temperatura + Mes + Conductividad + Prof_max	-0.0782	0.63209
Temperatura + Mes + Conductividad	-0.0121	0.45157
Temperatura + Mes	0.04234	0.26898

```
        caption=
          "Correlación proporción marismas respecto al clima sin imputar")
```

El pvalor se va reduciendo, pero incluso con un 10 % de significación, en todos los casos se acepta que los coeficientes sean nulos.

```
err_reg_occ_mar_sin <- occ_mar_sin_imp_ent %>% select(proporcion) %>%
  cbind(pred = predict(lm(proporcion ~Temperatura + Mes,
                          data = occ_mar_sin_imp_ent))) %>%
  rmse(truth = proporcion, estimate = pred) %>%
  pluck(3)

occ_mar_sin_imp_test %>% select(proporcion) %>%
  cbind(pred = predict(lm(proporcion ~ Temperatura + Mes,
                          data = occ_mar_sin_imp_ent),
                       occ_mar_sin_imp_test)) %>%
  ggplot(aes(x=proporcion, y = pred)) + geom_point() +
  geom_abline() + theme_minimal()
```




```
## [1] "El error cometido es del 23.77 %"
```

B.2.2. Datos imputados, por hábitat

```
summary(lm(proporcion ~ ., data = occ_lag_imp_ent %>%
             select_if(is.numeric)))
```

```
##
## Call:
## lm(formula = proporcion ~ ., data = occ_lag_imp_ent %>% select_if(is.numeric))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -27.4777  -5.6715   0.1159   5.5986  28.8033
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    49.6798     1.4893  33.357   <2e-16 ***
## Ano             2.0684     2.4238   0.853   0.3976
## Mes             0.7498     2.0352   0.368   0.7142
## Prof_max       -0.7697     1.9697  -0.391   0.6977
## Temperatura     5.3478     2.4976   2.141   0.0373 *
## Conductividad  -1.3794     1.3756  -1.003   0.3209
## O2_abs          0.2038     1.7868   0.114   0.9096
```

Tabla B.13: Correlación proporción lagunas respecto al clima imputado

Temperatura	Amonio	Ano	Clorofila	Turbidez	Prof_max
0.327	0.185	0.165	0.148	0.123	0.078

Tabla B.14: Regresiones proporción lagunas, clima imputado

Variables	R^2 ajustado	p-valor
Temperatura + Amonio + Ano + Clorofila + Turbidez + Prof_max	0.06363	0.14383
Temperatura + Amonio + Ano + Clorofila + Turbidez	0.07571	0.0956
Temperatura + Amonio + Ano + Clorofila	0.07801	0.07312
Temperatura + Amonio + Ano	0.0736	0.0617
Temperatura + Amonio	0.08325	0.03008

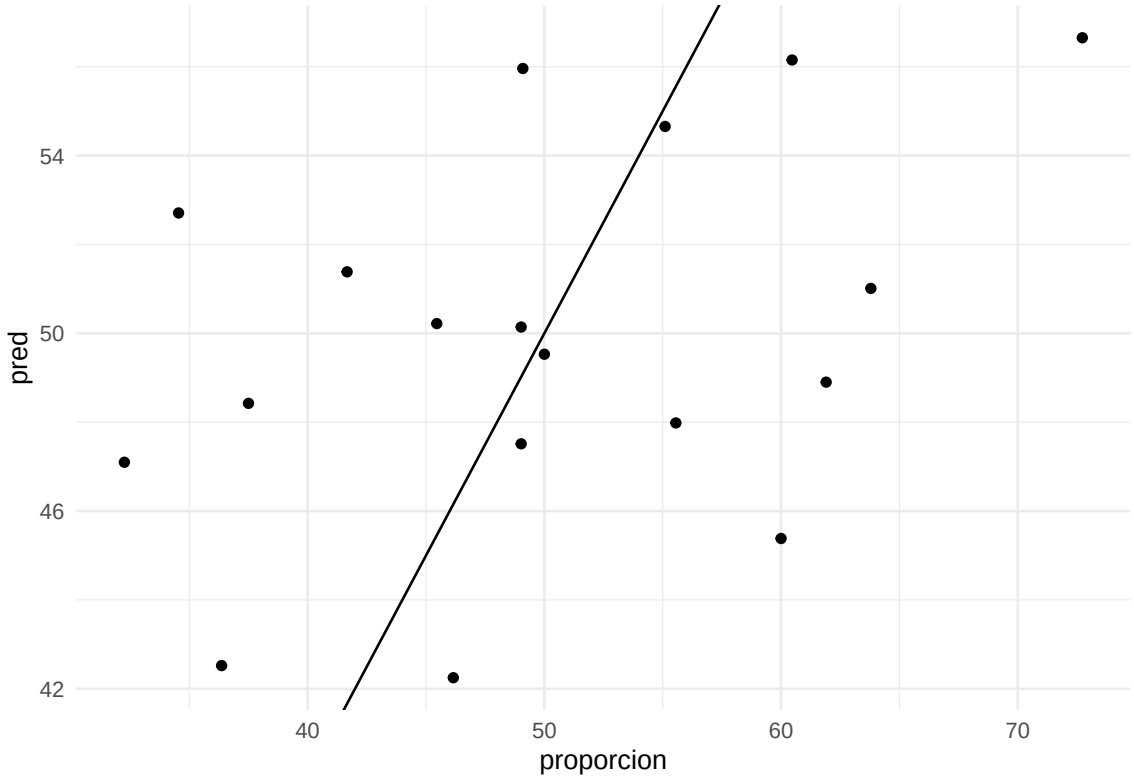
```
## pH             -1.0157     1.6985  -0.598   0.5526
## Turbidez       -3.1811     3.6010  -0.883   0.3813
## Clorofila      -1.4015     1.4344  -0.977   0.3333
## Amonio         -0.3144     2.8742  -0.109   0.9133
## Nitrato         0.7716     1.9054   0.405   0.6873
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.68 on 49 degrees of freedom
## Multiple R-squared:  0.1807, Adjusted R-squared:  -0.003208
## F-statistic: 0.9826 on 11 and 49 DF,  p-value: 0.4747
```

De nuevo no se rechaza la nulidad de coeficientes, y el R^2 ajustado es negativo.

```
prop_lag_imp_cor <- occ_lag_imp_ent %>% select_if(is.numeric) %>%
  cor(use = "pairwise.complete.obs")

prop_lag_imp_cor2 <- prop_lag_imp_cor["proporcion",] %>% abs() %>%
  sort(decreasing = TRUE)

prop_lag_imp_cor2[2:7] %>% t() %>%
  kable(digits=3,
        caption=
          "Correlación proporción lagunas respecto al clima imputado")
```

```
err_reg_occ_lag_imp <- occ_lag_imp_ent %>% select(proporcion) %>%
  cbind(pred = predict(lm(proporcion ~Temperatura + Amonio,
                          data = occ_lag_imp_ent))) %>%
  rmse(truth = proporcion, estimate = pred) %>%
  pluck(3)

occ_lag_imp_test %>% select(proporcion) %>%
  cbind(pred = predict(lm(proporcion ~ Temperatura + Amonio,
                          data = occ_lag_imp_ent),
```

```
                        occ_lag_imp_test)) %>%
  ggplot(aes(x=proporcion, y = pred)) + geom_point() +
  geom_abline() + theme_minimal()
```




```
## [1] "El error cometido es del 19.9 %"
```

```
summary(lm(proporcion ~ ., data = occ_mar_imp_ent %>%
             select_if(is.numeric)))
```

```
##
## Call:
## lm(formula = proporcion ~ ., data = occ_mar_imp_ent %>% select_if(is.numeric))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -38.241  -7.090   2.877   7.836  37.359
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   49.12322    2.19108  22.420   <2e-16 ***
## Ano            2.88403    2.29036   1.259    0.214
## Mes           -2.08175    2.13681  -0.974    0.335
## Prof_max      -0.90283    2.08567  -0.433    0.667
## Temperatura    1.55203    2.97622   0.521    0.604
## Conductividad -0.69122    2.60381  -0.265    0.792
```

Tabla B.15: Correlación proporción marismas respecto al clima imputado

Nitrato	Ano	Clorofila	Mes	pH	Amonio
0.203	0.173	0.114	0.113	0.062	0.055

Tabla B.16: Regresiones proporción marismas, clima imputado

Variables	R^2 ajustado	p-valor
Nitrato + Ano + Clorofila + Mes + pH + Amonio	-0.00974	0.49934
Nitrato + Ano + Clorofila + Mes + pH	0.00791	0.37022
Nitrato + Ano + Clorofila + Mes	0.02454	0.25348
Nitrato + Ano + Clorofila	0.02337	0.23003
Nitrato + Ano	0.02357	0.18731

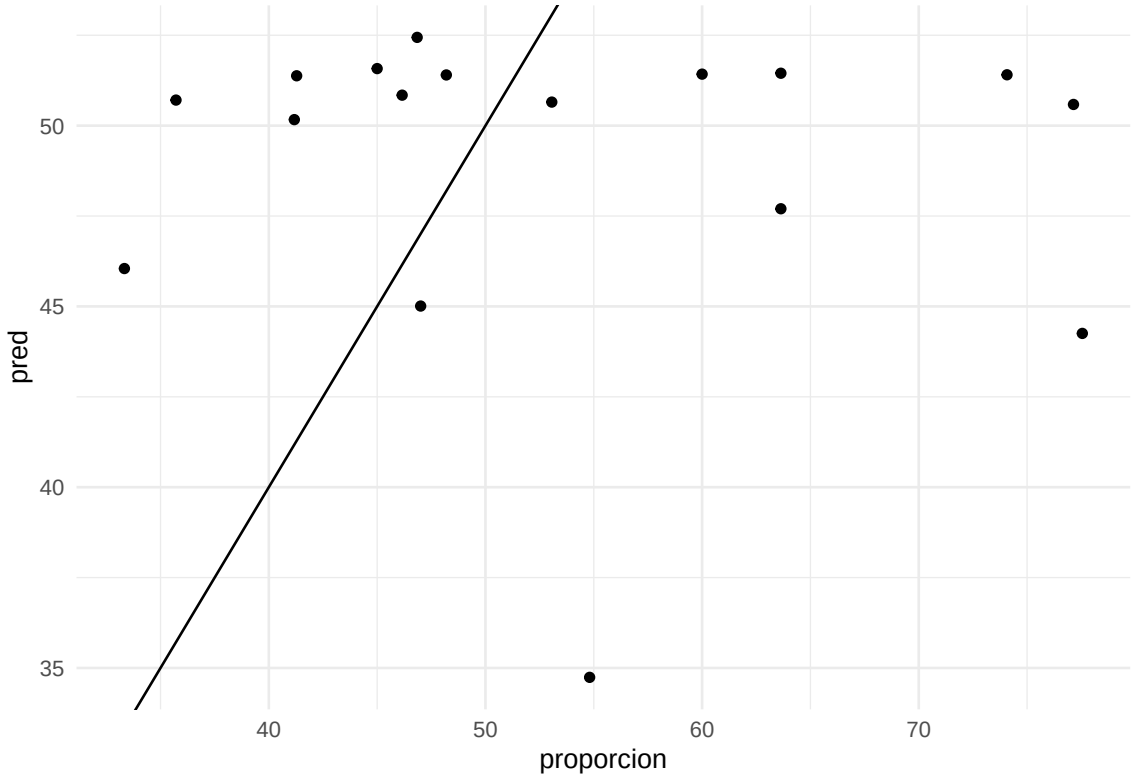
```
## O2_abs         2.55248    2.74880   0.929    0.358
## pH            -0.09848    2.61046  -0.038    0.970
## Turbidez      -1.72825    4.45794  -0.388    0.700
## Clorofila     -2.00667    2.94319  -0.682    0.499
## Amonio        -0.37035    2.65271  -0.140    0.890
## Nitrato       -4.75425    2.96670  -1.603    0.115
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.03 on 49 degrees of freedom
## Multiple R-squared:  0.1162, Adjusted R-squared:  -0.08216
## F-statistic: 0.5859 on 11 and 49 DF,  p-value: 0.831
```

Igual que en las lagunas.

```
prop_mar_imp_cor <- occ_mar_imp_ent %>% select_if(is.numeric) %>%
  cor(use = "pairwise.complete.obs")

prop_mar_imp_cor2 <- prop_mar_imp_cor["proporcion",] %>% abs() %>%
  sort(decreasing = TRUE)

prop_mar_imp_cor2[2:7] %>% t() %>%
  kable(digits=3,
        caption=
          "Correlación proporción marismas respecto al clima imputado")
```

```
err_reg_occ_mar_imp <- occ_mar_imp_ent %>% select(proporcion) %>%
  cbind(pred = predict(lm(proporcion ~ Nitrato + Ano,
                          data = occ_mar_imp_ent))) %>%
  rmse(truth = proporcion, estimate = pred) %>%
  pluck(3)

occ_mar_imp_test %>% select(proporcion) %>%
  cbind(pred = predict(lm(proporcion ~ Nitrato + Ano,
```

```
                          data = occ_mar_imp_ent),
                        occ_mar_imp_test)) %>%
  ggplot(aes(x=proporcion, y = pred)) + geom_point() +
  geom_abline() + theme_minimal()
```




```
## [1] "El error cometido es del 26.04 %"
```

B.3. Tuning bosques aleatorios

Inicialmente el tuning de árboles aleatorios no se llevaba a cabo por completo mediante **tidymodels** pues al indicar que el parámetro mtry se obtuviera mediante tuning el código producía un error. La forma de solucionarlo finalmente fue indicar los límites en los que debía buscarse el parámetro, pero antes de eso se aplicaba primero el tuning de mtry mediante **caret** y posteriormente utilizando el valor óptimo obtenido para mtry se aplicaba con **tidymodels** el tuning de trees y min_n. El código con el que se llevaba a cabo esto es el siguiente:

```
lag_imp_rf_mtry <-
  train(capt ~ ., data=lag_imp_ent, method = 'rf',
        na.action = na.exclude, tuneGrid = expand.grid(.mtry = 3:10),
        trControl = control <- trainControl(method='repeatedcv',
                                            number=10,
                                            search = 'grid'))
```

```
ggplot(lag_imp_rf_mtry) + theme_minimal() +
  xlab("Número de variables seleccionadas") +
  ylab("RMSE") + ggtitle("Lagunas")

rf_lag_capt_tune_model <-
  rand_forest(mtry = as.numeric(lag_imp_rf_mtry$bestTune),
              trees = tune(),min_n = tune()) %>%
  set_engine("randomForest") %>%
  set_mode("regression")

lag_capt_rf_tune_wkfl <- workflow() %>%
  add_model(rf_lag_capt_tune_model)  %>%
  add_formula(capt ~ .)

lag_capt_folds <- vfold_cv(lag_imp_ent, v=10, strata = capt)

rf_capt_grid <- grid_random(parameters(rf_lag_capt_tune_model),size=10)

lag_capt_rf_tuning <- lag_capt_rf_tune_wkfl %>%
  tune_grid(resamples= lag_capt_folds,
            grid=rf_capt_grid,
            metrics = metric_set(rmse))

lag_capt_rf_best_model <- lag_capt_rf_tuning %>%
  select_best(metric = "rmse")

lag_capt_rf_tune_wkfl_final <- lag_capt_rf_tune_wkfl %>%
  finalize_workflow(lag_capt_rf_best_model)

lag_capt_rf_tune_err <- lag_capt_rf_tune_wkfl_final %>%
  last_fit(split = lag_imp_split) %>%
  collect_metrics() %>% cbind(media = mean(lag_imp_test$capt)) %>%
  summarise(error = .estimate/media*100)


print(paste("El error cometido en las lagunas es del",
            round(lag_capt_rf_tune_err[1,1],2), "%"))
```

```
## R version 4.2.1 (2022-06-23 ucrt)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19045)
##
## Matrix products: default
##
## Random number generation:
##  RNG:     Mersenne-Twister
##  Normal:  Inversion
##  Sample:  Rounding
##
## locale:
## [1] LC_COLLATE=Spanish_Spain.utf8  LC_CTYPE=Spanish_Spain.utf8
## [3] LC_MONETARY=Spanish_Spain.utf8 LC_NUMERIC=C
## [5] LC_TIME=Spanish_Spain.utf8
##
## attached base packages:
## [1] grid      stats     graphics  grDevices utils     datasets  methods
## [8] base
##
## other attached packages:
##  [1] kknn_1.3.1           corrplot_0.92        DAAG_1.25.4
##  [4] magrittr_2.0.3       randomForest_4.7-1.1 C50_0.1.8
##  [7] rpart_4.1.16         caret_6.0-93         lattice_0.20-45
## [10] yardstick_1.1.0      workflowsets_1.0.1   workflows_1.1.3
## [13] tune_1.1.1           rsample_1.1.1        recipes_1.0.5
## [16] parsnip_1.1.0        modeldata_1.1.0      infer_1.0.4
## [19] dials_1.2.0          scales_1.2.1         broom_1.0.1
## [22] tidymodels_1.0.0     vcd_1.4-11           kableExtra_1.3.4
## [25] lubridate_1.8.0      knitr_1.42           readxl_1.4.1
## [28] skimr_2.1.5          forcats_0.5.2        stringr_1.5.0
## [31] dplyr_1.1.1          purrr_1.0.1          tidyr_1.3.0
## [34] tibble_3.2.1         ggplot2_3.4.0        tidyverse_1.3.2
## [37] readr_2.1.3
##
## loaded via a namespace (and not attached):
##   [1] backports_1.4.1      systemfonts_1.0.4    igraph_1.4.2
##   [4] plyr_1.8.7           repr_1.1.6           splines_4.2.1
##   [7] listenv_0.9.0        digest_0.6.29        foreach_1.5.2
##  [10] htmltools_0.5.4      fansi_1.0.3          googlesheets4_1.1.0
##  [13] tzdb_0.3.0           globals_0.16.2       modelr_0.1.11
##  [16] gower_1.0.0          vroom_1.6.1          svglite_2.1.1
##  [19] hardhat_1.3.0        jpeg_0.1-10          colorspace_2.0-3
##  [22] rvest_1.0.3          rbibutils_2.2.13     haven_2.5.1
##  [25] xfun_0.38            crayon_1.5.2         jsonlite_1.8.2
##  [28] libcoin_1.0-9        survival_3.3-1       zoo_1.8-12
##  [31] iterators_1.0.14     glue_1.6.2           gtable_0.3.3
##  [34] gargle_1.4.0         ipred_0.9-13         webshot_0.5.4
```

```
##  [37] future.apply_1.10.0  mvtnorm_1.1-3        DBI_1.1.3
##  [40] Rcpp_1.0.9           viridisLite_0.4.1    Cubist_0.4.2.1
##  [43] GPfit_1.0-8          bit_4.0.5            proxy_0.4-27
##  [46] Formula_1.2-5        stats4_4.2.1         lava_1.7.2.1
##  [49] prodlim_2019.11.13   httr_1.4.5           RColorBrewer_1.1-3
##  [52] ellipsis_0.3.2       pkgconfig_2.0.3      farver_2.1.1
##  [55] deldir_1.0-6         nnet_7.3-17          dbplyr_2.3.2
##  [58] utf8_1.2.2           tidyselect_1.2.0     labeling_0.4.2
##  [61] rlang_1.1.0          DiceDesign_1.9       reshape2_1.4.4
##  [64] munsell_0.5.0        cellranger_1.1.0     tools_4.2.1
##  [67] cli_3.6.1            generics_0.1.3       evaluate_0.20
##  [70] fastmap_1.1.0        yaml_2.3.5           ModelMetrics_1.2.2.2
##  [73] bit64_4.0.5          fs_1.5.2             future_1.32.0
##  [76] nlme_3.1-157         xml2_1.3.3           compiler_4.2.1
##  [79] rstudioapi_0.14      png_0.1-8            e1071_1.7-11
##  [82] reprex_2.0.2         lhs_1.1.6            stringi_1.7.8
##  [85] Matrix_1.5-3         vctrs_0.6.1          pillar_1.9.0
##  [88] lifecycle_1.0.3      furrr_0.3.1          Rdpack_2.4
##  [91] lmtest_0.9-40        data.table_1.14.2    latticeExtra_0.6-30
##  [94] R6_2.5.1             parallelly_1.35.0    codetools_0.2-19
##  [97] MASS_7.3-58.1        withr_2.5.0          mgcv_1.8-42
## [100] parallel_4.2.1       hms_1.1.3            modelenv_0.1.1
## [103] timeDate_4022.108    class_7.3-20         rmarkdown_2.21
## [106] inum_1.0-5           googledrive_2.1.0    partykit_1.2-20
## [109] pROC_1.18.0          base64enc_0.1-3      interp_1.1-4
```

Bibliografía

[1] Alcorlo, Paloma y Diéguez-Uribeondo, Javier (2014). «El cangrejo señal y el declive de las poblaciones de cangrejo autóctono». *Ambienta: La revista del Ministerio de Medio Ambiente*, **(109)**, pp. 52–61.

[2] Alcorlo, Paloma; Geiger, Walter y Otero, Marina (2008). «Reproductive biology and life cycle of the invasive crayfish Procambarus clarkii (Crustacea: Decapoda) in diverse aquatic habitats of South-Western Spain: Implications for population control». *Fundamental and Applied Limnology*, **173(3)**, p. 197.

[3] Alcorlo, Paloma; Noguerales, Victor; Mollá, Salvador et al. (2019). «Modeling the population dynamics of the Red Swamp Crayfish (Procambarus clarkii) in Doñana: Application to the harvesting strategies». *Journal of FisheriesSciences. com*, **13(2)**, pp. 001–011.

[4] Blanco, Juan A. (2008). «Pasos básicos en la modelización ecológica», **33(Especial 2014)**, pp. 471–484. ISSN 2078-7235.

[5] Bravo, Miguel A; Andreu, Ana C; Román, Isidro; Arribas, Rosa; Márquez-Ferrando, Rocío; Díaz-Delgado, Ricardo y Bustamante, Javier (2023). «Long-term monitoring in the biometrics of the red-swamp crayfish (Procambarus clarkii, Girard 1852) in Doñana Wetlands 2004-2022».

[6] Colomer, Mª Àngels; Margalida, Antoni; Valencia, Luis y Palau, Antoni (2014). «Application of a computational model for complex fluvial ecosystems: The population dynamics of zebra mussel Dreissena polymorpha as a case study». *Ecological Complexity*, **20**, pp. 116–126.

[7] Desconocido (2000). «Linear regression». https://web.archive.org/web/20080222195200/http://www.curvefit.com/linear_regression.htm.

[8] —— (2018). «Conejos, Cangrejos y Camalote». http://admreportaje.blogspot.com/2018/02/conejos-cangrejos-y-camalote.html.

[9] —— (2023). «Random Forest: Bosque aleatorio. Definición y funcionamiento». https://datascientest.com/es/random-forest-bosque-aleatorio-definicion-y-funcionamiento.

[10] Elin Waring, Amelia McNamara, Michael Quinn y Zhu, Hao (2022). *skimr: Compact and Flexible Summaries of Data*. https://cran.r-project.org/web/packages/skimr/. R package version 2.1.5.

[11] Hosmer Jr, David W; Lemeshow, Stanley y Sturdivant, Rodney X (2013). *Applied logistic regression*, tomo 398. John Wiley & Sons.

[12] Kuhn y Max (2008). «Building Predictive Models in R Using the caret Package». *Journal of Statistical Software*, **28(5)**, p. 1–26. doi: 10.18637/jss.v028.i05. https://www.jstatsoft.org/index.php/jss/article/view/v028i05.

[13] Kuhn, Max (2008). «Building predictive models in R using the caret package». *Journal of statistical software*, **28**, pp. 1–26.

[14] Kuhn, Max y Silge, Julia (2022). *Tidy Modeling with R.* " O'Reilly Media, Inc.".

[15] Kuhn, Max y Wickham, Hadley (2020). *Tidymodels: a collection of packages for modeling and machine learning using tidyverse principles.*. https://www.tidymodels.org.

[16] Liaw, Andy y Wiener, Matthew (2002). «Classification and Regression by randomForest». *R News*, **2(3)**, pp. 18–22. https://CRAN.R-project.org/doc/Rnews/.

[17] López, José Francisco (2017). «Coeficiente de determinación (R cuadrado)». https://economipedia.com/definiciones/r-cuadrado-coeficiente-determinacion.html.

[18] Luque-Calvo, Pedro L. (2017). *Escribir un Trabajo Fin de Estudios con R Markdown*.

[19] Masip Rodó, D; Escudero Bakx, G; Benítez Iglesias, R y Kanaan Izquierdo, S (2013). «Inteligencia artificial avanzada». *Cataluña: Editorial UOC*.

[20] Max Kuhn, Steve Weston y Quinlan, Ross (2023). *C50: C5.0 Decision Trees and Rule-Based Models*. https://cran.r-project.org/web/packages/C50/. R package version 0.1.8.

[21] Pannekoek, Jeroen (2001). «TRIM 3 manual (trends & indices for monitoring data)». *http://www. ebcc. info/trim. html*.

[22] Rodrigo, Joaquín Amat (2017). «Árboles de decisión, random forest, gradient boosting y C5.0». https://www.cienciadedatos.net/documentos/33_arboles_decision_random_forest_gradient_boosting_c50.

[23] Ruiz-Olmo, Jordi y Clavero, Miguel (2008). «Los cangrejos en la ecología y recuperación de la nutria en la Península Ibérica».

[24] Terry Therneau, Beth Atkinson y Ripley, Brian (2022). *rpart: Recursive Partitioning and Regression Trees*. https://cran.r-project.org/web/packages/rpart/. R package version 4.1.19.

[25] Torres, S Palomar; Suárez, E; Haro, A; Fresno, Ana y Ramos, C (2011). «Impacto de Procambarus clarkii en arrozales españoles y en la evaluación medioambiental de productos fitosanitarios». *Phytoma España: La revista profesional de sanidad vegetal*, **(234)**, pp. 76–78.

[26] Wickham, H; François, R; Henry, L y Müller, K (2020). *dplyr: A Grammar of Data Manipulation (2020)*.
https://cran.r-project.org/web/packages/dplyr/. R package version 1.1.2.

[27] Wickham, Hadley (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. ISBN 978-3-319-24277-4.
https://ggplot2.tidyverse.org.

[28] Wickham, Hadley; Averick, Mara; Bryan, Jennifer; Chang, Winston; McGowan, Lucy D'Agostino; François, Romain; Grolemund, Garrett; Hayes, Alex; Henry, Lionel; Hester, Jim; Kuhn, Max; Pedersen, Thomas Lin; Miller, Evan; Bache, Stephan Milton; Müller, Kirill; Ooms, Jeroen; Robinson, David; Seidel, Dana Paige; Spinu, Vitalie; Takahashi, Kohske; Vaughan, Davis; Wilke, Claus; Woo, Kara y Yutani, Hiroaki (2019). «Welcome to the tidyverse». *Journal of Open Source Software*, **4(43)**, p. 1686. doi: 10.21105/joss.01686.

[29] Wickham, Hadley y Grolemund, Garrett (2016). *R for data science: import, tidy, transform, visualize, and model data*. " O'Reilly Media, Inc.".

[30] Wickham, Hadley y Hester, Jim (2023). *readr: Read Rectangular Text Data*.
https://cran.r-project.org/web/packages/readr/. R package version 2.1.4.

[31] Wickham, Hadley; Vaughan, Davis y Girlich, Maximilian (2023). *tidyr: Tidy Messy Data*.
https://cran.r-project.org/web/packages/tidyr/. R package version 1.3.0.

[32] Winston Chang, JJ Allaire, Joe Cheng y Co (2022). *shiny: Web Application Framework for R*.
https://cran.r-project.org/web/packages/shiny/. R package version 1.7.4.

[33] Xie, Yihui (2023). *knitr: A General-Purpose Package for Dynamic Report Generation in R*.
https://yihui.org/knitr/. R package version 1.42.

[34] Zhu, Hao (2021). *kableExtra: Construct Complex Table with 'kable' and Pipe Syntax*.
https://cran.r-project.org/web/packages/kableExtra/. R package version 1.3.4.