



UNIVERSITAT OBERTA DE CATALUNYA (UOC)

MÁSTER UNIVERSITARIO EN CIENCIA DE DATOS (*Data Science*)

TRABAJO FINAL DE MÁSTER

ÁREA 5

Análisis y modelización predictiva de accidentes de tráfico con animales involucrados en vías interurbanas de España

Autor: Alba Gómez Varela

Tutor: Sergio Trilles Oliver

Profesor: Albert Solé Ribalta

Madrid, 12 de enero de 2023

Créditos/Copyright



Esta obra está sujeta a una licencia de Reconocimiento - NoComercial - CompartirIgual [4.0 CreativeCommons International](#) (CC BY-NC-SA 4.0).

El [repositorio en GitHub](#)¹ con todo el código de este *Trabajo Final de Máster* y el [proyecto en RPubs](#)² que contiene la fase de análisis están sujetos al mismo tipo de licencia.

¹<https://github.com/alba620/analisis-prediccion-accidentes-trafico-animales>

²https://rpubs.com/alba_gvarela/animal_car_accident_analysis

FICHA DEL TRABAJO FINAL

Título del trabajo:	Análisis y modelización predictiva de accidentes de tráfico con animales involucrados en vías interurbanas de España
Nombre del autor:	Alba Gómez Varela
Nombre del colaborador/a docente:	Sergio Trilles Oliver
Nombre del PRA:	Albert Solé Ribalta
Fecha de entrega (mm/aaaa):	01/2023
Titulación o programa:	Máster Universitario en Ciencia de Datos
Área del Trabajo Final:	Área 5
Idioma del trabajo:	Español
Palabras clave	predicción, accidente tráfico, animal

Abstract

The combination of technologies related to Data Science and open data provides a new approach to solving problems that have not been addressed to date. Moreover, it helps to tackle traditional issues more effectively. Consequently, the benefits that could be brought to society thanks to machine learning (ML) techniques are exponential in many different fields.

One of the areas to be deeper explored is road safety in Spain. However, it is worth to remark that accelerated steps have been taken in recent years due to investments of resources in projects such as the DGT 3.0 platform and new regulations on connected car which, in turn, will generate a greater volume of data.

This work aims to extract as much knowledge as possible from existing data on wildlife–vehicle collisions (WVC) in Spain. By creating ML models that identify risk areas based on context, it is confirmed that it is possible to predict them and to generate a methodology for this purpose. The good results, with 92 % accuracy and more than 91 % performance in the rest of metrics, show that this project could lead to new research and implementation pathways.

Therefore, as a final step, in order to foster the transfer of knowledge between the academic world and society, the appropriate documentation on the ML models and other findings that facilitate their deployment on the DGT 3.0 platform is delivered. Furthermore, the code, tools and datasets generated during the process are published under an open licence that allows to share, reproduce and enhance the knowledge that has been developed.

Keywords: prediction, machine learning, road accident, road safety, animal, DGT, DGT 3.0, connected car, wildlife–vehicle collisions, WVC.

Resumen

La aplicación de tecnologías relacionadas con la Ciencia de Datos a los conjuntos de datos abiertos disponibles abre un nuevo camino a la resolución de problemas no planteados hasta la fecha o cuyas propuestas para ser abordados se pueden mejorar. En consecuencia, el beneficio que se puede aportar a la sociedad gracias a técnicas de aprendizaje automático (*machine learning*), entre otras, es exponencial en muy diferentes ámbitos.

Una de las áreas por explorar más en profundidad es la de la seguridad vial en España. No obstante, cabe destacar que se están dando pasos acelerados en los últimos años gracias a inversiones de recursos en iniciativas como la plataforma DGT 3.0 y las normativas sobre coche conectado que, a su vez, generarán un volumen mayor de datos.

Este trabajo pretende extraer el mayor conocimiento posible de los datos existentes sobre accidentes de tráfico en España en los que hay animales involucrados. Mediante la creación de modelos que identifican zonas de riesgo en función del contexto, se confirma que se pueden predecir y generar una metodología para ello. Los buenos resultados, con un 92 % de exactitud y más de 91 % de rendimiento en el resto de métricas, muestran que el camino iniciado en este proyecto podría ser una vía de investigación e implementación a seguir.

Por ello, para fomentar el trasvase de conocimiento entre el mundo académico y la sociedad, se genera y entrega la documentación oportuna sobre los modelos y otros hallazgos que faciliten su despliegue en la plataforma DGT 3.0. Además, se publican el código, las herramientas y los conjuntos de datos generados durante el proceso bajo una licencia abierta, lo que permite compartirlos con toda la comunidad científica y reproducir el trabajo en otros entornos.

Palabras clave: predicción, machine learning, aprendizaje automático, accidente de tráfico, seguridad vial, animal, DGT, DGT 3.0, coche conectado.

Índice general

Abstract	IV
Resumen	V
Índice	V
Llistado de Figuras	VIII
Listado de Tablas	X
Acrónimos	XIV
1. Introducción	1
1.1. Descripción, contexto, justificación y relevancia	1
1.2. Motivación personal	3
1.3. Definición de los objetivos	3
1.3.1. Objetivo principal	3
1.3.2. Objetivos secundarios	4
1.4. Impacto del proyecto: sostenibilidad, ético-social y diversidad	4
1.5. Descripción de la metodología	6
1.6. Planificación del proyecto	8
2. Estado del arte	10
2.1. Características de los datos	11
2.1.1. Ámbito geográfico	11
2.1.2. Acotación temporal	12
2.1.3. Fuentes de los accidentes y tipo de fauna	12
2.2. Metodología de análisis y modelado	14
2.2.1. Preprocesado y test previos	14
2.2.2. Análisis espacio-temporal	15

2.2.3. Variables significativas: test y regresiones	16
2.2.4. Modelización predictiva: regresión logística	17
2.2.5. Tecnologías	18
2.3. Hallazgos y conclusiones	19
3. Diseño y requerimientos	23
3.1. Fuentes de datos	23
3.2. Flujo de trabajo	24
3.3. Tecnologías	25
3.3.1. Lenguajes de programación	25
3.3.2. Herramientas de terceros con licencia libre	25
3.3.3. Herramientas de elaboración propia	27
3.3.4. Hardware	29
4. Creación y análisis del <i>dataset</i> final	30
4.1. Captura, obtención y gestión de los datos	30
4.1.1. Registros de accidentes con víctimas	32
4.1.2. Registros de accidentes con animales involucrados	33
4.1.3. Vías bajo titularidad DGT	34
4.1.4. Todas las vías de España	35
4.1.5. Intensidad media diaria de las vías	35
4.1.6. Usos del suelo	37
4.1.7. Datos meteorológicos	37
4.1.8. Pendiente del terreno	38
4.1.9. Elevación del terreno	39
4.1.10. Fauna	40
4.2. Creación de nuevas variables	42
4.2.1. Longitud, latitud y geom	43
4.2.2. Día de la semana	43
4.2.3. Parte del día	44
4.2.4. Superficie de la Luna iluminada	45
4.2.5. Velocidad máxima de la vía	46
4.3. Integración de los datos de fuentes externas	47
4.3.1. Intensidad media diaria de la vía	48
4.3.2. Uso del suelo	48
4.3.3. Datos meteorológicos	49
4.3.4. Elevación del terreno	50

4.3.5. Pendiente del terreno	51
4.3.6. Velocidad máxima de la vía	51
4.4. <i>Dataset</i> final	52
4.4.1. Procesado y análisis del conjunto de datos	52
4.4.2. Descripción de los atributos	62
5. Generación y evaluación de los modelos	64
5.1. Creación del conjunto de datos para los modelos	64
5.1.1. Registros aleatorios	65
5.1.2. Accidentes con víctimas sin animales involucrados	65
5.1.3. Campos de los nuevos registros	65
5.1.4. <i>Dataset</i> completo en la base de datos	68
5.1.5. Conjuntos de entrenamiento y <i>test</i>	69
5.2. Generación de los modelos	72
5.2.1. Medidas de evaluación de los modelos	73
5.2.2. Regresión logística binaria	75
5.2.3. K vecinos más cercanos	76
5.2.4. Árbol de clasificación	77
5.2.5. Random forest	77
5.2.6. Gradient boosting	78
5.3. Análisis comparativo de los resultados	79
6. Producción de los resultados	82
6.1. Plataforma DGT 3.0	82
6.2. Contribuciones mediante otras vías de publicación	83
7. Conclusiones y líneas de trabajo futuras	84
7.1. Conclusiones	84
7.2. Líneas de trabajo futuras	85
Bibliografía	86

Índice de figuras

1.1.	Víctimas mortales en accidentes de tráfico desde 1980 en España.	2
1.2.	Fases de la metodología CRISP-DM.	7
1.3.	Diagrama Gantt con la planificación del proyecto.	9
2.1.	Posibles fases de las investigaciones de accidentes de tráfico con animales implicados.	11
3.1.	Flujo de trabajo del proyecto completo.	24
4.1.	Proceso de obtención y captura de datos de diferentes fuentes.	31
4.2.	Selección de <i>dashboards</i> de la visualización de todos los accidentes con víctimas.	32
4.3.	Vías de España bajo titularidad de la DGT (izq) y todas las vías según OSM (der).	34
4.4.	Mapa de tráfico en 2019 (izq) y resultado tras el proceso de <i>scraping</i> (der). . . .	36
4.5.	Usos del suelo en España, <i>zoom</i> sobre Calatayud y <i>buffer</i> sobre la A-2 en Calatayud.	38
4.6.	<i>Buffer</i> (arriba izq), registros GBIF (arriba der) y animales en cada accidente (abajo).	41
4.7.	Creación de la tabla con el vector final de campos.	42
4.8.	Creación de nuevas variables a partir de las tablas en la base de datos.	42
4.9.	Un registro (201.850.076.007) con los nuevos campos de localización.	43
4.10.	Un registro (201.850.076.007) con los nuevos campos referentes al día de la semana.	44
4.11.	Ejemplos con el campo sobre la parte del día según la fecha, hora y localización.	45
4.12.	Ejemplos con el campo sobre la Luna iluminada según la fecha y parte del día.	45
4.13.	Integración de los datos a partir de fuentes externas.	47
4.14.	Varios registros con el nuevo campo de IMD en función de la localización y año.	48
4.15.	Varios registros con el nuevo campo de uso del suelo en función de la localización.	49
4.16.	Registros con los nuevos campos sobre la meteorología en función del día y lugar.	50
4.17.	Varios registros con el nuevo campo de altitud en función de la localización.	50
4.18.	Varios registros con el nuevo campo de pendiente en función de la localización.	51

4.19. Varios registros con el nuevo campo de velocidad máxima según la localización.	52
4.20. Diagramas de caja de las temperaturas diarias.	54
4.21. Gráfico de densidad y Q-Q de la variable <code>tmin</code>	55
4.22. Correlogramas entre las variables numéricas.	56
4.23. Histogramas de las variables <code>luna</code> e <code>imd_total</code>	56
4.24. Gráficos de <code>maxspeed</code> , <code>uso_suelo</code> y <code>nombre_tipo_animal_1f</code>	57
4.25. Gráficos de <code>nombre_sentido</code> , <code>nombre_tipo_via</code> y <code>nombre_tipo_animal_2f</code> . . .	57
4.26. Frecuencia de <code>mes_1f</code> y <code>parte_dia</code>	58
4.27. Serie temporal completa y del año 2020.	59
4.28. Gráfico ACF 2016-2021.	59
4.29. Distribución de los accidentes por comunidades y provincias.	60
4.30. KDE de vacuno y de cabra montés.	61
 5.1. Matriz de confusión en este proyecto.	73
5.2. Evaluación de los resultados de la regresión logística.	75
5.3. Evaluación de los resultados del KNN.	76
5.4. Evaluación de los resultados del árbol de clasificación.	77
5.5. Evaluación de los resultados del random forest.	78
5.6. Evaluación de los resultados del gradient boosting.	79
5.7. Curva ROC de todos los modelos.	80

Índice de cuadros

3.1. Fuentes de datos empleadas en el proyecto.	24
4.1. Equivalencia entre el valor de <code>parte_dia</code> y el periodo del día referido.	45
4.2. Equivalencia entre el valor de <code>fclass</code> y la velocidad máxima de la vía (km/h).	46
4.3. Descripción de los atributos del dataset <code>accidentes_si</code>	63
5.1. Variables seleccionadas para la fase de modelado en <code>selected_data</code>	71
5.2. Tabla comparativa de los resultados de todos los modelos.	80
5.3. Peso de las variables más importantes en todos los modelos.	81

Acrónimos

ACM MCA	Análisis de Correspondencias Múltiples. Siglas en español o en inglés.
AEMET	Agencia Estatal de Meteorología.
API	Interfaz de Programación de Aplicaciones. Siglas en inglés.
AUC	Área bajo la curva. Siglas en inglés.
Bash	Bourne-again shell.
CCEG	Competencia de Compromiso Ético y Global.
CD	Descenso por Coordenadas. Siglas en inglés.
CKAN	Comprehensive Knowledge Archive Network.
CLI	Interfaz de Línea de Comandos. Siglas en inglés.
CNIG	Centro Nacional de Información Geográfica.
CRISP-DM	Cross Industry Standard Process for Data Mining.
CSV	Comma-Separated Values.
DGC	Dirección General de Carreteras.
DGT	Dirección General de Tráfico.
DGT 3.0	Plataforma de la Dirección General de Tráfico de coche conectado.
DDL	Lenguaje de Definición de Datos. Siglas en inglés.
FN	Falso negativo. Siglas en inglés.
FOSS	Free and Open Source Software.
FP	Falso positivo. Siglas en inglés.
GBIF	Global Biodiversity Information Facility.
GUI	Interfaz Gráfica de Usuario. Siglas en inglés.
IDE	Integrated Development Environment.
JSON	JavaScript Object Notation.
KDE	Kernel Density Estimation.
KNN	K-Nearest Neighbors.
ML	Machine Learning.
MDT	Modelo Digital del Terreno.
NND	Nearest Neighbour Distance.

ODS	Objetivos de Desarrollo Sostenible.
OSM	OpenStreetMap.
PCA	Análisis de componentes principales. Siglas en inglés.
PEC	Prueba de Evaluación Continua.
PPV	Valor positivo predictivo. Siglas en inglés.
ROC	Receiver-Operating-Characteristic.
SIG GIS	Sistema de Información Geográfica. Siglas en español o en inglés.
SQL	Structured Query Language.
TFM	Trabajo Final de Máster.
TLC	Teorema del Límite Central.
TN	Verdadero negativo. Siglas en inglés.
TNR	Tasa de verdaderos negativos. Siglas en inglés.
TP	Verdadero positivo. Siglas en inglés.
TPR	Tasa de verdaderos positivos. Siglas en inglés.
UOC	Universitat Oberta de Catalunya.

Capítulo 1

Introducción

En este capítulo se presenta el proyecto que se desarrollará en el presente *Trabajo Final de Máster* (TFM), indicando su contexto y el porqué de esta propuesta, tanto por su relevancia social (CCEG y ODS) como por motivos personales. Del mismo modo, se definen los objetivos, así como la metodología y planificación para su consecución con éxito dentro de los plazos establecidos. De este modo, se aplican todos los conocimientos adquiridos en el Máster Universitario de Ciencia de Datos (*Data Science*) de la Universitat Oberta de Catalunya (UOC).

1.1. Descripción, contexto, justificación y relevancia

En España, en 2021 perdieron la vida en **accidentes de tráfico** 1.533 personas, 1.116 de ellas en vías interurbanas. Además, 7.784 resultaron heridas graves y 110.378 heridas leves, tal y como indican los datos consolidados publicados por la Dirección General de Tráfico (DGT) en agosto de 2022 [1]. A pesar de la tendencia de claro descenso de la mortalidad en carreteras, como se puede observar en la Figura 1.1, y de que los accidentes de tráfico ya no son la primera **causa de muerte no natural** en España como ocurría entre 1980 y 2007, en la última década estos siniestros han seguido ocupando el cuarto o quinto lugar en la clasificación [2].

En este contexto, son **muchos los estudios realizados** tanto por la DGT como por otros organismos públicos y privados sobre los accidentes de tráfico centrados en **todo tipo de características**, como el medio de desplazamiento, la edad de las víctimas, el día de la semana, los accesorios de seguridad o la comunidad autónoma en la que ha tenido lugar el siniestro [3].

Entre los **factores de riesgo** más destacados por la propia DGT se encuentran la velocidad, la conducción bajo los efectos del alcohol y otras drogas, la conducción distraída, la infraestructura vial insegura, los vehículos inseguros, la atención inapropiada tras el accidente y el incumplimiento de las normas [4]. Sin embargo, desde esta Dirección General del Ministerio del Interior no se han destinado tantos recursos a analizar e intentar prevenir aquellos accidentes



Figura 1.1: Víctimas mortales en accidentes de tráfico desde 1980 en España.

Fuente: INE. Elaboración propia.

en los que hay **animales involucrados**, que en 2021 ascendieron a 31.991 [5] y cuyo coste medio anual es de más de **100 millones de euros al año**, según el primer y único estudio que data de 2015 [6], si bien es cierto que hay una estadística anual sobre esta problemática [7].

Ante esta necesidad, en el presente trabajo se analizarán los **accidentes de tráfico** con animales implicados producidos en vías interurbanas de España y se desarrollará un **modelo predictivo** en función del contexto, proyecto en el que está interesada la propia DGT. Con el objetivo de que el resultado de la investigación sea útil en el futuro, se analizarán los diferentes vectores que afectan al modelo y se publicarán tanto los conjuntos de datos resultantes como las herramientas desarrolladas durante el proyecto así como los resultados, de forma que el conocimiento esté **accesible a la comunidad científica**.

El interés por parte de DGT, además, no solo es por el propio trabajo, sino que tiene intención de aplicar esta metodología dentro de la **plataforma de coche conectado DGT 3.0**, lo que le da a este TFM un cariz de mayor relevancia al plantear el trasvase de conocimiento del mundo académico a la sociedad. Para posibilitarlo, dentro del presente proyecto se especificará el formato final de los datos para que puedan ser consumidos por la nueva plataforma.

Así, se trata de una propuesta completa e innovadora no trabajada hasta la fecha en España en la que se desarrollan todas las fases del **ciclo de vida de los datos** y se tienen en cuenta las necesidades del **cliente final (DGT)** dando lugar a un proyecto relevante para la sociedad, aunque en este caso no existe una contraprestación económica. Todo ello ofrece la oportunidad de consolidar y aplicar los conocimientos adquiridos durante el Máster Universitario en Ciencia de Datos (*Data Science*).

1.2. Motivación personal

Las razones por las que he diseñado este proyecto como *Trabajo Final de Máster* se pueden dividir en:

- **Beneficio social.** Siempre he estudiado en centros educativos públicos o financiados gracias a los impuestos de la ciudadanía. Es por ello que uno de mis intereses al plantear mi primer proyecto independiente y libre en el campo de la Ciencia de Datos es que el esfuerzo repercuta en el beneficio del conjunto de la sociedad e intentar devolver de algún modo esa inversión que se ha hecho en mí.
- **Empleo de datos abiertos.** España ha impulsado en los últimos años la transparencia y la reutilización de datos abiertos. Aunque como periodista soy consciente de que este esfuerzo podría ser bastante mayor, lo que me interesa en este punto es el desarrollo de herramientas que permitan gestionar tal cantidad de datos y extraer información relevante, aportando el conocimiento específico adquirido durante el máster, sin el cual sería imposible haber planteado este trabajo.
- **Interés personal.** Dentro de los dos puntos anteriores encajarían muchas problemáticas que podrían haber dado lugar a un TFM interesante. Sin embargo, lo que hace inclinar la balanza hacia el presente proyecto es el atropello que sufrió mi madre el 31 de octubre de 2020, convirtiéndose ella en una gran lesionada y yo en una persona trabajadora a tiempo completo y estudiante de un máster recién empezado a la que se le sumaba el rol de cuidadora. Tanto para mi familia como para mí, no puede haber mejor forma de cerrar esta etapa que con una investigación que pueda ayudar a evitar accidentes de tráfico.

1.3. Definición de los objetivos

La **hipótesis** de este *Trabajo Final de Máster* es que es posible la predicción de accidentes de tráfico en los que esté involucrada la fauna y se puede desarrollar una metodología para ello.

1.3.1. Objetivo principal

En línea con la hipótesis, el **objetivo principal** de este proyecto es **analizar los accidentes de tráfico** en vías interurbanas de España que hayan contado con la presencia de animales y desarrollar un **modelo predictivo** que identifique zonas de riesgo para la seguridad vial con fauna implicada en función del contexto.

1.3.2. Objetivos secundarios

Para conseguir el objetivo final, se establecen los siguientes objetivos parciales:

- El **análisis y comparación** de las investigaciones realizadas hasta la fecha sobre los accidentes de tráfico con animales involucrados, incluidas las técnicas y algoritmos empleados. En este **estado del arte** es clave extraer el mayor conocimiento posible sobre cómo se han abordado los proyectos previos para poder adaptar los éxitos y evitar los fracasos de las aproximaciones anteriores.
- La **creación del conjunto de datos** a partir de la integración de diferentes fuentes de datos abiertos y su posterior preprocesado hasta obtener un *dataset* óptimo para crear los modelos.
- La **generación de varios modelos** atendiendo a las características específicas del proyecto, que se ajusten y comparen para evaluar su eficiencia, así como el análisis de los diferentes **vectores** que les afectan, atendiendo especialmente a su componente **espacio-temporal** debido al tipo de problema que se pretende resolver.
- La **transferencia a la sociedad** del conocimiento generado mediante la **entrega del modelo predictivo** y la preparación de los ***datasets resultantes*** para implementarlo en la plataforma nacional de tráfico **DGT 3.0** [8]. Además, este objetivo incluye la puesta de los resultados y todos desarrollos al servicio de la comunidad científica. En este sentido, será de vital importancia que el proyecto sea **reproducible**.

1.4. Impacto del proyecto: sostenibilidad, ético-social y diversidad

A lo largo de todo este proyecto de investigación se actuará de manera honesta, ética, sostenible, socialmente responsable y respetuosa con los derechos humanos y la diversidad, tal y como marca la **competencia de compromiso ético y global (CCEG)** con la que está comprometida la UOC [9]. De hecho, la solución que se desarrollará tiene un claro compromiso de responsabilidad social y es el principal motivo de esta propuesta, como ya se ha abordado ampliamente en puntos anteriores.

Del mismo modo, aunque la perspectiva de **género** no se trata de manera directa en este TFM, es cierto que se consideran todas las aportaciones con independencia del género de la persona que las haya producido, al igual que el modelo y la documentación resultante serán

adecuados para cualquier tipo de usuario o usuaria y **no se reforzarán los rasgos ni estereotipos** de ningún tipo, no solamente de género.

En cualquier caso, dada la envergadura del proyecto y el compromiso adquirido al proponerlo, este punto merece una reflexión profunda contextualizándolo en el marco de los **Objetivos de Desarrollo Sostenible** (ODS) [10], cuyas conclusiones se exponen a continuación:

- **Dimensión de sostenibilidad.** El presente trabajo se alinea con el principio de seguridad sostenible que se ve reflejado en el número 1 de la Estrategia de Seguridad Vial 2030 [11]. El objetivo que persigue es el de “defender el derecho de la ciudadanía a moverse en unas condiciones de movilidad adecuada y segura, con el mínimo impacto ambiental posible”, lo que también está en consonancia con el ODS número 9, es decir, construir infraestructuras resilientes, promover la industrialización sostenible y fomentar la innovación, y número 11, entendido como lograr que las ciudades sean más inclusivas, seguras, resilientes y sostenibles.

En contrapartida, debido a la energía que se necesitará durante todo el proyecto y los posibles desplazamientos que serán requeridos, durante el desarrollo del TFM se impactará en el medio ambiente y el propio Ministerio para la Transición Ecológica y el Reto Demográfico facilita una calculadora con una metodología [12] que permite tomar conciencia de la huella ecológica del mismo. Por ello, durante todo el trabajo se hará un especial esfuerzo por economizar el coste computacional de los cálculos y tomar decisiones responsables, como es la de establecer contactos por videoconferencia en lugar de acudir presencialmente a las reuniones o, en su defecto, optar por el transporte público.

- **Dimensión de comportamiento ético y de responsabilidad social (RS).** También expuesto con anterioridad, este proyecto es muy ambicioso en cuanto al impacto positivo que puede suponer para la seguridad de las personas, tanto en el ámbito privado, relacionado con el ODS 3 que tiene por objetivo garantizar una vida sana y promover el bienestar para todos en todas las edades, como en la prevención de riesgos laborales, es decir, el ODS 8, que promueve el crecimiento económico inclusivo y sostenible, el empleo y el trabajo decente para todos.
- **Dimensión de diversidad, género y derechos humanos.** El primer principio de la Estrategia de Seguridad Vial 2030 [11] también establece la necesidad de integrarla en otras políticas públicas (como transportes, movilidad, salud, medioambiente, agenda urbana, igualdad de género, equidad, educación, empleo, seguridad laboral e industria), de modo que todas se complementen, y es en este sentido en el que el proyecto tiene la pretensión de impactar positivamente en esta dimensión. Así, al igual que en la Estrategia,

en este trabajo se prestará una especial atención al desarrollo de la visión de perspectiva de género en las decisiones en favor de la seguridad vial (el ODS 5 propone lograr la igualdad entre los géneros y empoderar a todas las mujeres y las niñas) y a cómo las aportaciones que la mejora de la seguridad vial puede proporcionar en favor de la disminución de las desigualdades económicas y sociales, íntimamente relacionada con el ODS 10, que se centra en reducir la desigualdad en y entre los países.

Además, este proyecto está muy condicionado por el ODS 17, consistente en la necesidad de **alianzas para conseguir los objetivos**. Al mismo tiempo, se enmarca en el principio 1 de la Estrategia de la DGT bajo el epígrafe de *responsabilidad compartida*, que promueve entre todos los actores “el enfoque integrado de seguridad vial y comprender que los objetivos sólo se podrán conseguir con la implicación de todos”. Como ya se ha especificado, en este TFM se trabajará teniendo a la DGT como cliente sin contraprestación de ningún tipo porque el principal objetivo es contribuir a la seguridad vial, y con ello, a la mejora del bienestar social, teniendo en cuenta todas las dimensiones ya mencionadas.

1.5. Descripción de la metodología

El proyecto que se desarrolla en este *Trabajo Final de Máster* sobre accidentalidad de tráfico sigue todo el ciclo de vida de los datos, desde su obtención y preprocesado hasta la publicación y producción de los resultados. En este contexto, tal y como ocurre en la mayoría de proyectos de minería de datos, *machine learning* y otros trabajos analíticos [13][14], una metodología que se adapta a las necesidades de esta investigación es la llamada ***Cross Industry Standard Process for Data Mining* (CRISP-DM)**, que busca el compromiso de ejecutar un proyecto de calidad. Se ha seleccionado esta y no otras, como las ágiles [15], debido a que tiene en cuenta los **requerimientos iterativos** con ciclos pequeños de planificación, ejecución y revisión, **respetando sus diferentes etapas** sin necesidad de que en cada iteración el resultado deba ser un producto.

En la Figura 1.2 se puede observar el diagrama que recoge las **seis fases** en las que se divide todo proyecto siguiendo la metodología elegida. Estas etapas, que no deben ser consideradas como compartimentos estancos, sino que se puede estar trabajando en más de una a la vez, avanzar o retroceder dependiendo de las necesidades del momento, se desarrollan a continuación:

1. **Entender el negocio.** Se trata de la fase que se trabaja en este capítulo, en la que se han definido los objetivos y requerimientos, así como descrito y estructurado las próximas etapas. Para ello, también se ha recopilado la información necesaria de la DGT a través del Observatorio Nacional de Seguridad Vial y se ha estudiado la bibliografía relacionada con la problemática.

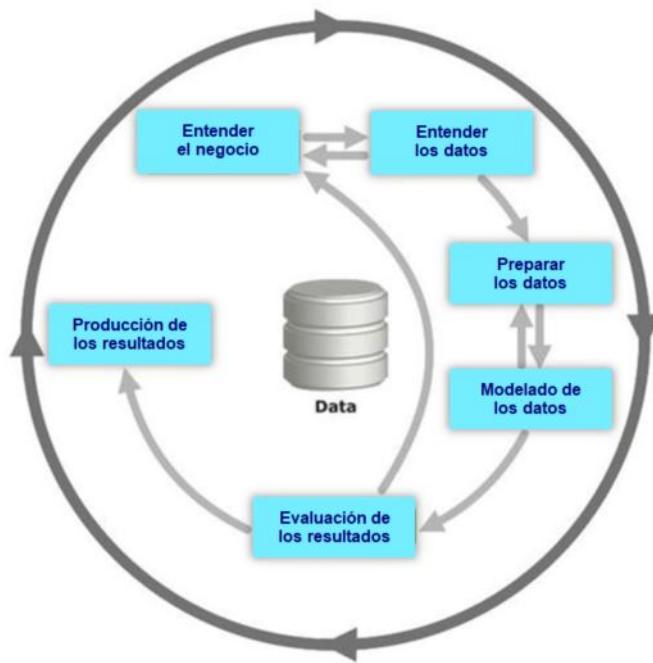


Figura 1.2: Fases de la metodología CRISP-DM.

Fuente: Dipanjan Sarkar, Raghav Bali y Tamoghna Ghosh. Elaboración propia.

2. **Entender los datos.** Es una etapa crítica en la que se identifican, recogen y analizan los datos, profundizando en su conocimiento. Así, los datos recogidos se exploran y visualizan para identificar patrones ocultos o posibles problemas e inconsistencias, entre otros. En este caso, se llevará a cabo una labor de recopilación y se trabajará con datos de las fuentes principales, como los de la DGT, y de otras de contexto, como OpenStreetMap (OSM) y Global Biodiversity Information Facility (GBIF). Además, se realizarán entrevistas con el personal de la DGT para incorporar su conocimiento experto no recogido en los *datasets*.
3. **Preparar los datos.** Una vez se tienen todos los datos, este es el momento de preprocesarlos con el objetivo de aumentar su calidad, limpiándolos, creando nuevos atributos o corrigiendo los problemas de formato que puedan presentar. En este proyecto, además, se desarrollarán las herramientas necesarias para poder georreferenciar los accidentes de tráfico y se extraerá la geometría de carreteras, sus atributos e información de la fauna existente por localizaciones.
4. **Modelado de los datos.** El objetivo de esta fase es el de obtener un modelo de datos que se ajuste a los objetivos iniciales. Por ello, en este proyecto se diseñará un modelo, con un flujo de entrada, un análisis y un flujo de salida, y se dividirán los conjuntos de datos en entrenamiento y *test*.

5. **Evaluación de los resultados.** En esta etapa se comparan los modelos resultantes entre sí según la métrica que se haya establecido previamente y la capacidad de dicho modelo de generalizar correctamente, repitiendo la fase de modelado de datos si fuese necesario. Por las características concretas de esta propuesta, se prevé que los datos resultantes de la evaluación sean validados, además, por el personal de la DGT.
6. **Producción de los resultados.** Es la última fase del proyecto en la que se presentan los resultados y las conclusiones sobre los mismos. En este proyecto, una vez validado el modelo, se hará entrega de los conjuntos de datos y cartografía que se consensúe con el personal técnico de DGT 3.0 para que puedan implementar una solución basada en este material que se integre dentro de la lógica de la plataforma. Además, se generará un enlace con un código DOI dentro de la plataforma Zenodo o se publicará el *dataset* en CKAN, dependiendo de las decisiones tomadas. Asimismo, se creará un repositorio en GitHub y un proyecto en RPubs con una licencia abierta para poner a disposición de la comunidad los resultados obtenidos.

Por último, aunque sale del alcance de este TFM, esta prevista una fase operativa de despliegue y revisión posterior del proyecto.

1.6. Planificación del proyecto

Para la planificación de este proyecto se ha de tener en cuenta que se enmarca dentro de un programa académico en el que se deben cumplir con unas fechas límite de entrega de diferentes **Pruebas de Evaluación Continua** (PEC), por lo que esos requerimientos temporales marcan los hitos del mismo. En cualquier caso, y teniendo esto en consideración, las tareas en las que se ha dividido el TFM están siempre orientadas a cumplir con los objetivos establecidos.

De este modo, las tareas que deben ser completadas para finalizar con éxito este TFM se dividen en **cinco bloques principales**, en los que las fechas finales coinciden con los plazos marcados desde la universidad¹. En este sentido, la redacción de la memoria del proyecto se plantea como una acción continua que se realizará en paralelo con el desarrollo del proyecto.

En la Figura 1.3 se presenta el **diagrama de Gantt** en el que se recogen las tareas, algunas de ellas ya mencionadas en el apartado de [1.5 Descripción de la metodología](#), y los tiempos asignados² a cada una de ellas dentro de sus bloques correspondientes.

¹La fecha de defensa pública provisional se ha establecido el 3 de febrero a la espera de la asignación oficial.

²Por las características de la vida laboral que se compatibiliza con este proyecto, en la planificación los fines de semana no son días hábiles de trabajo ni de entrega.

1.6. Planificación del proyecto

9

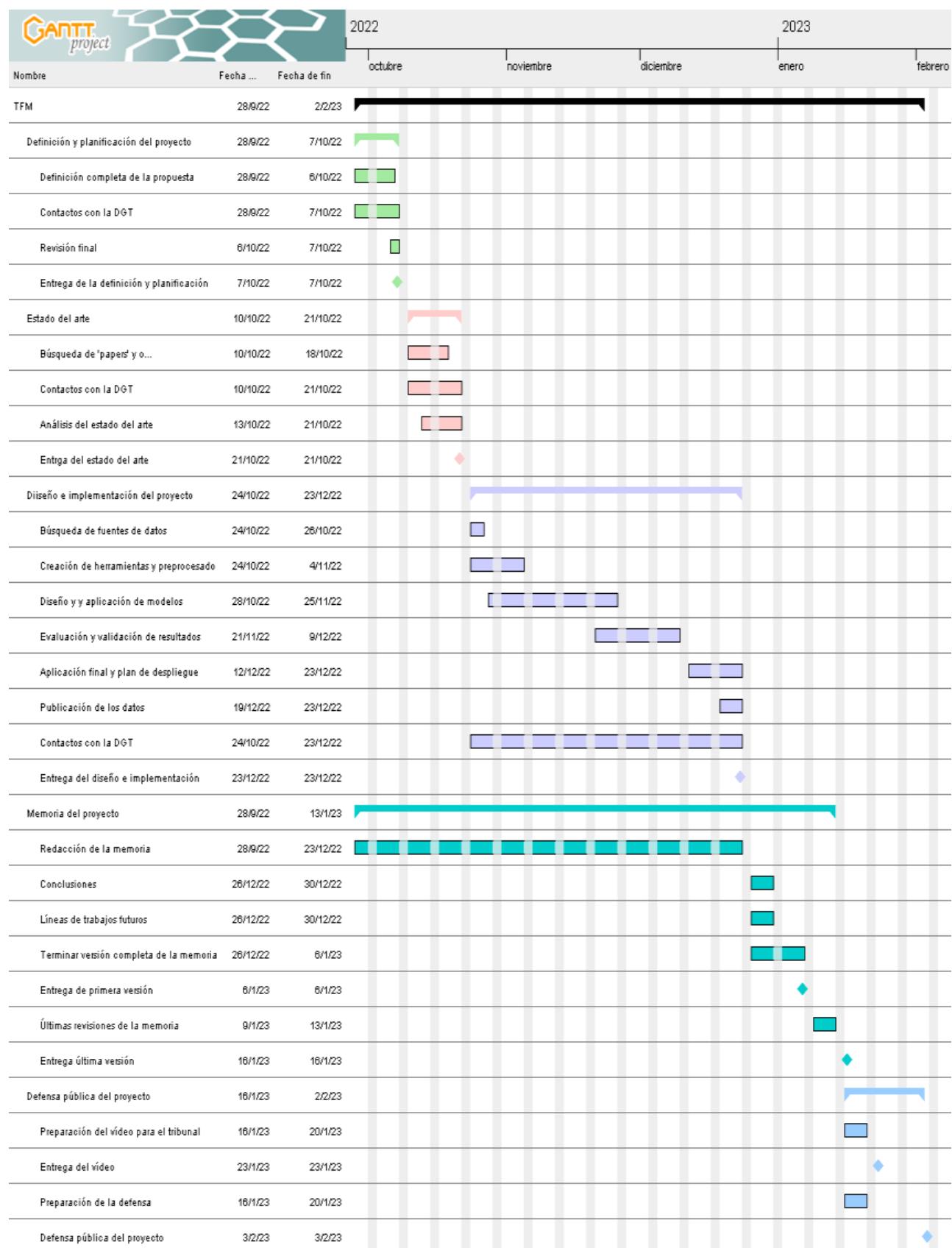


Figura 1.3: Diagrama Gantt con la planificación del proyecto.
Fuente: elaboración propia.

Capítulo 2

Estado del arte

La bibliografía y trabajos sobre accidentes de tráfico con animales implicados es muy abundante y aborda el problema desde diferentes perspectivas. Por tanto, como parte esencial de este *Trabajo Final de Máster*, en este capítulo se presenta un **estudio del estado del arte**, en el que se expone la información más relevante en relación con la investigación que se va a desarrollar y los objetivos marcados en el proyecto.

Como contexto, se ha de tener presente que las investigaciones analizadas se pueden dividir en dos tipos: aquellas publicaciones en las que se hace una **revisión bibliográfica** a modo de *review* [16] [17] y los **trabajos de campo** sobre los accidentes de tráfico con fauna implicada.

Debido a las características de esta área temática, en la Figura 2.1 se representan las fases que pueden incluir las investigaciones del segundo tipo, es decir, aquellas en las que se **analizan los datos** [6] [18] [19] [20] [21] [22] [23] [24] y las que dan un paso más y, además, los **modelizan** [25] [26]. Todas las fases de dicha figura son consecutivas y, debido al alcance de este proyecto, no se han estudiado trabajos que tuvieran como finalidad la implementación de nuevas medidas y la evaluación de las mismas.

Así, este estado del arte se divide en tres grandes bloques:

1. **Características de los datos.** Se analiza la recogida de los datos y sus características originales en las diferentes investigaciones, es decir, la primera fase.
2. **Metodología de análisis y modelado.** Se estudian las diferentes estrategias de agrupamiento, análisis y modelado de las investigaciones, es decir, las segunda, tercera y cuarta fases.
3. **Hallazgos y conclusiones.** Se extraen los hallazgos de las diferentes investigaciones estudiadas y se sintetizan a modo de conclusiones teniendo en cuenta los objetivos del presente proyecto.

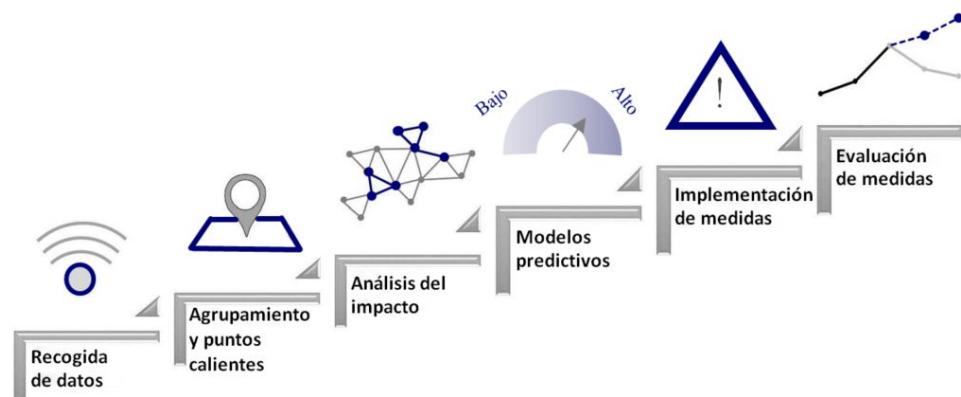


Figura 2.1: Posibles fases de las investigaciones de accidentes de tráfico con animales implicados.

Fuente: Raphaela Pagan. Adaptación y elaboración propia.

2.1. Características de los datos

2.1.1. Ámbito geográfico

En **España**, país objeto de estudio en este *Trabajo Final de Máster*, la mayoría de las investigaciones se centran en **áreas geográficas específicas**, sobre todo el noroeste [22] debido a que son análisis de la Universidad de Santiago de Compostela financiados por la Xunta de Galicia, aunque hay excepciones [25] [26]. Además, en esta clasificación también se encuentran aquellos trabajos en los que solo se analiza la accidentalidad por tipo de animales específicos, como el lobo en Castilla y León [25] o los jabalíes y corzos en Galicia [27].

Hay que buscar en estudios **fuerza de España** si se quieren encontrar proyectos que traten el problema con datos estadísticos que abarquen **zonas más amplias**, como todo un país completo [6] [17] [18] [19] [23] o una extensión de grandes dimensiones. En este último caso, destaca el centrado en una región de Australia con una extensión en torno a los 660.000 km² [20] frente a los 505.983 km² de España [28].

También es interesante el análisis de investigaciones que optan por poner el **zoom espacial**, lo cual se aleja de este proyecto pero se han de tener en cuenta porque se pueden encontrar soluciones válidas. En este sentido, existen trabajos centrados en tramos de carretera completos, accidentes geográficos [24] o incluso en grandes ciudades [21], aunque este tipo de trabajos son menos comunes debido a la frecuencia menor de colisiones entre animales y vehículos en esos espacios, así como a la menor gravedad de los mismos por una menor velocidad, factor [16] que se verá más adelante.

2.1.2. Acotación temporal

La extensión temporal que abarcan las investigaciones varía mucho entre ellas y, en general, cuanto **mayor es la horquilla analizada, más precisos** son los resultados en el caso de los estudios que incluyen modelos predictivos [16] [26].

El estudio que más años comprende en **España** es el de la provincia de Soria [26], con una serie histórica entre enero de 1988 y febrero de 2001, aunque solo se recogen el 63 % de las colisiones. También en España, la investigación centrada en lobos en Castilla y León [25] se acota entre 2001 y 2007 y un lustro (2006-2010) es el que igualmente se estudia en el de jabalíes y corzos en Lugo [27]. Mucho menos extenso en el tiempo se encuentra el de Lugo general [22], con los datos de 2006 y 2007 en la provincia.

En el **ámbito internacional**, el estudio **más extenso** en el tiempo [18] analiza tendencias entre 1990 y 2008 y el que le sigue de cerca entre 1990 y 2004 [17] analiza las consecuencias y cómo se pueden evitar estos siniestros.

Ya como estudios **menos amplios**, se pueden destacar el de Lituania [19] entre 2014 y 2018 y el de Lublin [21] (novena ciudad más grande de Polonia) entre 2009 y 2012 [21]. También en Polonia se sitúa el estudio [6] más actual entre los más completos que se han publicado sobre esta problemática, con registros de entre 2016 y 2020.

Por último, el ejemplo **más particular** es de tres años, entre septiembre de 1997 y agosto de 2000 [24], aunque en este caso de manera discontinua puesto que no se tienen en cuenta los casos entre diciembre y febrero por las condiciones climatológicas. El **mínimo temporal**, por su parte, se encuentra en un año, concretamente 2010 [23]. Además, se ha de subrayar que uno de los estudios [20] no indica la delimitación temporal de la recogida de datos, aunque por las referencias se puede intuir que es en 2007.

2.1.3. Fuentes de los accidentes y tipo de fauna

En la mayoría de las investigaciones analizadas, los datos de los accidentes de tráfico con animales involucrados provienen de **fuentes oficiales**. El elemento común más importante es que en todos los registros a excepción de un trabajo [24] se tiene, como mínimo, la información de **lugar, fecha y hora**. Sin embargo, sí que existen claras diferencias entre las investigaciones que discriminan y estudian solo accidentes con **algunas especies** implicadas y aquellas en las que **no se hace esta criba** inicial.

Como en apartados anteriores, debido a los objetivos del proyecto, es interesante extenderse más en los trabajos que tratan datos de **España** aunque se hayan analizado todos en profundidad. Por un lado, se encuentran aquellas investigaciones en las que se emplean datos oficiales, aunque no publicados, como la de Soria [26] que hace uso de la información de la Jefatura

Provincial de Tráfico (que depende del Ministerio del Interior). Se trata de un total de 2.067 registros atendiendo a las **tres especies** en las que están implicadas en el 98 % de las colisiones, concretamente el 38 % de corzo capreolus capreolus (Linnaeus, 1758) [29], el 35 % de jabalí sus scrofa (Linnaeus, 1758) [30] y el 25 % de ciervo cervus elaphus (Linnaeus, 1758) [31]. También en Castilla y León, el de la comunidad autónoma completa es un trabajo mucho más reducido [25] porque solo trabaja con los datos de 82 atropellos a **lobos**, información que se extrae de informes presentados por las autoridades de seguridad vial y datos de la Sección de Espacios Naturales y Especies Protegidas de la Junta de Castilla y León.

En una de las investigaciones en Lugo [22] se centran en las mismas **tres especies** de animales analizadas en la de Soria [26], por lo que solo se trabaja el 75,4 % de los 377 accidentes registrados en los que había animales involucrados. También en Lugo [27] se encuentra el único estudio que no indica claramente el origen de los datos [27], aunque se da a entender que los 3.037 accidentes analizados proceden de los registros oficiales de la Xunta de Galicia.

Fuera de España se puede encontrar el **caso más completo** que existe [17] en cuanto al origen de datos de los accidentes, ya que emplea las estadísticas de accidentes en Estados Unidos (de la Policía y de las patrullas de carretera), el recuento de cadáveres en carretera, la información sobre siniestros de la industria aseguradora y entrevistas con testigos, independientemente del animal involucrado. En este mismo país, se ubica otra investigación completa [18] con datos de todos los accidentes registrados en el Sistema de Informes de Análisis de Fatalidad (FARS, por sus siglas en inglés) de la Administración Nacional de Seguridad Vial (NHTSA, por sus siglas en inglés) y del Sistema de Estimaciones Generales (GES, por sus siglas en inglés) de Estados Unidos. Todos los estudios en Polonia [6] [23], por su parte, emplean datos policiales recogiendo hasta 128.878 colisiones. Mientras, en Lituania [19] los 14.427 registros proceden de fuentes oficiales de Tráfico y **divide a las especies** entre salvajes, domésticos o no identificados.

Otra investigación que hace un esfuerzo especial en la recogida de datos similar a la ya mencionada [17] se ubica en Australia [21], puesto que la detección de **deficiencias de la información oficial** se intenta suplir con datos de otros orígenes, como parques nacionales, oficinas forestales y asociaciones de cazadores, dando lugar a una base de datos de 930 accidentes con todos los animales divididos en tres grupos (mamíferos grandes, mamíferos pequeños y murciélagos).

La **investigación más restrictiva** de todas [20], por otro lado, analiza los datos procedentes de una fuente oficial (Rural and Remote Road Safety Study), pero delimita los accidentes únicamente a aquellos en los que haya habido muertos o heridos graves mayores de 16 años, definiéndolos como los que necesitan más de 24 horas de hospitalización. Los datos de estas 532 colisiones graves que dieron lugar a 600 víctimas humanas se complementan, además, con entre-

vistas personales a 33 de las víctimas que sobrevivieron y datos procedentes de cinco hospitales, siendo la única publicación que amplía la información disponible de este modo.

Por último, destaca uno de los casos [24], en el que la recogida de los datos se hizo de **forma directa** mediante el muestreo de las carreteras en los días sin nieve entre los meses de abril y noviembre, centrando la atención específicamente en la **fauna pequeña**, definida como “especies de vertebrados terrestres del tamaño del coyote canis latrans y más pequeños”.

2.2. Metodología de análisis y modelado

Llegados a este punto del proyecto, está claro que el problema es de **clasificación supervisada**, debido a que se conoce la ‘solución’ que se desea predecir. Concretamente, el objetivo es el de desarrollar un **modelo que identifique zonas de riesgo** para la seguridad vial con fauna implicada, por lo que la clase, *target* o *label* a predecir, que se denomina diferente dependiendo del manual [32] [33], indica precisamente si el registro es el de un accidente con animal.

Es por este motivo por el que las investigaciones analizadas en este estado del arte que llegan a la fase de modelado hacen uso de técnicas de *supervised machine learning* o **aprendizaje automático supervisado** y conviene recoger cuáles son las más empleadas en proyectos similares al que se desarrolla en este TFM. Antes de llegar a ese punto, se estudia el **preprocesado** de los datos y los **análisis** que se realizan en las investigaciones que no llegan a la fase de modelaje para, por último, estudiar los que sí que han llegado hasta un modelo predictivo. Así, se podrán potenciar las posibilidades de éxito y se evitarán aproximaciones que no hayan tenido resultados óptimos con anterioridad.

2.2.1. Preprocesado y test previos

En primer lugar, todos los trabajos realizan una **limpieza de datos** que se explicita en mayor o menor medida, como la eliminación de aquellos registros no válidos porque, por ejemplo, no están dentro del área geográfica de estudio [19]. En cualquier caso, se trata de un preprocesado sin diferencias con lo que se aconseja en los manuales de la temática [32] [33] [34] [35], por lo que no es necesario extenderse.

Antes de trabajar con los datos, muchas de las investigaciones especifican que **comprueban su normalidad** mediante pruebas estadísticas como la de Shapiro-Wilk [27], o su simplificación Shapiro-Francia [6]. Alternativamente, también se emplean otros test [6] para reforzar las conclusiones obtenidas, como las pruebas de Anderson-Darling (AD), de Cramér-von Mises (CvM) o de Lilliefors. Asimismo, evalúan si se da **homocedasticidad**, es decir, homogeneidad de las varianzas mediante pruebas como el test de Levene [27]. La verificación de la hipótesis se realiza utilizando la prueba no paramétrica de Wilcoxon-Mann-Whitney [6].

No obstante, como la normalidad y la homocedasticidad de los datos son **requisitos previos** [33] para emplear algunas de las pruebas estadísticas necesarias en el análisis exploratorio, que no se explice en la metodología empleada no permite suponer que estas condiciones no hayan sido evaluadas en cada uno de los trabajos.

2.2.2. Análisis espacio-temporal

Dada la **temporalidad** de los datos de los accidentes en los *dataset* sobre esta problemática, en todas las investigaciones se aborda esta dimensión de una manera u otra. Mientras que unas optan por una **aproximación** mediante gráficos de frecuencia y de araña que permiten identificar una cierta estacionalidad [27], otros añaden o emplean alternativamente una prueba de Tukey junto con el análisis de la varianza (ANOVA) [27] para encontrar **diferencias estadísticamente significativas** entre temporadas, días de la semana o momentos del día.

Otra forma [18] de abordar el análisis temporal es en la que la proporción de todos los accidentes mortales y no mortales se calcula por hora y por mes para permitir **comparaciones directas** entre los dos tipos de accidentes. Para el análisis estacional, la proporción representa el número de colisiones ocurridas en un mes seleccionado, dividido por todas las colisiones de ese tipo en todos los meses.

La otra gran dimensión de los datos es la **espacial**, por lo que la mayoría de las investigaciones analizadas emplean **métodos para clusterizar** los datos de manera que se facilite su posterior introducción en los modelos. El algoritmo más empleado es el método de Monte-carlo [26], algunos de ellos combinando estadísticos como Nearest Neighbour Distance (NND) para posibilitar la determinación de los conjuntos de puntos con una agregación significativa no aleatoria [27].

Asimismo, algunos de los estudios [27] aplican un análisis con una **estimación de densidad de Kernel** (KDE, por sus siglas en inglés) cuando se encuentra una agregación espacio-temporal significativa con el objetivo de elaborar mapas de trama de densidad de puntos. En esos casos, hay que tener en cuenta en el momento de la representación que, debido al sesgo bidimensional del cálculo de la KDE, las áreas son mayores de lo que deberían ser en la representación de los *clusters*. Este método estadístico de **análisis de puntos calientes** se plantea en otros trabajos [19] con pruebas estadísticas para determinar si el uso del suelo está asociado o no a las colisiones con animales, por ejemplo. Además, varios estudios determinan los puntos calientes de accidentes espacialmente y diariamente, semanalmente o estacionalmente [16].

Uno de los proyectos [18] **simplifica** este análisis espacial en extremo. Todos los accidentes de tráfico con fauna involucrada se agregan en los 19 años de periodo de análisis, agrupando los datos por estados y clasificándolos según el recuento absoluto de accidentes con animales, el recuento del total de accidentes y la proporción global de estos siniestros con animales.

También de forma muy simplificada, otras investigaciones [21] realizan un **análisis descriptivo** de las zonas en las que más accidentes se han producido, cotejando las ubicaciones con las características de esa localización en el momento del suceso, como una zona en obras o una nueva carretera.

2.2.3. Variables significativas: test y regresiones

La búsqueda de las **variables significativas** o que más pueden influir en los accidentes de tráfico con animales involucrados es un elemento común en todos los trabajos.

Con este objetivo, en ocasiones no muy explícito, las investigaciones hacen uso de **pruebas estadísticas**, como la de χ^2 de Pearson [6]. En uno de los análisis más sencillos que se plantea [20] solo se hace una **descripción** de los datos recogidos, extrayendo estadísticas como las condiciones lumínicas o el tipo de conductor pero sin profundizar, únicos casos en los que sí que se realiza un test de significancia.

En otra aproximación algo más interesante [6], se emplea el **coeficiente de correlación de Pearson o Spearman**, dependiendo de la normalidad de los datos previamente analizada, para evaluar si existe una correlación estadísticamente significativa entre el número de choques y otras 9 variables, como el día de la semana, la hora, la especie, etc., con un nivel de significación de 0,05.

Como un paso más en profundidad para identificar cuáles de las variables predictoras seleccionadas difieren entre las distribuciones de puntos de atropello y las aleatorias, algunos de los estudios [25] han desarrollado una **comparación de medias univariante** utilizando una prueba t de muestras no pareadas. Para las variables categóricas se utilizan, además, regresiones logísticas binarias. Así, las variables con diferencias significativas entre ambas distribuciones ($\alpha > 0,1$) se incluyen en los modelos posteriores.

Otra opción que va más allá en la exploración de estas variables con mayor peso en los accidentes de tráfico con fauna es la generación de **modelos logísticos** por pasos con correcciones de eventos raros y el cálculo de **correlaciones bivariadas** para detectar efectos de multilinealidad no deseados. Esto permite la eliminación de una de las variables cuando la correlación es superior a un porcentaje, como 70 % [25], seleccionando la variable con un estadístico t más bajo en la prueba univariante para su eliminación.

En el estudio analizado más extenso en el tiempo [18], se realizan **análisis de regresión por separado** sobre los recuentos de colisiones mortales, las tasas de accidentes y las proporciones de colisiones utilizando el año como predictor durante el periodo de 19 años. Las probabilidades de que se produzca una colisión en la oscuridad se modela primero en una regresión logística que examina la influencia del límite de velocidad en las probabilidades de una colisión mortal en la oscuridad y, después, según su gravedad.

Por último, aunque el objetivo de [24] es exclusivamente animalista, es interesante el empleo que hace de la **regresión logística mediante estimaciones de máxima verosimilitud** para predecir la probabilidad de ocurrencia de los atropellos en función de las variables del paisaje y de la carretera. De este modo, las variables introducidas al modelo describen atributos específicos de la localización del accidente y son la distancia media a la vegetación, la distancia a la estructura de cruce de fauna o alcantarilla de drenaje más cercana, la distancia a la población más cercana, la distancia a la fuente de agua más cercana, el tipo de hábitat y la topografía de la carretera.

Así, se puede confirmar si variables como el tráfico, uso de la tierra, paisaje, carretera, infraestructura a lo largo de la carretera, clima y aspectos temporales [16] realmente influyen en los tipos de accidentes analizados, pero **no permiten predecirlos**, lo cual se verá en el siguiente apartado.

2.2.4. Modelización predictiva: regresión logística

Los enfoques de **aprendizaje automático rara vez se prueban** para la investigación de accidentalidad de tráfico con fauna involucrada. Mientras que las estadísticas comunes, como la regresión o el análisis de componentes tienen por objetivo principal encontrar vínculos entre variables, el aprendizaje automático se centra en la capacidad de predicción previa de los algoritmos [16]. En las pocas investigaciones que han llegado hasta este punto, de hecho, solo se han empleado **regresiones logísticas**. En todos los casos se justifica esta decisión en que las variables empleadas son continuas y categóricas.

El caso más interesante con este modelo es el realizado en la provincia de Soria [26], debido al **éxito de los resultados y a la similitud de los datos** empleados con los que se tendrán para el presente proyecto. Antes de realizar la regresión logística, se definen los **tramos de carreteras con alta siniestralidad** mediante la detección de agrupaciones de lugares de colisión de animales comparando el patrón espacial de colisiones con el esperado en una situación aleatoria, en cuyo caso la probabilidad de colisiones para cada tramo de carretera mostraría una distribución de Poisson [36]. Sin embargo, la falta de datos sobre la intensidad de tráfico en la provincia provoca que esta variable no se tenga en cuenta, lo que se suple con información de cobertura forestal (tipos de bosque, matorrales, praderas, cultivos, ríos y presas, urbanizados e improductivos), además de información recogida directamente en el entorno, como la presencia de vallas.

En concreto, en este estudio español [26], los modelos predictivos para la localización de tramos/puntos con y sin colisiones se generan mediante una **regresión logística binaria** y posteriormente se validan con datos independientes. Así, se ajustan dos modelos en cada análisis: uno completo con todas las variables y otro con la versión reducida que solo incluye las

variables explicativas más significativas. En este sentido, los **modelos completos** únicamente se utilizan como referencia para mostrar la importancia relativa de los diferentes predictores, mientras que los **modelos reducidos** pueden considerarse como las mejores herramientas estadísticas de predicción que pueden producirse con un número mínimo de parámetros [37].

También en España se localiza la segunda investigación que trata de crear un modelo predictivo de los datos [25], concretamente en Castilla y León y emplea la misma metodología, dividiendo el conjunto de datos inicial en un 80 % para el entrenamiento y 20 % de *test*. En concreto, se evalúan diferentes modelos para caracterizar las localizaciones de colisión utilizando **regresiones logísticas con correcciones por eventos raros**, como se ha explicado en el paso previo del apartado anterior, dando como conclusión que los mejores modelos incluyen parámetros de tráfico y perturbaciones humanas. Para ello, se generan varios modelos con los diferentes tipos de variables que podrían explicar la distribución espacial de las colisiones entre lobos y vehículos: características del tráfico y de la carretera, paisaje circundante o grado de perturbación por elementos humanos. Cada modelo se prueba primero por separado y, posteriormente, todas las combinaciones posibles entre ellos. El nivel de significación para entrar en los modelos es de 0,05. El modelo final se obtiene utilizando el **criterio de información de Akaike (AIC)**, que es una medida de la bondad de ajuste escogida como el mejor método para seleccionar qué variables deben incluirse o excluirse en los modelos. Este criterio considera tanto el ajuste como la complejidad y permite la comparación simultánea entre varios modelos. Por último, el valor predictivo de los modelos se evalúa mediante una **curva ROC** (Receiver Operating Characteristic) en el subconjunto de datos de *test*.

2.2.5. Tecnologías

Para la implementación de las metodologías descritas, en todas las investigaciones se han hecho uso de diferentes tecnologías, entre las que destacan las siguientes:

- **Lenguajes de programación** como Python, R [6] [25], la plataforma de *software* estadístico SPSS [26] en una de las investigaciones o, también solo uno de los casos estudiados, AvenueTM [24].
- **Softwares de Sistemas de Información Geográfica (SIG)** [19], como ArcView[®] [24] [26] [25] [27], ArcMap [19] y ArcGIS [21], las tres propiedad de la empresa ESRI, o QGIS [6], de código abierto.
- Uso de **bases de datos con soporte geoespacial**, no mencionadas explícitamente en las investigaciones pero sin las cuales no se podría haber desarrollado ninguna de ellas.

2.3. Hallazgos y conclusiones

Teniendo en cuenta los objetivos de este proyecto, es interesante cerrar el estado del arte con la extracción de los hallazgos y conocimiento más relevantes de las investigaciones realizadas hasta la fecha sobre los accidentes de tráfico con animales implicados debido a que existen conclusiones similares analizando datos de diferentes países. Con la siguiente sintetización, además, se facilita la reutilización de esta información en las próximas fases de este *Trabajo Final de Máster*:

- El problema de la **infranotificación de accidentes viales con animales** involucrados constituye una de las limitaciones de los estudios [6] [20] y es difícil estimar el número de colisiones no declaradas pero podría ser significativo [25], lo que provoca que estas investigaciones sean difíciles metodica y logísticamente [21]. En Estados Unidos, por ejemplo, las bases de datos suelen excluir los accidentes con daños materiales inferiores a 1.000 dólares, no todos los conductores informan de las colisiones con animales y no todas las fuerzas del orden, los organismos naturales o los agentes de seguridad registran los accidentes [17]. En consecuencia, es probable que las colisiones con animales pequeños no se notifiquen [17]. Además, los datos pueden estar en ocasiones **mal georreferenciados**, lo que genera distorsiones en los resultados [21]. Por todo ello, es necesario disponer de datos fiables que permitan realizar análisis multivariantes [27].
- La decisión de evaluar un gran número de variables explicativas potencialmente interrelacionadas permite **encontrar los mejores predictores** posibles de los factores hipotéticamente vinculados a las colisiones [26]. Asimismo, es importante analizar cada una de las variables en un modelo completo para determinar cuáles son los predictores más óptimos para un modelo final [26].
- Como las colisiones tienden a **agruparse en el espacio** [16] [19], el método estadístico de **análisis de puntos calientes** es una herramienta valiosa para identificar las secciones de las carreteras más asociadas con colisiones entre vehículos y vida silvestre, pero es mucho menos sensible para especies raras [19]. En cualquier caso, en algunas ocasiones la **diversidad del hábitat** también es algo mayor en las zonas con más colisiones que en las zonas de bajo riesgo [26], por lo que es razonable investigar si la población de animales de una especie determinada en un área afecta al número de choques viales [6] cada vez que se inicie un estudio de estas características. Para ello, el **desarrollo de cartografía** permite desglosar los picos específicos de accidentes en cada momento particular en un tramo de carretera concreto [27].

- El análisis de puntos calientes puede servir como trabajo básico, pero solo unos pocos estudios tienen como objetivo el **modelado predictivo** [16], por lo que es necesario un **mayor nivel de progreso** en la investigación [16]. Para evaluar estos modelos, tal y como indica también la teoría [32] [33], es necesario dejar **datos independientes fuera** de los mismos que permitan su validación [26]. Como punto positivo, la poca literatura al respecto determina que **es posible** crear modelos que permitan identificar y predecir los puntos calientes de las colisiones con animales en las carreteras [19] [26]. En este sentido, la investigación de Soria [26] predijo correctamente el 74 % de los casos (79,2 % de las secciones de alta colisión y 68,1 % de las de baja colisión), por lo que su metodología se tendrá especialmente en cuenta en este proyecto, aunque deberá ser adaptada a los datos disponibles para el conjunto de España y las características propias del país en su conjunto.
- La presencia de **cruces, pasos subterráneos y guardarrail**s se ha asociado significativamente con la ausencia de colisiones [24] [26]. El vallado también puede ser un predictor clave decisivo y, por ejemplo, los atropellos fueron proporcionalmente menores a lo largo de las autopistas valladas que en las carreteras principales similares que carecían de estos elementos [23] [25].
- Existe una **clara temporalidad** que se puede estudiar tanto por la **época del año**, como en el **día de la semana y momento del día** [16] [17] [22]. Esta diferencia se puede dar en términos generales y también estudiando cada especie por separado [18] [23] [25] [27]. Por ello, es necesario averiguar el patrón de choque temporal particular de **cada especie** para predecir los períodos críticos en los que se producirán estos accidentes en una región específica [27] y que puede estar asociada a varios motivos, como migraciones, épocas de celo de los animales o temporada de caza [21]. Asimismo, se detecta que en la mayoría de los casos estos choques ocurren al amanecer y al anochecer [6] o cuando hay menos luz [18]. Alguna investigación llega a determinar, de hecho, que los **viajes nocturnos** son un factor de riesgo significativo al comparar choques relacionados con animales con otros choques por lesiones graves [20]. Además, las distribuciones estacionales parecen depender en cierta medida de la **zona geográfica** específica [6] [17]. Por último, como era previsible, durante el confinamiento en el periodo inicial de la pandemia de COVID-19 (marzo-abril de 2020), el número total de choques viales de este tipo disminuyó significativamente [6].
- Los requisitos específicos de **velocidad** influyen en la cantidad y gravedad de colisiones con animales [6] [17] [18] [22] [25]. En relación con el punto anterior, las probabilidades de que se produzca una colisión mortal o con heridos en la oscuridad aumentan aproximadamente un 1,5 % por el mismo aumento de la velocidad establecida [18].

- Los datos relativos a las **especies animales afectadas** varían considerablemente según la región [17] o zona [25]. En cualquier caso, la mayoría de los estudios se centran en colisiones con especies representadas como ciervos, alces u otros ungulados [16].
- La probabilidad de que se produzca un accidente con fauna involucrada también podría depender de la **calidad de la carretera** [22] o del **tipo** [19]. Así, el número de choques varía para carreteras de diferentes categorías por razones como el ancho de la carretera o el número de calzadas [6] [16]. Por ejemplo, estos accidentes son más comunes en las carreteras rurales de dos carriles [17].
- En cuanto a la influencia del **nivel del tráfico** existe menos consenso en general [17] y, por ir a un caso concreto, la gran mayoría de los accidentes con jabalíes se sitúan dentro del horario nocturno aunque sea durante el día cuando más coches circulan, mientras que los corzos se reparten de forma más homogénea a lo largo del día [27]. Otros estudios encuentran una clara dependencia entre el volumen del tráfico y los accidentes de este tipo [6] [19] [22].
- Tampoco existe acuerdo entre los que concluyen que las **variables del paisaje** apenas mejoran los modelos [25] y aquellas investigaciones en las que la distancia a ciertos elementos naturales sí que es un factor importante [6].
- La importancia del **uso del suelo** varía también en función de los diferentes estudios [6]. Por un lado, se encuentran aquellos que concluyen que los elementos de uso del suelo y su distribución desempeñan papeles importantes en los modelos [16] [26], sobre todo si se analizan accidentes con especies concretas, que puede ser aplicable en diferentes tipos de carreteras y en países desarrollados y menos desarrollados [19]. Igualmente, algunas investigaciones tienen en cuenta los distritos de caza con resultados positivos en la incorporación de esta variable [6], aunque no existe una relación directa entre el tamaño de la población de especies de caza en particular en un área determinada y el número de choques viales [6]. Por otro lado, hay proyectos en los que ni los usos del suelo ni los índices de estructura del paisaje, independientemente de la escala del análisis, son útiles en los modelos logísticos [25].
- La **meteorología** sí que parece influir [6] en este tipo de accidentes de tráfico. La probabilidad de que se produzcan es mayor cuando el tiempo es seco, tal vez debido a que los animales son menos propensos a moverse durante las inclemencias del tiempo [17].
- En cuanto a la **gravedad**, los accidentes con animales involucrados son menos graves que otros siniestros [17]. No obstante, al no haber testigos supervivientes ni pruebas que

aporten detalles, también es probable que algunas salidas de vía de un solo vehículo que resultan mortales sean el resultado de un intento de la persona que conduce de evitar un animal en la calzada [18].

- La distancia a los **elementos antropogénicos** tiene una correlación significativa positiva con la ocurrencia de colisiones en las que hay animales implicados [16] [25]. Así, la cercanía del límite municipal y la lejanía de los asentamientos humanos se presentan como buenos indicadores [25].
- En la mayoría de accidentes de este tipo solo hay **un vehículo involucrado** [17]. Además, otra de las investigaciones [20] señala una proporción significativamente mayor de motociclistas (51,7 %) en choques graves relacionados con animales en comparación con todos los demás choques graves con lesiones, por lo que el tipo de vehículo también se puede considerar un factor a tener en cuenta [6].
- Como **fuentes de datos externas** que pueden ayudar en el preprocesado y modelado de los datos, se extraen los siguientes como los posibles más útiles para este *Trabajo Final de Máster*: el Visual Road Catalogue para transformar los datos localizados por su mojón en España a su longitud y latitud [22], mapa de volumen de tráfico y velocidad de las diferentes comunidades autónomas o del Ministerio de Fomento, los usos del suelo del Mapa Forestal Nacional realizado a partir de fotografía aérea por el Ministerio de Medio Ambiente español a escala 1:50.000, la pendiente del modelo digital de elevaciones (resolución de 25 m) elaborado por el Instituto Geográfico Nacional de España a partir de curvas de nivel y puntos de elevación contenidos en el Mapa Topográfico Nacional a escala 1:25.000 [25], además de los datos de la red vial obtenidos de OpenStreetMap [6].

Capítulo 3

Diseño y requerimientos

En este capítulo se especifican las fuentes de datos del proyecto y se expone el flujo de trabajo a alto nivel para tener una visión global antes de profundizar en cada fase. Asimismo, se describen las tecnologías empleadas en este *Trabajo Final de Máster*, tanto de terceros como de desarrollo propio, y se exponen los requerimientos de *hardware* para reproducirlo.

3.1. Fuentes de datos

La obtención y captura de datos es uno de los puntos más importantes de este proyecto y del que **depende su éxito** porque dejar fuera del análisis un factor de riesgo puede suponer que el modelo final no cumpla los objetivos, centrándose el mayor esfuerzo en encontrar los **mejores predictores**, al igual que otras investigaciones con buenos resultados [26]. Por ello, se trabaja exhaustivamente en obtener un *dataset* final muy completo y pertinente que permita llegar al modelo más preciso dentro de las posibilidades. Por este motivo, se estudian en profundidad multitud de **fuentes de datos** y cómo extraer la información de cada una de ellas de manera que se puedan integrar para resolver el problema de este proyecto. Se tienen en consideración tanto variables que se emplean en otras investigaciones detalladas en el estado del arte (Capítulo 2) como otras que no se han estudiado hasta la fecha.

Aunque este asunto se trata en profundidad en la captura y obtención de los datos (Capítulo 4), es importante tener en cuenta en esta fase de diseño cuáles son las fuentes de datos seleccionadas entre todas las posibles y qué **conjuntos de datos** en concreto se trabajan de cada una de ellas, como se muestra en la Tabla 3.1.

Además de las fuentes indicadas en la Tabla 3.1, cabe destacar que también se crean nuevas variables de potencial interés mediante la **combinación de más de una fuente de datos**, como ocurre con el caso de la velocidad máxima de la vía, tal y como se ve más adelante.

Fuente	Datos obtenidos
Dirección General de Tráfico (DGT)	1. Ficheros de microdatos de accidentes con víctimas y su diccionario [38] [39] [40] [41] [42] [43] [44] 2. Ficheros de microdatos de accidentes con animales y su diccionario [45] [46] [47] [48] [49] [5] [50] 3. Vías bajo titularidad de la DGT
OpenStreetMap (OSM)	4. Todas las vías de España [51] [52]
Dirección General de Carreteras (DGC)	5. Intensidad media diaria [53] [54] [55] [56] [57] [58]
Dirección General de Desarrollo Rural, Innovación y Política Forestal	6. Mapa Forestal de España [59]
Agencia Estatal de Meteorología (AEMET)	7. Datos meteorológicos [60]
Centro Nacional de Información Geográfica (CNIG)	8. Modelo Digital del Terreno - MDT200 [61] 9. Modelo Digital de Pendientes - MDP05 [62]
Global Biodiversity Information Facility (GBIF)	10. Ficheros de la distribución de cada animal involucrado en accidentes en España [63] [64] [65] [66] [67] [68] [69] [70] [71] [72] [73] [74] [75] [76] [77] [78] [79] [80] [81] [82]

Cuadro 3.1: Fuentes de datos empleadas en el proyecto.

Fuente: elaboración propia.

3.2. Flujo de trabajo

El flujo de trabajo, tal y como se observa en la Figura 3.1, sigue el de cualquier otro proyecto que abarque el **ciclo de vida completo de los datos** tras la comprensión del problema a resolver. Por tanto, va desde la captura u obtención de los datos mediante la técnica más adecuada en cada caso hasta el modelado, evaluación y producción de los resultados, tal y como se describe en la **metodología CRISP-DM** en el Capítulo 1. A partir de esta visión global, en los siguientes capítulos se baja de nivel para explicar cada una de las fases en profundidad.

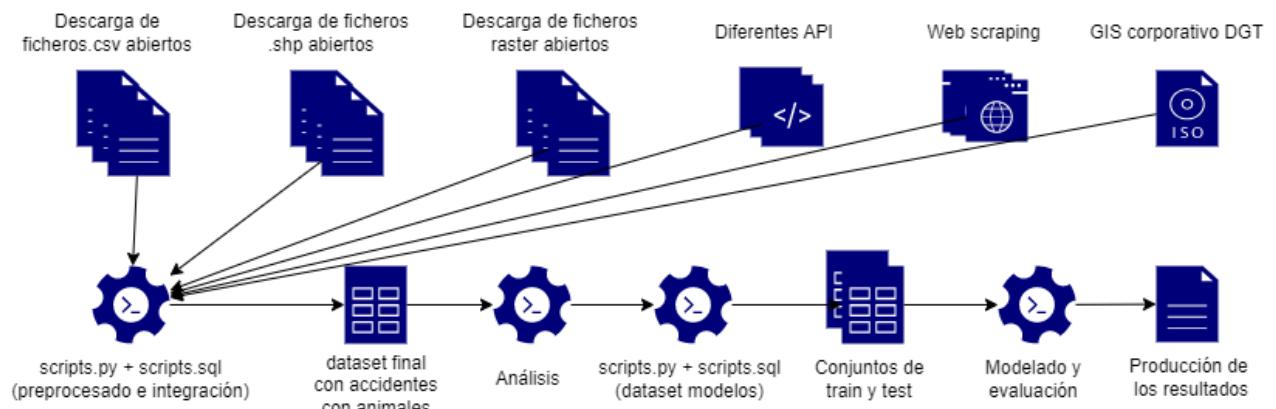


Figura 3.1: Flujo de trabajo del proyecto completo.

Fuente: elaboración propia.

3.3. Tecnologías

3.3.1. Lenguajes de programación

Los lenguajes de programación empleados, y sin entrar en detalle en los paquetes y librerías puesto que todo el código está documentado en [el repositorio](#) en GitHub, son los siguientes:

- **Python.** Lenguaje de programación de código abierto, interpretado, multiplataforma y de alto nivel creado en 1990 por Guido van Rossum [83]. Se emplea en todas las fases del proyecto, concretamente la versión Python 3.7.9 junto con el estándar PEP 8 [84].
- **SQL.** Structured Query Language (SQL) es un lenguaje que nació a mediados de la década de 1980 para manipular datos en bases de datos relacionales y ahora es muy potente para manipular datos en diversas tecnologías de bases de datos [85]. Se usa en todas las fases de este *Trabajo Final de Máster*, de manera aislada o combinada con Python, y en diferentes entornos (DBeaver, QGIS y Visual Studio Code).
- **R.** Es un lenguaje de programación de código abierto, interpretado y de alto nivel inicialmente diseñado por Ross Ihaka y Robert Gentleman para la computación estadística y la visualización, aunque ha evolucionado desde 1993 hasta ser ampliamente empleado para proyectos de aprendizaje automático [86]. Se utiliza en algunos momentos del trabajo para procesar y transformar los datos, así como en toda la fase de análisis de los mismos.
- **Bash.** Bourne-again shell (Bash), que empezó a ser codificado en 1988 por Brian Fox [87], se emplea en este proyecto como un lenguaje de *scripting* en momentos puntuales de las fases de obtención e integración de los datos para agilizar procesos concretos.
- **LaTex.** No es un lenguaje de programación, sino un sistema de composición tipográfica de alta calidad, específico para documentación técnica y científica, por lo que se ha convertido en un estándar para la comunicación y publicación de documentos científicos [88]. Este es el motivo, a pesar de la curva de aprendizaje, por el que se emplea para esta memoria.

3.3.2. Herramientas de terceros con licencia libre

Las herramientas empleadas son de código abierto o de licencia gratuita, reutilizando el conocimiento ya generado por la comunidad y reduciendo los costes:

- **DBeaver Community.** Es una herramienta de gestión de bases de datos multiplataforma, gratuita y de código abierto para desarrolladores y administradores de bases de datos [89]. Se escoge este gestor por su capacidad y sencillez para trabajar y visualizar bases de datos espaciales. La versión empleada es DBeaver 22.2.1.

- **PostgreSQL** con la extensión espacial **PostGIS**. PostgreSQL es un potente sistema de base de datos objeto-relacional de código abierto con más de 35 años de desarrollo activo, estable y robusto [90] y PostGIS es una extensión espacial también de código abierto para PostgreSQL [91], por lo que añade soporte para objetos geográficos permitiendo ejecutar consultas de localización en SQL. La versiones utilizadas son PostgreSQL 12.10 y PostGIS 3.0.3.
- **Visual Studio Code**. Es un editor de código optimizado, gratuito aunque propiedad de Microsoft Corporation, con soporte para operaciones de desarrollo como depuración, ejecución de tareas y control de versiones [92]. Por las necesidades de este proyecto, se trabaja con el apoyo de varias extensiones, como edit csv, json, Jupyter, PostgreSQL, Pylance, Python, XML to JSON o YAML, además de con la versión 1.56.2 del propio programa.
- **Jupyter Notebook**. Software libre, de código abierto sin ánimo de lucro y servicio web para la informática interactiva en todos los lenguajes de programación. La ejecución se realiza mediante la comunicación con un núcleo (*Kernel*) de cálculo [93]. Se elige su uso para la fase de modelado debido a que facilita su rápida consulta y entendimiento para cualquier persona ajena al proyecto porque se puede describir cada parte del análisis. Se trabaja con la versión 6.3.0.
- **Anaconda Navigator**. Es una interfaz gráfica de usuario (GUI) de escritorio incluida en Anaconda® Distribution que permite lanzar aplicaciones y gestionar paquetes, entornos y canales conda sin utilizar comandos de la interfaz de línea de comandos (CLI) [94]. En este proyecto se usa la versión 2.3.1 para lanzar Jupyter Notebook.
- **RStudio**. Es un entorno de desarrollo integrado (IDE, por sus siglas en inglés) gratuito y de código abierto para el lenguaje de programación R [95]. La versión con la que se trabaja es la de escritorio 2022.2.2.485, lanzada con el nombre Prairie Trillium.
- **RPubs**. Plataforma web de código abierto vinculada a RStudio que permite la publicación de documentos escritos con R Markdown [96], combinando la escritura con la salida de código R, lo que facilita que toda la comunidad pueda ver de forma sencilla los HTML generados de una forma que GitHub no permite.
- **GitHub Inc.** Para hacer uso de Git, *software* de control de versiones distribuido, se emplea su implementación en GitHub con una cuenta gratuita, donde además queda disponible en abierto el repositorio final de este proyecto.

- **QGIS.** Es una aplicación profesional de SIG que está construida sobre software libre y de código abierto (FOSS, por sus siglas en inglés) [97]. Se trabaja con la versión QGIS 3.22 LTR porque es la más estable en el momento del inicio del proyecto.
- **API de la AEMET.** Sistema para la difusión y reutilización de la información de Agencia Estatal de Meteorología. El *token* (API key) empleado en el proyecto se obtiene mediante el registro en la página web [60].
- **API del GIS corporativo de la DGT.** Sistema para la reutilización de la información geográfica de la Dirección General de Tráfico [98]. El *token* (API key) empleado lo proporciona la propia DGT para este proyecto.
- **Postman.** Es una plataforma para crear y utilizar diferentes API [99], lo que en este trabajo se usa para probar y optimizar las consultas antes de ser integradas en los respectivos *scripts*, también en los procesos de *scraping*. Se trabaja con una cuenta gratuita y la versión Desktop Platform Version 9.31.0.
- **Tableau Public.** Plataforma gratuita para explorar, crear y compartir visualizaciones de datos en línea propiedad de Salesforce [100].
- **Overleaf.** Editor de LaTeX online propiedad de Digital Science UK Limited con el que se trabaja esta memoria, permite compilar de forma rápida, facilita el control de versiones y la revisión por parte del director en un proyecto compartido [101].
- **Photopea.** Editor de imágenes avanzado, que permite trabajar tanto con gráficos rasterizados como vectoriales en el propio navegador desarrollado por Ivan Kutsir [102]. Este *software* se financia con publicidad, por lo que es considerado la versión alternativa de Adobe Photoshop. Se trabaja con la versión Photopea 5.0.
- **Diagram.** Software gratuito y de código abierto propiedad de la empresa JGraph Ltd que permite la creación de diagramas [103]. Se usa la versión de navegador 15.5.2.
- **Google Chrome.** Navegador web de código cerrado creado por Google, aunque su uso es gratuito y su desarrollo es derivado de proyectos de código abierto [104]. La versión empleada es la 108.0.5359.94.

3.3.3. Herramientas de elaboración propia

Además del empleo de diferentes API cuyas consultas se integran en los *scripts* en los que se combina Python y SQL, es importante destacar cuatro desarrollos de elaboración propia por su novedad en el campo del análisis espacial y su utilidad para otros proyectos:

- Desarrollo para convertir el vector { `provincia`, `carretera`, `km` } a { `longitud`, `latitud`, `geom` }, siendo `geom` la representación espacial de tipo `POINT` de la **localización de cualquier punto kilométrico** para poder ser insertado como un campo *geometry* dentro de la base de datos, según el estándar del Open Geospatial Consortium (OGC WKT) [105]. Como referencia de especificación para el desarrollo, se toma la descripción de la API proporcionada por el GIS corporativo de la DGT, descrita en el documento ‘Obtener las coordenadas de un punto proporcionando una provincia, carretera y punto kilométrico’ [98]. Como requisito previo, se ha de normalizar la información relativa al campo `provincia` y se puede consultar el *script* completo en `pk2loc.py` y `pk2locbd.py`, que también hacen uso de `models.py`.
- Herramientas para **procesar la altitud y la pendiente** a partir de los datos del IGN. En ambos casos se ha utilizado la información procedente del CNIG, concretamente la de los modelos MDT200 [61] y MDP05 [62]. Para ello, se descargan todos los ficheros (1.524 y 1.180, respectivamente) en formato ASC mediante la ejecución del programa proporcionado por el propio centro de descargas. Para acceder a estos ficheros, se tiene como referencia el MTN50, como se describe en los metadatos de ambas capas. Existen dos casos diferentes:

Altitud. La implementación completa se puede consultar en el *script* `elevationsdb.py`, que hace uso de `utils.py` y `models.py`, y sigue la siguiente lógica:

1. Se crean las tablas `mtn50` con `creacion_mtn50.sql` y `mtn50_hojas_pendientes` mediante `creacion_mtn50_hojas_pendientes.sql` y se importan los datos oficiales a dichas tablas en la base de datos `tfm`.
2. A partir de la localización del accidente, se consulta el MTN50 para obtener la hoja correspondiente.
3. Se busca el fichero correspondiente a dicha hoja cuyo nombre contiene el sistema de proyección EPSG.
4. El programa accede al interior del fichero y lee su cabecera.
5. En función de la cabecera y la transformación de coordenadas al sistema de referencias del fichero, se interpola el valor correspondiente a la altitud.

Pendiente. El proceso es el mismo al descrito en Altitud tras la creación `mtn50_hojas_alturas` mediante `creacion_mtn50_hojas_alturas.sql`, con la modificación de que en vez de obtener un valor discreto referido a la posición del accidente, se define un *buffer* de 30 metros a cada lado de la posición. Se calcula la mediana (más robusta que la media ante los *outliers* [34]) de las pendientes dentro de dicho *buffer*, con

el objetivo de no tener solamente en cuenta la pendiente de la carretera en sí, sino el terreno en el que realmente viven los animales. La implementación se puede consultar en el script `slopesdb.py`, que también integra su respectivos `utils.py` y `models.py`.

Asimismo, destaca el desarrollo de un proceso de **optimización de los algoritmos** descritos anteriormente de forma que en vez de hacer el cálculo para cada localización dentro de cada hoja, este se ejecuta de forma conjunta para todas las localizaciones dentro de esa hoja, respondiendo a la siguiente expresión:

$$\forall id_i \in \{H_j\} \implies f(id_i, H_j) = \lambda, \Phi, sheetdata \implies \text{altitud or pendiente}$$

- Script para calcular un **buffer de 500 metros a cada lado de las carreteras** (1 kilómetro en total) a partir del mapa forestal [59] para tomar la información sobre los usos del suelo gracias a la combinación de Python y SQL. El objetivo es obtener una capa geográfica con el **uso del suelo** en las zonas más próximas a las carreteras y, posteriormente, calcular el uso del suelo mayoritario en la proximidad a cada uno de los accidentes. Se puede consultar el código de generación de la capa en `landbuffer.py` y del cálculo en cada punto del accidente en el script `landusedb.py`, que hace uso de su respectivo `utils.py`.

3.3.4. Hardware

Los elementos físicos a tener en cuenta para un trabajo de estas características, además de los evidentes como un ratón o pantalla, son los siguientes:

- **PC.** Este proyecto no se puede desarrollar con cualquier ordenador debido a la alta demanda de capacidad de computación requerida. En este caso, las especificaciones del portátil empleado son:
 - Intel(R) Core(TM) i7-6700HQ CPU con 8 núcleos, @ 2.60GHz.
 - Unidades de disco: KINGSTON SHFS37A480G y SanDisk SD8SNAT-128G-1006 (SSD).
 - Sistema operativo Windows 10 Home de 64 bits con procesador basado en x64.
 - Tarjeta gráfica NVIDIA GeForce GTX 960M.
- **Disco duro externo.** Existen colecciones de ficheros que ocupan mucho espacio, como los modelos digitales del terreno [61] y de pendientes [62], por lo que se decide almacenarlos y trabajar con ellos desde un disco duro externo. Entre las opciones disponibles, se escoge el modelo WD My Passport 25E2 con una capacidad de 4 TB.

Capítulo 4

Creación y análisis del *dataset* final

En este capítulo se abordan las fases del entendimiento de los datos y su preparación, que comprende desde su captura, obtención y gestión a partir de diferentes fuentes hasta su integración en un *dataset* final, el cual se preprocesa y analiza en profundidad para extraer información importante para el desarrollo del proyecto. Debido a la complejidad del trabajo con diferentes fuentes de datos, el flujo de trabajo a bajo nivel se muestra en un diagrama para cada una de ellas. Además de estar enlazado en cada apartado, todo el código empleado para este capítulo se encuentra dentro del directorio `src` del repositorio.

4.1. Captura, obtención y gestión de los datos

Para gestionar todos los conjuntos de datos que serán necesarios hasta llegar a un *dataset* final optimizado de cara a conseguir los objetivos de este proyecto, se trabaja en una **base de datos espacial**, llamada `tfm`. En el momento de su creación, esta base de datos está vacía y se va nutriendo a lo largo de los siguientes apartados, importando los datos mediante el gestor de bases de datos DBeaver, ejecutando los *scripts* o a través del programa QGIS, al que está conectada, dependiendo del caso. El flujo completo de esta primera fase de captura, obtención y gestión de los datos se puede ver en la Figura 4.1 y se explica de forma pormenorizada en los siguientes apartados para cada una de las fuentes.

Debido a las restricciones de extensión por los requisitos de la presente memoria en el marco académico en el que se desarrolla, no se incluye una descripción exhaustiva de cada conjunto de datos ni de sus atributos, aunque dicho **análisis se ha realizado** para tomar las decisiones oportunas en cada momento.

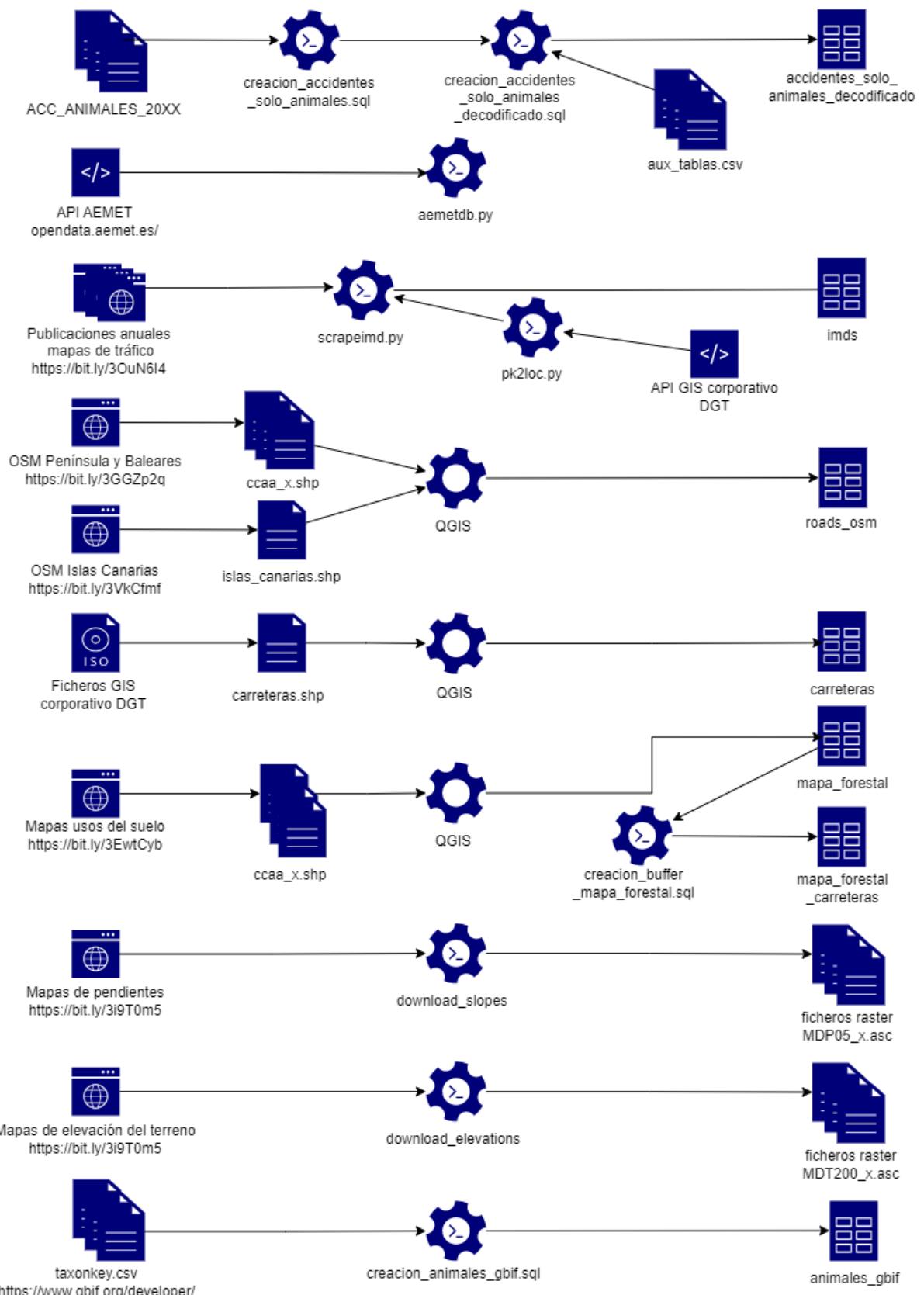


Figura 4.1: Proceso de obtención y captura de datos de diferentes fuentes.

Fuente: elaboración propia.

4.1.1. Registros de accidentes con víctimas

Hasta el 25 de noviembre de 2022, los únicos registros de accidentes con los que se puede trabajar porque no hay otros publicados son los relativos a los **microdatos de todos los accidentes de tráfico con víctimas** de los años 2016 [38], 2017 [39], 2018 [40], 2019 [41] y 2020 [42], publicados en ficheros excel en el apartado ‘DGT en Cifras’ del portal web de la Dirección General de Tráfico [106], espacio en el que también se incluyen el 16 de noviembre de 2022 los correspondientes a 2021 [43]. Tal y como se especifica en la documentación de dichos datos [44], el atributo `tipo_accidente` tiene una de las opciones que es ‘Atropello a animales’. Para poder avanzar en este *Trabajo Final de Máster*, se decide tomarlos como referencia y, en última instancia, que fueran sobre los que se construyan los modelos si no se consiguieran datos más específicos.

Una vez se publican los datos que realmente son pertinentes para este proyecto, el 25 de noviembre, todo el trabajo de decodificado, transformación e integración de los accidentes con víctimas carece de relevancia alguna en este TFM. No obstante, ya que uno de los objetivos del proyecto es la transferencia a la sociedad del conocimiento generado, se decide **publicar el dataset tratado** en Zenodo y, además, se crea un **dashboard** en Tableau [107] que aporta valor añadido al permitir explorar los datos de una forma interactiva e intuitiva, tanto por parte de personas con un conocimiento específico como para aquellas que no lo poseen. Se puede acceder al **proyecto completo**, el primero de estas características en España, en [este enlace](#) y observar algunas de las opciones en la Figura 4.2.



Figura 4.2: Selección de dashboards de la visualización de todos los accidentes con víctimas.
Fuente: *proyecto de elaboración propia tras el tratamiento de los datos de la DGT*.

4.1.2. Registros de accidentes con animales involucrados

Para solicitar los registros de todos los accidentes con animales y no solo aquellos en los que haya habido víctimas humanas, se realiza el **proceso burocrático** completo por el que se cumple el compromiso de confidencialidad, el consentimiento para inscribirse en el Registro Nacional de Investigadores, los datos del investigador principal y una breve memoria de este proyecto de investigación, incluyendo objetivos, metodología, fuente de financiación y plan de difusión de los resultados.

El 25 de noviembre de 2022, tras reuniones y la aceptación de la petición por parte de la DGT, la propia Dirección General publica en el apartado ‘DGT en Cifras’ de su portal web los **microdatos de todos los accidentes de tráfico con animales**, causaran víctimas o solo daños materiales, de los años 2016 [45], 2017 [46], 2018 [47], 2019 [48], 2020 [49] y 2021 [5] junto a un fichero Excel con el diccionario para decodificar las tablas [50]. Se trata de los conjuntos de datos más detallados y relevantes publicados hasta la fecha sobre accidentes de tráfico en los que hay animales involucrados, por lo que se convierten en el núcleo del proyecto a partir del cual se construye el *dataset* final.

Para ello, en primer lugar se crea la tabla `accidentes_solo_animales` en la base de datos `tfm` mediante `creacion_accidentes_solo_animales.sql` y se importan los ficheros respectivos a cada uno de los años publicados tras haber sido transformados a formato CSV, puesto que es el único que admite la versión gratuita de DBeaver. En este punto, la tabla completa contiene 165.452 registros descritos por 20 variables: `id_num`, `ind_accda`, `ind_acciv`, `total_mu30df`, `total_hg30df`, `total_hl30df`, `fecha_accidente`, `hora_accidente`, `mes_1f`, `anyo`, `ccaa_1f`, `provincia_1f`, `cod_municipio`, `carretera`, `km`, `sentido_1f`, `tipo_via_3f`, `titularidad_via_2f`, `tipo_animal_1f` y `tipo_animal_2f`.

Muchos de los campos están codificados, por lo que se **decodifican para facilitar su interpretabilidad** de forma directa y reproducible exportando a CSV cada una de las hojas del libro Excel de la documentación [50] e importándolas como tablas auxiliares a la base de datos, precedidas por el prefijo `aux_` para evitar confusiones. Cabe destacar las excepciones de la tabla `aux_codigo_municipio_ine`, que se construye manualmente con las indicaciones de la documentación uniendo el código de la provincia y del municipio, y de la tabla `aux_tipo_animal_1`, a la que se añade una columna con su equivalencia en `taxonkey` de GBIF por su utilidad en fases futuras. Por último, se ejecuta `creacion_accidentes_solo_animales_decodificados.sql` que crea la tabla `accidentes_solo_animales_decodificado` con todos los campos antes mencionados y los nuevos decodificados gracias a las 11 tablas auxiliares, añadiendo las siguientes 12 variables al conjunto de datos: `nombre_ind_accd`, `nombre_ind_acciv`, `nombre_mes`, `nombre_ccaa`, `nombre_provincia`, `nombre_municipio`, `nombre_sentido`, `nombre_tipo_via`, `nombre_titularidad_via`, `nombre_tipo_animal_1f`, `nombre_tipo_animal_2f` y `taxonkey`.

4.1.3. Vías bajo titularidad DGT

La base de datos de **vías bajo titularidad de la DGT** no se encuentra accesible a través de internet, pero son datos públicos que desde la propia Dirección General facilitan para este proyecto. Consta de 15.417 registros descritos con los siguientes atributos: `id`, `geom`, `objectid`, `ddtram_cod`, `ddcar_codi`, `ddcar_deno`, `ddtram_den`, `ddtram_tip`, `ddtram_tit`, `ddtram_t_1`, `ddtram_lon`, `pk_ini`, `pk_fin`, `idtipovian`, `ddprov_pro`, `ddcom_comu`, `color`, `ddcar_carr`, `ddtvia_ord`, `pkinigis`, `pkfingis` y `shape_leng`.

Para importar estos datos a la base de datos `tfm`, se opta por hacerlo con el fichero **shapefile** mediante **QGIS**, de manera que se pueda detectar de forma intuitiva si la carga de los datos falla para alguna de las regiones de España gracias al visor. Una vez cargados los datos en el programa, se observa (Figura 4.3) que en **Cataluña** la proporción de vías contenidas en este conjunto de datos es mucho menor que en otras zonas. Explorando los datos en profundidad, gracias a los atributos `ddprov_pro` y `ddcom_comu` se verifica que solo hay 249 vías incluidas de esta comunidad. Consultada la DGT por el asunto, se verifica que los datos son correctos debido a las competencias autonómicas sobre las carreteras, diferente a las de otras comunidades.

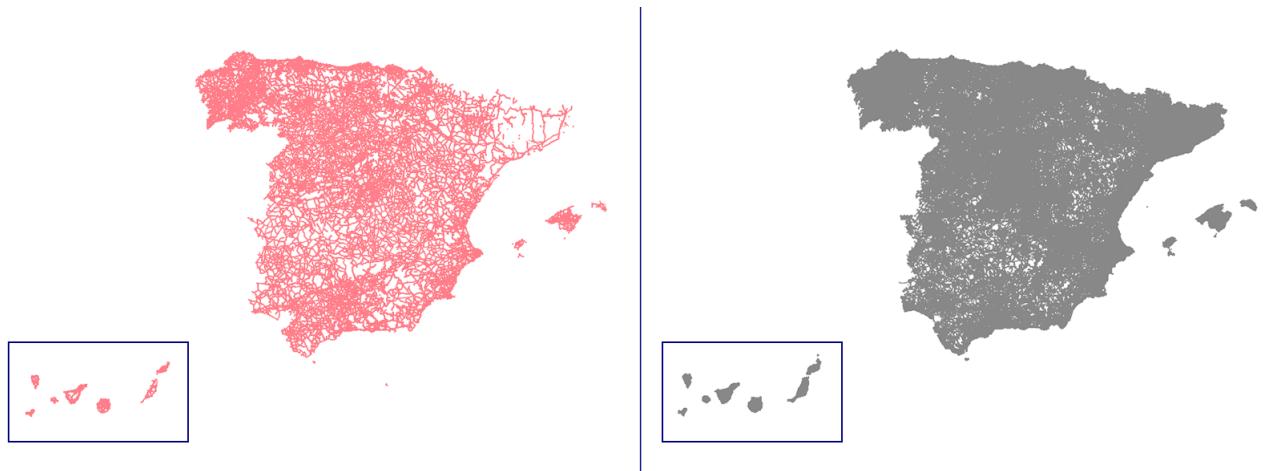


Figura 4.3: Vías de España bajo titularidad de la DGT (izq) y todas las vías según OSM (der).

Fuente: *DGT y OSM. elaboración propia mediante el programa QGIS y Photopea.*

Finalmente, **se importan los datos** cargados a la base de datos `tfm` en una tabla llamada `carreteras` con la DDL que se encuentra en `creacion_carreteras.sql` y se realizan las comprobaciones oportunas para verificar que no se han producido errores en el proceso.

4.1.4. Todas las vías de España

Los datos proporcionados por la DGT tienen carencias que se han de suplir para enriquecer el conjunto de datos final. Con este objetivo, se encuentra útil la **base de datos de OpenStreetMap (OSM)**, puesto que contiene atributos con los que se puede trabajar para obtener información de valor, concretamente de los datos relativos al territorio nacional en la Península Ibérica e Islas Baleares [51] y en las Islas Canarias [52].

En esta ocasión también se hace la importación a través de QGIS, no solo para detectar posibles fallos en la carga de los datos, sino para **combinar la capa vectorial** de la Península y Baleares con la de Canarias como proceso previo a dicha importación. Como se observa en la Figura 4.3, esta nueva capa **contiene información de más carreteras**. Por último, se importan los datos a la tabla `roads_osm` creada previamente con la DDL `creacion_roads_osm.sql` en la base de datos `tfm` y se repiten las verificaciones oportunas.

En total, son 2.818.615 registros, descritos por las variables `id`, `geom`, `osm_id`, `code`, `fclass`, `name`, `ref`, `oneway`, `maxspeed`, `layer`, `bridge`, `tunnel`, cuyo significado se puede consultar en la documentación [108] para no extender más este apartado.

Por último, se debe mencionar que, con el objetivo de cumplir con la **normativa de la Unión Europea sobre protección de datos** [109], no se han importado los nombres de usuario, los identificadores de usuario y los identificadores de cambios de los objetos de OSM porque pueden contener información personal de las personas que han colaborado.

4.1.5. Intensidad media diaria de las vías

La **intensidad media diaria** (IMD) es el número de vehículos que pasa de media diaria por la sección de carretera de cada una de las estaciones de aforo de la Red de Carreteras de España en el periodo de un año, según define el Ministerio de Transportes, Movilidad y Agenda Urbana [110].

Aunque se trata de una información que puede ser relevante para los objetivos de esta proyecto, tal y como se ha visto en el estado del arte (Capítulo 2), **no existe una documentación oficial** ni una base de datos equivalente de ningún tipo con esta información de forma accesible, al igual que ocurre con la velocidad máxima. Sin embargo, con una exploración en profundidad se han encontrado mapas en sitios web oficiales de este Ministerio que sí que muestran este tipo de información, aunque solamente para la España peninsular, por lo que **los datos existen**.

En este sentido, se pueden diferenciar dos períodos. La información en la página web [110] relativa a 1960, 1965, 1970, 1975, 1980, 1985, 1990, 1995, 2000, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013 y 2014 se muestra simplemente en un mapa en formato PDF que no se puede extraer a otro formato digital manipulable y tampoco son años relevantes para este

proyecto. No obstante, sí que se puede encontrar un **visor interactivo** para los años 2015 [53], 2016 [54], 2017 [55], 2018 [56], 2019 [57] y 2020 [58] desarrollado por la Dirección General de Carreteras del Ministerio de Fomento, como se muestra a la izquierda de la Figura 4.4.

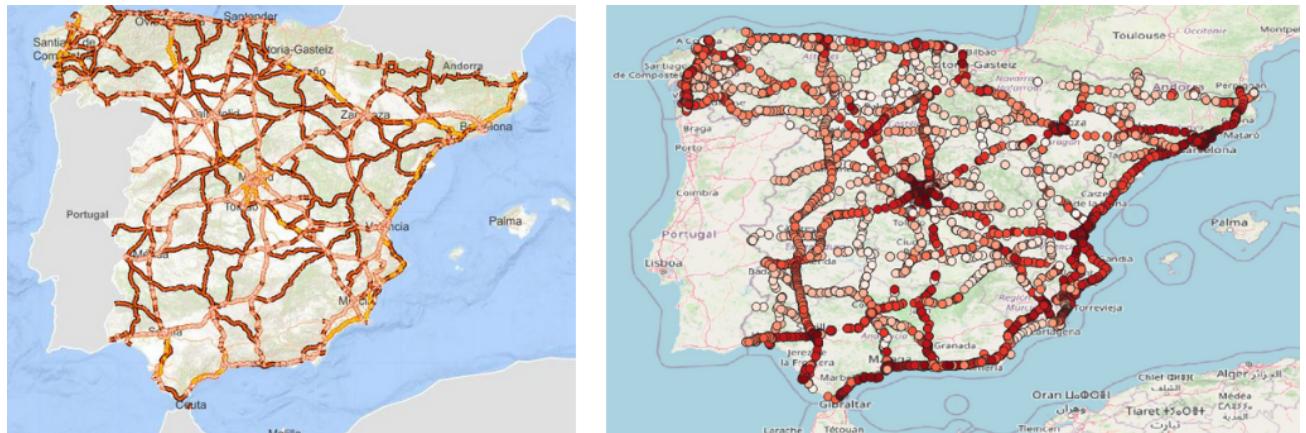


Figura 4.4: Mapa de tráfico en 2019 (izq) y resultado tras el proceso de *scraping* (der).

Fuente: Dirección General de Carreteras y elaboración propia.

Así, se podría entender que esos datos deberían ser más fácilmente accesibles, pero no lo son y se decide buscar una **opción legal** que acorte los plazos de petición por las vías oficiales con el objetivo de cumplir con los hitos en la planificación de este *Trabajo Final de Máster*. Por ello, la alternativa seleccionada es la de realizar un **proceso de scraping** que permita obtener los datos directamente de la página web que contiene el visor de cada uno de los años entre 2015 y 2020, asegurando que está permitido tras el análisis del fichero `robots.txt` [111].

Para ello, primero se crea la tabla vacía `imds` en la base de datos `tfm` mediante `creacion_imds.sql` y posteriormente con `creacion_importacion_provincias.sql` la tabla `provincias`, ya poblada con los datos de las provincias para calcular la longitud y latitud. Por último, se ejecuta el script `scrappeimd.py`, que hace uso de `utils.py` y la herramienta de desarrollo propio `pk2loc.py` ya explicada en el Capítulo 3 y adaptada a este caso.

En cualquier caso, se recomienda consultar el largo y complejo código al completo, desde la construcción de las URL a las que se llama el programa para hacer la descarga hasta el cálculo del `geom` de cada valor como el punto medio entre el `pk_inicio` y el `pk_fin` dentro de la geometría de cada carretera.

En total, tras 20 horas de ejecución, se obtiene un conjunto de datos con 6.000 registros, 1.000 correspondientes a cada año, descritos por los atributos `id`, `objectid`, `provincia`, `nombre`, `year`, `tipo_carretera`, `pk_inicio`, `pk_fin`, `longitud`, `vh_km_total`, `vh_km_ligeros`, `vh_km_pesados`, `imd_total`, `imd_ligeros`, `imd_ligeros`, `imd_pesados` y `geom`. Existen 1.010 observaciones en las que no se ha podido calcular su `geom` debido a cambios en los nombres de las carreteras, cuya evolución no está actualizada en la API del GIS corporativo

de la DGT. A la derecha de la Figura 4.4, se observa una aproximación visual a los datos obtenidos, con tonos **rojos más intensos aquellos tramos con mayor densidad** puesto que en el mapa de la DGC la interpretación es inversa y puede dar lugar a confusiones.

4.1.6. Usos del suelo

Los **usos del suelo** tampoco están incluidos en los datos proporcionados por la DGT, pero pueden ser un factor relevante [6] [16] [19] [26], por lo que es interesante enriquecer el *dataset* final con esta información.

En esta ocasión, sí que existe documentación oficial al respecto, por lo que se opta por emplear los datos del **Mapa Forestal de España** [59], a escala 1:50.000, de la Dirección General de Desarrollo Rural y Política Forestal, que depende del Ministerio de Agricultura, Alimentación y Medio Ambiente. Aunque es un proyecto realizado entre 1997 y 2006, es el más reciente de estas características.

Para importar los datos a la base de datos `tfm`, se hace uso de QGIS, que permite **unir todas las capas vectoriales** de una forma sencilla. Esto se debe a que la información no se encuentra toda en un mismo fichero *shapefile*, por lo que se procede en primer lugar a la descarga de todos conjuntos de datos de **cada una de las provincias** de España para, a continuación, cargar cada uno de ellos como una nueva capa vectorial y, por último, combinar todas las capas en una, cuya información es la que se importa a la tabla `mapa_forestal`, previamente creada con [creacion_mapa_forestal.sql](#) en la base de datos sobre la que se está trabajando.

Se identifican un total de 874.979 áreas descritas mediante los atributos `id`, `geom`, `poligon`, `prov_mfe50`, `ccaa_mfe50`, `tfcctot`, `tfccarb`, `fcc_pond`, `tipestr`, `distrib`, `for_man`, `sp1`, `o1`, `e1`, `sp2`, `o2`, `e2`, `sp3`, `o3`, `e3`, `definicion`, `clas_ifn`, `usos_suelo`, `usos_gener`, `tsp1`, `tsp2`, `tsp3`, `tipo_bosqu`, `id_forarb`, `cla_forarb`, `nom_forarb`, `regbio`, `shape_leng`, `shape_area`, `layer` y `"path"`.

Finalmente, se crea la capa `mapa_forestal_carreteras` y se puebla con el resultado de calcular un **buffer de 500 metros** a cada lado de la carretera mediante el script [creacion_buffer_mapa_forestal.sql](#), tal y como se explica en el epígrafe de ‘Herramientas de elaboración propia’ (Capítulo 3). En la figura 4.5 se observan diferentes detalles del resultado.

4.1.7. Datos meteorológicos

La **meteorología** ha demostrado ser un **factor de protección** en algunas de las investigaciones anteriores [6] [17], aunque no hay muchos análisis que la tengan en cuenta. Es por este motivo por el que se decide enriquecer el *dataset* final con este tipo de información.

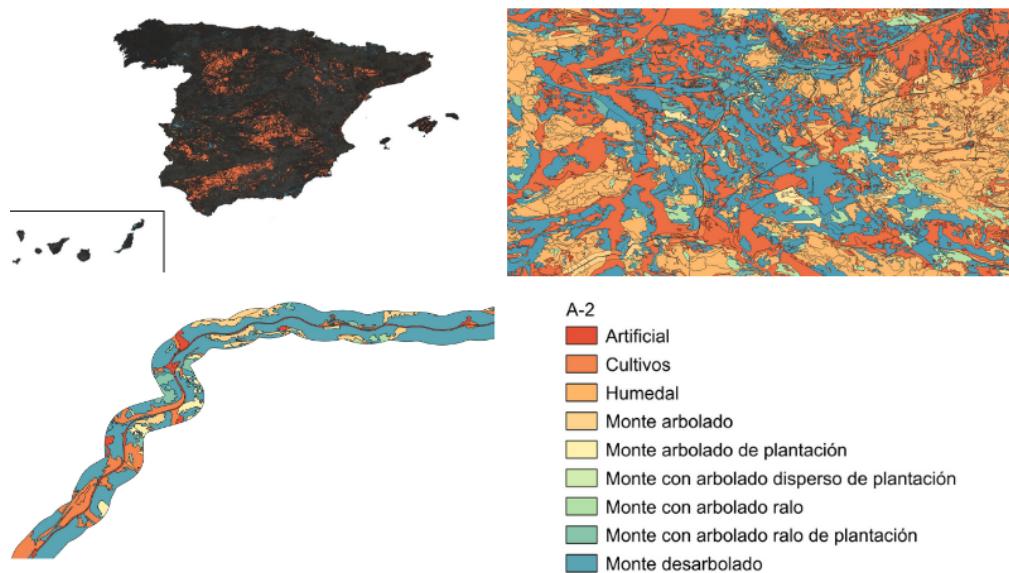


Figura 4.5: Usos del suelo en España, *zoom* sobre Calatayud y *buffer* sobre la A-2 en Calatayud.
Fuente: elaboración propia con datos de la DGT y el Mapa Forestal de España.

Como fuente oficial, se prevé la extracción de los datos directamente de la **Agencia Estatal de Meteorología** (AEMET), que posee una API muy completa [60] que permite consultar los datos recogidos por estaciones meteorológicas concretas y que devuelve un JSON con los siguientes campos: `fecha`, `indicativo`, `nombre`, `provincia`, `altitud`, `tmed`, `prec`, `tmin`, `horatmin`, `tmax`, `horatmax`, `dir`, `velmedia`, `racha`, `horaracha`, `sol`, `presMax`, `horaPresMax`, `presMin` y `horaPresMin`.

En este punto, se obtiene el ***token*** en la página del sitio web correspondiente [112] y se estudia el funcionamiento de la API mediante pruebas con el programa Postman. Además, como las peticiones se realizan a una **estación meteorológica** en concreto, se descargan todas las estaciones clasificadas como ‘automáticas’ por la AEMET, se cargan los ficheros en QGIS para **fusionar todas las capas** y, por último, se importan los datos en una única tabla llamada `aemet_estaciones` a la base de datos `tfm`, cuya DDL se encuentra en `creacion_aemet_estaciones.sql`. De este modo, se facilita la futura fase de integración.

4.1.8. Pendiente del terreno

En cuanto a la **pendiente del terreno** en el que se producen los siniestros, no existen estudios sobre su posible relevancia, por lo que se considera pertinente introducir esta información en el *dataset* final para enriquecerlo.

Como fuente oficial, se emplean los datos recogidos en el **modelo digital de pendientes** realizado a partir de las nubes de puntos LIDAR de la primera cobertura con **paso de malla**

de 5 metros [62] del Centro Nacional de Información Geográfica (CNIG), que depende de la Dirección General del Instituto Geográfico Nacional (IGN), y que fue publicado en enero de 2020, según indica el análisis de los metadatos.

El conjunto de datos completo consta de 1.180 ficheros en formato ASC y pesa 15 gigabytes, motivo que lleva a la decisión de integrar esta información directamente en el dataset final atendiendo a cada una de las localizaciones en las que se han registrado accidentes de tráfico con animales involucrados en lugar de realizar la importación completa a la base de datos tfm. Por tanto, en este punto solo se descargan los ficheros mediante la ejecución del programa del propio centro de descargas del CNIG y se crean las tablas mtn50 con creacion_mtn50.sql y mtn50_hojas_pendientes mediante creacion_mtn50_hojas_pendientes.sql, poblándolas con los datos oficiales en la base de datos tfm. De esta manera, ya está todo preparado para ejecutar slopesdb.py en la fase de integración, tal y como se ha explicado en el apartado de ‘Herramientas de elaboración propia’ (Capítulo 3).

4.1.9. Elevación del terreno

Con la altitud del terreno se da una situación similar a la pendiente del mismo. Se trata de un factor que se tiene en cuenta de forma residual en alguna investigación, según se ha estudiado en el estado del arte (Capítulo 2) y, de hecho, tampoco existen estudios sobre su posible relevancia. Así, se considera positivo introducir esta información en el dataset final con el objetivo de valorar todos los posibles factores en la problemática que ocupa al proyecto para maximizar la probabilidad de precisión en los posibles modelos.

Existen datos oficiales en abierto con esta información con diferente nivel de precisión. Por motivos de coste computacional, se opta por el modelo digital del terreno 1ª Cobertura con paso de malla de 200 metros [61] del Centro Nacional de Información Geográfica (CNIG), que depende de la Dirección General del Instituto Geográfico Nacional (IGN), con información recogida entre 2009 y 2016.

El conjunto de datos completo consta de 1.524 ficheros en formato ASC y pesa 9,31 gigabytes, así que una vez más se decide integrar esta información directamente en el dataset final en función de la localización en la que se ha registrado cada accidente, por lo que se evita la importación completa a la base de datos tfm, ahorrando tiempo, espacio y energía. En consecuencia, en este punto se procede a la descarga de los ficheros siguiendo el proceso anterior y, como la tabla mtn50 ya está en la base de datos tfm, solo es necesario crear mtn50_hojas_alturas mediante creacion_mtn50_hojas_alturas.sql. De esta manera, ya está todo preparado para ejecutar elevationsdb.py en la fase de integración, como se ha visto en ‘Herramientas de elaboración propia’ (Capítulo 3).

4.1.10. Fauna

En un proyecto de accidentes de tráfico con animales, se puede entender que la fauna debe ser tomada, *a priori*, como un elemento clave. Sin embargo, no se han encontrado investigaciones que tengan en cuenta la **distribución de los animales** en la zona objeto de estudio. Entrando en profundidad en este tema, se detecta que uno de los motivos puede ser debido a su complejidad, por lo que en este proyecto se aborda la situación de la forma más precisa posible.

Por ello, se escoge como fuente de datos la **Global Biodiversity Information Facility (GBIF)**, la base de datos más grande y consolidada del mundo al respecto, definida por ellos mismos como “una organización internacional y una red de datos financiada por gobiernos de todo el mundo, destinada a proporcionar a cualquier persona, en cualquier lugar, acceso abierto y gratuito a datos sobre cualquier tipo de forma de vida que hay en la Tierra” [113]. El único requisito para descargar la información necesaria es el de registrarse, por lo que es el primer paso que se da.

En principio, se baraja la opción de usar la **API de la propia GBIF**, que es muy potente y permite la descarga de los datos y campos necesarios, en este caso las especies de animales en función de las diferentes taxonomías y su extensión dentro del territorio nacional. Sin embargo, **se decide descargar manualmente** la distribución de cada una de las especies de interés porque es un proceso mucho más rápido, lo que obliga a realizar un preprocesado antes de su importación a la base de datos `tfm`. En este sentido, el mayor problema se encuentra en que dos ficheros CSV de los 20 descargados tienen distinto tipo de separador dentro del mismo archivo, por lo que se hacen comprobaciones de forma para evitar fallos o pérdida de información. Como los datos se tratan en R, se aprovecha para exportar finalmente únicamente aquellos registros que tengan el valor ‘ES’ referente a España en el campo `countrycode`, ya que son los únicos que interesan.

El **criterio para seleccionar los animales** de los que se descarga la información es que hayan estado involucrados en algún accidente de tráfico entre 2016 y 2021, según la clasificación recogida en el campo `tipo_animal_1f` de la tabla `accidentes_solo_animales_decodificado`. Así, se busca en internet su nombre científico para hallar el *taxonkey*. Las etiquetas de las que no se busca la distribución de los animales en España son ‘animal no identificado’, ‘ave’ y ‘otro animal’ por ser muy genéricos, así como ‘canino’ porque solo está clasificado como doméstico y ‘nutria’ porque no hay información. Por tanto, los conjuntos de datos con los que se trabaja son los correspondientes a las clasificaciones de ‘cabra montés’ [63], ‘ciervo’ [64], ‘gamo’ [65], ‘gato montés’ [66], ‘conejo’ [67], ‘corzo’ [68], ‘lince ibérico’ [69], ‘liebre’ [70], ‘jabalí’ o ‘cerdo’ [71], ‘lobo’ [72], ‘muflón’ [73], ‘oso pardo’ [74], ‘rebeco’ [75], ‘caprino’ [76], ‘equino’ [77], ‘tejón’ [78], ‘zorro’ [79], ‘felino’ [80], ‘ovino’ [81] y ‘vacuno’ [82].

Una vez tratados los datos, se importan a la tabla `animales_gbif`, que previamente se ha

creado en la base de datos `tfm` con `creacion_animales_gbif.sql`. Una vez poblada la tabla, el último paso para dejarla completa de cara a fases futuras es la creación de un *buffer* con el script `geom_buff_animales_gbif.sql` que imputa al campo `buff` el cálculo la **extensión completa de cada animal** a partir de su punto de observación (`geom`) y la incertidumbre registrada sobre dicha ubicación (`coordinateuncertaintyinmeters`). El resultado del proceso se puede ver en la Figura 4.6.

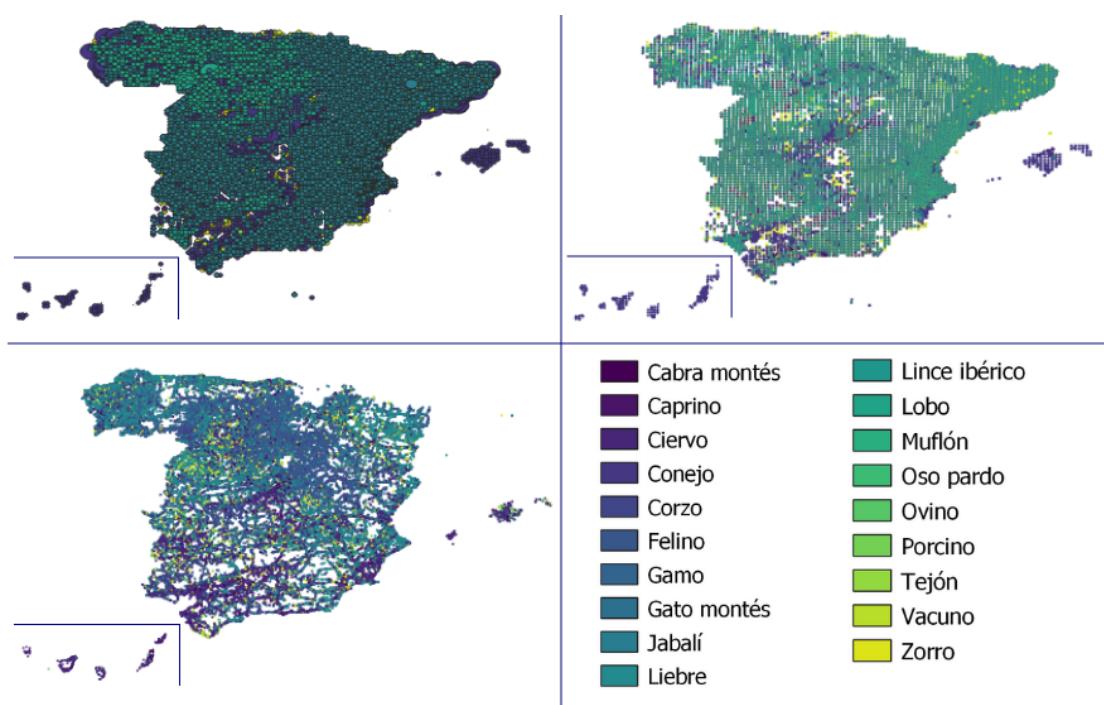


Figura 4.6: *Buffer* (arriba izq), registros GBIF (arriba der) y animales en cada accidente (abajo).
Fuente: elaboración propia a partir de datos de GBIF y DGT.

Finalmente, la tabla `animales_gbif` consta de 123.068 registros, descritos por los campos `gbifid`, `datasetkey`, `occurrenceid`, `kingdom`, `phylum`, "class", "order", "family", `genus`, `species`, `infraspecificepithet`, `taxonrank`, `scientificname`, `verbatimscientificname`, `verbatimsientificnameauthorship`, `countrycode`, `locality`, `stateprovince`, `occurrencestatus`, `individualcount`, `publishingorgkey`, `decimallatitude`, `decimallongitude`, `coordinateuncertaintyinmeters`, `coordinateprecision`, `elevation`, `elevationaccuracy`, "depth", `depthaccuracy`, `eventdate`, "day", "month", "year", `taxonkey`, `specieskey`, `basisofrecord`, `institutioncode`, `collectioncode`, `catalognumber`, `recordnumber`, `identifiedby`, `dateidentified`, `license`, `rightsholder`, `recordedby`, `typestatus`, `establishmentmeans`, `lastinterpreted`, `mediatype`, `issue`, `geom` y `buff`.

4.2. Creación de nuevas variables

Ya están todas las fuentes de datos trabajadas y se ha realizado un análisis pormenorizado de cada una de ellas, detectando todas las necesidades, aunque no esté recogido en esta memoria. Para trabajar a partir de ahora, se crea **una copia de accidentes_solo_animales_decodificado**, duplicado al que, además, se añaden los **campos que se completarán** para obtener el conjunto de datos final, todo ello con el *script* `creacion_accidentes_animales_final.sql`, tal y como se observa en la Figura 4.7. Las nuevas variables son `longitud`, `latitud`, `geom`, `dia_semana`, `nombre_dia_semana`, `tipo_dia`, `parte_dia`, `luna`, `prec`, `tmed`, `tmin`, `tmax`, `sol`, `uso_suelo`, `altitud`, `pendiente`, `imd_total` y `maxspeed`,

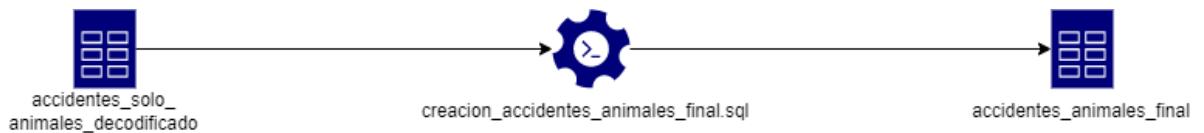


Figura 4.7: Creación de la tabla con el vector final de campos.

Fuente: elaboración propia.

Con la tabla `accidentes_animales_final` ya en la base de datos `tfm`, se crean **nuevos atributos a partir de otros ya existentes**. El flujo de trabajo se puede observar en la Figura 4.8 y en cada uno de los apartados de esta sección se detalla el proceso realizado.

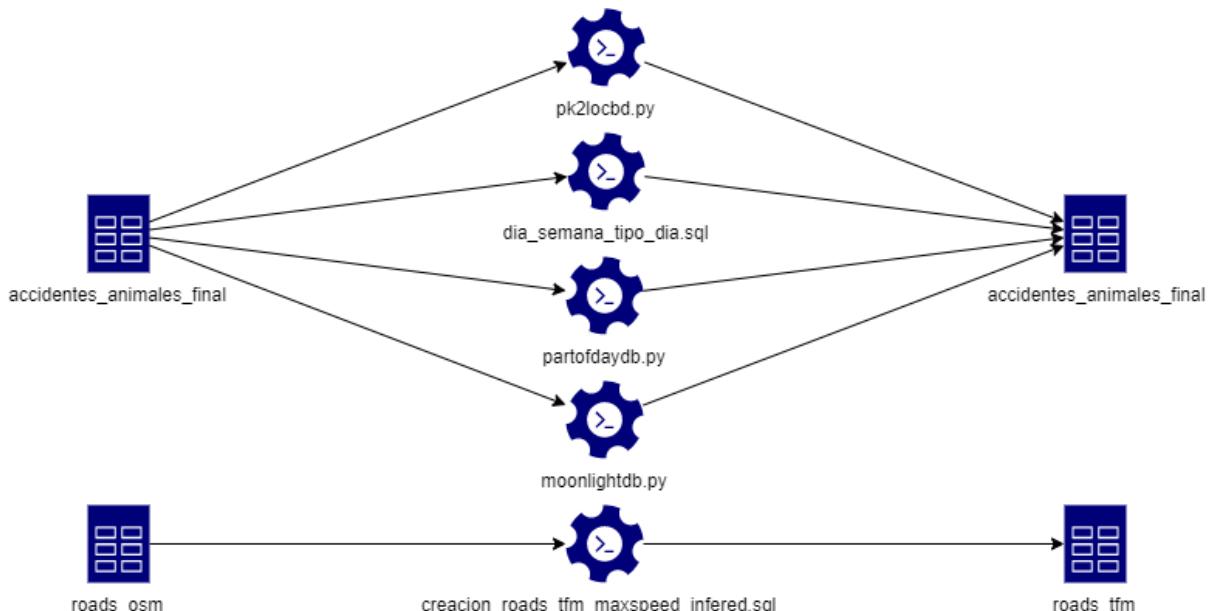


Figura 4.8: Creación de nuevas variables a partir de las tablas en la base de datos.

Fuente: elaboración propia.

4.2.1. Longitud, latitud y geom

La conversión de la información sobre la **ubicación del accidente a un formato adecuado** para su análisis es uno de los puntos clave de este proyecto. Es imposible abordarlo sin este dato y en este momento no está disponible ni en los conjuntos de la DGT ni en el resto de fuentes. Por ello, tal y como se explica en el epígrafe de ‘Herramientas de elaboración propia’ (Capítulo 3), **una de las herramientas más importantes** de este TFM es la que se desarrolla para convertir el vector { provincia, carretera, km } a { longitud, latitud, geom }, puesto que permite ser aplicada a todos los *dataset* que publica la DGT según su normativa.

Así, para completar los campos de `longitud`, `latitud` y `geom`, simplemente se ejecuta el script `pk2locdb.py`, que hace uso de `pk2loc.py` y `models.py` además de la API del GIS corporativo de la DGT. El mayor inconveniente es precisamente esta API, puesto que procesa una media de 500 registros cada hora, lo que suma un total de **13 días ininterrumpidos** para conseguir todas las ubicaciones en el nuevo formato. En cualquier caso, se consiguen transformar todos los registros que tienen una carretera no clasificada como ‘No inventariada’ y cuyo nombre sigue estando en la base de datos de la DGT, como se puede ver en la Figura 4.9.

<code>id_num</code>	<code>nombre_provincia</code>	<code>carretera</code>	<code>km</code>	<code>longitude</code>	<code>latitude</code>	<code>geom</code>
201.850.076.007	Zaragoza	N-II	229,5	-1,69903	41,33261	POINT (-1.69903 41.33261)

Figura 4.9: Un registro (201.850.076.007) con los nuevos campos de localización.

Fuente: elaboración propia.

4.2.2. Día de la semana

El **día de la semana** en el que se ha producido el accidente también puede ser un factor de riesgo, según se ha visto en otras investigaciones [17] [18] [22] [23] [25] [27] analizadas en el estado del arte (Capítulo 2), pero es un dato que no se tiene en este momento.

A partir de la **fecha del siniestro** sí que se puede calcular esta información. Para ello, se crea la tabla auxiliar `aux_dia_semana` con los campos de `valor` con los números del 1 al 7, `etiqueta` que contiene el literal del día de la semana y `tipo_dia` que toma el valor ‘diario’ o ‘finde’ en función del día que sea.

Con todo preparado, se ejecuta `dia_semana_tipo_dia.sql` que, gracias a las funciones de tiempo de PostgreSQL [114], permite extraer el día de la semana en numérico, dato que se incluye en el campo hasta ahora vacío de `dia_semana`, tras lo cual se transforma el 0 por el 7 en el caso de los domingos. Así, a partir de la columna `dia_semana` en la tabla y mediante *joins* con `aux_dia_semana` se decodifica dicha variable y se imputan los respectivos valores en los campos `nombre_dia_semana` y `tipo_dia`, como se puede ver en el ejemplo de un registro en la Figura 4.10.

<code>id_num</code>	<code>fecha_accidente</code>	<code>dia_semana</code>	<code>nombre_dia_semana</code>	<code>tipo_dia</code>
201.850.076.007	2018-02-16	5	Viernes	Diario

Figura 4.10: Un registro (201.850.076.007) con los nuevos campos referentes al día de la semana.

Fuente: elaboración propia.

4.2.3. Parte del día

El estado del arte (Capítulo 2) arroja como conclusión también importante que en algunas investigaciones la mayoría de los casos estos choques con animales ocurren al **amanecer** y al **anochecer** [6] o **cuando hay menos luz** [18]. Por este motivo y debido a que las horas de luz son muy diferentes dependiendo del punto de España y el momento del año, se busca la forma más precisa de determinar este aspecto.

Entre las opciones, se opta por la librería Astral [115] de Python para **automatizar el cálculo** tras verificar que arroja resultados correctos. Para la clasificación de las partes del día en este proyecto, se tienen en cuenta las siguientes definiciones, atendiendo la **clasificación astronómica** en lugar de la naval o la civil:

- **Dawn.** Momento del comienzo del crepúsculo antes del amanecer, con aparición de luz solar indirecta que se dispersa en la atmósfera de la Tierra, cuando el centro del disco solar ha alcanzado los 18° por debajo del horizonte del observador.
- **Sunrise.** Hora de la mañana en la que la parte superior del sol rompe el horizonte (suponiendo un lugar sin elementos que lo oculten).
- **Sunset.** Momento del anochecer en el que el sol está a punto de desaparecer por debajo del horizonte (suponiendo que el lugar no presenta elementos que lo oculten).
- **Dusk.** Momento del fin del anochecer, con la desaparición de luz solar indirecta que se dispersa en la atmósfera de la Tierra, cuando el centro del disco solar ha alcanzado los 18° por debajo del horizonte del observador.

Siguiendo esta lógica, se establecen **cuatro posibles partes del día**, según la luz solar. Tal y como se especifica en la Tabla 4.1, se añade un margen de 15 minutos al inicio y final del amanecer y anochecer debido a que lo que finalmente se estudia es el comportamiento de los animales en cada momento y este no es tan exacto como un reloj.

Finalmente se realiza el cálculo atendiendo al **horario de verano o invierno**, la diferencia horaria entre **Canarias** y el resto de España, el **día** del año, la **hora** y la ubicación exacta según la **latitud** y **longitud** halladas con anterioridad. Tras 96 minutos de ejecución

parte_dia	Periodo	Duración
Amanecer	Entre Dawn - 15 minutos y Sunrise + 15 minutos	1 hora
Día	Entre Sunrise + 15 minutos y Sunset - 15 minutos	11 horas
Anochecer	Entre Sunset - 15 minutos y Dusk + 15 minutos	1 hora
Noche	Entre Dusk + 15 minutos y Dawn - 15 minutos	11 horas

Cuadro 4.1: Equivalencia entre el valor de `parte_dia` y el periodo del día referido.

de `partofdaydb.py`, que hace uso de su respectivo `utils.py`, se verifica que los cálculos e inserción en la tabla son correctos, como se puede ver en varios ejemplos en la Figura 4.11.

123 id_num	fecha_accidente	hora_accidente	123 longitud	123 latitud	parte_dia
201.744.045.208	2017-11-30	20:30	-1,19641	40,83266	Noche
201.744.045.102	2017-08-02	21:30	-0,16832	40,99479	Anochecer
201.744.045.163	2017-09-09	19:40	-1,57793	40,49036	Día

Figura 4.11: Ejemplos con el campo sobre la parte del día según la fecha, hora y localización.
Fuente: elaboración propia.

4.2.4. Superficie de la Luna iluminada

Aunque las investigaciones no suelen tener en cuenta la **Luna** en el momento de un accidente de tráfico, parece razonable que pueda tratarse de un factor relevante en los que hay animales involucrados y se producen de noche. Esto es así porque, entre otros motivos, el **comportamiento de los animales** varía en función de la luz nocturna y la fase lunar, por lo que se debe estudiar si esta información puede mejorar la capacidad predictiva del modelo.

Para extraer esta información a partir de la fecha del accidente, se hace uso del paquete **PyEphem** [116] de Python. Como lo que importa para este proyecto no es la fase de la Luna en sí, sino la **superficie iluminada**, es este el valor que se imputa, representando 0 la oscuridad total (Luna nueva) y 100 toda su superficie (Luna llena). Puesto que solo se realiza el cálculo en los accidentes que han sido clasificados como ‘Noche’ en el campo `parte_dia`, el tiempo de ejecución del script `moonlightdb.py`, que hace uso de su respectivo `utils.py`, se reduce a una hora y, además, se permite distinguir entre el 0 de Luna nueva y NULL de un momento diferente a la noche. El resultado de algunos registros de ejemplo se observa en la Figura 4.12.

123 id_num	fecha_accidente	parte_dia	123 luna
201.744.045.208	2017-11-30	Noche	82
201.744.045.102	2017-08-02	Anochecer	[NULL]
201.744.045.163	2017-09-09	Día	[NULL]

Figura 4.12: Ejemplos con el campo sobre la Luna iluminada según la fecha y parte del día.
Fuente: elaboración propia.

4.2.5. Velocidad máxima de la vía

No existe una documentación oficial ni una base de datos equivalente sobre la velocidad máxima en cada uno de los tramos de cada una de las vías interurbanas de España. Como alternativa para suplir esta falta de información, se decide crear un **nuevo atributo a partir de otra de las fuentes de datos**. Para ello, de forma general, se toma el valor de la variable `maxspeed` de la tabla `roads_osm` en la base de datos `tfm` como velocidad máxima en kilómetros por hora de ese tramo, pero en la mayoría de los casos no tiene valor. En el supuesto de ausencia, se realiza ese cálculo a partir del valor de `fclass` contenido en la base de datos de OSM, según las características de cada tramo especificadas en su documentación [108]. Siguiendo el **reglamento de circulación** de España [117], las equivalencias de velocidad máxima calculadas para este proyecto en función de la categoría de la vía se muestran en la Tabla 4.2.

<code>fclass</code>	velocidad máxima
motorway	120
trunk	120
primary	90
secondary	80
tertiary	80
unclassified	80
residential	50
motorway_link	60
trunk_link	60
primary_link	60
secondary_link	60

<code>fclass</code>	velocidad máxima
tertiary_link	60
living_street	30
service	30
track	30
pedestrian	-
bus_guideway	-
escape	-
raceway	-
road	-
busway	-

Cuadro 4.2: Equivalencia entre el valor de `fclass` y la velocidad máxima de la vía (km/h).

Fuente: elaboración propia.

Como se aprecia en dicha Tabla 4.2, a pesar de la gran diversidad de tipos de vía, se opta únicamente por realizar la conversión de aquellas clasificadas como *roads*, *link roads* o *special road types* porque son las que se encuentran en zonas **interurbanas**. Las *links roads* se unifican a 60 kilómetros por hora y las *tracks* en 30, homogeneizando el dato en función de la **Normativa Técnica de Carreras** [118]. En este sentido, aunque la norma especifique una velocidad, la empresa constructora finalmente toma la última decisión y esta información no se encuentra recogida y unificada en ningún documento a nivel estatal, por lo que el análisis

exhaustivo para cada tramo de vía de España queda fuera del alcance de este proyecto.

La ejecución del script `creacion_roads_tfm_maxspeed_infered.sql` permite la realización de este proceso y crea la tabla `roads_tfm` en la base de datos `tfm`, con índice espacial e imputando el valor correspondiente en el nuevo campo `maxspeed_infered`.

Tras el cálculo, el número de tramos a los que se imputa una velocidad máxima inferida por tratarse de vías interurbanas es de **2.381.951** del total de 2.818.615 en todo el territorio nacional contenidos en la tabla, en la que también hay espacios urbanos que no se han tratado.

4.3. Integración de los datos de fuentes externas

En este momento, lo único que falta para terminar esta etapa de construcción del *dataset* final es la **integración de los datos procedentes de fuentes externas**. El flujo de trabajo se puede observar en la Figura 4.13 y a continuación se detalla el proceso realizado en función de la fuente de datos en cada caso.

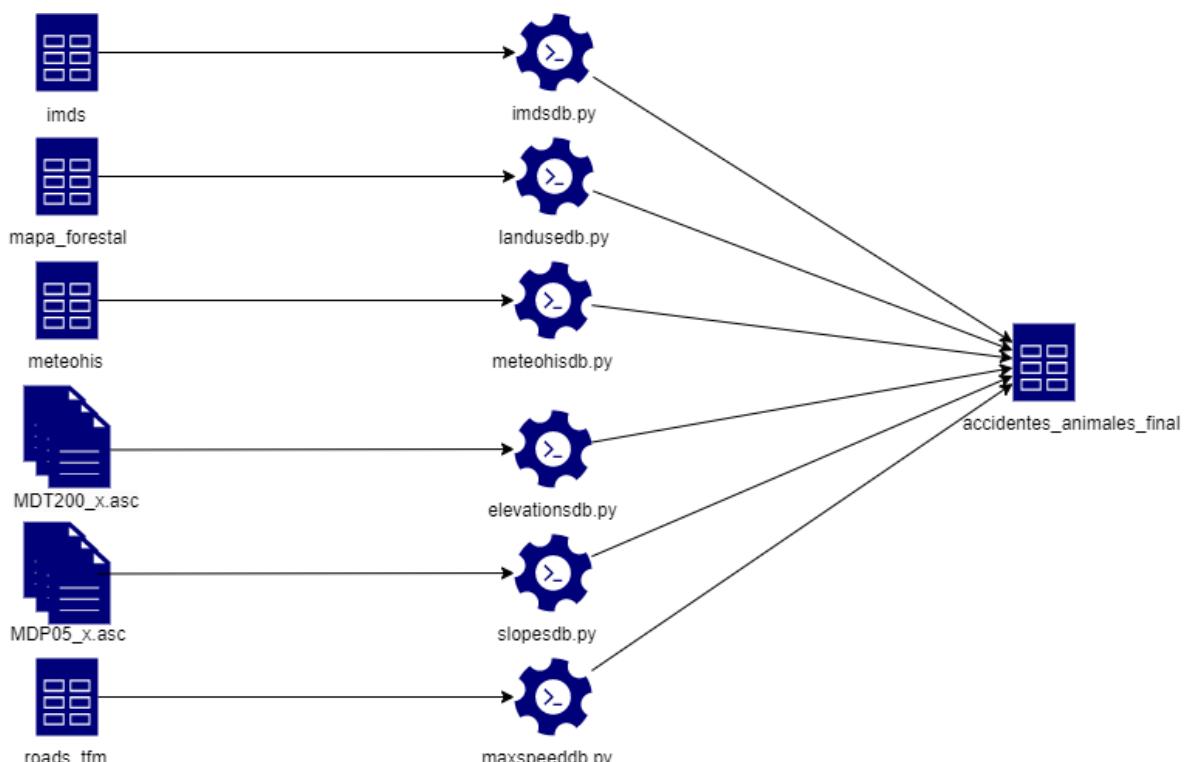


Figura 4.13: Integración de los datos a partir de fuentes externas.

Fuente: elaboración propia.

4.3.1. Intensidad media diaria de la vía

En la base de datos `tfm` ya se encuentra la tabla `imds` con la información de la intensidad media diaria de cada vía (IMD) desglosada por años entre 2015 y 2020 [53] [54] [55] [56] [57] [58], resultado del proceso de *scraping* en la fase anterior. Además, en ese mismo proceso se integró la conversión del vector `{ provincia, carretera, km }` a `{ longitud, latitud, geom }`, campos que también se encuentran en la tabla `accidentes_animales_final`, por lo que todos los datos están correctamente preparados para integrar los IMD en el *dataset* final.

Como último paso, se crea el *script* `imdsdb.py` que, haciendo uso de su respectivo `utils.py`, evalúa el **año del accidente y la carretera** en la que ha ocurrido para buscar en `imds` el punto más cercano del que se tenga el dato de dicha vía dentro de su geometría. Como solo hay datos hasta 2020, si el siniestro tuvo lugar en 2021, se toman como referencia los datos de 2019 como último año sin pandemia disponible. Finalmente, imputa el valor de `imd_total` en la columna homónima de `accidentes_animales_final`, puesto que se desconoce el tipo de vehículo que se ha visto involucrado en cada siniestro. La ejecución del proceso completo es de alrededor de media hora si se crea un índice espacial en la tabla `imds` y en la Figura 4.14 se muestran algunos registros de ejemplo.

<code>id_num</code>	<code>anyo</code>	<code>carretera</code>	<code>longitud</code>	<code>latitud</code>	<code>imd_total</code>
201.606.001.402	2.016	EX-209	-6,76817	38,92256	20.364
201.610.003.474	2.016	N-521	-6,65334	39,42684	2.910
201.636.016.188	2.016	PO-332	-8,77999	42,14218	15.353

Figura 4.14: Varios registros con el nuevo campo de IMD en función de la localización y año.

Fuente: elaboración propia.

4.3.2. Uso del suelo

En la base de datos `tfm` también se encuentran ya las tablas de `mapa_forestal` y `mapa_forestal_carreteras` con la información de los usos del suelo en toda España [59] y en un *buffer* de 500 metros a cada lado de las carreteras, respectivamente.

Mediante el *script* `landusedb.py`, que hace uso de su respectivo `utils.py`, se evalúa la `latitud` y `longitud` donde ha ocurrido cada accidente y se calcula, entre todos los usos contenidos en el área alrededor de 500 metros de ese punto, cuál es el uso del suelo mayoritario, que es el valor que finalmente se imputa en el campo `uso_suelo` de `accidentes_animales_final`, como se ve en la Figura 4.15. En este caso, el tiempo total de ejecución asciende a una hora y media debido a la complejidad del cálculo dentro de cada área, aunque es relativamente rápido gracias al índice espacial en `mapa_forestal`, sin el cual habrían hecho falta tres días.

123 id_num	123 longitud	123 latitud	uso_suelo
201.602.000.160	-2,14156	39,19824	Cultivos
201.602.000.177	-1,91961	39,24525	Cultivos
201.602.000.190	-2,57266	38,6174	Cultivos
202.015.111.814	-8,19321	43,26869	Monte arbolado de plantación
202.016.113.232	-2,36438	40,46427	Monte arbolado de plantación
202.024.116.453	-5,09397	42,64095	Monte arbolado

Figura 4.15: Varios registros con el nuevo campo de uso del suelo en función de la localización.

Fuente: elaboración propia.

4.3.3. Datos meteorológicos

Para extraer los datos meteorológicos del **lugar** y **momento** del accidente, la opción trabajada como prioritaria es la de emplear la **API de la AEMET** [112], que ha sido creada con esta finalidad. Para ello, se desarrolla el *script* `aemetdb.py`, que hace uso de su respectivo `utils.py`. Previamente, con el *token* proporcionado por la Agencia se realizan las pruebas en Postman para verificar que las llamadas son correctas.

Para determinar sobre qué **estación meteorológica** hacer la consulta, se hace el cálculo de cuál es la más cercana al lugar del accidente, comparando la `longitud` y `latitud` de cada registro con el `geom` de la tabla `aemet_estaciones`, que se importó a la base de datos `tfm` en la fase anterior. De este modo, se realiza la consulta de los datos recogidos por esa estación, en la `fecha_accidente` y `hora_accidente` siguiendo la documentación de la API, aunque igualmente la información proporcionada por la AEMET es diaria.

Sin embargo, al correr el *script* sobre todo el conjunto de datos, se comprueba que en la mayoría de los casos la consulta no devuelve valores, por lo que se hace un análisis en profundidad de la situación. El origen del problema es que muchas estaciones trabajan durante **algunos períodos de tiempo** y luego quedan inutilizadas, pero no hay forma de saber cuáles. Preguntando el servicio técnico de la AEMET sobre una solución para buscar solo en las estaciones que tengan información de cada fecha en concreto, **no aportan ninguna opción**, por lo que se opta por buscar una alternativa viable dentro de los plazos de este proyecto.

Por ello, se decide hacer uso de los **datos ya descargados** de la AEMET en diciembre de 2021 mediante otro proceso para hacer una Prueba de Evaluación Continua (PEC) de temática libre de la asignatura de Visualización de Datos, que en ese caso se centró en el cambio climático y cuyo resultado se puede ver en [este enlace](#) [119]. Se trata de todos los registros de todas las estaciones meteorológicas de España hasta 2021. En cada uno de los 292 ficheros se encuentran los atributos `fecha`, `indicativo`, `nombre`, `provincia`, `altitud`, `tmedia`, `precipitacion`, `tmin`, `horatmin`, `tmax`, `horatmax`, `dir`, `velmedia`, `racha`, `horaracha`, `sol`, `presmax`, `horapresmax`, `presmin` y `horapresmin`.

Para gestionar todos estos *datasets* y facilitar las consultas, se crea en la base de datos la tabla `meteohis` con la sentencia recogida en `creacion_meteohis.sql` y se importan todos los datos a la misma tras haber **unido todos los ficheros con bash**, quedando una tabla con 628.076 registros. En este punto, se crea el *script* `meteohisdb.py` que calcula la estación meteorológica más cercana al punto del accidente con datos para esa fecha y vuelca los valores correspondientes para dicho accidente en la tabla `accidentes_animales_final` en los campos correspondientes a `prec`, `sol`, `tmin`, `tmed` y `tmax`, tal y como se observa en la Figura 4.16.

<code>123 id_num</code>	<code>fecha_z</code>	<code>nombre</code>	<code>longitud</code>	<code>latitud</code>	<code>sol</code>	<code>prec</code>	<code>tmin</code>	<code>tmed</code>	<code>tmax</code>
201.649.021.303	2016-10-24	Zamora	-6,54966	42,05413	0,5	8,4	5,1	8,8	12,5
202.044.129.879	2020-11-02	Teruel	-0,55283	40,85299	9,9	0	12,9	18,95	25
201.927.091.229	2019-04-03	Lugo	-7,47713	42,52805	3,2	0,8	0,8	4,65	8,5

Figura 4.16: Registros con los nuevos campos sobre la meteorología en función del día y lugar.

Fuente: elaboración propia.

4.3.4. Elevación del terreno

Los datos sobre la altitud del terreno que se tienen que integrar en el *dataset* final **no se encuentran en la base de datos `tfm`**, sino en un disco duro externo por motivos de espacio como se especifica en apartados anteriores, pero sí que están las tablas `mtn50` y `mtn50_hojas_alturas`.

Tal y como se explica en ‘Herramientas de elaboración propia’ (Capítulo 3), ejecutando el *script* `elevationsdb.py`, que hace uso de su respectivo `utils.py`, se realiza el proceso completo a partir de la localización del accidente. La secuencia es la siguiente: consulta el MTN50 para obtener la hoja correspondiente, busca el fichero correspondiente a dicha hoja cuyo nombre contiene el sistema de proyección EPSG, accede al interior del fichero y lee su cabecera y, en función de la cabecera y la transformación de coordenadas al sistema de referencias del fichero, se interpola el valor correspondiente a la altitud, que es el que introduce en el campo `altitud` de la tabla `accidentes_animales_final`.

Toda esta fase se alarga una media de dos horas si se paraleliza y se lanza un proceso de cálculo por cada año. En la Figura 4.17 se observa el ejemplo de varios registros, que verifican que el resultado es correcto.

<code>123 id_num</code>	<code>nombre_provincia</code>	<code>longitud</code>	<code>latitud</code>	<code>altitud</code>
201.610.003.556	Cáceres	-5,85689	39,27016	487,134
201.632.014.103	Ourense	-7,46413	41,92246	849,729
201.806.050.592	Badajoz	-6,67574	39,12015	229,979

Figura 4.17: Varios registros con el nuevo campo de altitud en función de la localización.

Fuente: elaboración propia.

4.3.5. Pendiente del terreno

Los datos relativos a la pendiente del terreno **tampoco se encuentran en la base de datos `tfm`**, sino en un disco duro externo por motivos, pero sí que están las tablas `mtn50` y `mtn50_hojas_pendientes`.

Ejecutando `slopesdb.py`, que hace uso de su respectivo `utils.py`, el proceso es totalmente idéntico al anterior, con la excepción de que no se imputa el valor de la posición del accidente en el campo `pendiente` de la tabla `accidentes_animales_final`, sino la mediana resultante de los valores discretos que se encuentran en un *buffer* de 30 metros a cada lado de dicha posición.

Toda esta fase se alarga una media de más de un día. En la Figura 4.18 se observa el ejemplo de varios registros, que verifican que el resultado es correcto.

123 id_num ↴	abc nombre_provincia ↴	123 longitud ↴	123 latitud ↴	123 pendiente ↴
201.802.048.579	Albacete	-1,34037	39,10184	9,6945
201.802.048.578	Albacete	-1,3426	39,10137	8,576
201.802.048.582	Albacete	-1,90227	39,00128	0,856

Figura 4.18: Varios registros con el nuevo campo de pendiente en función de la localización.

Fuente: elaboración propia.

4.3.6. Velocidad máxima de la vía

El dato que se ha inferido de la **velocidad máxima en cada tramo** de cada vía se encuentra en el campo `maxspeed_inferred` de la tabla `roads_tfm` en la base de datos `tfm`, que es con la que se está trabajando. Se estudia la opción de integrar esta información en la tabla `carreteras`, ya que es la que procede de la base de datos oficial de la DGT. Para ello, se ejecuta `creacion_max_speed_carreteras.sql`, que imputa a cada vía su velocidad máxima, evaluando a qué carretera se refiere en ambas tablas en función de su geometría. Sin embargo, se observa que hacer este proceso provoca que finalmente el dato que se obtiene en la velocidad máxima en el punto del accidente sea menos preciso porque se pierde la granularidad del dato de cada tramo de esa carretera en favor del máximo en la vía completa.

Por tanto, aunque computacionalmente sea más costoso, finalmente se decide hacer la consulta directamente contra la tabla `roads_tfm`. Así, el script `maxspeeddb.py`, que hace uso de su respectivo `utils.db`, calcula, a partir de la longitud y la latitud de cada registro, el punto de carretera al que corresponde porque es el más cercano. Por último, extrae la velocidad máxima de ese punto, que es el dato que se introduce en el campo `maxspeed` de la tabla `accidentes_animales_final`, como se puede observar en el ejemplo de la Figura 4.19

<code>id_num</code>	<code>nombre_provincia</code>	<code>longitud</code>	<code>latitud</code>	<code>maxspeed</code>
201.603.000.378	Alicante/Alacant	-0,91824	38,10489	80
201.902.076.333	Albacete	-2,16892	38,42499	80
201.933.095.773	Asturias	-5,76034	42,9932	120
201.749.047.104	Zamora	-6,05332	41,90102	90
201.942.100.760	Soria	-2,33993	41,93731	80
201.746.046.218	Valencia/València	-0,70307	39,58535	90
202.010.110.022	Cáceres	-5,80361	39,91893	120

Figura 4.19: Varios registros con el nuevo campo de velocidad máxima según la localización.

Fuente: elaboración propia.

4.4. Dataset final

4.4.1. Procesado y análisis del conjunto de datos

Tal y como se expone en el Capítulo 3, se aprovecha la potencialidad de **R** para realizar toda la fase de procesado de los datos del conjunto y análisis del mismo. Se recomienda consultar el proceso completo, paso a paso, incluyendo **infografías interactivas**, **test estadísticos** y **cuadros** en el proyecto en RPubs, disponible en [este enlace](#), y a continuación se recogen los aspectos y conclusiones más importantes.

4.4.1.1. Limpieza de datos

Tras la carga de la tabla `accidentes_animales_final` en un *dataframe* gracias a la conexión a la base de datos `tfm` desde RStudio, se verifica que dicha carga ha sido correcta mediante la observación de registros aleatorios y comprobando que el número total de registros y campos coincide: **165.452 observaciones y 50 variables**. Al examinar el **tipo de dato** de cada atributo, debido al conocimiento previo que se tiene de la información que contienen, se detecta que hay algunos que deben ser **factorizados** para optimizar el análisis, por lo que se procede a esa transformación. Finalmente, como primera aproximación se estudian los **datos resumen** de cada variable del *dataset*, del que se pueden extraer algunas conclusiones rápidas, como que aparentemente no hay valores extremos, puntos que se analizan en profundidad en cada apartado correspondiente.

En cuanto a la limpieza de los datos propiamente entendida como tal, primero se comprueba que **no existen registros duplicados**.

El **tratamiento de los nulos** [120], por su parte, es una de las fases más complejas debido a la toma de decisiones que obliga a realizar. En primer lugar, tras haber leído la documentación de todas las fuentes de datos de las que proceden, se imputan como valores ausentes en el campo `altura` y `pendiente` aquellos registros con -999 o -9999 en dichas columnas, respectivamente.

A continuación, se estudia el valor absoluto de *missing data* en cada uno de los atributos y su proporción. Tras su análisis pormenorizado se toman las siguientes acciones:

- `longitud`, `latitud` y `geom`. Se **eliminan los registros** que no contienen esta información, puesto que es información espacial esencial. Se trata solo del 5 % del conjunto de datos, por lo que esta decisión no distorsiona el análisis y optimiza procesos futuros.
- `nombre_municipio`. A pesar del 69 % de nulos, se trata de una información espacial genérica si se compara con la `longitud` y `latitud` o el `geom`, por lo que su mantenimiento **no afecta** porque realmente no se empleará en los modelos.
- `nombre_titularidad_via` y `cod_municipio`: 0.00064 % es una proporción despreciable, por lo que **se mantienen** todos los registros.
- `prec`, `tmin`, `tmed`, `tmax` y `sol`. Son datos meteorológicos que podrían llegar a ayudar en el análisis y predicción, pero **su ausencia no compromete** el proyecto además de tratarse solo de un 3.59 % de los registros.
- `luna`. El campo es nulo cuando el accidente no ha ocurrido por la noche, por lo que incluso **ser nulo aporta información** relevante y se decide no eliminarlos.
- `pendiente` y `altitud`. Es información proporcionada por el CNIG y el valor es nulo simplemente porque no se ha documentado dicha información para esa localización. Así, **se mantienen** todos los registros porque su ausencia no compromete la calidad del proyecto.

El **análisis de los valores extremos o *outliers*** [34] también es importante porque su existencia puede sesgar los análisis y, lo que es peor, introducir ruido en la entrada de los algoritmos de clasificación que se emplearán en la fase de modelado. Por tanto, se analizan todas las variables numéricas mediante la técnica más adecuada en cada uno de los casos:

- Máximo y mínimo en `total_mu30df`, `total_hg30df`, `total_hl30df`, `pendiente`, `altitud` e `imd_total`. Los totalizadores indican el número de fallecidos, heridos hospitalizados y no hospitalizados, respectivamente, en un cómputo de 30 días para cada uno de los accidentes. Se observa que no existe ningún dato negativo (lo que indicaría un error) y que el valor más alto es el de heridos no hospitalizados, que en, al menos, un accidente asciende a 21. El mismo proceso de comprobación se hace con las otras tres variables debido a que se desconoce su dominio exacto, pero por el conocimiento del campo se concluye que el rango sí que es plausible. Por tanto, en todos los casos los valores entran dentro de lo posible y **no se trata ninguna de las variables**.

- Diagramas de caja con `prec`, `tmin`, `tmed`, `tmax` y `sol`. En cuanto a la meteorología registrada el día del accidente, se decide comparar los *boxplots* interactivos como el recogido en la Figura 4.20 de manera que se puede consultar directamente el valor de los valores extremos. Como el alcance del proyecto es el territorio español, estos datos coinciden con las temperaturas registradas en España en los últimos años, a pesar de que las cálidas parezcan muy altas, puesto que han sido los años más extremos de la serie histórica. Del mismo modo, las temperaturas mínimas se acercan a los récords registrados desde 2016, por lo que **no hay motivos para sospechar** que se trate de datos erróneos. Por otro lado, los valores mínimos de sol y lluvia son 0, por lo que no hay errores en ese sentido, y los máximos de lluvia pueden encontrar explicación en las precipitaciones torrenciales que se han dado en varios momentos en los últimos años.

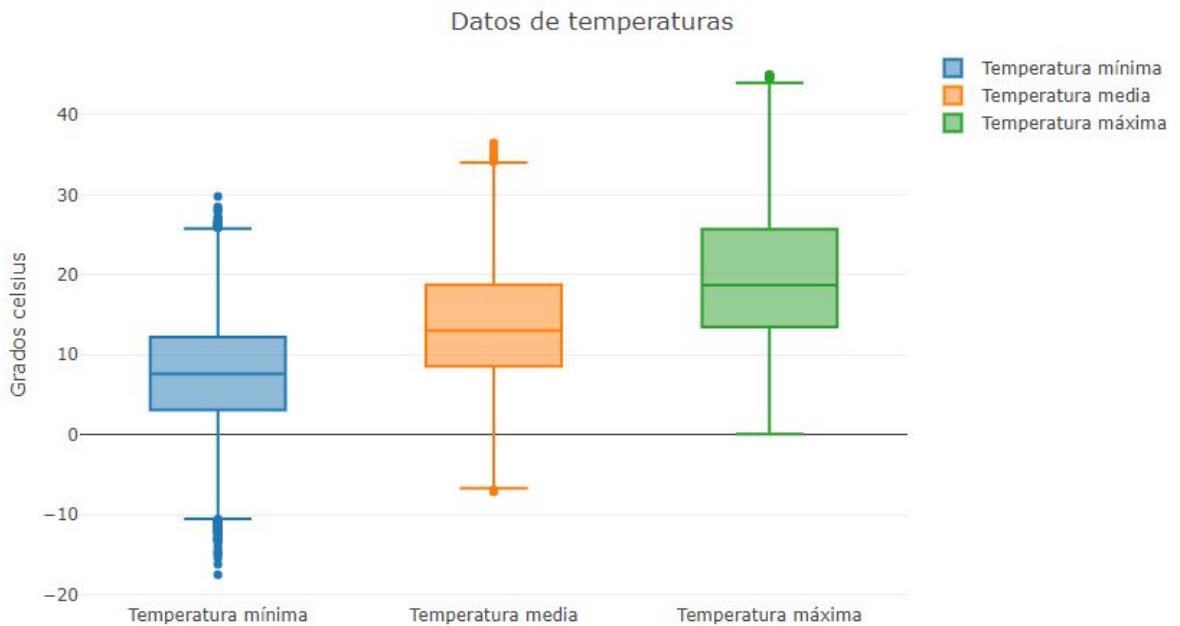


Figura 4.20: Diagramas de caja de las temperaturas diarias.
Fuente: elaboración propia.

- Verificación del dominio en `km`, `luna`, `longitud`, y `latitud`. De todas esas variables se conoce su dominio de antemano y, efectivamente, se verifica que **todos los valores están dentro** de él.

En cuanto a las **variables categóricas**, se comprueba que todas ellas sean correctas y los **niveles** de los factores también. Existen solo dos excepciones que se tratan, para lo que se elimina el único retorno de carro que existe para que todos los niveles tengan la misma forma y se homogeneiza la posibilidad de una velocidad máxima de 48 kilómetros por hora a 50, puesto que 48 es imposible según la legislación española.

4.4.1.2. Análisis de normalidad

Debido a que todas las variables contienen muchas más de 30 observaciones, se considera que **su tamaño es lo suficientemente grande como para asumir normalidad** aplicando el teorema del límite central (TLC) [35].

Con el objetivo de afinar todos los cálculos al máximo, se decide evaluar cada una de las variables numéricas, descartando la `longitud` y `latitud`. En una primera evaluación visual se observa que probablemente ninguna de las variables siga una distribución normal, siendo el caso más dudoso el de `tmin`, como se aprecia en la Figura 4.21, extraída del [análisis completo](#).

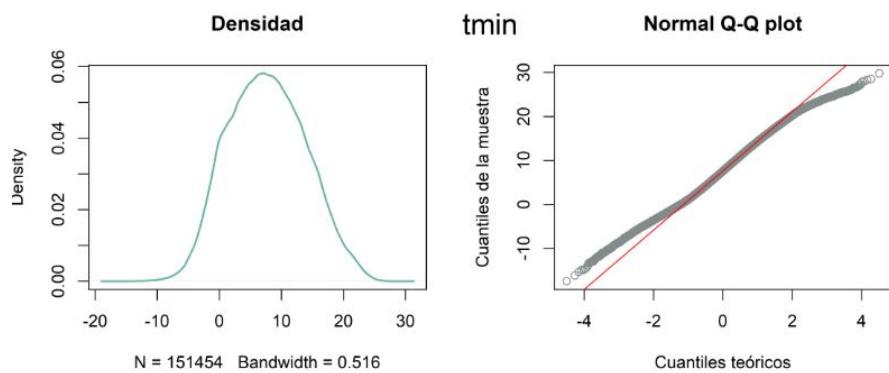


Figura 4.21: Gráfico de densidad y Q-Q de la variable `tmin`.

Fuente: elaboración propia.

Como existen dudas sobre algunas variables, se realiza el contraste de normalidad de **Lilliefors** en todas ellas. Debido a que el *p-value* es muy inferior $\alpha = 0.05$ en todos los casos, se rechaza la hipótesis nula y se concluye que **ninguna variable procede de una distribución normal**, tal y como se intuyó tras la inspección visual.

4.4.1.3. Análisis de correlación

Para analizar la correlación [120] se usa el coeficiente de **correlación de Spearman**, dado que se trata de un test no paramétrico y los datos no siguen una distribución normal. Sin embargo, también se realizan las pruebas con el coeficiente de **correlación de Pearson**, ya que se puede aplicar el teorema del límite central (TLC) sobre la muestra al tener un tamaño suficientemente grande. Las diferencias que se aprecian entre ambos correlogramas son insignificantes, como se puede ver en la Figura 4.22 por lo que se puede llegar a las mismas conclusiones.

Las únicas variables que parecen correlacionadas son las relativas a las temperaturas y se verifica que dicha conclusión sea correcta mediante dos test de correlación por cada par de atributos con un nivel de significancia de $\alpha = 0.05$. Los test arrojan que las variables `tmin`, `tmed` y `tmax` están altamente correlacionadas de forma positiva, por lo que no podrán

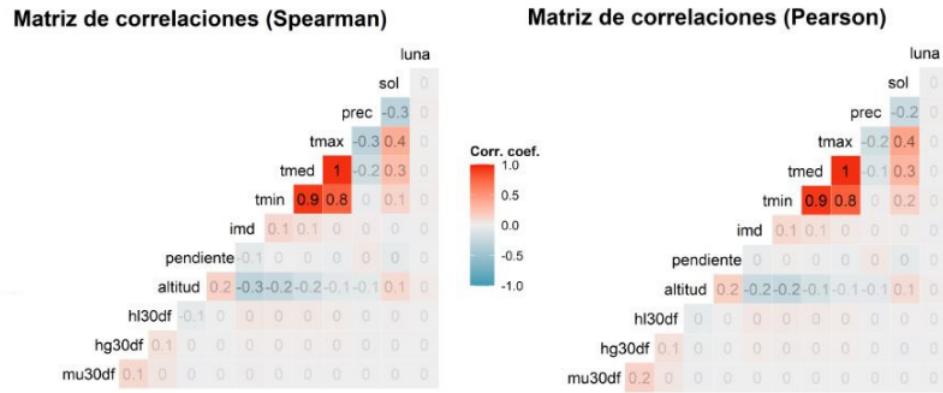


Figura 4.22: Correlogramas entre las variables numéricas.

Fuente: elaboración propia.

ser introducidas las tres en los modelos de forma simultánea. Este resultado era esperable, sobre todo el de `tmed` puesto que es resultado de una combinación lineal de las otras dos.

Otra conclusión sorprendente es que no se encuentran correlaciones entre variables que la intuición podría haber señalado erróneamente, como entre `maxspeed` y los fallecidos o heridos.

4.4.1.4. Características generales de los accidentes

Como una primera aproximación a las características de los accidentes, se estudia su **frecuencia**, para lo que la inspección visual sirve de apoyo, aunque por motivos de espacio se recomienda consultarlos en el [análisis completo](#). Para ello, se realiza un histograma por cada una de las **variables continuas** y las dos que más llaman la atención son la de `luna` e `imd_total`, como se ve en la Figura 4.23.

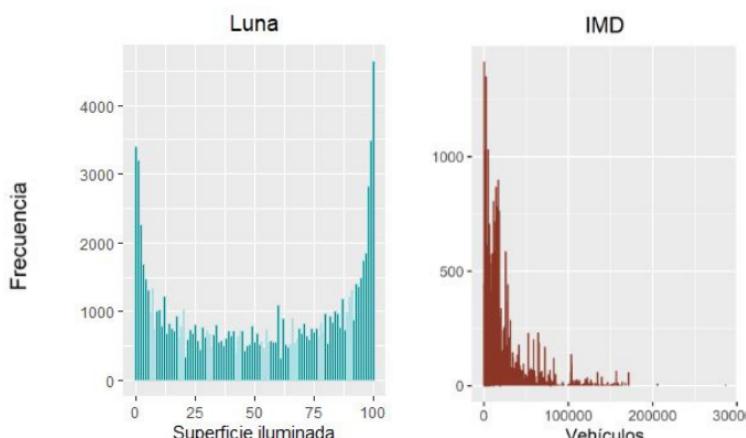


Figura 4.23: Histogramas de las variables `luna` e `imd_total`.

Fuente: elaboración propia.

En cuanto a las **variables categóricas**, y teniendo en cuenta que en la temporalidad y espacialidad se profundiza en apartados específicos, aquellas con más niveles se estudian con **gráficos de barras interactivos**, que facilitan su interpretación y selección, apoyados en una leyenda. En la Figura 4.24 se muestran los que podrían ser más relevantes para este proyecto.

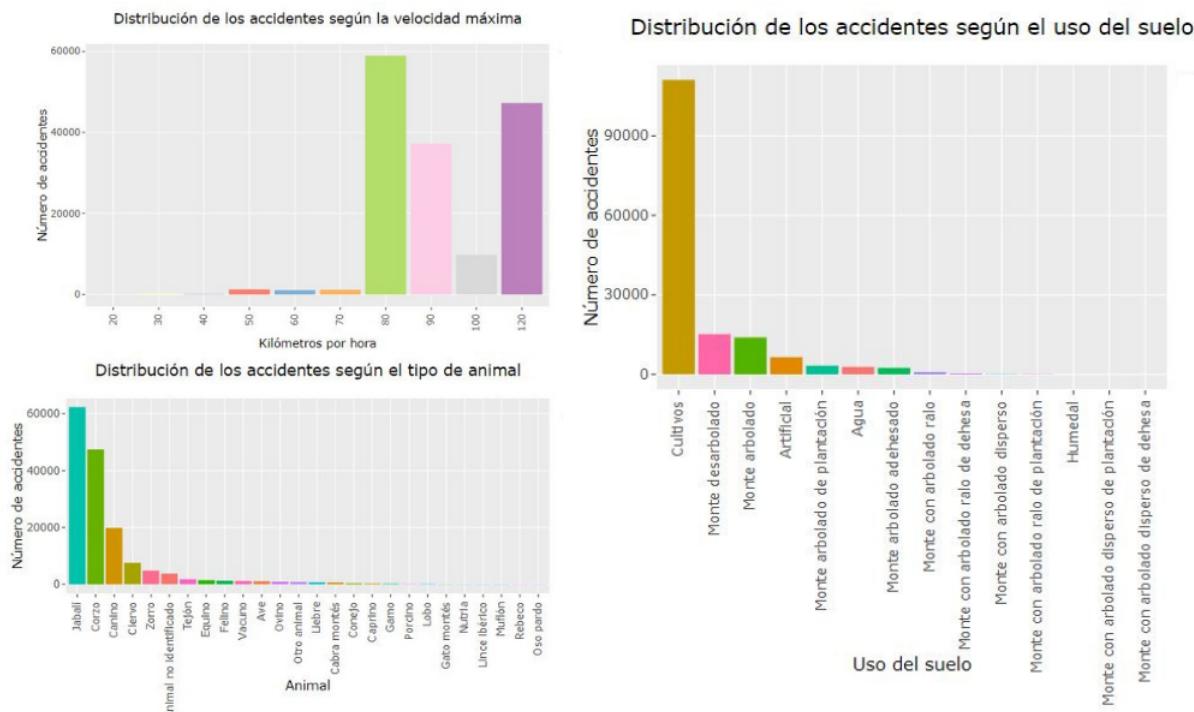


Figura 4.24: Gráficos de `maxspeed`, `uso_suelo` y `nombre_tipo_animal_1f`.
Fuente: elaboración propia.

Por último, aquellas variables con menos opciones de clasificación se estudian mediante **gráficos circulares**, que permiten detectar qué parte del todo constituye cada categoría, como se ve en los ejemplos de la Figura 4.25.



Figura 4.25: Gráficos de `nombre_sentido`, `nombre_tipo_via` y `nombre_tipo_animal_2f`.
Fuente: elaboración propia.

Las **conclusiones** que se obtienen de esta parte del análisis son que en la mayoría de accidentes con animales, las consecuencias no conllevan víctimas mortales (`total_mu30df`), ni **víctimas de ningún tipo** (`total_hg30df` y `total_h130df`), afirmación que se confirma con el análisis de las variables `nombre_ind_accd` y `nombre_ind_acciv`.

Del análisis de las variables continuas también se comprueba, como se sospechaba, que la proporción de superficie de Luna iluminada (`luna`) en el momento del accidente puede llegar a ser un factor a tener en cuenta, sobre todo en fechas cercanas a **Luna nueva y Luna llena**. Además, al contrario de lo que ocurre con otros tipos de accidentes de tráfico, se descubre que la mayor frecuencia de accidentes se da cuando el `imd_total` es más bajo, es decir, en las vías cuya media de **densidad circulatoria es menor**.

Por otro lado, los **jabalíes, corzos y ciervos** son los animales presentes en la mayoría de los accidentes, como ya adelantaban los estudios analizados en el estado del arte (Capítulo 2). Sin embargo, aquellos clasificados como ‘caninos’ se sitúan en el número tres, por lo que hay que tenerlo en cuenta ya que, al ser también clasificados como domésticos por la DGT, no se tienen sus datos de distribución del GBIF.

Además, se descubre que la mayoría de los accidentes se producen en vías cuya **velocidad máxima permitida** `maxspeed` es de 80 kilómetros por hora o más. Asimismo, la gran mayoría de los accidentes se producen en áreas rodeadas de **cultivos**.

Por último, la **titularidad de la vía**, el **tipo** y el **sentido** no aportan información que parezca que llegue a ser relevante en este proyecto.

4.4.1.5. Análisis temporal

Para realizar el análisis temporal se emplean varias técnicas. Los gráficos de barras ayudan a **comparar períodos de tiempo** entre ellos, como los de la Figura 4.26.

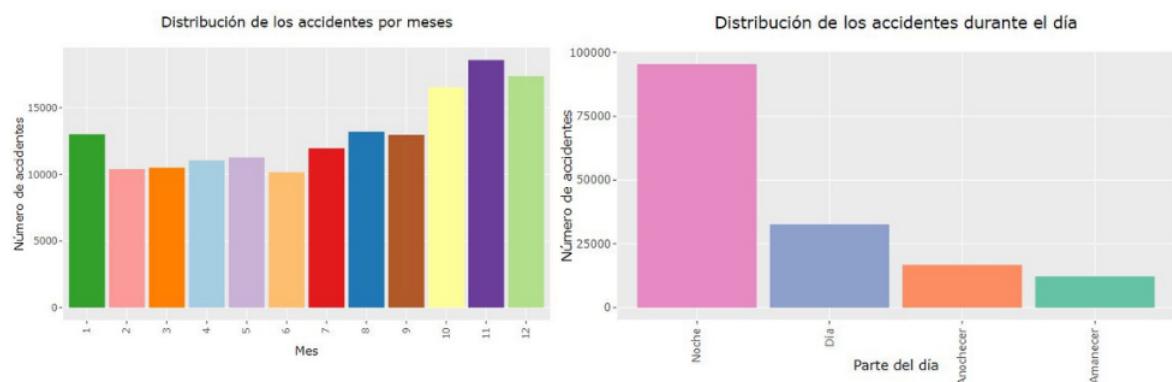


Figura 4.26: Frecuencia de `mes_1f` y `parte_dia`.

Fuente: elaboración propia.

Además, se estudian las **series temporales**, tanto completa como por años individuales, mediante **gráficos interactivos** que permiten seleccionar la **granularidad** del periodo en cada año para compararlos entre ellos de forma cómoda e intuitiva. Como la la Figura 4.27 es estática por las características de este soporte, se recomienda explorarlos en el [análisis completo](#).

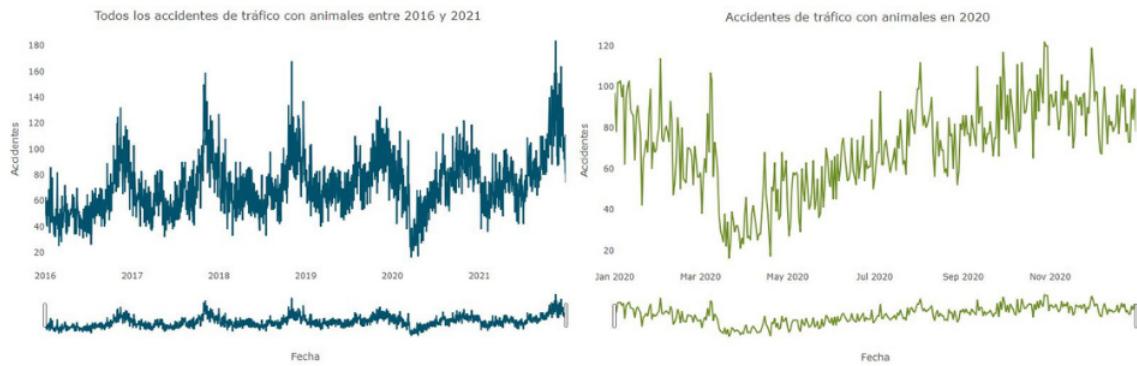


Figura 4.27: Serie temporal completa y del año 2020.

Fuente: elaboración propia.

Asimismo, se estudian los resultados de la **función de autocorrelación (ACF)** con un intervalo de confianza del 95 %, representado por las dos líneas horizontales discontinuas en los gráficos como el de la Figura 4.28. Por último, con el test de **Raíz Unitaria de Kwiatkowski (KPSS)** se estudia si los datos son estacionarios tanto en su conjunto como para cada uno de los años.



Figura 4.28: Gráfico ACF 2016-2021.

Fuente: elaboración propia.

Las conclusiones del análisis temporal son muy interesantes. De **octubre a enero** son los meses en los que más accidentes con animales involucrados se registran. No existe tal diferencia si se comparan **días de la semana** entre ellos. Lo más destacado, sin embargo, es que el 61 % de los accidentes se producen de **noche** mientras que solo el 21 % de **día** (10 horas), que es poco más que la suma del 11 % del **anochecer** (1 hora) y el 8 % del **atardecer** (1 hora).

En cuanto a la evolución de los accidentes en el tiempo, se pueden observar **tendencias similares comparando todos los años** entre sí. Como era previsible por los efectos de la pandemia de coronavirus, **2020 es un año anómalo**.

El análisis de las autocorrelaciones arroja que la **serie temporal no es aleatoria** ni en el estudio de su conjunto ni para cada uno de los años, incluido el 2020. Además, existe un **alto grado de autocorrelación** entre todas las observaciones adyacentes, casi adyacentes y no adyacentes.

A pesar de esta no aleatoriedad y fuerte autocorrelación, se verifica estadísticamente que los datos **no son estacionales**.

4.4.1.6. Análisis espacial

Se aborda el análisis de la componente espacial de más a menos área de estudio. Por ello, se estudia la distribución de los accidentes **por comunidades autónomas** y por **provincias**. A partir de esa investigación surge la duda de si los accidentes se distribuyen igual en todas las partes de España **según el tipo de animal**, gráficos que se pueden ver en la Figura 4.29.

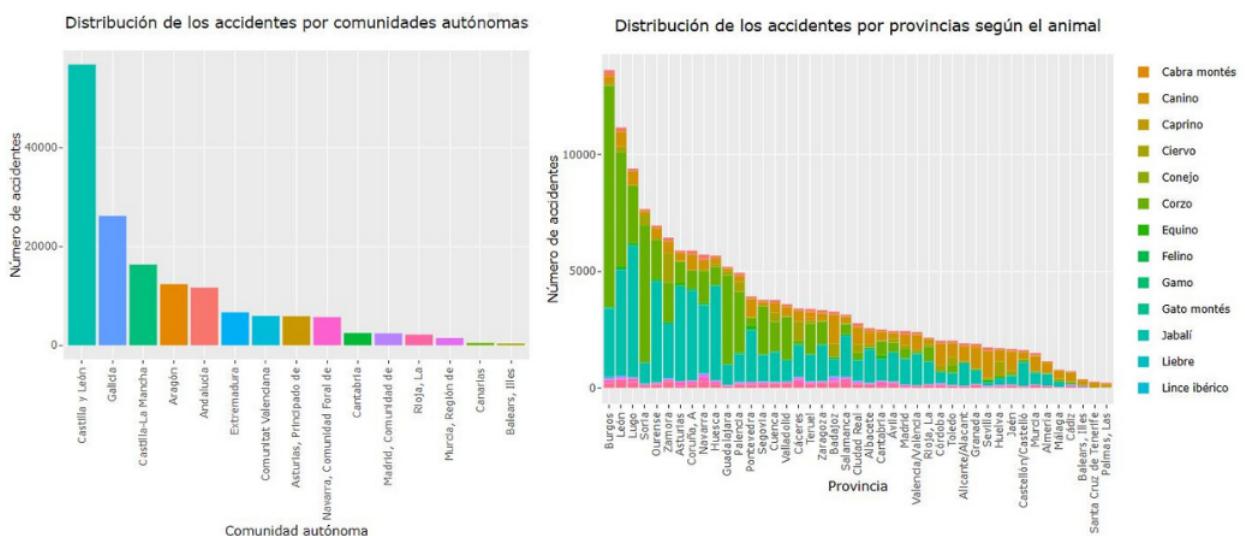


Figura 4.29: Distribución de los accidentes por comunidades y provincias.

Fuente: elaboración propia.

Además, como hay muchos **municipios** diferentes, se opta por mostrar en el **análisis completo** las frecuencias en forma de tabla por si hay que necesitar recurrir a ella en el futuro.

A la luz de toda la información obtenida, surgen dos necesidades: concretar mejor las **zonas con alta probabilidad** de este tipo de colisiones y hacerlo teniendo en cuenta el **tipo de animal** en cada caso. Este es el motivo por el que se decide desarrollar diferentes **capas de**

estimación de densidad de Kernel (KDE), lo que permite calcular la función de densidad de probabilidad de los accidentes con animales involucrados.

Esta información es muy relevante y se va a **productivizar** para poder emplearse, por lo que para seguir con la línea del proyecto, se deciden hacer los **cálculos en Python**, que se pueden consultar en el directorio `kde` del repositorio en GitHub. De este modo, además de la visualización para cada tipo de animal, el programa guarda en un fichero para cada animal un *array* de tres dimensiones con la localización x , y y la densidad de probabilidad z para todo el país. Esto permite disponer de un modelo digital del terreno (MDT) en el que la coordenada z representa la probabilidad de tener un accidente con el tipo de animal asociado. Para la generación de este fichero, se ha utilizado un algoritmo *kernel density* [121] [122], empleando como entrada la localización (longitud, latitud) y el tipo de animal.

En la Figura 4.30, se muestran dos ejemplos a partir de la visualización del MDT obtenido anteriormente, aunque todos se encuentran recogidos en el [análisis completo](#), teniendo en cuenta que cada imagen no es la información más relevante del proceso.

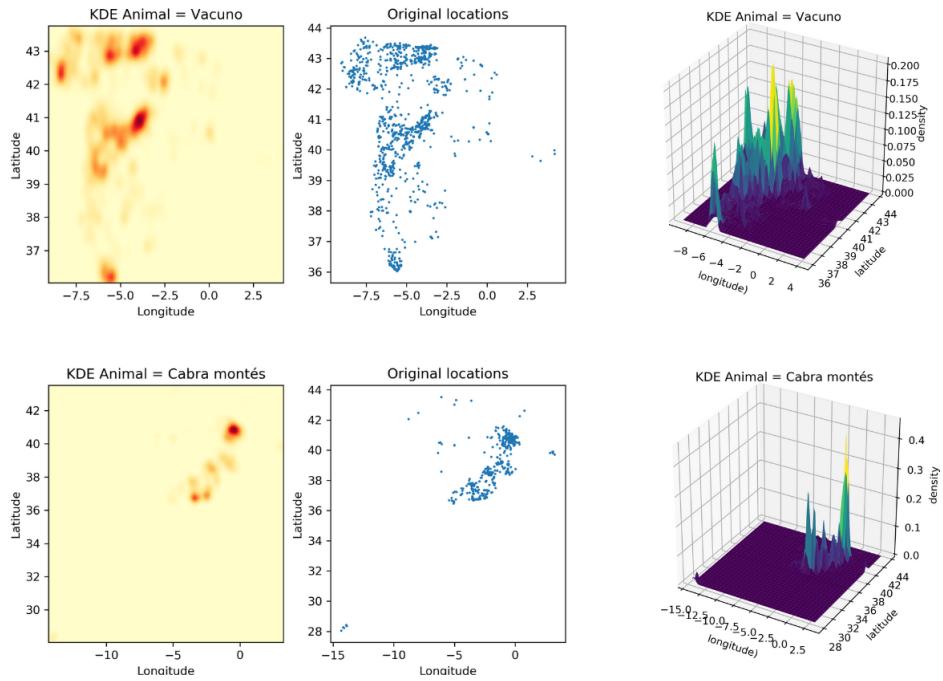


Figura 4.30: KDE de vacuno y de cabra montés.

Fuente: elaboración propia.

En conclusión, los datos tienen una **componente espacial** que puede ser crítica para el proyecto. Por tanto, no es tan importante conocer en qué comunidades autónomas, provincias o municipios se produce cada accidente con animales, sino el análisis de cada registro en una **ubicación concreta**. Esto es así porque, aunque el **jabalí** es el animal más involucrado en accidentes de tráfico, la probabilidad de accidente con un tipo de animal u otro varía en función

del lugar. Por ejemplo, en Soria es mucho más probable chocarse con un corzo. Además, se observa que en **País Vasco** y **Cataluña** no hay registros y esto es debido a que los datos no han sido proporcionados por la DGT.

Así, la **distribución de los accidentes depende de cada tipo de animal**, lo cual era previsible porque la fauna no se distribuye de forma homogénea por España, tal y como se pudo comprobar al analizar los datos procedentes de GBIF. Por tanto, el conocimiento más relevante que se extrae de este punto son los *arrays* resultantes que contienen la **estimación de densidad de kernel** de los accidentes para cada animal.

4.4.2. Descripción de los atributos

En la Tabla 4.3 se detallan las 50 variables que describen cada uno de los 157.094 accidentes de tráfico con animales recogidos en el conjunto de datos `accidentes_si` que se encuentra en la base de datos `tfm`, resultado de la fase de procesado y análisis.

Atributo	Descripción	Ejemplo
<code>id_num</code>	Identificador único	201822058460
<code>ind_accda</code> <code>nombre_ind_accda</code>	Valor 1 si hay daños materiales exclusivamente y 0 si no Valores anteriores decodificados	0 No es un accidente de daños exclusivamente
<code>ind.acciv</code> <code>nombre_ind.acciv</code>	Valor 1 si hay víctimas exclusivamente y 0 si no Valores anteriores decodificados	1 Accidente con víctimas exclusivamente
<code>total_mu30df</code> <code>total_hg30df</code> <code>total_hl30df</code>	Número total de fallecidos (mu), hospitalizados (hg) y no hospitalizados (hl) registrados en el accidente, a 30 días	0 0 3
<code>fecha_accidente</code>	Fecha del accidente	2018-03-22
<code>hora_accidente</code>	Hora del accidente	22:40
<code>mes_1f</code> <code>nombre_mes</code>	Número del mes: de 1 enero a 12 diciembre Número del mes decodificado	3 Marzo
<code>anyo</code>	Año en el que se produce el accidente	2018
<code>ccaa_1f</code> <code>nombre_ccaa</code>	Código de la comunidad autónoma, normalizado por el INE Nombre de la comunidad autónoma, decodificando <code>ccaa_1f</code>	2 Aragón
<code>provincia_1f</code> <code>nombre_provincia</code>	Código de la provincia, normalizado por el INE Nombre de la provincia, decodificando <code>provincia_1f</code>	22 Huesca
<code>cod_municipio</code> <code>nombre_municipio</code>	Código del municipio, normalizado por el INE Nombre del municipio, decodificando <code>cod_municipio</code>	22048 Barbastro
<code>carretera</code>	Denominación de la carretera	N-240
<code>km</code>	Punto kilométrico	163.85
<code>sentido_1f</code> <code>nombre_sentido</code>	Valor 1 si es ascendente, 2 descendente, 3 ambos o 4 se desconoce Valores anteriores decodificados	2 Descendente
<code>tipo_via_3f</code> <code>nombre_tipo_via</code>	Valor 1 para autopista y autovía o 2 para el resto vías interurbanas Valores anteriores decodificados	2 Resto vías interurbanas
<code>titularidad_via_2f</code> <code>nombre_titularidad_via</code>	Valor 1 para estatal, 2 para autonómica, 3 para provincial, cabildo/conSELL, 4 para municipal, 5 para otra o 999 sin especificar Valores anteriores decodificados	1 Estatal

Atributo	Descripción	Ejemplo
<code>tipo_animal_1f</code>	Valor 0 para animal no identificado, 1 ave, 2 cabra montés, 3 ciervo, 4 conejo, 5 corzo, 6 gamo, 7 gato montés, 8 jabalí, 9 liebre, 10 lince ibérico, 11 lobo, 12 muflón, 13 nutria, 14 oso pardo, 15 rebeco, 16 tejón, 17 zorro, 18 otro animal, 19 ave, 20 canino, 21 caprino, 22 equino, 23 felino, 24 ovino, 25 porcino, 26 vacuno o 27 otro animal Valores anteriores decodificados	8 Jabalí
<code>tipo_animal_2f</code>	Valor 0 para animal no identificado, 1 para silvestre o 2 para doméstico Valores anteriores decodificados	1 Silvestre
<code>longitud</code>	Longitud en formato decimal	0.08136
<code>latitud</code>	Latitud en formato decimal	42.03089
<code>geom</code>	Representación espacial de tipo POINT	POINT (0.08136 42.03089)
<code>dia_semana</code>	Número del día de la semana: de 1 lunes a 7 domingo	4
<code>nombre_dia_semana</code>	Número del día de la semana decodificado	Jueves
<code>tipo_dia</code>	Tipo del día de la semana: diario o finde	Diario
<code>parte_dia</code>	Parte del día: amanecer, día, anochecer o noche	Noche
<code>luna</code>	Superficie de Luna iluminada de 0 a 100 si es nocturno y NULL si es diurno	22
<code>prec</code>	Precipitación diaria de 07h a 07h (mm = 1/m2)	0
<code>tmin</code>	Temperatura mínima del día (°C)	-1.5
<code>tmax</code>	Temperatura máxima del día (°C)	13.5
<code>tmed</code>	Temperatura media diaria (°C)	6
<code>sol</code>	Horas de sol ese día	11.2
<code>uso_suelo</code>	Uso mayoritario del suelo en el área de 500 metros	Cultivos
<code>altitud</code>	Altitud en metros sobre el nivel del mar	487.202
<code>pendiente</code>	Mediana de la pendiente en un buffer de 30 metros	0.8585
<code>taxonkey</code>	Valor 2441054 para cabra montés, 2440958 ciervo, 2436940 conejo, 5220126 corzo, 5220136 gamo, 7964291 gato montés, 7705930 jabalí o porcino, 2436691 liebre, 2435261 lince ibérico, 5219173 lobo, 2441116 muflón, 2433433 oso pardo, 5220170 rebeco, 2433875 tejón, 5219243 zorro, 2441056 caprino, 2440886 equino, 2435035 felino, 2441110 ovino o 2441022 vacuno	7705930
<code>imd_total</code>	Intensidad media diaria de circulación en esa vía ese año	236
<code>maxspeed</code>	Velocidad máxima en el tramo de carretera	100

Cuadro 4.3: Descripción de los atributos del dataset `accidentes_si`.

Capítulo 5

Generación y evaluación de los modelos

En este capítulo se construye y preprocesa el conjunto de datos final que se introduce en los algoritmos de aprendizaje supervisado, el cual se divide en los subconjuntos de entrenamiento y de *test*. Además, se generan los modelos de predicción con la mejor combinación de hiperparámetros en cada caso y se evalúa el rendimiento de cada uno de ellos así como el peso de las variables. Por tanto, se trabajan las fases de modelado de datos y evaluación de los resultados. Todo el código empleado para este capítulo se encuentra dentro del directorio `models` del repositorio.

5.1. Creación del conjunto de datos para los modelos

Antes de generar cualquier modelo se ha de tener un conjunto de entrenamiento (*train*) y de validación (*test*) correctamente construidos para que sea capaz de **generalizar y poder verificarlo**, evitando posibles sesgos [13].

En el capítulo anterior se ha trabajado en el *dataset* con todos los accidentes de tráfico con animales involucrados en España entre 2016 y 2021 y, tras haber procesado todos los datos, se tienen 157.094 registros geolocalizados en total. Para la construcción del conjunto de datos necesario en esta fase, por tanto, se decide introducir otros **157.094 registros falsos pero posibles**, es decir, puntos con características reales en los que podría haber habido un accidente con un animal involucrado pero realmente no ha sido así. Para ello, se emplean dos métodos:

1. Creación de **registros aleatorios**.
2. **Extracción de registros** de los conjuntos de datos de accidentes con víctimas, descartando aquellos en los que sí que hubiera animales involucrados.

5.1.1. Registros aleatorios

Para balancear el conjunto de datos final, se decide introducir registros en los que no se haya producido un accidente de tráfico según los datos, pero sea posible que hubiera ocurrido. Para ello, es necesario que esté ubicado **dentro de las red de carreteras** bajo titularidad de la DGT, que es el dominio de este proyecto, cuyas geometrías están en la tabla `carreteras` ya importada en la base de datos.

El proceso que se sigue comienza con la creación de la tabla `puntosnoaccidentes` en la base de datos `tfm` con la DDL `creacion_puntosnoaccidentes.sql`. A continuación, se ejecuta cinco veces el *script* `puntosnoaccidentes.sql`, que genera el `geom` de 73.875 puntos aleatorios, cinco en cada una de las 14.775 vías contenidas en el conjunto de datos de las geometrías de las carreteras de la DGT. La segunda sentencia de dicho *script* imputa el valor de la longitud y latitud de cada registro en los respectivos campos. Por último, con `accidentes_no.sql`, se crea la tabla `accidentes_no`, que sigue la misma estructura que la tabla `accidentes_si`, y se vuelcan los campos `id`, `geom`, `lon` y `lat` a la nueva tabla ejecutando la última sentencia de `puntosnoaccidentes.sql`.

5.1.2. Accidentes con víctimas sin animales involucrados

Por otro lado, es interesante introducir en el conjunto de datos que se está construyendo registros de **accidentes de tráfico reales sin animales** involucrados con el objetivo de crear modelos capaces de diferenciar aquellos en los que sí que ha habido fauna. Para ello, se vuelve a trabajar con la tabla `accidentes_decodificado` que se elaboró hasta el 25 de noviembre de 2022 con los datos de los accidentes con víctimas [38] [39] [40] [41] [42] [43]. En total, hay 200.785 accidentes registrados con localización en los que no hay animales involucrados.

Por tanto, se seleccionan aleatoriamente 13.869 registros de 2016 y 13.870 del resto de años (83.219 en total) y se introducen en la tabla `accidentes_no` con el *script* `accidentes_victimas_accidentes_no.sql`.

De esta forma el *dataset* final de registros no verdaderos contiene un total de 157.094 registros, los mismos que el de accidentes reales con animales.

5.1.3. Campos de los nuevos registros

Ya se tiene la tabla `accidentes_no` con todos los registros que la conformarán, pero se pueden enriquecer completando la **información de los campos para que sean de utilidad** en los modelos. Para ello, o bien se **reutilizan** los *scripts* del Capítulo 4 adaptados o se hace mediante un **nuevo código**, dependiendo de las necesidades en cada caso:

- `id_num`. El campo se ha generado automáticamente al volcar los datos de los registros.
- `longitud`, `latitud` y `geom`. Todos los registros tienen ya su correspondiente información espacial, como se ha señalado en el proceso de creación.
- `fecha_accidente` y `hora_accidente`. En el caso de los registros que se han generado de forma aleatoria, se obtienen también aleatoriamente estos dos campos. Para aquellos cuyo origen son los accidentes con víctimas, se decide generar la fecha completa a partir del campo `mes_1f` y `año`, puesto que sí que tienen valores, y para la hora simplemente se atribuyen aleatoriamente los minutos para que el campo esté completo, ya que esta también se encuentra documentada. Todo este proceso se hace con el *script* `fecha_hora_accidente.sql`, en el que se pueden consultar todas las sentencias.
- `dia_semana`, `nombre_dia_semana`, `tipo_dia`, `mes_1f`, `nombre_mes` y `año`. Los datos procedentes de los accidentes con víctimas ya tienen estos atributos completos con información real y para aquellos generados aleatoriamente se extrae dicha información de la `fecha_accidente` recientemente creada y se decodifica con las tablas `aux_dia_semana` y `aux_mes`, subidas a la base de datos en fases anteriores. Además, se clasifican los días de todos los registros en ‘Diario’ o ‘Finde’, según de cuál se trate. La adaptación completa se puede consultar en el *script* `dia_mes_anho.sql`.
- `parte_dia`. Con la adaptación de `partofdaydb.py` y su `utils.py`, se clasifican los registros en ‘Amanecer’, ‘Día’, ‘Atardecer’ y ‘Noche’, según el criterio explicado en el Capítulo 4. Para ello, se atiende a su `hora_accidente`, `fecha_accidente`, `longitud` y `latitud`.
- `luna`. La información de la superficie de la Luna iluminada en el caso de que el accidente se haya producido por la noche se calcula con el *script* `moonlightdb.py`, que hace uso de `utils.py` y que ha sido adaptado para este *dataset*, para lo que tiene en cuenta la `fecha_accidente`.
- `maxspeed`. Se adaptan `maxspeeddb.py` y su `utils.py` para que a partir de la `longitud` y `latitud` de cada registro, se calcule la velocidad máxima en dicho punto de la carretera mediante el proceso que se explica en el Capítulo 4.
- `imd_total`. La adaptación que se recoge en el *script* `imdsdb.py` y el `utils.py` permite nutrir el campo a partir de los datos existentes de `longitud`, `latitud` y `año`, afinando la precisión en el caso de que se desconozca la `carretera`.

- `uso_suelo`. Se calcula el uso del suelo en un *buffer* a 500 metros a cada lado de las carreteras y se imputa el valor de mayor extensión mediante `landusedb.py`, adaptación del *script* homónimo de la fase anterior del que también se edita su `utils.py`.
- `altitud`. Además de adaptar el código a la tabla `accidentes_no`, en `elevationsdb.py` y su `utils.py` se incluye el tratamiento de un *bug* no detectado antes debido a que ningún registro se encontraba en los límites de la hoja con un formato diferente ('1105/1108'), por lo que también se actualiza el código en el *script* inicial desarrollado en el Capítulo 4.
- `pendiente`. Al igual que con la altitud, se adapta el código de `slopesdb.py` y su `utils.py` a la nueva tabla y se arregla el *bug* detectado para evitar incompatibilidades.
- `sol`, `prec`, `tmin`, `tmed` y `tmax`. Como la API de la AEMET quedó descartada, directamente se adapta `meteohisdb.py` y su `utils` a la tabla `accidentes_no` para completar los campos a partir de la información temporal y espacial de cada registro.
- `ind_accda`, `nombre_ind_accd`, `ind_acciv`, `nombre_ind_acciv`, `total_mu30df`, `total_hg30df` y `total_hl30df`. Son campos que describen consecuencias de los accidentes, por lo que no se pueden emplear para predecirlos, de manera que no se completan. En cualquier caso, los totalizadores `total_mu30df`, `total_hg30df` y `total_hl30df` sí que tienen valor para los registros procedentes de los accidentes con víctimas.
- `ccaa_1f`, `nombre_ccaa`, `provincia_1f`, `nombre_provincia`, `cod_municipio`, `nombre_municipio`, `carretera` y `km`. En el caso de los registros procedentes de los accidentes de tráfico con víctimas, estos datos se han importado porque ya existían, a excepción de la comunidad autónoma. Sin embargo, aunque esta información es sencilla de calcular y en el caso de los registros que se crearon aleatoriamente también se podría obtener esos campos a partir de su `longitud` y `latitud` o de su `geom`, se decide no hacerlo porque el dato que se tiene de localización es mucho más preciso que una región tan grande como las mencionadas.
- `tipo_animal_1f`, `nombre_tipo_animal_1f`, `tipo_animal_2f`, `nombre_tipo_animal_2f` y `taxonkey`. Son atributos relacionados con el animal con el que se tiene el accidente en caso de haber alguno involucrado, por lo que se mantienen como nulos.
- `sentido_1f`, `nombre_sentido`, `tipo_via_3f`, `nombre_tipo_via`, `titularidad_via_2f` y `nombre_titularidad_via`. Son características de la carretera en la que se ubica el accidente, pero según la documentación de los accidentes con animales [50] y los accidentes con víctimas [44], los criterios empleados en ambos casos son diferentes, por lo que no se pueden unificar y se opta por descartar estos campos.

Por último, se lleva a cabo el mismo procesado de los datos que con el conjunto de datos de accidentes con animales. Como debido al proceso de construcción no hay ningún registro con el `geom` sin valor, únicamente se imputa como nulo el valor -999 del campo `altitud` y -9999 del campo `pendiente`. Para ello se emplean las sentencias de `procesado_accidentes_no.sql`.

5.1.4. Dataset completo en la base de datos

En primer lugar, se ejecuta `creacion_datos.sql`, que contiene las sentencias para generar la tabla `datos`, producto de la unión de `accidentes_sí` y `accidentes_no`. Para ello, mediante ese mismo *script* previamente se crea la variable `target`, que toma el valor 0 en el caso de `accidentes_no` y 1 en el de `accidentes_sí`, quedando así definida la **variable objetivo, target o clase** a predecir.

Una última variable que se trata todavía en la tabla `datos` dentro de la base de datos `tfm` es la de `geom`, puesto que su formato no es adecuado para ser introducido en los modelos y las de `longitud` y `latitud` podrían meterse, pero los algoritmos no sabrían discernir que se trata de **datos espaciales**. Por tanto, es el momento de buscar una alternativa que permita mantener esta información y que ayude a potenciar la capacidad predictora de los modelos. Para ello, se estudian dos opciones:

1. **H3.** Sistema de indexación geoespacial jerárquica hexagonal patentado por Uber Technologies, Inc. que permite una serie de algoritmos y optimizaciones basados en la malla, como los vecinos más próximos, el camino más corto, el suavizado de gradientes, etc [123]. El problema es que los hexágonos se superponen en las fronteras, lo que tiene sentido para no perder el seguimiento de los coches en movimiento cuando pasan de una celda a otra, como es el negocio de Uber, pero en este proyecto se incuraría en el riesgo de que un accidente esté localizado en dos hexágonos al mismo tiempo, por lo que se descarta.
2. **Geohash.** Sistema de geocódigos de dominio público creado por Gustavo Niemeyer en 2008 que permite la codificación de la ubicación geográfica en un texto corto de letras y números, es decir, la conversión da lugar a un código alfanumérico. Una característica importante es que los *geohashes* ofrecen propiedades como precisión arbitraria, prefijos similares para posiciones cercanas y la posibilidad de eliminar gradualmente caracteres del final del código para reducir su tamaño (y perder gradualmente precisión) [124]. Se opta por su uso porque cada registro puede tener únicamente un *geohash* asociado y ya se ha empleado con éxito, tanto en investigación con áreas espaciales [125] como en otros problemas de tráfico [126]. Con las sentencias finales de `creacion_datos.sql` se crean las columnas `geohash5`, `geohash6` y `geohash7`, que recogen la información equivalente a 5, 6 y 7 dígitos respectivamente, es decir, de celdas más extensas (5) a menos (7).

5.1.5. Conjuntos de entrenamiento y *test*

Para facilitar el seguimiento del desarrollo del resto del proyecto, a partir de este punto se recoge paso a paso en el **Jupyter Notebook** `models.ipynb` dentro del mismo directorio homónimo en el que se está trabajando. En cualquier caso, a continuación se realiza un resumen de las decisiones tomadas y los pasos más importantes.

5.1.5.1. Tratamiento de los datos previo a su división

Una vez realizada la conexión a la base de datos `tfm`, se carga la tabla `datos`, la última que se ha trabajado, en el *dataframe* `data`. Aunque ya se ha hecho el preprocesado, es necesario realizar unas **transformaciones finales** para que tengan el formato adecuado:

- `geohash`. Estos campos contienen información alfanumérica que no se puede introducir en los modelos, por lo que se busca una solución para que sea numérica. Se crea una función que convierte cada carácter a **formato ASCII** y se imputa el resultado a una nueva columna llamada `ascii_geohash` junto al número de referencia.
- `fecha_accidente`. Es de tipo `datetime`, por lo que se busca una solución para que sea de tipo numérica, probándose tres opciones:
 1. Emplear `año`, `mes` y `dia` como campos separados. Se crea la columna `dia`.
 2. Concatenar el año, el mes y el día. Se quitan los guiones de `fecha_accidente`.
 3. Convertir `fecha_accidente` a su fecha en el calendario juliano.
- `luna`. El problema de este campo numérico reside en que no toma valor cuando el accidente no ha sido por la noche, sumando 169.163 NaN. Sin embargo, en la fase de análisis se vio que contenía información que podía ser relevante, por lo que se discretiza para después binarizarla (*dummy*). Aquellos registros que no hayan sido de noche tendrán un 0 en todas las categorías y un 1 en la correspondiente los que fueran por la noche.
- `altitud`. Presenta un 5,33 % de nulos. Como todos los datos están localizados, se imputa la media de la altitud en función de su `geohash6`. Así, no se sesga la información puesto que no se distingue entre los registros verdaderos y falsos y a todos se les da la altura más aproximada a su situación geográfica en lugar de una media genérica.
- `pendiente`. La proporción de nulos asciende a 21,59 %. Sin embargo, puede ser un dato importante dado el comportamiento de algunos animales. Por ello, se sigue el mismo sistema que en `altitud` para no distorsionar la entrada a los modelos.

- `sol`, `prec` y `tmed`. Contienen un 26,42% de nulos cada una de ellas, pero también representan información que puede ser vital. No se puede imputar el valor medio de esa fecha concreta en el `geohash` de esa ubicación porque en esas zonas concretas no existen estaciones meteorológicas activas para el día requerido. Finalmente se opta por imputar el valor de la media en la fecha dada sin tener en cuenta la ubicación.
- `uso_suelo`, `tipo_dia`, `nombre_dia_semana`, `parte_dia`, `nombre_mes`. Se crean múltiples variables binarias *dummy* de cada uno de los atributos categóricos para que la información pueda ser correctamente interpretada por cada algoritmo.

5.1.5.2. Selección de los datos

Antes de dividir el *dataset* es importante **seleccionar qué datos** se van a introducir finalmente en los modelos. En este punto se tiene especial cuidado con no incurrir en el error conocido como ***data leakage***, que consiste en utilizar información sobre la clase que se intenta predecir para entrenar el modelo [127]. También se descartan variables de las que se posea información más específica, la misma información en otro formato o estén muy correlacionadas (multicolinealidad), consiguiendo una **reducción de la dimensionalidad**. Con la descripción de cada atributo realizada en el Capítulo 4 presente, las variables eliminadas son:

- `id_num`, `ind_accda`, `nombre_ind_accd`, `ind_acciv`, `nombre_ind_acciv`, `total_mu30df`, `total_hg30df`, `total_hl30df`, `tipo_animal_1f`, `nombre_tipo_animal_1f`, `taxonkey`, `tipo_animal_2f`, `nombre_tipo_animal_2f`, `tipo_via_3f`, `nombre_tipo_via`, `titularidad_via_2f` y `nombre_titularidad_via`. Introducirían en los modelos información de la clase, bien sea por el dato en sí o por su formato, debido a que en la integración de registros falsos y verdaderos había columnas codificadas de forma diferente según su origen.
- `fecha_accidente`, `hora_accidente`, `dia`, `mes_1f`, `ccaa_1f`, `nombre_ccaa`, `provincia_1f`, `nombre_provincia`, `cod_municipio`, `nombre_municipio`, `carretera`, `km`, `sentido_1f`, `nombre_sentido`, `longitud`, `latitud`, `geom`, `dia_semana`, `tmin`, `tmax`, `fecha_juliana`, `ascii_geohash5`, `ascii_geohash7`, `geohash5`, `geohash6`, `geohash7` y `luna`. La información de estos atributos ya se encuentra recogida en otros que permanecen en el conjunto de datos.

Así, el *dataframe* `selected_data` esta formado por las variables recogidas en la Tabla 5.1, transformadas en múltiples variables binarias las señaladas como categóricas (cat):

anyo	prec	sol	tmed
altitud	pendiente	maxspeed	fecha_num
nombre_mes (cat)	parte_dia (cat)	tipo_dia (cat)	nombre_dia_semana (cat)
luna_cat (cat)	uso_suelo (cat)	imd_total	ascii_geohash6
target			

Cuadro 5.1: Variables seleccionadas para la fase de modelado en `selected_data`.

Fuente: elaboración propia.

Por último, tal y como se ha visto en el análisis, el **año 2020** muestra un comportamiento anómalo debido a la pandemia de coronavirus, casuística extraordinaria que provocó el descenso brusco de los accidentes de tráfico con animales involucrados a consecuencia del confinamiento generalizado. Por tanto, se decide eliminarlo puesto que son circunstancias que no se prevé que se repitan. Así, el porcentaje de **accidentes cada año** queda relativamente homogéneo: 2016 registró un 18,29 %, 2017 un 19,58 %, 2018 un 19,96 %, 2019 un 20,53 % y 2021 un 21,64 %.

De este modo, el *dataframe* `selected_data` está compuesto por los **314.188 registros**, **157.094** de accidentes con animales y otros **157.094** que no son de accidentes con animales correctamente identificados, por lo que los datos están **balanceados** y se evitan posibles sesgos de algunos algoritmos en favor de la clase que más instancias tiene [32]. Además, se han evitado los errores de **leakage** y se ha reducido la dimensionalidad, quedando un conjunto de datos de **17 variables**, que en realidad son 56 por el proceso de conversión de las categóricas en binarias. Debido a este tamaño, no es necesario aplicar otras estrategias de reducción de dimensionalidad, como la selección de registros aleatorios o el análisis de componentes principales (PCA) [32].

5.1.5.3. Separación en *train* y *test*

Debido al problema que se intenta resolver, se decide que el conjunto de **test** sea **2021**, el último año del que existen registros completos, en lugar de un porcentaje de registros aleatorios. En el futuro se tendrá la necesidad de predecir períodos concretos y no fechas salteadas dentro de un periodo. Como en este momento los registros están desordenados al proceder de dos fuentes de datos diferentes, se crea la función `train_test_split_sorted`, que ordena estos **datos por fechas y los separa** ya en orden. Así, el subconjunto de *train* contiene el **78,36 % de los registros** (de 2016 a 2019) y el de *test* un **21,65 %**, cumpliendo con la proporción de entre un 20 y un 30 % para validar que marcan los manuales [128].

Por último, **se escalan los datos** que están incluidos en los subconjuntos de `X_train` y `X_test` y se convierten a un rango [-1, 1]. Esto es necesario para algunos de los algoritmos que se van a utilizar debido a que trabaja con más de una dimensión a la vez, lo que evitará posibles

sesgos si los atributos no se mueven en las mismas escalas. Sin embargo, haberlo hecho antes también habría afectado a la variable objetivo, produciendo el efecto indeseado de *leakage*.

Consecuencia de todo este proceso, los datos quedan preparados de la siguiente forma:

- `X_train`. Registros entre 2016 y 2019 sin la variable `target`.
- `X_train_norm`. Registros entre 2016 y 2019 sin `target` con los valores escalados.
- `y_train`. Variable `target` para los registros contenidos en `X_train` y `X_train_norm`.
- `X_test`. Registros de 2021 sin la variable `target`.
- `X_test_norm`. Registros de 2021 sin la variable `target` con los valores escalados.
- `y_test`. Variable `target` para los registros contenidos en `X_test` y `X_test_norm`.

5.2. Generación de los modelos

Esta es la fase de modelización predictiva. Como ya se ha abordado en puntos anteriores, este proyecto se centra en un **problema de clasificación** y, como ya se conoce la variable a predecir, se emplean algoritmos de tipo **supervisado**. La lógica es repetitiva para cada uno de los modelos resultantes, por lo que se crean dos funciones que se emplean en todos los casos:

1. `best_params` :
 - Para cada algoritmo, estudia la **combinación de hiperparámetros** con mayor precisión en el subconjunto de *train* con una **validación cruzada** de cinco particiones estratificadas. De esta forma se evitan problemas como el *overfitting*, es decir, que el modelo no generalice bien, o el *underfitting*, que consiste en que ni tiene buenos resultados en el entrenamiento ni en la generalización en *test* [129].
 - Muestra la **precisión media** de la validación cruzada y su **desviación** para cada par de hiperparámetros así como un **mapa de calor** con la información.
 - Guarda en una variable global la **mejor combinación** de hiperparámetros.
2. `evaluate_best_params` :
 - Define el modelo con la mejor combinación de hiperparámetros hallada en `best_params` y evalúa su **capacidad de generalización** en el subconjunto de *test*, mostrando el informe completo de la clasificación, lo que incluye ***precision*, *recall*, *f1-score* y *accuracy***.
 - Muestra la **matriz de confusión** en el subconjunto de *test*.

- Muestra el peso de las **20 variables** que más influencia tienen.
- Crea una variable global para calcular la **curva ROC** y el **área bajo la curva**.

Por último, cuando se terminan los cálculos para todos los modelos, que se hacen **de forma consecutiva** como se puede comprobar en el orden de ejecución de `models.ipynb`, se muestra en una gráfica todas las curvas ROC junto a la tabla con el valor del área bajo la curva de cada modelo. Además, se crea una **tabla con todas las medidas** de todos los modelos para facilitar las comparaciones y se compara en otra cuáles son las **características** con más peso.

5.2.1. Medidas de evaluación de los modelos

Antes de entrar en cada uno de los modelos, como todos se miden igual para que sean comparables, se definen las métricas que se emplean en la evaluación de los resultados [129]:

- **Matriz de confusión.** Una de las formas más completas de representar el resultado de la evaluación de una clasificación binaria. En la Figura 5.1 se definen gráficamente los conceptos de **verdadero negativo (TN)**, **falso positivo (FP)**, **falso negativo (FN)** y **verdadero positivo (TP)**, todos abreviados por sus siglas en inglés.

		Predicción	
		No accidente	Sí accidente
		con animal (0)	con animal (1)
Realidad	No accidente	Verdadero negativo (TN)	Falso positivo (FP)
	Sí accidente	Falso negativo (FN)	Verdadero positivo (TP)

Figura 5.1: Matriz de confusión en este proyecto.

Fuente: elaboración propia a partir de [129].

- **Accuracy.** Traducido por algunos manuales como ‘precisión’ o ‘exactitud’, es el número

de predicciones correctas (TP y TN) dividido por el número de todas las muestras (todas las entradas de la matriz de confusión sumadas):

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

- **Precision.** También se conoce como **valor predictivo positivo (PPV)**, mide cuántas de las muestras predichas como positivas son realmente positivas. Se emplea como una métrica de rendimiento cuando el objetivo es limitar el número de falsos positivos:

$$Precision = \frac{TP}{TP+FP}$$

- **Recall.** También llamado **sensibilidad o tasa de verdaderos positivos (TPR)**, mide cuántas de las muestras realmente positivas son predichas como positivas. Se utiliza como métrica de rendimiento cuando se necesita identificar todas las muestras positivas, es decir, cuando es importante evitar los falsos negativos, como es el caso de este proyecto:

$$Sensibilidad = Recall = \frac{TP}{TP+FN}$$

- **F1-score.** Es una forma de resumir las medidas anteriores, ya que consiste en la media armónica de la precisión y el recall:

$$F1 - score = \frac{2*Precision*Recall}{Precision+Recall} = \frac{2*TP}{2*TP+FP+FN}$$

- **Tasa de verdaderos negativos (TNR)**, por sus siglas en inglés). También llamado **especificidad**, mide cuántas de las muestras realmente negativas son predichas como negativas. Se utiliza como métrica de rendimiento cuando se necesita identificar todas las muestras negativas, es decir, cuando es importante evitar los falsos positivos:

$$Especificidad = \frac{TN}{FP+TN}$$

- **Curva ROC.** Considera todos los umbrales posibles para un clasificador determinado, mostrando la tasa de falsos positivos (FPR, por sus siglas en inglés), que es la fracción de falsos positivos de todas las muestras negativas, frente a la tasa de verdaderos positivos (TPR, por sus siglas en inglés).
- **Área bajo la curva (AUC,** por sus siglas en inglés). Es el resultado del cálculo del área bajo la curva ROC.

5.2.2. Regresión logística binaria

Este algoritmo de regresión estima la **probabilidad de una instancia de pertenecer a una clase** en concreto, determinando que la clase es positiva si la probabilidad es mayor al 50 % y negativa en caso contrario. Para ello, aplica una función sigmoide al producto interno de los pesos de la función de aprendizaje con los valores de entrada y un término *bias*. La ecuación del modelo es la siguiente, siendo $P(Y = 1/X)$ la probabilidad de que Y tome el valor 1, en presencia de las variables independientes X [130]:

$$P(Y = 1/X) = \frac{e^{b_0 + \sum_{i=1}^n (b_i x_i)}}{1 + e^{b_0 + \sum_{i=1}^n (b_i x_i)}}$$

Teóricamente, se ve muy afectado por la **multicolinealidad**, motivo por el que se eliminaron las variables correlacionadas, y en este proyecto en concreto también por las diferentes escalas, por lo que se emplean los **datos escalados**. En cuanto a los hiperparámetros, se trabaja con diferentes opciones de **iteraciones máximas y de ‘solver’** [131], que es el algoritmo de Descenso por Coordenadas (CD, por sus siglas en inglés) [132]. Los valores cambian en cada iteración puesto que no se ha fijado una semilla y en esta la mejor combinación es `max_iter = 2000` y `solver = sag`, siendo la precisión media de la **validación cruzada de un 79.66 %** con una **desviación de +/-0.07 %**. Los resultados de la evaluación en el subconjunto de *test* se observan en la Figura 5.2.

	precision	recall	f1-score	support	Predicción
	0	1			
no animal (0)	0.71	0.90	0.79	28205	
sí animal (1)	0.86	0.63	0.73	28523	
accuracy			0.76	56728	
macro avg	0.79	0.77	0.76	56728	
weighted avg	0.79	0.76	0.76	56728	
					Realidad
					0
				25380	2825
					1
				10514	18009

Figura 5.2: Evaluación de los resultados de la regresión logística.

Fuente: elaboración propia.

Los resultados pueden ser considerados ‘**buenos**’ por estar el *accuracy* por encima del 75 %, aunque existe una gran diferencia en la *precision* y *recall* entre ambas clasificaciones y un ligero sobreentrenamiento. Así, existe margen de mejora que se puede conseguir optimizando la entrada gracias al empleo únicamente de variables que sean significativas. Sin embargo, se opta por usar **otros algoritmos** y, en caso de fracaso, se revisaría este modelo.

5.2.3. K vecinos más cercanos

El KNN o K vecinos más cercanos (*k nearest neighbours*, en inglés) calcula la distancia con todas las instancias de entrenamiento y se seleccionan las k **instancias más cercanas** por cada registro a clasificar. Este **aprendizaje perezoso** o *lazy learning methods* sucede en el mismo momento en el que se pide clasificar una nueva instancia, lo que lo diferencia de otros algoritmos de aprendizaje supervisado.

Se emplean los datos escalados porque para el KNN es muy importante expresar los atributos en valores que sean comparables [32]. Los hiperparámetros evaluados en el entrenamiento son el **número de vecinos** (k) y los **pesos** [133], dando como la mejor combinación en esta iteración `n_neighbors = 6` y `weights = distance`, siendo la precisión media de la **validación cruzada** de un **78.17 %** con una **desviación de +/-0.01 %**. Los resultados de la evaluación en el subconjunto de test se muestran en la Figura 5.3.

	precision	recall	f1-score	support	Predicción	
					0	1
no animal (0)	0.70	0.77	0.74	28205		
sí animal (1)	0.75	0.68	0.71	28523		
accuracy			0.72	56728	Realidad	
macro avg	0.73	0.72	0.72	56728		
weighted avg	0.73	0.72	0.72	56728		
					0	21781 6424
					1	9252 19271

Figura 5.3: Evaluación de los resultados del KNN.

Fuente: elaboración propia.

Los resultados muestran un sobreentrenamiento mayor que en la regresión logística, lo que se ve en un *accuracy* que es un 6 % inferior a la media de la validación cruzada. Por tanto, se puede concluir que el modelo resultante de este algoritmo tiene **problemas para generalizar** porque los resultados con el conjunto de *train* son bastante superiores a los de *test*. Además, también hay mucha diferencia en el *recall*, por lo que la tasa de verdaderos positivos varía mucho de una clase a otra. En cualquier caso, aunque esta situación no se diera, **no se puede considerar un modelo válido** sin explorar otras opciones porque el *accuracy* es de tan solo un 72 %, por lo que el algoritmo clasifica mal más de un cuarto de los registros.

5.2.4. Árbol de clasificación

Es un tipo de árbol de decisión y su funcionamiento se basa en la subdivisión del espacio de datos de entrada para generar **regiones disjuntas** para que todos los datos de esa región (llamada hoja o nodo) sean de la misma clase. Si esto no ocurre, el árbol se sigue subdividiendo hasta que acabe o se le aplique la parada. El punto más diferente al resto de algoritmos de clasificación es el **criterio de clasificación de una hoja**, siendo la clase escogida la que satisfaga la siguiente ecuación [32]:

$$c(t) = \operatorname{argmin}_j \sum_{i=1}^k p_i(t) C_{i,j}$$

Por tanto, los hiperparámetros estudiados son la **profundidad máxima** (niveles antes de parar) y el número de **instancias mínimas** que contiene una hoja [134], dando como mejor resultado `max_depth` = 9 y `min_samples_split` = 2. La precisión media de la validación cruzada sobre el subconjunto de *train* es de **89.47 %** con una desviación de **+/-0.02 %** y el resultado de la evaluación en el subconjunto de test se puede ver en la Figura 5.4.

	precision	recall	f1-score	support	Predicción	
					0	1
no animal (0)	0.91	0.88	0.89	28205		
sí animal (1)	0.88	0.92	0.90	28523		
accuracy			0.90	56728	Realidad	
macro avg	0.90	0.90	0.90	56728		
weighted avg	0.90	0.90	0.90	56728		
					0	24705 3500
					1	2310 26213

Figura 5.4: Evaluación de los resultados del árbol de clasificación.

Fuente: elaboración propia.

A la luz del buen rendimiento de este algoritmo en todas las métricas, con un *accuracy* del 90 % se puede considerar que su **capacidad predictora es muy buena**. Por tanto, se decide seguir el camino de la lógica de los árboles de clasificación, buscando algoritmos que aprovechen su potencialidad a la vez que disminuyan su error para optimizar los resultados.

5.2.5. Random forest

Funciona de forma muy similar a los árboles de clasificación, aumentando la complejidad. La diferencia es que suele tener **más capacidad de predicción** porque utiliza un método de

ensemble, es decir, **combina varios árboles de clasificación** para aprovechar la potencialidad de todos ellos de forma conjunta y predecir la clase final de cada instancia, de ahí su nombre. En cambio, aunque reduce la posibilidad de **sobreentrenamiento** porque cada árbol se entrena con un muestreo tanto de los elementos del subconjunto original de *train* como de sus variables, puede aumentar el error en el conjunto de entrenamiento [32] [135].

Por ello, la combinación de hiperparámetros analizada incluye la **profundidad máxima** y el **número de árboles** en el bosque [136], arrojando como la mejor en esta iteración la que emplea `max_depth = 12` y `n_estimators = 100`. La precisión media de la validación cruzada es de **90.28 %** con una desviación de **+/-0.04 %** y este dato se mantiene en la validación en el subconjunto de *test*, como se observa junto al resto de métricas en la Figura 5.5.

	precision	recall	f1-score	support	Predicción	
					0	1
no animal (0)	0.92	0.88	0.90	28205		
sí animal (1)	0.88	0.93	0.90	28523		
accuracy			0.90	56728		
macro avg	0.90	0.90	0.90	56728		
weighted avg	0.90	0.90	0.90	56728		
					Realidad	
					0	24766 3439
					1	2129 26394

Figura 5.5: Evaluación de los resultados del random forest.

Fuente: elaboración propia.

Los resultados son muy similares al árbol del clasificación, aunque **muy ligeramente superiores**, lo que lleva a dar un pasos en esta dirección, trabajando en la disminución del error.

5.2.6. Gradient boosting

Se basa también en los árboles de clasificación y el *ensemble* pero con una gran diferencia, ya que la idea clave es el uso del *gradient descent* para **minimizar los errores de los residuos**. Consigue mejorar las predicciones porque el aumento de gradiente se ajusta a cada nuevo árbol en función de los errores de las predicciones del árbol anterior. Es decir, para cada nuevo árbol, el aumento de gradiente analiza los errores y luego crea un nuevo árbol alrededor de estos errores, y a este árbol **le dan igual las clasificaciones que ya son correctas**, lo que lo hace muy robusto al sobreentrenamiento. Por último, el aumento de gradiente calcula los residuos de las predicciones de cada árbol y suma todos los residuos para puntuar el modelo [137].

Por tanto, en la búsqueda la mejor combinación de hiperparámetros también se estudia la **tasa de aprendizaje** que reduce la contribución de cada árbol y el **número de etapas de refuerzo** a realizar [138], entendiendo conceptualmente cada etapa como un árbol. La mejor combinación de hiperparámetros, con un `learning_rate` = 1 y `n_estimators` = 200 da como media de la validación cruzada una precisión de **92.15 %** con una desviación de **+/-0.16 %**, lo que se convierte en cifras muy similares en la validación final, tal y como recoge la Figura 5.6.

	precision	recall	f1-score	support	Predicción
					0 1
no animal (0)	0.93	0.91	0.92	28205	
sí animal (1)	0.91	0.93	0.92	28523	
accuracy			0.92	56728	
macro avg	0.92	0.92	0.92	56728	
weighted avg	0.92	0.92	0.92	56728	
					Realidad
					0 1
					25717 2488
					1864 26659

Figura 5.6: Evaluación de los resultados del gradient boosting.

Fuente: elaboración propia.

Los resultados obtenidos son **muy buenos**, con un rendimiento por encima del 90 % en todas las métricas y un *accuracy* del 92 %, por lo que se pasa a comparar todos los modelos.

5.3. Análisis comparativo de los resultados

Haber definido anteriormente el **significado de cada métrica** y en **cómo funciona cada algoritmo** de clasificación empleado permite llegar a conclusiones y entender el motivo de las mismas sin extenderse, ajustando los comentarios a las limitaciones de esta memoria.

En primer lugar, se compara la **curva ROC** de cada uno de los modelos generados a partir de la mejor combinación de hiperparámetros. Como se observa en la Figura 5.7, el **gradient boosting** es el algoritmo que mejores resultados arroja, seguido del random forest y árbol de clasificación, la regresión logística y el KNN.

Aunque esta información es importante, es preciso **comparar el rendimiento de cada modelo** generado atendiendo a cada una de las métricas estudiadas para tomar decisiones sobre cuál es el mejor para satisfacer las necesidades de este proyecto, puesto que se puede dar el caso de que algunos tengan una mayor sensibilidad y otros especificidad, por ejemplo. Para ello, se elabora la Tabla 5.2, también recogida el *Jupyter Notebook* `models.ipynb`.

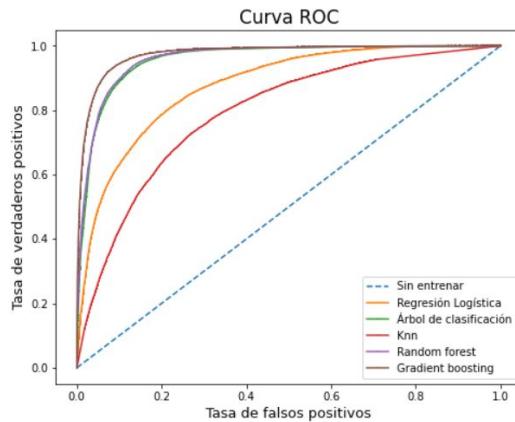


Figura 5.7: Curva ROC de todos los modelos.

Fuente: elaboración propia.

Modelo	Regresión logística	KNN	Árbol de clasificación	Random forest	Gradient boosting
Accuracy	0.76	0.72	0.90	0.90	0.92
Precisión no animal (0)	0.71	0.70	0.91	0.92	0.93
Precisión sí animal (1)	0.86	0.75	0.88	0.88	0.91
Recall no animal (0)	0.90	0.77	0.88	0.88	0.91
Recall sí animal (1)	0.63	0.68	0.92	0.93	0.93
F1-score no animal (0)	0.79	0.74	0.89	0.90	0.92
F1-score sí animal (1)	0.73	0.71	0.90	0.90	0.92
AUC	0.88	0.79	0.96	0.96	0.97

Cuadro 5.2: Tabla comparativa de los resultados de todos los modelos.

Fuente: elaboración propia.

El análisis en conjunto no deja lugar a dudas. El mejor modelo es el **gradient boosting**, que supera a todos los demás en todas las métricas. Debido a su lógica interna ya explicada, era esperable que fuera el algoritmo que mostrara un mayor rendimiento general, pero no implicaba que fuera superior en todas las clasificaciones.

Además, cabe destacar el **muy buen resultado del árbol de clasificación y del random forest**, según los criterios establecidos por los manuales de aprendizaje automático [32] [129] [135] [130] [137], con un *accuracy* del 90 %. Además, están por debajo del gradient boosting en todas las métricas analizadas, pero a una escasa distancia. Asimismo, la poca diferencia entre el árbol de clasificación y el random forest es sorprendente debido a la mayor complejidad de este último, que es un compuesto de los mejores modelos del anterior.

La **regresión logística**, por su parte, podría clasificarse como con una exactitud ‘buena’ según los manuales al estar su *accuracy* por encima del 75 %, pero está muy alejada de los tres mejores modelos. Se podría haber buscado su optimización, por ejemplo, mediante la inclusión de únicamente variables significativas, pero no habría llegado al rendimiento de los árboles. Por

último, los peores resultados los presenta el **KNN**, que además sufre un claro *overfitting*, por lo que no generaliza bien, a pesar de que su AUC sí que es mejor que el de la regresión logística.

Así, los buenos resultados y la poca diferencia entre los mejores modelos descartan otras técnicas como ***stacking*** o ***cascading***, que solo añadirían coste computacional [32].

Otro conocimiento valioso para el proyecto es la detección de las **variables más influyentes** en los modelos, por lo que en la Tabla 5.3 se recogen las que tienen un peso igual o superior a 0.001 en el redondeo a tres decimales.

Variable	Gradient boosting	Random forest	Árbol de clasificación	KNN	Regresión logística
tmed	1 (0.0845)	1 (0.0727)	1 (0.2085)	4 (0.0121)	2 (0.1023)
maxspeed	2 (0.0804)	3 (0.0615)	2 (0.1000)	1 (0.0320)	1 (0.1150)
fecha_num	3 (0.0271)	8 (0.0055)	6 (0.0139)		
parte_dia_Dia	4 (0.0226)	4 (0.0373)	3 (0.0453)	3 (0.0124)	3 (0.0309)
ascii_geohash6	5 (0.0204)	13 (0.0027)	8 (0.0032)		
prec	6 (0.0142)	2 (0.0659)			5 (0.0027)
imd_total	7 (0.0136)	7 (0.0089)	5 (0.0165)	6 (0.0018)	6 (0.0020)
altitud	8 (0.0108)	6 (0.0103)	4 (0.0171)	7 (0.0017)	9 (0.0010)
anyo	9 (0.0090)	9 (0.0052)	9 (0.0030)		
pendiente	10 (0.0041)	15 (0.0014)	10 (0.0021)		
uso_suelo_Artificial	11 (0.0032)	12 (0.0042)	7 (0.0049)	5 (0.0079)	11 (0.0006)
sol	12 (0.0022)	10 (0.0045)	12 (0.0007)	2 (0.0234)	4 (0.0059)
parte_dia_Anochece	13 (0.0006)	14 (0.0016)			7 (0.0018)
parte_dia_Noche		5 (0.0208)			
luna_cat_21_80		11 (0.0043)			
luna_cat_95_10		16 (0.0009)			
uso_suelo_Cultivos		17 (0.0009)			
parte_dia_Amanecer		18 (0.0008)			
uso_suelo_Monte_arbolado_adehesado			11 (0.0010)		
nombre_mes_Octubre				8 (0.0012)	
uso_suelo_Monte_desarbolado					10 (0.0006)

Cuadro 5.3: Peso de las variables más importantes en todos los modelos.

Fuente: elaboración propia.

Se observa que la clave del **éxito del proyecto** reside en el enriquecimiento de los conjuntos de datos publicados por la DGT [45] [46] [47] [48] [49] [5] con la integración de **otras fuentes de datos**. Además, la construcción de un conjunto de datos para los modelos muy completo con **registros de diferente naturaleza** ha optimizado su potencialidad de clasificación.

Entre otros, la meteorología (`tmed`, `prec` y `sol`) tiene un papel crucial, así como la velocidad máxima (`maxspeed`), la fecha (`fecha_num` y `anyo`), la parte del día (`parte_dia`), la ubicación (`ascii_geohash6`), la densidad circulatoria (`imd_total`) y las condiciones geográficas (`altitud`, `pendiente` y `uso_suelo`). No debe pasar desapercibido, no obstante, que `nombre_mes`, `tipo_dia`, `nombre_dia_semana` y `luna_cat` no tienen relevancia alguna.

Por tanto, las **características físicas del lugar, la carretera y algunas del momento** son clave para predecir los accidentes de tráfico con animales involucrados y mejorar los modelos podría pasar por **aumentar dicha información** con datos, por ejemplo, como tramos vallados.

Capítulo 6

Producción de los resultados

En este capítulo se aborda la última fase de este *Trabajo Final de Máster*, la de producción de los resultados, siguiendo la planificación y la metodología CRISP-DM definidas en el Capítulo 1. Por un lado, a continuación se detalla el material que se entrega a la DGT y, por el otro, los conocimientos y materiales que quedan a disposición de la comunidad y en qué plataformas.

6.1. Plataforma DGT 3.0

Uno de los marcos en el que este proyecto seguirá su vida es el de la **plataforma de coche conectado DGT 3.0**, creada para la gestión de la información relacionada con seguridad vial. Su objetivo es recopilar toda la información relacionada con riesgos en la carretera, normalizarla mediante el uso de estándares y publicarla de forma gratuita a través de interfaces de comunicación basados en API REST y MQTT, utilizando mensajería en formato JSON y XML.

En este sentido, es importante poner sobre la mesa que uno de los servicios de la plataforma es un **mapa de movilidad** [139] consistente en un conjunto de *endpoints* donde se publica información de la ubicación y características de elementos de riesgo para el tráfico en formato GeoJSON. También existe una API REST, llamada **Bandeja de Salida** [140], en la que se publican en formato JSON los eventos de riesgo en tiempo real.

El objetivo en este punto es, por tanto, facilitar el uso por parte de la plataforma DGT 3.0 de toda la información generada que pudiera ser de utilidad. Para ello, se entregará lo siguiente:

- **Modelos generados y evaluación de variables de peso.** Código fuente en formato *Jupyter Notebook* ([models.ipynb](#)).
- **Modelo digital del terreno (MDT).** Ficheros generados en el análisis (Capítulo 3) con las capas de estimación de densidad de Kernel (KDE) por cada especie en CSV, lo que permite disponer de un MDT con la probabilidad de accidente con el animal asociado.

- **Usos del suelo.** Mapa con su distribución en el entorno de las vías en formato *shapefile*. Con estos datos, la plataforma DGT 3.0 podrá publicar los siguientes contenidos:
- **Mapa de movilidad:**
 - Capa geográfica con la distribución de usos del suelo.
 - Capa geográfica con el modelo digital del terreno de la densidad de probabilidad de accidente en función del modelo *Kernel Density*.
- **Bandeja de Salida.** Desarrollo de una lógica que proporcione el nivel de riesgo en función del momento y la localización de un vehículo a partir de los modelos proporcionados.

6.2. Contribuciones mediante otras vías de publicación

Aunque la sociedad en su conjunto se beneficiará del proyecto a través de la plataforma DGT 3.0, existen otras vías para hacer la **transferencia del conocimiento generado**. Además, es importante que se pueda continuar la investigación en esta área o se reutilice para mejorar otras. Por ello, se realizan las siguientes publicaciones, subiendo 1.000 registros de cada *dataset* y Aragón del *shapefile* por su tamaño y proporcionando el conjunto completo a quien lo solicite:

1. Repositorio en **GitHub**, en [este enlace](#). Destacan como desarrollos independientes:
 - `elevations`. Herramienta para hallar la **altitud** de cualquier punto de España.
 - `slopes`. Permite calcular la mediana de las **pendientes** cercanas a cualquier punto.
2. Registros del *dataset* decodificado y enriquecido de los accidentes de tráfico con y sin víctimas con **animales involucrados** en España (2016-2021). DOI: [10.5281/zenodo.7523379](https://doi.org/10.5281/zenodo.7523379)
3. **Análisis completo** de los accidentes de tráfico con animales involucrados en España entre 2016 y 2021, proyecto disponible en [este enlace](#).
4. **Modelo digital del terreno** (MDT) completo con la probabilidad de accidente por cada especie animal en función del modelo *Kernel Density*. DOI: [10.5281/zenodo.7522024](https://doi.org/10.5281/zenodo.7522024)
5. Registros del conjunto de datos completo, decodificado y enriquecido de todos los accidentes de tráfico **con víctimas** en España entre 2016 y 2021. DOI: [10.5281/zenodo.7523402](https://doi.org/10.5281/zenodo.7523402)
6. Proyecto completo e interactivo en **Tableau** sobre todos los accidentes con víctimas en España entre 2016 y 2021, accesible en [este enlace](#).
7. **Shapefile del mapa forestal** alrededor de las vías de Aragón. DOI: [10.5281/zenodo.7522758](https://doi.org/10.5281/zenodo.7522758)
8. Registros de **animales** involucrados en accidentes y los *buffer*. DOI: [10.5281/zenodo.7523015](https://doi.org/10.5281/zenodo.7523015)

Capítulo 7

Conclusiones y líneas de trabajo futuras

En este capítulo se presentan las conclusiones de este *Trabajo Final de Máster*, haciendo una reflexión y evaluación de todo el proceso. Además, se establecen posibles líneas de trabajo futuro a partir de necesidades e ideas surgidas durante el desarrollo del proyecto.

7.1. Conclusiones

Este TFM se ha **confirma la hipótesis** de que es posible la predicción de accidentes de tráfico con fauna involucrada y se puede desarrollar una metodología para ello. De hecho, el **objetivo principal** del desarrollo de un modelo predictivo que identifique zonas de riesgo en función del contexto se ha **superado**. Es capaz de predecir, con un 92 % de *accuracy* y más de un 90 % de rendimiento en el resto de métricas, si en cualquier punto habrá un accidente con presencia de animales o no en función de sus características y las de cada momento concreto.

En este sentido, todos los **objetivos secundarios o parciales** también se han cumplido y han sido esenciales para la consecución del objetivo final. El completo estudio del **estado del arte** ha sido clave para llevar este proyecto a buen puerto, puesto que sentó las bases y el *dataset inicial* se enriqueció con los datos que investigaciones anteriores habían tenido en consideración, así como las aproximaciones iniciales a su análisis. En la **creación de conjunto de datos** a partir de la integración de diferentes fuentes de datos abiertos, además, se tuvieron en cuenta otros posibles factores no estudiados hasta la fecha. La **generación de varios modelos predictivos** y el análisis de las características que les afectan, por su parte, también ha sido muy importante, puesto que todos los estudios anteriores se quedaban en la regresión logística, que es la que peores resultados dio en este proyecto. Por último, también se ha conseguido el objetivo de **transferencia a la sociedad** del conocimiento generado a través de

la plataforma DGT 3.0, lo que se ha ampliado con la publicación con licencias abiertas *datasets* y herramientas desarrolladas, esperando que estas contribuciones sean de utilidad.

En el ámbito académico, para la consecución de los objetivos y desarrollo completo del proyecto, se han empleado **todos los conocimientos adquiridos** en el Máster Universitario de Ciencia de Datos, tanto teóricos como prácticos, teniendo que ampliar en muchas áreas al pasar por todas las fases del ciclo de vida de los datos con nuevas necesidades en cada una. Para ello, el empleo de la **metodología CRISP-DM** ha sido un acierto, guiando todo el proceso. Sin embargo, uno de los puntos incumplidos es el de la **planificación**. La fase más extensa y en la que más esfuerzos se han invertido ha sido en la de búsqueda de fuentes de datos, creación de herramientas y preprocesado. En cada intento de integración han surgido obstáculos para los que ha habido que buscar soluciones y desarrollos nuevos, a lo que se ha sumado que los datos de accidentes no estuvieron disponibles hasta el 25 de noviembre. En contraprestación, también ha sido la fase de **mayor aprendizaje**. Otro punto a destacar es la preparación personal que este proyecto ha supuesto para gestionar relaciones con un **cliente**, adaptarse a plazos, etc.

En cuanto a los éxitos, más allá de los mencionados, uno de los mayores ha sido la **liberación de los microdatos** de accidentes con animales involucrados, porque no solo son para este proyecto, sino que la DGT los publicó en su portal, por lo que ahora están disponibles para la comunidad científica y la sociedad. Además, el TFM pretende dar solución a una problemática real y manifiesta por las entidades públicas. Muestra de ello fue que en noviembre, a mitad del desarrollo del proyecto, el Ministerio de Transportes, Movilidad y Agenda Urbana publicó una convocatoria de **Compra Pública de Innovación en Carreteras** [141], cuyo Reto 10 consiste en ‘Medidas de protección para usuarios vulnerables y para accidentes con fauna’ [142].

Por último, poniendo el proyecto en su **contexto global** y habiendo trabajado siempre de una manera honesta, ética, sostenible, socialmente responsable y respetuosa con los derechos humanos y la diversidad, como recoge la CCEG, todo lo aquí descrito tiene una relación positiva directa con el **impacto del proyecto** estudiado al inicio, sobre todo en las dimensiones de sostenibilidad, responsabilidad social y derechos humanos, siendo la sociedad en su conjunto el grupo de personas beneficiadas. En concreto, a pesar de su huella ecológica, destaca su contribución al principio de seguridad sostenible y a los ODS 3, 5, 8, 9, 10, 11 y 17, descritos ampliamente en el Capítulo 1. Del mismo modo, todo el TFM se ha enfocado atendiendo a la diversidad de género, lo que se refleja en cómo está escrita esta memoria.

7.2. Líneas de trabajo futuras

Durante el desarrollo de este proyecto, se han detectado posibles nuevas líneas de trabajo para mejorar la detección y predicción de accidentes de tráfico con animales y, en general,

mejorar la seguridad vial. Además, han surgido ideas de trabajo colaborativo con otras áreas de conocimiento que sería positivo explorar:

- **Enriquecimiento del *dataset* final.** Aunque se han trabajado múltiples fuentes de datos, ha quedado otro conocimiento sin codificar, como es la presencia de vallas, guardarráles, etc. Que sean relevantes o no solo se podrá determinar si se estudian. Además, el **vector final** con el que se ha realizado este proyecto podría aplicarse a otros tipos de accidentes de tráfico para evaluar si también es válido en esos casos.
- Registro y estudio de las **características de las personas que conducen** los vehículos, línea interesante para cualquier tipo de accidente. Además del género, la edad o el consumo de sustancias prohibidas, que ya se recogen aunque no están publicados los datos desagregados, sería relevante otro tipo de información. Por ejemplo, la duración del trayecto hasta el momento del accidente, los años de experiencia al volante y el grado de experiencia conduciendo en la zona del siniestro.
- Creación de **modelos predictivos para cada tipo de animal**, puesto que pueden provocar consecuencias muy diferentes en los accidentes. Además, aunque no se ha detectado una estacionalidad general, es previsible que sí que la haya por especie. Esta nueva línea ayudaría a desarrollar medidas más específicas y adecuadas dependiendo del contexto.
- Desarrollo de un **modelo digital del terreno** (MDT) con la densidad de probabilidad de accidente, además de por especies como ya se entrega, **por períodos**. En este sentido, una primera aproximación muy útil sería la diferenciación entre partes del día, como se ha visto en el análisis y resultado de los modelos predictivos. Además, si se realiza en conjunto a la propuesta anterior, también se podría realizar en función de su estacionalidad.
- **Sistema de seguimiento y evaluación** del proyecto en la plataforma DGT 3.0. Al no estar contemplado en los pliegos ni en ningún documento oficial, es necesario tomar conciencia de que el seguimiento y evaluación de los resultados tras su implementación tendrá que planificarse. También es interesante que llegue el conocimiento generado a manos de las personas a las que se adjudique el proyecto del Reto 10 descrito anteriormente.
- En cuanto a la colaboración entre áreas, se propone la creación de un sistema automatizado que **notifique a GBIF la presencia del animal** involucrado y de su especie justo en esa posición. De este modo, actualizaría su registros sobre especies que ya tienen ubicadas con datos oficiales y, quizás, permitiría detectar la presencia de otros animales que hasta ahora no estaban en la base de datos para esa ubicación. Esto beneficiaría a toda persona, institución u organización que haga uso de estos datos.

Bibliografía

- [1] Observatorio Nacional de Seguridad Vial (Dirección General de Tráfico). “Anexo estadístico. Agosto 2022”. En: *Balance de las cifras de siniestralidad vial 2021* (ago. de 2022). URL: https://www.dgt.es/export/sites/web-DGT/.galleries/downloads/nota_prensa/descienden-un-20-los-fallecidos-por-siniestro-de-trafico-en-las-ciudades/anexo-estadistico-vf-nuevos-terminos-rev.pdf (visitado 29-09-2022).
- [2] INE. *Defunciones por causas (lista reducida) por sexo y grupos de edad.* 2022. URL: <https://www.ine.es/jaxiT3/Tabla.htm?t=7947> (visitado 03-10-2022).
- [3] La Moncloa. *Los accidentes de tráfico se cobraron la vida de 1.004 personas el pasado año.* 7 de ene. de 2022. URL: <https://www.lamoncloa.gob.es/serviciosdeprensa/notasprensa/interior/Paginas/2022/070122-balance-siniestralidad-2021.aspx> (visitado 29-09-2022).
- [4] Dirección General de Tráfico. *Educación Vial. Factores y Valores de riesgo.* 2014. URL: <https://www.dgt.es/estaticos/PEVI/contenidos/Externos/recursos/jovenes/factores-y-valores-de-riesgo.pdf> (visitado 29-09-2022).
- [5] Dirección General de Tráfico. *Ficheros de microdatos de accidentes con animales 2021.* 25 de nov. de 2022. URL: <https://www.dgt.es/menusecundario/dgt-en-cifras/dgt-en-cifras-resultados/dgt-en-cifras-detalle/?id=00842> (visitado 11-2022).
- [6] Tomasz Krukowicz, Krzysztof Firlag y Paweł Chrobot. “Spatiotemporal Analysis of Road Crashes with Animals in Poland”. En: *Sustainability* 14.3 (ene. de 2022). ISSN: 2071-1050. DOI: [10.3390/su14031253](https://doi.org/10.3390/su14031253). URL: <https://www.mdpi.com/2071-1050/14/3/1253> (visitado 10-2022).
- [7] Ministerio para la Transición Ecológica y el Reto Demográfico. “Informe metodológico estandarizado”. En: *Estadística anual de accidentes con víctimas con animales implicados* (2022). URL: https://www.miteco.gob.es/es/biodiversidad/temas/inventarios-nacionales/19-accidentes-con-victimas-animales-implicados_tcm30-207427.pdf (visitado 28-09-2022).

- [8] Dirección General de Tráfico. *Qué es DGT 3.0*. 2022. URL: <https://www.dgt.es/muevete-con-seguridad/tecnologia-e-innovacion-en-carretera/dgt-3.0/> (visitado 28-09-2022).
- [9] Universitat Oberta de Catalunya. *Impacto global #Agenda2030*. 2022. URL: <https://www.uoc.edu/portal/es/compromis-social/index.html> (visitado 06-10-2022).
- [10] Organización de Naciones Unidas. *Objetivos de Desarrollo Sostenible*. 2022. URL: <https://www.un.org/sustainabledevelopment/es/objetivos-de-desarrollo-sostenible/> (visitado 06-10-2022).
- [11] Dirección General de Tráfico. *Misión, visión y principios de la Estrategia*. 2022. URL: <https://seguridadvial2030.dgt.es/vision-2030/mision-vision-y-principios-de-la-estrategia/> (visitado 05-10-2022).
- [12] Ministerio para la Transición Ecológica y el Reto Demográfico. *Calculadoras*. 2022. URL: <https://www.miteco.gob.es/es/cambio-climatico/temas/mitigacion-politicas-y-medidas/calculadoras.aspx> (visitado 06-10-2022).
- [13] Dipanjan Sarkar, Raghav Bali y Tamoghna Ghosh. “Machine Learning Fundamentals”. En: *Hands-on Transfer learning with python: Implement advanced deep learning and neural network models using TensorFlow and keras*. 1.^a ed. Birmingham, UK: Packt Publishing, ago. de 2018. ISBN: 978-1-78883-130-7. URL: <https://learning.oreilly.com/library/view/hands-on-transfer-learning/9781788831307/>.
- [14] Andrea Cirillo. “The Data Mining Process - CRISP-DM Methodology”. En: *R data mining: Implement data mining techniques through practical use cases and real-world datasets*. 1.^a ed. Birmingham: Packt, nov. de 2017. ISBN: 978-1-78712-446-2. URL: <https://learning.oreilly.com/library/view/r-data-mining/9781787124462/>.
- [15] Project Management Institute. “Elementos Fundamentales”. En: *Guía de los Fundamentos para la dirección de Proyectos: (guía Del PMBOK®)*. 6.^a ed. Newtown Square, Pennsylvania, EEUU: Project Management Institute, 2017, págs. 4-36. ISBN: 978-1-62825-194-4.
- [16] Raphaela Pagany. “Wildlife-vehicle collisions - Influencing factors, data collection and research methods”. En: *Biological Conservation* 251 (sep. de 2020). DOI: [10.1016/j.biocon.2020.108758](https://doi.org/10.1016/j.biocon.2020.108758). URL: https://www.researchgate.net/publication/344285425_Wildlife-vehicle_collisions_-_Influencing_factors_data_collection_and_research_methods (visitado 06-10-2022).
- [17] M.P. Huijser y col. *Wildlife-Vehicle Collision Reduction Study: Report to Congress*. Ago. de 2008. URL: <https://www.fhwa.dot.gov/publications/research/safety/08034/08034.pdf> (visitado 10-2022).

- [18] John M. Sullivan. “Trends and characteristics of animal-vehicle collisions in the United States”. En: *Journal of Safety Research* 42.1 (feb. de 2011), págs. 9-16. ISSN: 0022-4375. DOI: <https://doi.org/10.1016/j.jsr.2010.11.002>. URL: <https://www.sciencedirect.com/science/article/pii/S0022437510001234> (visitado 10-2022).
- [19] Lina Galinskaité y col. “The Influence of Landscape Structure on Wildlife–Vehicle Collisions: Geostatistical Analysis on Hot Spot and Habitat Proximity Relations”. En: *ISPRS International Journal of Geo-Information* 11 (ene. de 2022), pág. 63. DOI: [10.3390/ijgi11010063](https://doi.org/10.3390/ijgi11010063). URL: https://www.researchgate.net/publication/357856739_The_Influence_of_Landscape_Structure_on_Wildlife-Vehicle_Collisions_Geostatistical_Analysis_on_Hot_Spot_and_Habitat_Proximity_Relations (visitado 10-2022).
- [20] Peter Rowden, Dale Steinhardt y Mary Sheehan. “Road crashes involving animals in Australia”. En: *Accident Analysis Prevention* 40.6 (nov. de 2008), págs. 1865-1871. ISSN: 0001-4575. DOI: <https://doi.org/10.1016/j.aap.2008.08.002>. URL: <https://www.sciencedirect.com/science/article/pii/S0001457508001371> (visitado 10-2022).
- [21] Katarzyna Tajchman y col. “Wildlife - Vehicle collisions in urban area in relation to the behaviour and density of mammals”. En: *Polish Journal of Natural Sciences* 32 (ene. de 2017), págs. 49-59. URL: http://www.uwm.edu.pl/polish-journal/sites/default/files/issues/articles/tajchman_et.al._2017.pdf (visitado 10-2022).
- [22] Emilio R. Diaz-Varela y col. “Assessing methods of mitigating wildlife–vehicle collisions by accident characterization and spatial analysis”. En: *Transportation Research Part D: Transport and Environment* 16.4 (jun. de 2011), págs. 281-287. ISSN: 1361-9209. DOI: <https://doi.org/10.1016/j.trd.2011.01.002>. URL: <https://www.sciencedirect.com/science/article/pii/S1361920911000034> (visitado 10-2022).
- [23] A. Czerniak y L. Tyburski. “Zdarzenia drogowe z udziałem zwierząt”. En: *Infrastruktura i Ekologia Terenów Wiejskich* 02 (feb. de 2011). URL: <https://bibliotekanauki.pl/articles/61127> (visitado 10-2022).
- [24] Anthony Clevenger, Bryan Chruszcz y Kari Gunson. “Spatial patterns and factors influencing small vertebrate fauna road-kill aggregations”. En: *Biological Conservation* 109 (ene. de 2003), págs. 15-26. DOI: [10.1016/S0006-3207\(02\)00127-1](https://doi.org/10.1016/S0006-3207(02)00127-1). URL: <https://www.webofscience.com/wos/woscc/full-record/WOS:000179138400002?SID=EUW1ED0EBDtKVx33ng84o6SsoiW6b> (visitado 10-2022).
- [25] Victor Javier Colino-Rabanal, Miguel Lizana y Salvador J. Peris. “Factors influencing wolf *Canis lupus* roadkills in Northwest Spain”. En: *A European Journal of Wildlife Research* 57 (1 de jun. de 2011), págs. 399-409. ISSN: 1439-0574. DOI: <https://doi.org/10.1007/s10651-011-0681-0>

- 10.1007/s10344-010-0446-1. URL: <https://link.springer.com/article/10.1007/s10344-010-0446-1> (visitado 10-2022).
- [26] Juan E. Malo, Francisco Suárez y Alberto Díez. “Can we mitigate animal–vehicle accidents using predictive models?” En: *Journal of Applied Ecology* 41.4 (jul. de 2004), págs. 701-710. DOI: <https://doi.org/10.1111/j.0021-8901.2004.00929.x>. eprint: <https://besjournals.onlinelibrary.wiley.com/doi/pdf/10.1111/j.0021-8901.2004.00929.x>. URL: <https://besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/j.0021-8901.2004.00929.x> (visitado 10-2022).
- [27] Beatriz Rodríguez-Morales, Emilio Rafael Díaz-Varela y Manuel Francisco Marey-Pérez. “Spatiotemporal analysis of vehicle collisions involving wild boar and roe deer in NW Spain”. En: *Accident Analysis Prevention* 60 (nov. de 2013), págs. 121-133. ISSN: 0001-4575. DOI: <https://doi.org/10.1016/j.aap.2013.07.032>. URL: <https://www.sciencedirect.com/science/article/pii/S000145751300300X> (visitado 10-2022).
- [28] INE. “Territorio y medio ambiente”. En: *España en cifras 2022* 121 (2022), págs. 2-7. ISSN: 2255-0410. URL: https://www.ine.es/ss/Satellite?L=es_ES&c=INEPublicacion_C&cid=1259924856416&p=1254735110672&pagename=ProductosYServicios%2FPYSLayout¶m1=PYSDetalleGratuitas (visitado 11-10-2022).
- [29] Cristina San José. “Capreolus capreolus (Linnaeus, 1758)”. En: *Atlas y Libro Rojo de los Mamíferos Terrestres de España*. 1.^a ed. Madrid, España: Dirección General para la Biodiversidad - SECEM-SECEMU, 2007, págs. 359-361. URL: https://www.miteco.gob.es/es/biodiversidad/temas/inventarios-nacionales/ieet_mami_capreolus_capreolus_tcm30-99790.pdf.
- [30] Carme Rosell y Juan Herrero. “Sus scrofa Linnaeus, 1758”. En: *Atlas y Libro Rojo de los Mamíferos Terrestres de España*. 1.^a ed. Madrid, España: Dirección General para la Biodiversidad - SECEM-SECEMU, 2007, págs. 348-351. URL: https://www.miteco.gob.es/es/biodiversidad/temas/inventarios-nacionales/ieet_mami_sus_scrofa_tcm30-99882.pdf.
- [31] Juan Carranza. “Cervus elaphus Linnaeus, 1758”. En: *Atlas y Libro Rojo de los Mamíferos Terrestres de España*. 1.^a ed. Madrid, España: Dirección General para la Biodiversidad - SECEM-SECEMU, 2007, págs. 352-355. URL: https://www.miteco.gob.es/es/biodiversidad/temas/inventarios-nacionales/ieet_mami_cervus_elaphus_tcm30-99791.pdf.
- [32] Jordi Gironés Roig y col. *Minería de datos. Modelos y algoritmos*. 1.^a ed. Barcelona, España: Editorial UOC (Oberta UOC Publishing, SL), jul. de 2017. ISBN: 978-84-9116-904-8.

- [33] Jiawei Han, Micheline Kamber y Jian Pei, eds. *Data Mining: Concepts and Techniques*. 3.^a ed. The Morgan Kaufmann Series in Data Management Systems. Boston, Massachusetts, Estados Unidos: Morgan Kaufmann, 2012. ISBN: 978-0-12-381479-1. DOI: <https://doi.org/10.1016/B978-0-12-381479-1.00020-4>. URL: <https://www.sciencedirect.com/science/article/pii/B9780123814791000204>.
- [34] Esteban Vegas Lozano. *Preprocesamiento de los datos*. Universitat Oberta de Catalunya (UOC). Barcelona, España, sep. de 2020. URL: https://materials.campus.uoc.edu/daisy/Materials/PID_00276230/pdf/PID_00276230.pdf.
- [35] Ester Bernadó Mansilla. *Contrastes de hipótesis*. Universitat Oberta de Catalunya (UOC). Barcelona, España, sep. de 2020. URL: https://materials.campus.uoc.edu/daisy/Materials/PID_00276233/pdf/PID_00276233.pdf.
- [36] Barry N Boots y Arthur Getis. *Point pattern analysis*. Vol. 8. SAGE Publications, Incorporated, 1988. URL: <https://researchrepository.wvu.edu/cgi/viewcontent.cgi?article=1013&context=rri-web-book> (visitado 10-2022).
- [37] Gerry P Quinn y Michael J Keough. *Experimental design and data analysis for biologists*. Cambridge university press, mar. de 2002. ISBN: 9780511806384. DOI: <https://doi.org/10.1017/CBO9780511806384>. URL: https://books.google.es/books?hl=en&lr=&id=VtU3-y7LaLYC&oi=fnd&pg=PR15&ots=cBu03zmkkG&sig=SXa6EpdU2TchwypJFAEsnacRkbw&redir_esc=y#v=onepage&q&f=false (visitado 10-2022).
- [38] Dirección General de Tráfico. *Ficheros microdatos de accidentes con víctimas 2016*. 2016. URL: <https://www.dgt.es/menusecundario/dgt-en-cifras/dgt-en-cifras-resultados/dgt-en-cifras-detalle/?id=00171> (visitado 10-2022).
- [39] Dirección General de Tráfico. *Ficheros microdatos de accidentes con víctimas 2017*. 2017. URL: <https://www.dgt.es/menusecundario/dgt-en-cifras/dgt-en-cifras-resultados/dgt-en-cifras-detalle/?id=00172> (visitado 10-2022).
- [40] Dirección General de Tráfico. *Ficheros microdatos de accidentes con víctimas 2018*. 2018. URL: <https://www.dgt.es/menusecundario/dgt-en-cifras/dgt-en-cifras-resultados/dgt-en-cifras-detalle/?id=00173> (visitado 10-2022).
- [41] Dirección General de Tráfico. *Ficheros microdatos de accidentes con víctimas 2019*. 2019. URL: <https://www.dgt.es/menusecundario/dgt-en-cifras/dgt-en-cifras-resultados/dgt-en-cifras-detalle/?id=00174> (visitado 10-2022).

- [42] Dirección General de Tráfico. *Ficheros microdatos de accidentes con víctimas 2020*. 19 de ago. de 2022. URL: <https://www.dgt.es/menusecundario/dgt-en-cifras/dgt-en-cifras-resultados/dgt-en-cifras-detalle/?id=00808> (visitado 10-2022).
- [43] Dirección General de Tráfico. *Ficheros microdatos de accidentes con víctimas 2021*. 16 de nov. de 2022. URL: <https://www.dgt.es/menusecundario/dgt-en-cifras/dgt-en-cifras-resultados/dgt-en-cifras-detalle/?id=00840> (visitado 11-2022).
- [44] Dirección General de Tráfico. *Ficheros microdatos de accidentes con víctimas. Diccionario*. 1 de ene. de 2016. URL: <https://www.dgt.es/menusecundario/dgt-en-cifras/dgt-en-cifras-resultados/dgt-en-cifras-detalle/?id=00840> (visitado 10-2022).
- [45] Dirección General de Tráfico. *Ficheros de microdatos de accidentes con animales 2016*. 25 de nov. de 2022. URL: <https://www.dgt.es/menusecundario/dgt-en-cifras/dgt-en-cifras-resultados/dgt-en-cifras-detalle/?id=00847> (visitado 11-2022).
- [46] Dirección General de Tráfico. *Ficheros de microdatos de accidentes con animales 2017*. 25 de nov. de 2022. URL: <https://www.dgt.es/menusecundario/dgt-en-cifras/dgt-en-cifras-resultados/dgt-en-cifras-detalle/?id=00846> (visitado 11-2022).
- [47] Dirección General de Tráfico. *Ficheros de microdatos de accidentes con animales 2018*. 25 de nov. de 2022. URL: <https://www.dgt.es/menusecundario/dgt-en-cifras/dgt-en-cifras-resultados/dgt-en-cifras-detalle/?id=00845> (visitado 11-2022).
- [48] Dirección General de Tráfico. *Ficheros de microdatos de accidentes con animales 2019*. 25 de nov. de 2022. URL: <https://www.dgt.es/menusecundario/dgt-en-cifras/dgt-en-cifras-resultados/dgt-en-cifras-detalle/?id=00844> (visitado 11-2022).
- [49] Dirección General de Tráfico. *Ficheros de microdatos de accidentes con animales 2020*. 25 de nov. de 2022. URL: <https://www.dgt.es/menusecundario/dgt-en-cifras/dgt-en-cifras-resultados/dgt-en-cifras-detalle/?id=00843> (visitado 11-2022).
- [50] Dirección General de Tráfico. *Diccionario tabla Accidentes animales*. 25 de nov. de 2022. URL: https://www.dgt.es/.galleries/downloads/dgt-en-cifras/publicaciones/Ficheros_de_microdatos_de_accidentes_con_animales/Diccionario_Tabla_Acc_Animales.xlsx (visitado 10-2022).
- [51] OpenStreetMap contributors. *Download OpenStreetMap data for this region: Spain*. 2022. URL: <https://download.geofabrik.de/europe/spain.html> (visitado 2022).
- [52] OpenStreetMap contributors. *Download OpenStreetMap data for this region: Canary Islands*. 2022. URL: <http://download.geofabrik.de/africa/canary-islands.html> (visitado 2022).

- [53] Ministerio de Fomento. Dirección General de Carreteras. *Mapa de Tráfico de la DGC. Año 2015*. 2015. URL: <https://mapas.fomento.gob.es/mapatrafico/2015/> (visitado 10-2022).
- [54] Ministerio de Fomento. Dirección General de Carreteras. *Mapa de Tráfico de la DGC. Año 2016*. 2016. URL: <https://mapas.fomento.gob.es/mapatrafico/2016/> (visitado 10-2022).
- [55] Ministerio de Fomento. Dirección General de Carreteras. *Mapa de Tráfico de la DGC. Año 2017*. 2017. URL: <https://mapas.fomento.gob.es/mapatrafico/2017/> (visitado 10-2022).
- [56] Ministerio de Fomento. Dirección General de Carreteras. *Mapa de Tráfico de la DGC. Año 2018*. 2018. URL: <https://mapas.fomento.gob.es/mapatrafico/2018/> (visitado 10-2022).
- [57] Ministerio de Fomento. Dirección General de Carreteras. *Mapa de Tráfico de la DGC. Año 2019*. 2019. URL: <https://mapas.fomento.gob.es/mapatrafico/2019/> (visitado 10-2022).
- [58] Ministerio de Fomento. Dirección General de Carreteras. *Mapa de Tráfico de la DGC. Año 2020*. 2020. URL: <https://mapas.fomento.gob.es/mapatrafico/2020/> (visitado 10-2022).
- [59] Alimentación y Medio Ambiente. Dirección General de Desarrollo Rural y Política Forestal. Subdirección General de Silvicultura y Montes. Área inventario y Estadística Forestal Ministerio de Agricultura. *Mapa Forestal de España. Escala 1:50.000*. 2006. URL: https://www.miteco.gob.es/es/biodiversidad/servicios/banco-datos-naturaleza/informacion-disponible/mfe50_descargas_ccaa.aspx (visitado 25-10-2022).
- [60] Agencia Estatal de Meteorología. *AEMET OpenData*. 2022. URL: <https://opendata.aemet.es/centrodedescargas/inicio> (visitado 11-2022).
- [61] Instituto Geográfico Nacional y Sistema Cartográfico Nacional Comunidades Autónomas FEGA. *Modelo Digital del Terreno con paso de malla de 200 metros (MDT200) de España*. 2006. URL: <https://centrodedescargas.cnig.es/CentroDescargas/index.jsp> (visitado 07-11-2022).
- [62] Instituto Geográfico Nacional y Sistema Cartográfico Nacional Comunidades Autónomas FEGA. *Modelo digital de pendientes realizado a partir de las nubes de puntos LIDAR de la primera Cobertura con paso de malla de 5 metros*. 2006. URL: <https://centrodedescargas.cnig.es/CentroDescargas/index.jsp> (visitado 08-11-2022).

- [63] GBIF.Org User. *GBIF Occurrence Download Capra pyrenaica Schinz, 1838*. 4 de dic. de 2022. doi: [10.15468/DL.H9W7FY](https://doi.org/10.15468/DL.H9W7FY). URL: <https://www.gbif.org/occurrence/download/0190305-220831081235567>.
- [64] GBIF.Org User. *GBIF Occurrence Download Cervus elaphus Linnaeus, 1758*. 4 de dic. de 2022. doi: [10.15468/DL.UF7BCW](https://doi.org/10.15468/DL.UF7BCW). URL: <https://www.gbif.org/occurrence/download/0190310-220831081235567>.
- [65] GBIF.Org User. *Occurrence Download Dama dama (Linnaeus, 1758)*. 4 de dic. de 2022. doi: [10.15468/DL.P8BZ7T](https://doi.org/10.15468/DL.P8BZ7T). URL: <https://www.gbif.org/occurrence/download/0190330-220831081235567>.
- [66] GBIF.Org User. *Occurrence Download Felis silvestris Schreber, 1777*. 4 de dic. de 2022. doi: [10.15468/DL.Y24KDF](https://doi.org/10.15468/DL.Y24KDF). URL: <https://www.gbif.org/occurrence/download/0190337-220831081235567>.
- [67] GBIF.Org User. *Occurrence Download Oryctolagus cuniculus (Linnaeus, 1758)*. 4 de dic. de 2022. doi: [10.15468/DL.MXJWR8](https://doi.org/10.15468/DL.MXJWR8). URL: <https://www.gbif.org/occurrence/download/0190323-220831081235567>.
- [68] GBIF.Org User. *Occurrence Download Capreolus capreolus (Linnaeus, 1758)*. 4 de dic. de 2022. doi: [10.15468/DL.78QA5S](https://doi.org/10.15468/DL.78QA5S). URL: <https://www.gbif.org/occurrence/download/0190327-220831081235567>.
- [69] GBIF.Org User. *Occurrence Download Lynx pardinus (Temminck, 1827)*. 4 de dic. de 2022. doi: [10.15468/DL.ZVW7N9](https://doi.org/10.15468/DL.ZVW7N9). URL: <https://www.gbif.org/occurrence/download/0190352-220831081235567>.
- [70] GBIF.Org User. *Occurrence Download Lepus Linnaeus, 1758*. 4 de dic. de 2022. doi: [10.15468/DL.ZRRUBF](https://doi.org/10.15468/DL.ZRRUBF). URL: <https://www.gbif.org/occurrence/download/0190347-220831081235567>.
- [71] GBIF.Org User. *Occurrence Download Sus scrofa Linnaeus, 1758*. 4 de dic. de 2022. doi: [10.15468/DL.T7TDAJ](https://doi.org/10.15468/DL.T7TDAJ). URL: <https://www.gbif.org/occurrence/download/0190346-220831081235567>.
- [72] GBIF.Org User. *Occurrence Download Canis lupus Linnaeus, 1758*. 4 de dic. de 2022. doi: [10.15468/DL.G5SUU6](https://doi.org/10.15468/DL.G5SUU6). URL: <https://www.gbif.org/occurrence/download/0190358-220831081235567>.
- [73] GBIF.Org User. *Occurrence Download Ovis musimon (Pallas, 1811)*. 4 de dic. de 2022. doi: [10.15468/DL.ZFAW7W](https://doi.org/10.15468/DL.ZFAW7W). URL: <https://www.gbif.org/occurrence/download/0190379-220831081235567>.

- [74] GBIF.Org User. *Occurrence Download Ursus arctos Linnaeus, 1758.* 4 de dic. de 2022. DOI: [10.15468 / DL .4E6KCQ](https://doi.org/10.15468/DL.4E6KCQ). URL: <https://www.gbif.org/occurrence/download/0190381-220831081235567>.
- [75] GBIF.Org User. *Occurrence Download Rupicapra rupicapra (Linnaeus, 1758).* 4 de dic. de 2022. DOI: [10.15468/DL.TX5UBB](https://doi.org/10.15468/DL.TX5UBB). URL: <https://www.gbif.org/occurrence/download/0190382-220831081235567>.
- [76] GBIF.Org User. *Occurrence Download Capra hircus Linnaeus, 1758.* 4 de dic. de 2022. DOI: [10.15468 / DL .TNMZ5Y](https://doi.org/10.15468/DL.TNMZ5Y). URL: <https://www.gbif.org/occurrence/download/0190394-220831081235567>.
- [77] GBIF.Org User. *Occurrence Download Equus caballus Linnaeus, 1758.* 4 de dic. de 2022. DOI: [10.15468 / DL .FWYWKR](https://doi.org/10.15468/DL.FWYWKR). URL: <https://www.gbif.org/occurrence/download/0190404-220831081235567>.
- [78] GBIF.Org User. *Occurrence Download Meles meles (Linnaeus, 1758).* 4 de dic. de 2022. DOI: [10.15468 / DL .F2CCPM](https://doi.org/10.15468/DL.F2CCPM). URL: <https://www.gbif.org/occurrence/download/0190390-220831081235567>.
- [79] GBIF.Org User. *Occurrence Download Vulpes vulpes (Linnaeus, 1758).* 4 de dic. de 2022. DOI: [10.15468/DL.DWNWXS](https://doi.org/10.15468/DL.DWNWXS). URL: <https://www.gbif.org/occurrence/download/0190391-220831081235567>.
- [80] GBIF.Org User. *Occurrence Download Felis catus Linnaeus, 1758.* 4 de dic. de 2022. DOI: [10.15468/DL.379A22](https://doi.org/10.15468/DL.379A22). URL: <https://www.gbif.org/occurrence/download/0190412-220831081235567>.
- [81] GBIF.Org User. *Occurrence Download Ovis aries Linnaeus, 1758.* 4 de dic. de 2022. DOI: [10.15468/DL.PK5JE9](https://doi.org/10.15468/DL.PK5JE9). URL: <https://www.gbif.org/occurrence/download/0190421-220831081235567>.
- [82] GBIF.Org User. *Occurrence Download Bos taurus Linnaeus, 1758.* 4 de dic. de 2022. DOI: [10.15468 / DL .8KP4AP](https://doi.org/10.15468/DL.8KP4AP). URL: <https://www.gbif.org/occurrence/download/0190424-220831081235567>.
- [83] Guido Van Rossum y Fred L Drake Jr. *Python reference manual.* Ámsterdam, Holanda: Stichting Mathematisch Centrum, 1995. URL: [https://ir.cwi.nl/pub/5008](https://www.python.org/doc/2.5.2/tut/node1.html).
- [84] Barry Warsaw y Nick Coghlan Guido van Rossum. *PEP 8 – Style Guide for Python Code.* 5 de jul. de 2001. URL: <https://peps.python.org/pep-0008/> (visitado 15-11-2022).
- [85] Alan Beaulieu. *Learning SQL.* 3.^a ed. California, Estados Unidos: O'Reilly Media, Inc, mar. de 2020. ISBN: 978-1-492-05761-1. URL: <https://learning.oreilly.com/library/view/learning-sql-3rd/9781492057604/>.

- [86] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria, 2022. URL: <https://www.R-project.org/> (visitado 15-11-2022).
- [87] Carl Albing y JP Vossen. “Chapter 1. Beginning bash”. En: *bash Cookbook*. 1.^a ed. California, Estados Unidos: O'Reilly Media, Inc, oct. de 2017. ISBN: 978-1-491-97533-6. URL: <https://learning.oreilly.com/library/view/bash-cookbook-2nd/9781491975329/>.
- [88] The LaTeX Project. *LaTeX – A document preparation system*. 3 de nov. de 2022. URL: https://www.google.com/intl/es_es/chrome/ (visitado 10-2022).
- [89] Serge Rider y DBeaver Community. *DBeaver Community. Free Universal Database Tool*. 2022. URL: <https://dbeaver.io/> (visitado 10-2022).
- [90] The PostgreSQL Global Development Group. *PostgreSQL 12.13 Documentation*. English. Ed. por The PostgreSQL Global Development Group. 2022. URL: <https://www.postgresql.org/files/documentation/pdf/12/postgresql-12-A4.pdf>.
- [91] The PostgreSQL Global Development Group. *PostGIS 3.0.9dev Manual*. English. Ed. por The PostgreSQL Global Development Group. 2022. URL: <https://postgis.net/stuff/postgis-3.0.pdf>.
- [92] Microsoft Corporation. *Visual Studio Code*. 2022. URL: <https://code.visualstudio.com/> (visitado 15-11-2022).
- [93] Jupyter Trademark. *About Us. Project Jupyter's origins and governance*. 2023. URL: <https://jupyter.org/about> (visitado 02-01-2023).
- [94] Jupyter Trademark. *About Us. Project Jupyter's origins and governance*. 2023. URL: <https://jupyter.org/about> (visitado 02-01-2023).
- [95] RStudio Team. *RStudio: Integrated Development Environment for R*. RStudio, PBC. Boston, MA, Estados Unidos, 2022. URL: <http://www.rstudio.com/> (visitado 10-2022).
- [96] *Easy web publishing from R*. RStudio. 2022. URL: <https://rpubs.com/> (visitado 16-12-2022).
- [97] QGIS Development Team. *QGIS Geographic Information System*. QGIS Association. 2022. URL: <https://www.qgis.org>.
- [98] VEXIZA. *Obtener las coordenadas de un punto proporcionando una provincia, carretera y punto kilométrico*. Publicación interna. Dirección General de Tráfico.
- [99] Abhinav Asthana, Ankit Sotgi and Abhijit Kane. *Postman*. 2022. URL: <https://www.postman.com/> (visitado 10-2022).
- [100] JGraph. *Bienvenido a Tableau Public*. 2022. URL: <https://public.tableau.com/app/discover> (visitado 10-2022).

- [101] Overleaf. *Overleaf, Online LaTeX Editor*. 2022. URL: <https://www.overleaf.com/> (visitado 10-2022).
- [102] Learn Photopea. Ivan Kutskir. 2022. URL: <https://www.photopea.com/learn/> (visitado 07-12-2022).
- [103] JGraph. *diagrams.net, draw.io*. 14 de oct. de 2021. URL: <https://www.diagrams.net/> (visitado 10-2022).
- [104] Google Chrome. *El navegador creado por Google*. 2022. URL: https://www.google.com/intl/es_es/chrome/ (visitado 10-2022).
- [105] Roger Lott. *Geographic information — Well-known text representation of coordinate reference systems*. Open Geospatial Consortium. Arlington, VA, Estados Unidos, 13 de ago. de 2019. URL: <http://docs.opengeospatial.org/is/18-010r7/18-010r7.html> (visitado 10-2022).
- [106] Dirección General de Tráfico. *Dirección General de Tráfico*. 2022. URL: <https://www.dgt.es/inicio/> (visitado 2022).
- [107] Alba Gómez Varela. *Accidentes de tráfico con víctimas en España (2016-2021)*. 2 de dic. de 2022. URL: https://public.tableau.com/app/profile/alba.gomez.varela/viz/accidentes_trafico_2016_2021/Accidentes (visitado 02-12-2022).
- [108] OpenStreetMap contributors. *OpenStreetMap*. 2022. URL: <https://www.openstreetmap.org> (visitado 2022).
- [109] Comisión Europea. *Data protection in the EU*. 2022. URL: https://commission.europa.eu/law/law-topic/data-protection/data-protection-eu_en (visitado 23-11-2022).
- [110] Movilidad y Agenda Urbana Ministerio de Transportes. *Publicación anual con el tráfico de la RCE*. 2022. URL: <https://www.mitma.es/carreteras/trafico-velocidades-y-accidentes-mapa-estimacion-y-evolucion/mapas-de-trafico> (visitado 10-2022).
- [111] Movilidad y Agenda Urbana Ministerio de Transportes. *robots.txt*. 2022. URL: <https://www.mitma.gob.es/robots.txt> (visitado 31-10-2022).
- [112] Agencia Estatal de Meteorología. *Obtención API Key*. 2022. URL: <https://opendata.aemet.es/centrodedescargas/altaUsuario> (visitado 27-10-2022).
- [113] Global Biodiversity Information Facility (GBIF). *¿Qué es GBIF?* 2022. URL: <https://www.gbif.org/es/what-is-gbif> (visitado 23-11-2022).
- [114] The PostgreSQL Global Development Group. *Date/Time Functions and Operators*. 2022. URL: <https://www.postgresql.org/docs/current/functions-datetime.html#FUNCTIONS-DATETIME-EXTRACT> (visitado 29-11-2022).

- [115] Simon Kennedy. *Astral v3.0*. 2022. URL: <https://astral.readthedocs.io/en/latest/> (visitado 09-12-2022).
- [116] Brandon Craig Rhodes. *PyEphem Quick Reference*. 2022. URL: <https://rhodesmill.org/pyephem/quick.html> (visitado 09-12-2022).
- [117] Ministerio de la Presidencia. “Real Decreto 1428/2003, de 21 de noviembre, por el que se aprueba el Reglamento General de Circulación para la aplicación y desarrollo del texto articulado de la Ley sobre tráfico, circulación de vehículos a motor y seguridad vial, aprobado por el Real Decreto Legislativo 339/1990, de 2 de marzo”. En: *Boletín Oficial del Estado (BOE)* 306 (ene. de 2004). URL: <https://www.boe.es/buscar/doc.php?id=BOE-A-2003-23514> (visitado 25-10-2022).
- [118] Ministerio de la Presidencia. “Real Decreto 1428/2003, de 21 de noviembre, por el que se aprueba el Reglamento General de Circulación para la aplicación y desarrollo del texto articulado de la Ley sobre tráfico, circulación de vehículos a motor y seguridad vial, aprobado por el Real Decreto Legislativo 339/1990, de 2 de marzo”. En: *Boletín Oficial del Estado (BOE)* 306 (ene. de 2004). URL: <https://www.mitma.gob.es/carreteras/normativa-tecnica> (visitado 25-10-2022).
- [119] Alba Gómez Varela. *Calentamiento global: historia de un fracaso*. 14 de dic. de 2021. URL: https://public.tableau.com/app/profile/alba.gomez.varela/viz/accidentes_trafico_2016_2021/Accidentes (visitado 15-12-2022).
- [120] Jiawei Han, Micheline Kamber y Jian Pei. “Chapter 3: Data Preprocessing”. En: *Data Mining*. 3.^a ed. Boston, Estados Unidos: Morgan Kaufmann, 2012. ISBN: 978-0-12-381479-1. DOI: <https://doi.org/10.1016/B978-0-12-381479-1.00003-4>. URL: <https://www.sciencedirect.com/science/article/pii/B9780123814791000034>.
- [121] Dani Arribas-Bel. *KDE for spatial data*. GitHub Repository. 2015. URL: <https://gist.github.com/darribas/9109901> (visitado 08-12-2022).
- [122] Jake VanderPlas. *Kernel Density Estimation in Python*. GitHub Repository. 2013. URL: <http://jakevdp.github.io/blog/2013/12/01/kernel-density-estimation/> (visitado 08-12-2022).
- [123] Inc Uber Technologies. *H3. Hexagonal hierarchical geospatial indexing system*. 2022. URL: <https://h3geo.org/> (visitado 20-12-2022).
- [124] *Geohash. Tips Tricks*. 2022. URL: <http://geohash.org/site/tips.html> (visitado 20-12-2022).
- [125] Julia Collins y Allison G. Gaylord. *Same Place, Different Name: The Case for Research Site Identifiers*. Sep. de 2016. DOI: <https://doi.org/10.5281/zenodo.2564110>. URL: <https://zenodo.org/record/2564110#.Y7MEy3bMKUk> (visitado 20-12-2022).

- [126] José Gómez Castaño. *Algorithm and architecture for the generation and mobile high-speed transmission of traffic lights information to connected and automated cars in real time*. Jul. de 2019. DOI: <https://doi.org/10.5281/zenodo.3291817>. URL: <https://zenodo.org/record/3291817#.Y7MLV3bMKUm> (visitado 20-12-2022).
- [127] Shachar Kaufman y col. “Leakage in Data Mining: Formulation, Detection, and Avoidance”. En: *ACM Trans. Knowl. Discov. Data* 6.4 (dic. de 2012). ISSN: 1556-4681. DOI: [10.1145/2382577.2382579](https://doi.org/10.1145/2382577.2382579). URL: <https://doi.org/10.1145/2382577.2382579>.
- [128] Ben Auffarth. *Machine Learning for Time-Series with Python*. 1.^a ed. Birmingham, UK: Packt Publishing, oct. de 2021. ISBN: 978-1-80181-962-6. URL: <https://learning.oreilly.com/library/view/machine-learning-for/9781801819626/>.
- [129] Andreas C. Müller y Sarah Guido. “Chapter 2. Supervised Learning y Chapter 5. Model Evaluation and Improvement”. En: *Introduction to Machine Learning with Python*. 4.^a ed. California, Estados Unidos: O'Reilly Media, sep. de 2016. ISBN: 9781449369415. URL: <https://learning.oreilly.com/library/view/introduction-to-machine/9781449369880>.
- [130] GUILLÉN ESTANY, Montserrat y ALONSO ALONSO, María Teresa. *Modelos de regresión logística*. Universitat Oberta de Catalunya (UOC). Barcelona, España, 2020.
- [131] Scikit-learn developers. *sklearn.linear_model.LogisticRegression*. 2022. URL: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html (visitado 12-2022).
- [132] Stanley Lemeshow David W. Hosmer Jr. y Rodney X. Sturdivant. “Chapter 4: Model-Building Strategies And Methods For Logistic Regression”. En: *Applied Logistic Regression, 3rd Edition*. 3.^a ed. Massachusetts, Estados Unidos: Wiley, abr. de 2013. ISBN: 978-0-470-58247-3. URL: <https://learning.oreilly.com/library/view/applied-logistic-regression/9781118548356/>.
- [133] Scikit-learn developers. *sklearn.neighbors.KNeighborsClassifier*. 2022. URL: <https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html> (visitado 12-2022).
- [134] Scikit-learn developers. *sklearn.tree.DecisionTreeClassifier*. 2022. URL: <https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html> (visitado 12-2022).
- [135] Aurélien Géron. “Chapter 2: End-to-End Machine Learning Project”. En: *Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems*. 2.^a ed. California, Estados Unidos: O'Reilly Media, sep. de

2019. ISBN: 978-1-492-03264-9. URL: <https://learning.oreilly.com/library/view/hands-on-machine-learning/9781492032632/>.
- [136] Scikit-learn developers. *sklearn.ensemble.RandomForestClassifier*. 2022. URL: <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html> (visitado 12-2022).
- [137] Aurélien Géron. “Chapter 4: From Gradient Boosting to XGBoost”. En: *Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems*. 2.^a ed. California, Estados Unidos: O'Reilly Media, sep. de 2019. ISBN: 978-1-492-03264-9. URL: <https://learning.oreilly.com/library/view/hands-on-machine-learning/9781492032632/>.
- [138] Scikit-learn developers. *sklearn.ensemble.GradientBoostingClassifier*. 2022. URL: <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html> (visitado 12-2022).
- [139] DGT3.0. *Caso de Uso 6. Mapa de movilidad*. GitLab Repository. Nov. de 2022. URL: https://gitlab.cs.cmobility30.es/dgt3.0_esp/caso-de-uso-6 (visitado 12-2022).
- [140] DGT3.0. *Caso de Uso 4. Panel de mensajes virtual*. GitLab Repository. Jul. de 2022. URL: https://gitlab.cs.cmobility30.es/dgt3.0_esp/caso-de-uso-4 (visitado 12-2022).
- [141] Movilidad y Agenda Urbana Ministerio de Transportes. *Compra Pública de Innovación en Carreteras*. Nov. de 2022. URL: <https://www.mitma.gob.es/carreteras/innovacion/compra-publica-de-innovacion> (visitado 01-12-2022).
- [142] Movilidad y Agenda Urbana Ministerio de Transportes. *Reto 10. Medidas de protección para usuarios vulnerables y para accidentes con fauna*. Dic. de 2022. URL: <https://www.mitma.gob.es/carreteras/innovacion/compra-publica-de-innovacion/reto-10-medidas-de-proteccion-para-usuarios-vulnerables-y-para-accidentes-con-fauna> (visitado 01-12-2022).