

# Inteligencia Artificial

## Práctica 3

Interpretabilidad

**Realizado por:**

Alberto González Fernández, N° EXP 21719077

# Índice

<b>Índice</b>	<b>2</b>
<b>Contexto</b>	<b>3</b>
<b>Objetivo</b>	<b>3</b>
<b>Base de Datos</b>	<b>4</b>
<b>Resolución</b>	<b>5</b>
<b>Librerías</b>	<b>6</b>
<b>Problemas encontrados</b>	<b>7</b>
<b>Conclusiones</b>	<b>7</b>

# Contexto

Los accidentes de tráfico son la primera causa de fallecimientos de jóvenes en España y desde hace tiempo se encuentra en el top diez de mayor número de muertes causadas en todo el mundo. Desde que el uso del automóvil comenzó a extenderse, este problema no ha hecho nada más que aumentar, por lo que, para frenarlo, en España, se hicieron numerosos cambios a lo largo de los años, el más notable fue la introducción del carnet por puntos en 2006, que, debido al miedo de los conductores de perder puntos y potencialmente el permiso completamente, bajó mucho el número de víctimas mortales en los siguientes años hasta quedar justo por debajo de los dos mil anuales. Pero esto no es suficiente, ya que aun así este número puede ser reducido con más medidas.

Otro causante de la reducción del número de víctimas mortales es la organización europea Euroncap, quien se dedica a hacer test de seguridad frente a distintos tipos de colisiones y dando una nota final del cero al cinco para informar a los posibles compradores si el vehículo es seguro en caso de accidente, produciendo un aumento en el concienciamiento de los fabricantes por la seguridad, y, aunque algunos modelos siguen suspendiendo el test, el número de casos en lo que esto pasa ni roza los números cuando se creó esta organización.

Por último, otro gran factor en la ayuda a la prevención es la mejora y el aumento de la infraestructura de carreteras, ya que hace años la gran mayoría de carreteras eran comarcales y secundarias, mientras que ahora, con el aumento del número de autopistas y autovías se ha aumentando la seguridad, y, aunque las carreteras secundarias son menos utilizadas en la actualidad, se siguen mejorando para evitar cualquier posible incidente debido a este motivo.

# Objetivo

El objetivo de este estudio es, haciendo uso de una base de datos que recoge información de hasta dos millones de accidentes en los Estados Unidos, aplicar el análisis a realizar a España, ya que no hay este tipo de información pública en el país.

Se ha usado una base de datos de los Estados Unidos ya que aunque se encuentre en otro continente, es otro país occidental con un contexto muy equiparable al contexto de España, por lo que con lo que se analice se podrán sacar conclusiones aplicables a cualquier país occidental.

Con ayuda de la interpretabilidad, que nos ayuda a dar contexto y nos ayuda a confiar en los atributos de la base de datos.

# Base de Datos

La base de datos recoge dos millones de datos de accidentes en Estados Unidos. Se divide en cuatro tipos de datos, que serán los que se analizan posteriormente.

El primero es la severidad, del uno al cuatro cuanto más grande el valor mayor impacto sobre el tráfico, siendo uno un accidente que no produce ningún tipo de retención, mientras que cuatro se trata de una retención de más de dos horas.

El segundo es la localización del accidente, ciudad, estado, etc.

El tercero son las condiciones meteorológicas, temperatura, humedad, clima, etc.

El último es la condición de la carretera, si el accidente se produce en un stop, un semáforo, un cruce, etc.

Cabe destacar que la base de datos solo almacena datos sobre circunstancias ajenas al conductor, ya que si se quisieran registrar datos sobre el coche se necesitaría la telemetría del propio vehículo, cosa que es muy difícil ya que es imposible tener acceso a los datos de todos los vehículos en circulación, por lo que la velocidad o el giro del coche no se tendrán en cuenta en este estudio. Por otra parte, los datos del propio conductor tampoco se pueden obtener, ya que se tendrían que hacer test a los conductores antes y después del accidente, otra vez algo imposible ya que hacer experimentos de este tipo es totalmente ilegal, si el conductor se despista o esté en estado de embriaguez son datos de los que no se disponen.

Por tanto, el estudio solo se realizará sobre circunstancias ajenas al conductor.

Estos datos han sido obtenidos desde la página web de [Kaggle](#).

# Resolución

El código para seguir la resolución se puede encontrar [aquí](#).

El primer paso para poder resolver la práctica será importar las librerías principales, ya mencionadas, y cargar la base de datos. Se hace como objetivo de la interpretabilidad la severidad de los accidentes.

Lo siguiente es pasar todos los datos de la base de datos a tablas, haciendo un cambio de tipo de valor a numérico para poder incluirlo en la tabla.

A continuación, se realiza un mapa de calor de correlación de las diferentes atributos de la base de datos, obteniendo la máxima relación entre temperatura y visibilidad y los cruces con los semáforos.

Y, por otra parte, se realiza una comparación de características bivariadas relacionando cuatro de los atributos más fuertes de la base de datos como son la temperatura, la humedad, en estado y si es de día o de noche.

Una vez se han realizado estas tablas de comparación, se procede a realizar el entrenamiento del modelo haciendo uso de la librería *XGBoost*, y se evalúa su rendimiento llegando a hacer uso de una matriz de confusión, en la que obtendremos los resultados de la clasificación del modelo y el porcentaje de acierto que ha tenido.

También se realiza una clasificación ROC y AUC, en la que se grafica los resultados de falsos positivos y negativos que ha tenido el modelo.

Es ahora, una vez acabado el entrenamiento del modelo que se entra en la materia de los métodos de interpretación.

Haciendo uso de la librería *XGBoost* lo primero que se hace es graficar la importancia que le da la librería dando mayor o menor valor al peso que le da la librería a los diferentes atributos.

También se hace uso de la librería *ELI5*, con la que, en vez de hacer una gráfica se muestra en forma de tabla el peso que le da a cada atributo. Esto se hace de forma normal y de forma permutada, dando dos resultados diferentes.

Lo próximo es realizar un gráfico de dependencia parcial haciendo uso de la librería *pdpbox*. Con esta gráfica se puede observar como un atributo afecta a la predicción del modelo, por ejemplo, en la gráfica del código se ha realizado una comprobación del impacto de la humedad, obteniendo de resultado que hay más posibilidades de gravedad cuanto más porcentaje de humedad hay.

También se realiza un gráfico ICE, similar al gráfico anterior pero con un nivel de detalle superior.

Por otra parte, el gráfico PD bivalente realiza un mapa de calor de las relaciones entre dos atributos, en el caso del código de la práctica, se ha realizado la comparación entre el estado y la humedad.

Otra forma de calcular la interpretabilidad es haciendo uso de la librería *SKATER*.

De la misma forma que hace *XGBoost*, se grafica el peso que se le da a los diferentes atributos de la base de datos.

También se realiza una *plot\_list*. en la que se ve en forma de mapa de calor y de gráfica 3d la relación entre dos atributos y su efecto en el resultado. En el caso de la resolución de la práctica se han usado el estado y la temperatura.

Lo siguiente que se realiza en el código es el uso de *LIME*, un algoritmo para poder observar el comportamiento de un modelo. Como ejemplo se han usado dos tipos de accidentes, en el primero el resultado es 100% leve, mientras que en el segundo, el modelo predice un 72% leve y un 28% grave.

Otra forma de observar la interpretación es mediante el uso de la librería *SHAP*. Como en las otras librerías, lo primero que se realiza es la tabla de impacto en el modelo. Y, después, se grafica la forma de decisión del modelo para determinar el resultado. También se grafica a puntos el impacto de cada atributo sobre el resultado. De igual forma se hace la comparación entre la dependencia de dos atributos, en el caso del código, se compara la temperatura y el estado.

Una última forma de calcular la interpretación es mediante la librería FairML.

## Librerías

- Pandas: Pandas se trata de una librería para la creación y manipulación de dataframes.
- Numpy: Numpy es una librería usada para realizar computación científica en Python.
- Matplotlib: Matplotlib es una librería de Python para crear visualizaciones estáticas, animadas e interactivas.
- Seabourn: Seabourn es una librería de Python para la visualización de datos basada en matplotlib. Provee una interfaz de alto nivel para el graficado de estadísticas.
- Sklearn: Sklearn es una librería de Python simple y eficiente empleada para realizar predicciones de datos.
- XGBoost: XGBoost es una librería de Python que implementa algoritmos de aprendizaje automático.
- ELI5: ELI5 es una biblioteca de Python que permite visualizar y depurar varios modelos de Machine Learning.
- SKATER: Skater es un marco unificado para habilitar la interpretación de modelos y ayudar a construir un sistema de aprendizaje automático interpretable que a menudo se necesita para casos de uso del mundo real.
- Shap: SHAP es una librería de Python usada para explicar el resultado de cualquier modelo de aprendizaje automático. Conecta la asignación óptima de créditos con explicaciones locales utilizando los valores clásicos de Shapley de la teoría de juegos y sus extensiones relacionadas.
- FairML: FairML, se trata de una caja de herramientas integral para auditar modelos predictivos mediante la cuantificación de la importancia relativa de las entradas del modelo. FairML aprovecha la compresión del modelo y cuatro algoritmos de clasificación de entrada para cuantificar la dependencia predictiva relativa de un modelo en sus entradas.

# Problemas encontrados

En la realización de esta práctica se han encontrado tres problemas.

Al hacer la evaluación del rendimiento del modelo se realiza con éxito el cálculo de la precisión, mientras que, a la hora de hacer la matriz de confusión, al tratarse de cuatro posibles soluciones, da error de que no está apoyada la multiclase.

Al realizar el gráfico ICE univariable, en mitad del proceso da error debido a un parámetro `n_jobs` dentro de la librería el cual no se ha podido modificar satisfactoriamente.

Por otro lado, el uso de la librería SHAP ha dado bastantes problemas, ya que al hacer uso de dataframes en 3d, debido a las columnas, filas y las cuatro posibles salidas de la base de datos, no se le puede pasar un dataframe en 2d como pide la librería.

Y, por último, la librería FairML, al calcular las importancias salta el error '*Division by zero*', a lo que no se le ha encontrado solución, ya que la base de datos tiene que tener valores iguales a cero si o si, debido a haber entradas booleanas.

## Conclusiones

Tras realizar este estudio de interpretabilidad se puede concluir que, aunque esta base de datos esté muy completa con hasta más de treinta parámetros diferentes, se puede observar mediante los diferentes usos de las librerías que se han visto anteriormente, muchos de estos diferentes atributos que se almacenan no tienen nada que ver con la implicación en el resultado final de si el accidente será grave o leve, como la condición del tiempo o la velocidad del viento, comparado con otros atributos que tienen un peso mucho mayor, como la humedad o la visibilidad.

También se ha sacado como conclusión que hay que tener en cuenta el tipo de datos y la forma con la que se van a trabajar, ya que puede que, como en este caso, alguna librería de problemas con el formato.