FACULTAD DE ESTUDIOS ESTADÍSTICOS

MÁSTER EN MINERÍA DE DATOS E INTELIGENCIA DE NEGOCIOS

Curso 2019/2020

Trabajo de Fin de Máster

TITULO: *Aplicación de Técnicas de Minería de Datos para la Clasificación de la Gravedad de los Accidentes de Tráfico en Motocicletas*

***Alumno*:** Mario Roberto Cerda García

***Tutor*:** Ramón Alberto Carrasco González

Septiembre de 2020

UNIVERSIDAD COMPLUTENSE
MADRID

Índice de contenidos

Índice de ilustraciones

1. Introducción

1.1. Contexto general de los accidentes de tráfico

Un accidente de tráfico es aquel que se produce en las vías o terrenos incluidos en el ámbito de aplicación de la legislación sobre tráfico, circulación de vehículos a motor y seguridad vial.

En el ámbito de la salud, los accidentes de tráfico constituyen una tragedia en nuestra sociedad moderna. Si bien es cierto que en los últimos años se ha tomado mayor consciencia de la importancia que representa este tema, aún queda mucho camino por recorrer en este sentido.

Por este motivo, la OMS (Organización Mundial de la Salud) se ha centrado especialmente en estudiar y evaluar este tipo de eventos que cada año terminan con la vida o afectan a la salud de millones de personas en el mundo. De hecho, la OMS estima que cada año mueren 1.35 millones de personas en todo el mundo como consecuencia de los accidentes de tráfico y hasta 50 millones de personas más sufren algún tipo de lesión en esta clase de accidentes. A nivel mundial, como muestra la ilustración 1, la OMS sitúa a los accidentes de tráfico como la octava causa de muerte, siendo responsables del 2.5% de las muertes en el mundo. Sin embargo, si se consideran únicamente a las personas entre 5 y 29 años, los accidentes de tráfico son la principal causa de muerte en este grupo (World Health Organization et al., 2018). En definitiva, se pierden muchas vidas cada año y, probablemente, muchas de ellas pueden ser evitadas si se tomasen mayores precauciones.

Rank	Cause	% of total deaths
	All Causes	
1	Ischaemic heart disease	16.6
2	Stroke	10.2
3	Chronic obstructive pulmonary disease	5.4
4	Lower respiratory infections	5.2
5	Alzheimer's disease and other dementias	3.5
6	Trachea, bronchus, lung cancers	3.0
7	Diabetes mellitus	2.8
8	Road traffic injuries	2.5
9	Diarrhoeal diseases	2.4
10	Tuberculosis	2.3

2016 WHO Global Health Estimates

Ilustración 1. Principales causas de muerte en el mundo (WHO, 2018)

Por otra parte, desde un punto de vista económico, los accidentes de tráfico también tienen un impacto profundo tanto en la economía familiar como en la economía nacional. En su informe sobre la situación mundial de la seguridad vial 2015, la OMS estima un impacto de las muertes y las lesiones causadas por los accidentes de tráfico del 3% del PIB (Producto Interior Bruto)

de forma general, aunque en países en vías de desarrollo pude llegar hasta el 5% de su PIB (World Health Organization, 2015). En España, según datos de la OCDE (Organización para la Cooperación y el Desarrollo Económicos), en 2013 los accidentes de tráfico le costaron al país 9640 millones de euros (0.94% del PIB) en atención a los accidentados (Gálvez, 2015).

A nivel europeo, en 2018 murieron 25100 personas en accidentes de tráfico en carreteras de la Unión Europea. En este contexto, España registró el sexto mejor puesto en víctimas por millón de habitantes. Con 39 muertes por millón de habitantes, España y Alemania comparten esta sexta posición situándose por detrás de Reino Unido (28), Dinamarca (30), Irlanda (31), Suecia (32) y Malta (38) (Expansión, 2019).

Centrándonos en el contexto del objeto de estudio, en lo que respecta a los motociclistas, en Europa hay tres datos importantes a destacar según datos del Observatorio Europeo de la Seguridad Vial (ERSO por sus siglas en inglés), organismo dependiente de la Comisión Europea. En primer lugar, en 2016 el 15% de las muertes en accidentes de tráfico en Europa correspondieron a conductores de motocicletas. En segundo lugar, como muestra la ilustración 2, en el periodo comprendido entre 2007 y 2016, las muertes en accidentes de motocicletas han sido la categoría de vehículo que menos se ha reducido porcentualmente. Por último, éste es el medio de transporte en el que más estacionalidad se observa en las víctimas mortales, siendo los meses de verano los meses en los que se concentran la mayor parte de las muertes en accidentes viales en motocicleta (European Commission, 2018).

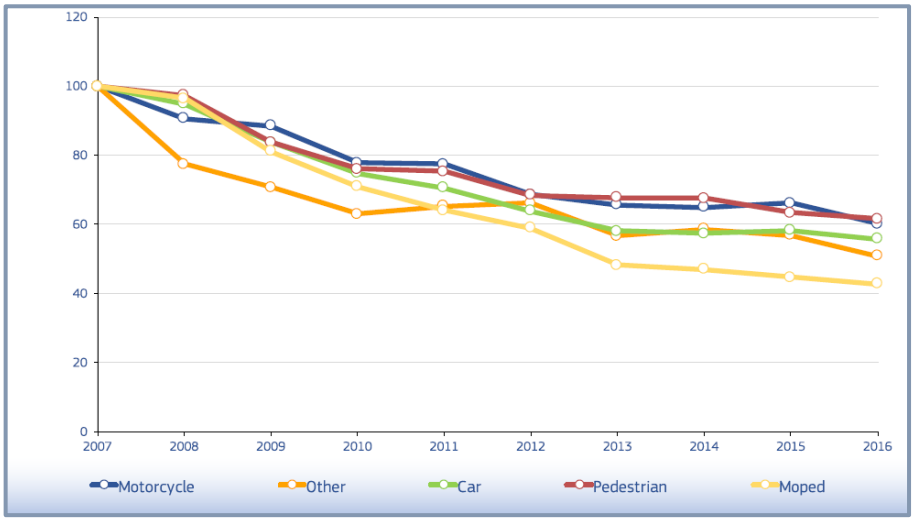


Ilustración 2. Evolución de accidentes mortales según medio de transporte. (European Comission, 2018)

En España, los datos de siniestralidad son bastante agridulces. Por un lado, hay que destacar que 2019 se cerró con la mejor cifra de muertes en carretera, alcanzando un mínimo histórico de 1098 fallecidos. Sin embargo, las muertes de motoristas alcanzaron la cifra más alta desde

2009, con un total de 264 motoristas fallecidos (Espinosa, 2020). Estos datos demuestran que ha habido un avance respecto a los accidentes en líneas generales debido a la mejora en los sistemas de seguridad (principalmente en vehículos de 4 ruedas) y a la mejora de las condiciones de la vía. Sin embargo, parece que los motociclistas aún son muy vulnerables debido a la menor estabilidad del vehículo que conducen y los pocos elementos de seguridad pasiva con los que cuentan, por lo que, en caso de colisión, es su cuerpo el que absorbe la fuerza del impacto (RACE, 2019).

1.2. Motivación de la investigación

La gestión de los accidentes de tráfico en términos generales continúa siendo una asignatura pendiente y, como se ha visto anteriormente, puede englobarse como un tema de salud pública al ser una problemática que anualmente provoca grandes cantidades muertos y heridos con secuelas de por vida.

Además, si bien es cierto que ha habido grandes mejoras en lo que respecta a la siniestralidad a nivel general, en el caso de los usuarios más vulnerables, como el caso de los motoristas, se ha visto que este avance no sigue el mismo ritmo.

Con este trabajo de investigación se centra el foco de la accidentalidad en estos usuarios vulnerables y se tratará de investigar todo lo relacionado a este tipo accidentes para poder observar posibles patrones que ayuden a entender mejor los accidentes de motociclistas.

1.3. Objetivo de la investigación

El objetivo principal que busca este trabajo de investigación es obtener un modelo que permita clasificar a los motociclistas accidentados en un accidente de tráfico entre dos posibles categorías: accidentado grave y accidentado leve. En este trabajo, se ha considerado que el accidentado grave es aquella persona que muere o sufre heridas graves que requieren hospitalización durante el accidente o durante los treinta primeros días tras el accidente.

La primera consideración importante fue, por tanto, agrupar a los muertos y heridos graves en accidentes de motocicletas en una sola categoría. El motivo para realizar la investigación con este enfoque es que este tipo de accidentes es muy característico, ya que se trata de vehículos que alcanzan velocidades iguales o superiores a los turismos, pero que, en caso de accidente, no existe un chasis que proteja al conductor y/o pasajeros. Por ello, la probabilidad de morir o sufrir lesiones que deriven en una minusvalía es superior a la de otros vehículos, por lo que ambos resultados serían bastante perjudiciales tanto si sucede uno como si sucede otro.

Además, como se emplearán técnicas de minería de datos para el desarrollo de este trabajo, se desprenden varios objetivos secundarios:

- Lograr un tratamiento adecuado de los datos con los que se va a realizar el proyecto.
- Entrenar diferentes modelos utilizando varios algoritmos de machine learning para poder obtener aquel que clasifique mejor que el resto.
- Comprender aquellas variables que más influyen en el tipo de consecuencia que sufren los motoristas accidentados.

2. Estado del arte

Antes de continuar, se ha analizado aquellas investigaciones previas que hayan tenido como objetivo desarrollar modelos de predicción o clasificación empleando técnicas de machine learning o cualquier otra basada en minería de datos.

Ya en 2005 se puede encontrar una investigación que usa técnicas de machine learning para analizar accidentes de tráfico (Chong et al., 2005). En esta investigación, para poder clasificar la severidad de una lesión que constaba de cinco clases, se emplearon redes neuronales, árboles de decisión, Support Vector Machines (SVM) con kernel polinomial y radial, así como un modelo híbrido de redes neuronales con árboles de decisión (DTANN). Se trabajó con un dataset de 417670 casos con accidentes reportados en Estados Unidos entre 1995 y 2000. Se concluyó que los mejores modelos (evaluados por accuracy) a la hora de clasificar fueron el modelo híbrido y el árbol de decisión. Además, se pone de manifiesto que si se hubiera podido disponer de la información relativa a la velocidad de los vehículos los modelos habrían presentado una mejor performance.

En 2011, en un artículo de investigación se desarrolla un modelo de clasificación binario (lesionado o ileso) utilizando algoritmos de SVM (Montt et al., 2011). En esta ocasión, se busca clasificar el grado de severidad de las lesiones en personas involucradas en accidentes de tráfico y se modeliza concretamente una clasificación LS-SVM (Least-Squares Support-Vector Machine) con PSO (Particle Swarm Optimization) para estimar los mejores parámetros del algoritmo SVM. Se eligió el kernel RBF (Radial Basis Function) al ser el que reporta mejores resultados en la literatura y en trabajos relacionados anteriores. El dataset original contiene 12 variables independientes y alrededor de 70000 observaciones de accidentes ocurridos entre 2003 y 2009 en Valparaíso (Chile), si bien la modelización se realizó con 3000 observaciones. Los resultados de la investigación permitieron concluir que el algoritmo SVM generaliza

adecuadamente la clasificación del estado de las personas que sufren un accidente de tráfico en Chile.

También en 2011 se realiza un estudio que clasifica los accidentes de tráfico en base a la severidad de las lesiones (lesionado leve o muerto/lesionado grave) en carreteras rurales de Granada (España) mediante el uso de redes bayesianas (de Oña et al., 2011). Dos tercios de los 1536 accidentes con los que cuenta el set de datos original se emplearon para entrenar las redes y el tercio restante para testearlo. En esta investigación, que contó con 18 variables predictoras, se concluyó que las redes bayesianas pueden ser útiles para clasificar este tipo de accidentes según el tipo de gravedad de la lesión.

Un par de años más tarde, en otro estudio se obtuvieron las causas más frecuentes de los accidentes en Chile en los últimos 8 años mediante inteligencia computacional (Montt et al., 2013). En esta ocasión se observó que las causas más frecuentes de accidentes de tráfico son circular sin mantener una distancia de seguridad razonable y la pérdida del control del vehículo. En los accidentes de peatones entre las causas más frecuentes se encuentran el no respetar el paso de cebra o el cruzar la calzada de forma sorpresiva o descuidada.

Ese mismo año, en un nuevo estudio se emplearon árboles de decisión para estudiar la severidad (herido leve o muerto/herido grave) de los accidentes de tráfico en carreteras rurales en Granada (España) desde 2003 hasta 2009, ambos incluidos (Abellán et al., 2013). Para este estudio se utilizaron únicamente los accidentes con un vehículo involucrado en carreteras rurales de doble sentido y sin considerar los accidentes en intersecciones. El dataset cuenta con 1801 accidentes y 19 variables. En este estudio se obtuvieron más de 70 reglas de decisión y se pudieron conocer algunos de los factores que más afectan a la accidentalidad en motocicletas, siendo la salida de la calzada uno de los tipos de accidentes que más lesiones graves o mortales causa.

Uno de los estudios más recientes lo encontramos en (Herceg & Yaman, 2019), donde se emplean métodos de clasificación mediante el clasificador bayesiano ingenuo y el algoritmo C4.5 (árbol de decisión) para poder comprender el rol del factor humano y el factor ambiente en la severidad de los accidentes de tráfico. Se concluye que el árbol de decisión C4.5 provee mejores resultados prediciendo la severidad en los accidentes y que el modelo bayesiano es un modelo que se construye más rápido. Además, se observó que el género, el tipo de persona, la posición en el asiento y el uso de los sistemas de retención están relacionados con el grado de la lesión. Por su parte, las condiciones del tiempo y la fracción horaria influyen en el número

de accidentes producidos. También destaca que el número de ocupantes del vehículo está correlacionado negativamente con el número de accidentes fatales.

Por último, ya en 2020 es necesario destacar un trabajo para predecir la gravedad de los accidentados de tráfico en Barcelona (Vila, 2020). En este trabajo se utiliza Regresión Logística y Random Forest para realizar las predicciones (herido leve o herido grave/muerto) en un dataset de 22356 observaciones. Cada técnica se aplica sobre el conjunto desbalanceado (típico en datos de accidentes) y sobre conjuntos balanceados bajo cuatro métodos: upsampling, downsampling, SMOTE y ROSE. Como resultado destacable se puede extraer que el balanceo de clases no supuso una mejora significativa en este tipo de datos.

3. Fundamento teórico

3.1. Metodología SEMMA

Para llevar a cabo este trabajo se plantea seguir la metodología SEMMA, que se compone de una serie de pasos secuenciales desarrollados por la compañía SAS Institute mostrados en la ilustración 3. Estos pasos son:

- Sample (muestrear): proceso de obtención del dataset con el que se va a trabajar.
- Explore (explorar): fase que busca entender la información que contienen los datos para detectar relaciones, anomalías y tendencias.
- Modify (modificar): supone realizar cambios en las variables de tal forma que la modelización se pueda llevar a cabo de una forma óptima.
- Model (modelizar): desarrollar todos aquellos modelos con técnicas de machine learning para poder clasificar o predecir la variable objetivo.
- Assess (evaluar): supone evaluar los modelos creados y compararlos entre sí bajo diversas métricas.

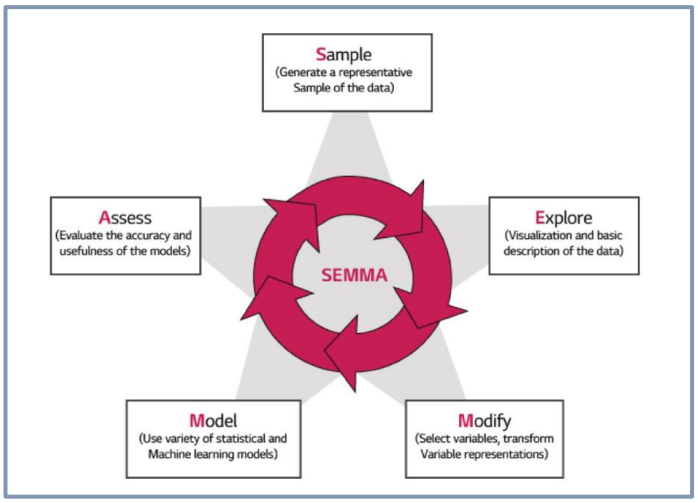


Ilustración 3. Metodología SEMMA (Calviño, 2019)

3.2. Algoritmos de Machine Learning

El aprendizaje automático o machine learning trata, en esencia, de extraer conocimiento de los datos, pues se trata de un campo de la investigación que conjuga estadística, inteligencia artificial y ciencias de la computación. Como su nombre indica, lo que se busca es que las máquinas aprendan por sí solas, y esto se consigue mediante el uso de algoritmos que permitan identificar patrones que se esconden en los datos.

Los algoritmos de machine learning se puede dividir en dos ramas principalmente: aprendizaje supervisado y aprendizaje no supervisado. En este trabajo se utilizarán los algoritmos pertenecientes a la primera rama y que se nombran a continuación.

3.2.1. Redes Neuronales

Así como un cerebro usa una red de células interconectadas llamadas neuronas para crear un procesamiento paralelo masivo, una red neuronal artificial utiliza una red de nodos para resolver problemas de aprendizaje (Lantz, 2015).

Los nodos input son las variables independientes de los modelos. El nodo output será la variable dependiente del modelo, aunque puede haber más de una. La capa oculta la conforman los nodos ocultos, siendo variables artificiales que no existen en los datos. Es por esta capa oculta por la que este algoritmo se considera un modelo de caja negra. Esta primera aproximación permite hacerse una idea de la arquitectura de una red neuronal, representada en la ilustración 4.

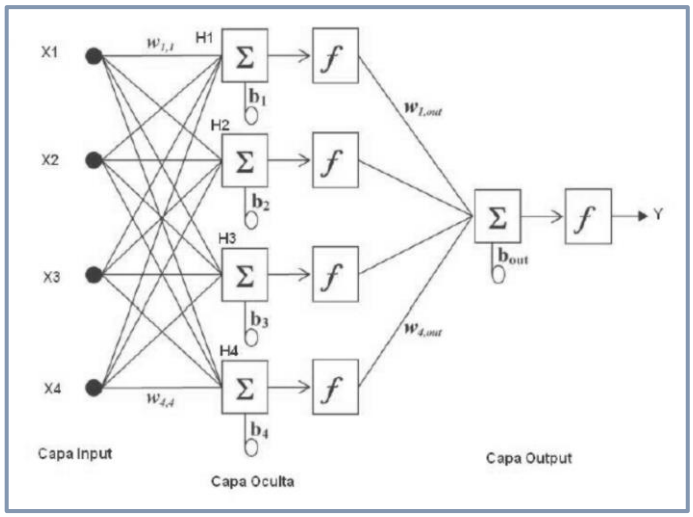


Ilustración 4. Arquitectura de una red neuronal (Portela, 2019)

En una red neuronal la capa input se conecta con la capa oculta mediante la función de combinación, representada por Σ, donde los pesos w_{ij} hacen el papel de parámetros a estimar. Posteriormente, tras aplicar la función de combinación, se aplica a cada nodo oculto la función de activación, representada por f, siendo la tangente hiperbólica la más utilizada. Por último, se aplica la combinación y la activación de la capa oculta a la capa output (Portela, 2019).

3.2.2. Random Forest

Random Forest se puede entender como un método de ensamblado de árboles de decisión, pues combina los principios básicos de Bagging (bootstrap averaging) con la selección aleatoria de variables, de manera que se pueda mejorar la diversidad de los árboles de decisión calculados (Lantz, 2015).

La fortaleza de este algoritmo predictivo está, por tanto, en que combina el resultado de muchos árboles de decisión permitiendo mejorar considerablemente el resultado que se obtendría si se emplean árboles individuales. El proceso que sigue el algoritmo se puede resumir de la siguiente forma:

Dados los datos de tamaño N,

1. Repetir m veces i), ii), iii):

   i) Seleccionar *N* observaciones con reemplazamiento de los datos originales.

   ii) Aplicar un árbol de la siguiente manera:

   En cada nodo, seleccionar *p* variables de las *k* originales y de las *p* elegidas, escoger la mejor variable para la partición del nodo.

   iii) Obtener predicciones para todas las observaciones originales *N*

2. Promediar las *m* predicciones obtenidas en el apartado 1 (Portela, 2019).

3.2.3. Gradient Boosting

Gradient Boosting es un método que consiste en repetir la construcción de árboles de decisión modificando ligeramente las predicciones iniciales cada vez, intentando ir minimizando los residuos en la dirección de decrecimiento.

Por tanto, este algoritmo se basa en ir actualizando las predicciones en la dirección de decrecimiento dada por el negativo del gradiente de la función de error $L(y_i, f(x_i))$. La función $f(x_i)$ es la función de predicción de y_i basada en los valores de x_i. El proceso que sigue el algoritmo es el siguiente:

0. $\hat{p}_i^0 =$ % de 1 en los datos.
1. Calcular el residuo actual $r_i^m = y - \hat{p}_i^m$ (este residuo es el gradiente, dada la función de error Deviance).
2. Ajustar mediante árbol de regresión los residuos $r_i^m =$ variable dependiente, $X =$ vector de variables predictoras $\hat{r}_i^m$
3. Actualizar f_i mediante $f_i^{m+1} = f_i^m + v \cdot r_i^m = \frac{1}{2}\log\left(\frac{\hat{p}_i^n}{1-\hat{p}_i^n}\right) + v \cdot r_i^m$.
4. Actualizar la probabilidad predicha mediante $\hat{p}_i^m$
5. Volver al paso 1. Se repiten los pasos 1 a 5 hasta la convergencia o bien sobreajuste (Portela, 2019).

3.2.4. Extreme Gradient Boosting

Está basado en Gradient Boosting básico. La principal novedad que presenta este algoritmo es la utilización de la regularización, que está orientada a reducir la varianza de los errores, evitando el sobreajuste que podían tener los modelos de Gradient Boosting.

La diferencia que aporta la regularización es que ésta interviene en la optimización interna del algoritmo, es decir, en este algoritmo se modifica el gradient boosting a la hora de construir cada árbol con una función de penalización basada en el número de hojas y el score-predicción en cada hoja. Es por ello por lo que el algoritmo XGBoost prefija dos parámetros de regularización: gamma y lambda.

$$\Omega(f_t) = \underbrace{\gamma T}_{\text{Number of leaves}} + \underbrace{\frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2}_{\text{L2 norm of leaf scores}}$$

Por tanto, a la hora de construir cada árbol, se tiene en cuenta esa penalización y se utiliza un algoritmo secuencial para evaluar cómo se hace cada división de manera que la función objetivo sea la habitual, pero penalizada por la función anterior (Portela, 2019).

3.2.5. Support Vector Machines

La idea detrás de este algoritmo radica en plantear el problema de separación lineal de clases mediante el uso de un hiperplano de separación. Este planteamiento se basa en tres ideas importantes:

1. Maximal margin. Se trata de construir un hiperplano que permita separar con el máximo margen posibles las dos clases. Esto a menudo mejora tanto el sesgo como la varianza de los resultados.

   Se trata, por tanto, de hallar el vector de parámetros w que maximice el margen. Las ecuaciones de los hiperplanos que delimitan el margen son $w \cdot x = 1$ y $w \cdot x = -1$. Denotando y como $(-1,1)$ en el problema de clasificación, hay que maximizar la distancia entre los dos hiperplanos de separación ($\frac{2}{\|w\|}$).
2. Soft Margin. Dado que no existe la separación perfecta, para evitar el sobreajuste es necesario permitir observaciones mal clasificadas. Esto supone introducir una variable ξ de residuo y una constante C de regularización del margen que está relacionada inversamente con la anchura del margen y, por tanto, el permiso para fallar. A mayor C menores residuos ξ y menor margen.
3. Kernel. Parte de que la separación entre clases no siempre es lineal. Para solventar este problema y poder aplicar un algoritmo de separación lineal en datos no separables linealmente se trabaja en un espacio de dimensión superior donde sí tenga sentido la separación lineal.

   El problema es que este aumento de dimensión se traduce en cálculos impracticables, por lo que se utiliza el denominado "truco Kernel" por el que cualquier algoritmo que dependa sólo de los productos escalares permite trabajar computacionalmente en una dimensión controlada a través de una función llamada Kernel que tiene que cumplir:

   $$K(x, y) = < \varphi(x), \varphi(y) >$$

   Sustituyendo en el problema de optimización x_i, y_i por su Kernel $K(x_i, y_i)$, implícitamente se está aumentando la dimensión del espacio de variables de cara a la construcción de hiperplanos de separación, sin pasar realmente por la creación de nuevas variables (Portela, 2019).

3.2.6. Ensamblado

Los métodos de ensamble consisten en la construcción de predicciones a partir de la combinación de varios modelos. En un problema de clasificación cada algoritmo que ya ha sido calculado arrojará una serie de probabilidades, con las cuales se puede trabajar de tres formas distintas:

- Promediado (averaging). Como indica el término consiste en promediar las probabilidades de cada uno de los algoritmos, si bien se puede ponderar cada uno de la forma que se considere oportuna.
- Voto. Consistiría en predecir el resultado con mayoría entre las predicciones.
- Combinación a partir de otro algoritmo. Se trata de introducir en uno de los modelos las predicciones de otros modelos como variables independientes (Portela, 2019).

Por último, es necesario indicar que para llevar a cabo el proyecto se utilizará, en primer lugar, el software SAS Enterprise Miner, ya que permite una rápida interpretación de las variables, así como su posterior tratamiento mediante el empleo de nodos de ejecución. En cuanto a la fase de modelización de algoritmos de machine learning se utilizará el software R, cuyo uso para este tipo de trabajos está ampliamente extendido gracias a su gran número de librerías, como es el caso de *Caret*.

3.3. Métricas de evaluación de modelos.

Para cada técnica de machine learning se calcularán varios modelos, pues el objetivo es obtener un modelo que sea el mejor para ese algoritmo. Para poder analizar la performance de cada uno de ellos se utilizarán la tasa de fallos y el área bajo la curva ROC.

- Tasa de fallos. Mide el porcentaje de predicciones erradas y se puede considerar como la opuesta a la exactitud del modelo.

$$Tasa\ de\ fallos = \frac{FP + FN}{TP + TN + FP + FN}$$

- Área bajo la curva (AUC). Esta métrica parte de la obtención de la curva ROC, la cual nos indica la capacidad de clasificación que tiene un modelo en todos los puntos de corte o umbrales de clasificación. La ventaja del AUC es que, al medir todo el área por debajo de la curva, permite obtener una métrica eficiente y comparar los modelos más fácilmente al ser invariable con respecto al umbral de clasificación.

Posteriormente, se presentará la matriz de confusión del modelo ganador para cada uno de los algoritmos mencionados anteriormente en el apartado 3.2. La matriz de confusión relaciona las

predicciones con los valores reales, por lo que permite generar distintas métricas acerca de la capacidad para detectar varios elementos en la matriz:

		Referencia	
		Sí	No
Predicción	Sí	TP	FP
	No	FN	TN

• Verdaderos positivos (TP): personas predichas como accidentadas graves que realmente lo fueron.
• Verdaderos negativos (TN): personas predichas como accidentados leves que realmente lo fueron.
• Falsos positivos (FP): personas predichas como accidentadas graves que realmente fueron leves.
• Falsos negativos (FN): personas predichas como accidentadas leves, que realmente fueron graves.

Las métricas que se pueden obtener a partir de la matriz de confusión son las siguientes:

- Exactitud (Accuracy). Indica en general el rendimiento del modelo, pues mide el porcentaje de aciertos.

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

- Sensibilidad (Sensitivity). Esta métrica mide, la capacidad del modelo de clasificar la clase positiva, en este caso, sufrir un accidente grave. Sería la proporción de accidentados graves que realmente han sido clasificados así.

$$Sensitivity = \frac{TP}{TP + FN}$$

- Especificidad (Specificity). Similar al anterior, permite medir la capacidad del modelo de clasificar la clase negativa, es decir, de no sufrir un accidente grave.

$$Specificity = \frac{TN}{TP + FP}$$

- Precisión (Possitive Predictive Value). Nos indica la capacidad del modelo para predecir la clase positiva. Muestra la proporción de los predichos como accidentados graves que realmente tuvieron un accidente grave.

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

- F1 Score. Permite relacionar la precisión y la sensibilidad del modelo. Por tanto, esta métrica nos indica qué tan bien predice y clasifica a la clase positiva.

$$F1 = 2 \cdot \frac{PPV \cdot Sensitivity}{PPV + Sensitivity}$$

Por último, a la hora de decidir qué modelo es el mejor se empleará el AUC, ya que definitivamente es una forma muy útil para decidir el modelo que mejor relación presenta en términos de sensibilidad y especificidad.

4. Exploración y procesamiento de datos

4.1. Conjunto de datos

Los datos provienen del Portal Estadístico de la Dirección General de Tráfico (DGT) (https://sedeapl.dgt.gob.es/WEB_IEST_CONSULTA/) mediante el cual se puede acceder a los archivos de microdatos de todos los accidentes registrados en España. En esta web se pueden obtener los datos por año, siendo 2015 el año con los registros más recientes.

En esta misma página web se puede descargar el "Diseño de registro año 2011 y anualidades posteriores" que contiene información relativa a los microdatos y ayuda, en parte, a su comprensión. Esta comprensión parcial se debe a que, en algunos campos, las categorías de las variables no coinciden con las descritas en el diccionario de variables.

Para poder compensar esta falta de información se debe acudir a la "Orden INT/2223/2014, de 27 de octubre, por la que se regula la comunicación de la información al Registro Nacional de Víctimas de Accidentes de Tráfico" disponible en el Boletín Oficial del Estado y observar en su Anexo I el formulario de accidentes con víctimas. También es necesario completar el significado de las variables empleando el "Anuario estadístico de accidentes 2015" de la DGT (DGT, 2015).

Con la descarga del archivo de microdatos se obtienen los tres archivos separados mostrados en la ilustración 5, que contienen diferente información, pero se relacionan entre sí.

Tabla	Registros	Campos
TABLA_ACCVICT_2015	97.756	39
TABLA_VEHIC_2015	170.749	14
TABLA_PERS_2015	238.476	31

Ilustración 5. Información de los ficheros de microdatos

4.2. Tratamiento preliminar

Para este trabajo es clave entender la información que proporcionan todas las variables presentes en los tres archivos de microdatos y seleccionar aquellas que serán relevantes para este caso. Por ello, el primer paso es entender cómo se relacionan las distintas tablas:

- La “TABLA_ACCVICT_2015” contiene información relativa a las circunstancias que envuelven un accidente de tráfico. Contiene la variable identificativa *ID_Accidente*, que conecta con las otras dos tablas. Esta variable ID contiene un único registro para cada accidente.
- La “TABLA_VEHIC_2015” contiene información relativa a las características de los vehículos implicados en un accidente de tráfico. Contiene la variable identificativa *ID_Accidente*, que conecta con las otras dos tablas. En esta tabla, esta variable ID puede hallarse repetida en varias ocasiones, ya que pueden encontrarse varios vehículos implicados en un mismo accidente. Para diferenciar los vehículos implicados se emplea la variable identificativa *ID_Vehiculo*.
- La “TABLA_PERS_2015” contiene información relativa a las características de las personas involucradas en un accidente de tráfico. Contiene la variable identificativa *ID_Accidente*, que conecta con las otras dos tablas. En esta tabla, esta variable ID puede hallarse repetida en varias ocasiones, ya que en un accidente pueden estar involucradas varias personas. Para diferenciarlos se emplea la variable identificativa *ID_Vehiculo*, permitiendo saber en qué vehículo implicado se encontraba, *ID_Persona*, para entender qué persona es dentro de un mismo vehículo, *ID_Conductor*, para entender si era el conductor del vehículo, *ID_Pasajero* para saber si era pasajero y para enumerar los posibles pasajeros involucrados en el accidente.

Tras entender la relación entre tablas se orientan todos los datos hacia el estudio de la siniestralidad en motocicletas, por lo que se filtró la información en función a este tipo de accidentes. Primero se filtró la tabla de vehículos, ya que si se filtra la variable Tipo_Vehiculo, donde sólo aparezca el valor 4 (motocicleta), se obtienen las 25291 motocicletas involucradas en un accidente de tráfico durante el 2015. Este valor corresponde a las motos accidentadas, pero en una moto pueden ir pasajeros y varias motos pueden estar involucradas en un mismo accidente.

Para abordar estos problemas, primero fue necesario convertir los registros de los acompañantes en nuevas variables empleando el ID_Accidente y el ID_Vehiculo para que los datos se crucen correctamente. Con esto se eliminan duplicidades en la tabla de personas al separar la información de conductor y pasajero. Para solucionar el problema de los accidentes con varias motocicletas involucradas se decidió eliminar estos registros, ya que la alternativa sería considerar a esas observaciones como registros separados cuando realmente son un mismo

accidente y tocaría duplicar la misma información relativa a las circunstancias del accidente, lo que quizá podría repercutir en la calidad de los modelos estudiados.

El siguiente paso es cruzar la información de las tres tablas utilizando la función BUSCARV de Excel empleando variable identificativa ID_Accidente. Se crea una nueva tabla que parte de la tabla de personas, para incorporar la información de los vehículos que conducían y, por último, la información que rodeaba al accidente. Para que este cruce de datos sea adecuado conviene emplear además otras variables identificativas para cerciorarse de que los datos correspondan a cada registro. Como resultado se obtiene una tabla con 22604 registros.

Por otra parte, para hacer más sencillo el trabajo computacional se decidió convertir 5 variables categóricas relativas a las anomalías en una sola que contenga a las 5. Se pasó de 5 variables con tres categorías cada una, a una variable con 7 niveles como muestra la ilustración 6.

Variables anteriores	Niveles	Nueva variable	Niveles
Anomalia_Ninguna	1. Sin anomalía 2. Con anomalía 3. Se desconoce	Anomalía	1. Ninguna 2. Neumático 3. Reventón 4. Dirección 5. Frenos 6. Varias/más de una 7. No consta
Anomalia_Neumatico	1. Neumáticos desgastados 2. Sin anomalías 3. Se desconoce		
Anomalia_Reventon	1. Reventón 2. Sin anomalías anteriores 3. Se desconoce		
Anomalia_Direccion	1. Anomalías en la dirección 2. Sin anomalías anteriores 3. Se desconoce		
Anomalia_Frenos	1. Anomalías en los frenos 2. Sin anomalías 3. Se desconoce		

Ilustración 6. Creación variable anomalía

Esta fase preliminar finalizó con el diseño de la variable objetivo “Accdo_Grave”, siendo una variable binaria que toma valor 1 cuando el accidente en moto se saldó con un muerto o accidentado grave (requiere hospitalización) y 0 en caso contrario. Para obtener esta variable se partió de las variables originales “Muerto_24h”, “Muerto_30d”, “Herido_Grave_24h”, “Herido_Grave30d” en la tabla de personas, asignando valor 1 en caso de que cualquiera de los

ocupantes de la motocicleta accidentada haya muerto o resultado herido grave en el momento del accidente o dentro de los 30 primeros días tras el accidente; y 0 en caso contrario.

Resultado de esta selección de variables y registros se obtienen las 45 variables de la ilustración 7 que pasarán a un proceso de depuración de datos más exhaustivo.

Variable	Tipo	Valores
ID_Accidente	ID	
Accdo_Grave	Binaria	0 Accidentado tuvo heridas leves 1 Accidentado resultó muerto o con heridas graves
Anio_Matricula_Veh	Intervalo	9999 Se Desconoce/Sin Especificar
Anio_Permiso	Intervalo	9999 Se Desconoce/Sin Especificar
Edad	Intervalo	999 Se Desconoce/Sin Especificar
Edad_P (pasajero)	Intervalo	999 Se Desconoce/Sin Especificar
Hora	Intervalo	n
Num_Ocupantes_Veh	Intervalo	999 Se Desconoce/Sin Especificar
Tot_Veh_Implicados	Intervalo	n
Anomalía	Clase	1 Sin anomalías 2 Neumáticos 3 Reventón 4 Dirección 5 Frenos 6 Varias/más de una 7 No consta
Carretera	Clase	Nombre de la carretera
Casco_P (pasajero)	Clase	1 Sí 2 No 3 Se desconoce o salió proyectado
Com_Aut	Clase	1 Andalucía 2 Aragón 3 Asturias 4 Illes Balears 5 Canarias 6 Cantabria 7 Castilla Y León 8 Castilla-La Mancha 9 Cataluña 10 Com. Valenciana 11 Extremadura 12 Galicia 13 Com. de Madrid 14 Reg. de Murcia 15 Com. F. Navarra 16 La Rioja 17 País Vasco 18 Ceuta Y Melilla
Dia_Sem	Clase	1 Lunes … 7 Domingo
Factores_Atmos	Clase	1 Buen Tiempo 2 Niebla Intensa 3 Niebla Ligera 4 Lloviznando 5 Lluvia Fuerte 6 Granizando 7 Nevando 8 Viento Fuerte 9 Otro

Infracc_Alumbrado	Clase	1 Incorrecta utilización del alumbrado 2 Ninguna infracción 3 Se desconoce		
Infracc_Carga_Veh	Clase	1 Exceso, mal acondicionamiento o desprendimiento de la carga 2 Ninguna infracción 3 Se desconoce		
Infracc_Cond	Clase	1 No respetar señal de STOP 2 No respetar paso para peatones 3 No respetar otra regulación de prioridad 4 Circular en sentido contrario o por lugar prohibido 5 Invadir parcialmente el sentido contrario 6 Adelantar antirreglamentariamente 7 No mantener el intervalo de seguridad 8 Otra infracción 9 Ninguna infracción 10 Se desconoce		
Infracc_Resumen	Clase	1 Ninguna infracción 2 Alguna infracción 3 Se desconoce		
Infracc_Velocidad	Clase	1 Infracción de velocidad 2 Marcha lenta 3 Ninguna 4 Se desconoce		
Luminosidad	Clase	1 Luz del día 2 Amanecer o atardecer 3 Sin luz y con iluminación artificial encendida 4 Sin luz		
Mes	Clase	1 Enero … 12 Diciembre		
Mes_Matricula_Veh	Clase	1 Enero … 12 Diciembre 99 Sin dato		
Municipio	Clase	Sin dato aaaaa Si el municipio tiene 5000 habitantes o más: código de municipio normalizado por el INE.		
Pasajero	Clase	0 No viaja pasajero 1 Sí viaja pasajero		
Prioridad_Ceda	Clase	0 No 1 Sí		
Prioridad_Marcas	Clase			
Prioridad_Otra	Clase			
Prioridad_Paso	Clase			
Prioridad_Semaforo	Clase			
Prioridad_Stop	Clase			
Provincia	Clase	1 Álava 2 Albacete 3 Alicante 4 Almería 5 Ávila 6 Badajoz	19 Guadalajara 20 Gipuzkoa 21 Huelva 22 Huesca 23 Jaén 24 León	36 Pontevedra 37 Salamanca 38 S.C.Tenerife 39 Cantabria 40 Segovia 41 Sevilla

		7 Islas Baleares 8 Barcelona 9 Burgos 10 Cáceres 11 Cádiz 12 Castellón 13 Ciudad Real 14 Córdoba 15 A Coruña 16 Cuenca 17 Girona 18 Granada	25 Lleida 26 La Rioja 27 Lugo 28 Madrid 29 Málaga 30 Murcia 31 Navarra 32 Ourense 33 Asturias 34 Palencia 35 Las Palmas	42 Soria 43 Tarragona 44 Teruel 45 Toledo 46 Valencia 47 Valladolid 48 Bizkaia 49 Zamora 50 Zaragoza 51 Ceuta 52 Melilla
Red_Carretera	Clase	1 Titularidad estatal 2 Titularidad autonómica 3 Titularidad provincial 4 Titularidad municipal 5 Otras titularidades		
Sexo	Clase	1 Hombre 2 Mujer 999 Sin Especificar		
Sexo_P	Clase	1 Hombre 2 Mujer 999 Sin Especificar		
Superficie_Calzada	Clase	1 Seca Y Limpia 2 Umbría 3 Mojada 4 Helada 5 Nevada 6 Barrillo 7 Gravilla Suelta 8 Aceite 9 Otro Tipo		
Tipo_Accidente	Clase	1 Frontal 2 Fronto-lateral 3 Lateral 4 Por alcance 5 Múltiple o en caravana 6 Colisión contra obstáculo o elemento de la vía 7 Atropello a personas 8 Atropello a animales 9 Vuelco 10 Caída 11 Sólo salida de la vía 12 Salida de la vía por la izquierda con colisión 13 Salida de la vía por la izquierda con despeñamiento 14 Salida de la vía por la izquierda con vuelco 15 Salida de la vía por la izquierda, otro tipo 16 Salida de la vía por la derecha con colisión 17 Salida de la vía por la derecha con despeñamiento 18 Salida de la vía por la derecha con vuelco 19 Salida de la vía por la derecha otro tipo 20 Otro tipo de accidente		

Tipo_Intersec	Clase		No aplica (no es intersección)
		1	En T ó Y
		2	En X ó +
		3	Giratoria
		4	Otro tipo
Tipo_Via	Clase	1	Autopista
		2	Autovía
		3	Vía para automóviles
		4	Vía convencional con carril lento
		5	Vía convencional
		6	Camino vecinal
		7	Vía de servicio
		8	Ramal de enlace
		9	Otro tipo
Trazado_No_Intersec	Clase		No aplica (es intersección)
		1	Recta
		2	Curva suave
		999	Se desconoce
Uso_Casco	Clase	1	Sí
		2	No
		3	Se desconoce o salió proyectado
Uso_Cinturon	Clase	1	Sí
		2	No
		3	Se desconoce
Visibilidad_Restringida	Clase	0	Sin Dato
		1	Edificios
		2	Configuración Del Terreno
		3	Vegetación
		4	Factores Atmosféricos
		5	Deslumbramiento
		6	Polvo O Humo
		7	Otra_Causa
		8	Sin Restricción
Zona	Clase	1	Carretera
		2	Zona Urbana
		3	Travesía
		4	Variante
Zona_Agrupada	Clase	1	Vías interurbanas
		2	Vías urbanas

Ilustración 7. Tabla de variables seleccionadas

Igualmente, resultado de la interpretación de cada variable se procedió a eliminar algunas de las variables originales que ya no iban a ser de utilidad en los siguientes pasos del estudio. Se eliminaron las variables identificativas, las utilizadas para calcular la variable objetivo antes mencionadas, aquellas variables que tomaban un único valor, variables relativas a peatones o que no correspondían con las motocicletas.

4.3. Análisis descriptivo de las variables

Antes de comenzar con la fase de depuración sería conveniente poder analizar los datos tal y como se han obtenido, ya que posteriormente podrían ser objeto de correcciones, transformaciones o agrupaciones.

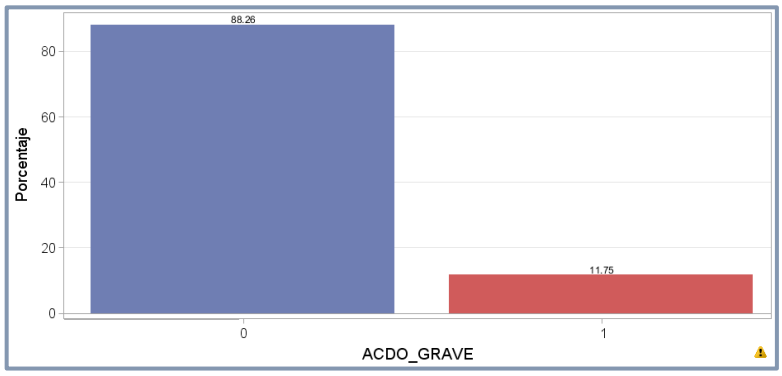


Ilustración 8. Porcentaje accidentados leves y graves

Analizando la variable objetivo, se observa que el 11.75% de los accidentes de motocicleta acabaron con al menos un accidentado grave o muerto. La ilustración 8 permite observar que existe una desproporción entre las clases de la variable objetivo, conocido también como desbalanceo. Este primer análisis lleva a considerar si se debe realizar balanceo de los datos. Sin embargo, la clase minoritaria cuenta con 2655 observaciones, un valor más que aceptable para que los algoritmos de machine learning puedan entrenar. Además, como se comprobó en el trabajo de (Vila, 2020) el uso de técnicas de balanceo en este tipo de datos no repercutía en un mejor performance de los algoritmos.

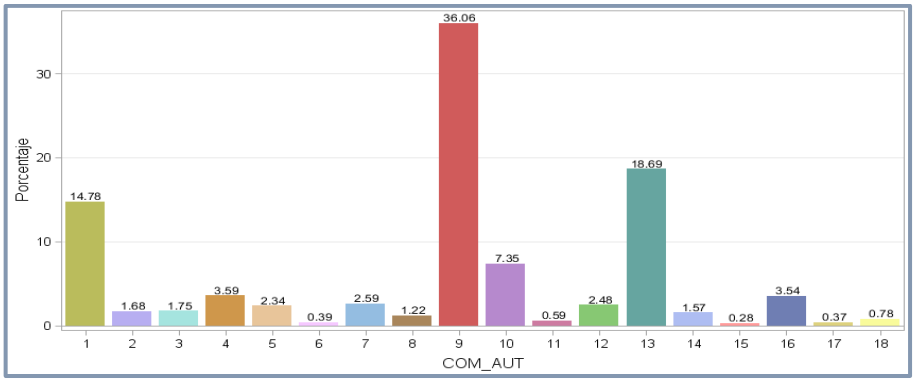


Ilustración 9. Variable Comunidad Autónoma

La variable Comunidad Autónoma de la ilustración 9 nos indica que aquellas que concentran la mayoría de accidentes son Cataluña (36.06%), Comunidad de Madrid (18.69%), Andalucía (14.78%) y Comunidad Valenciana (7.35%). De hecho, sólo las provincias de Barcelona y Madrid acumulan ya el 50.17% de los accidentes de motocicletas.

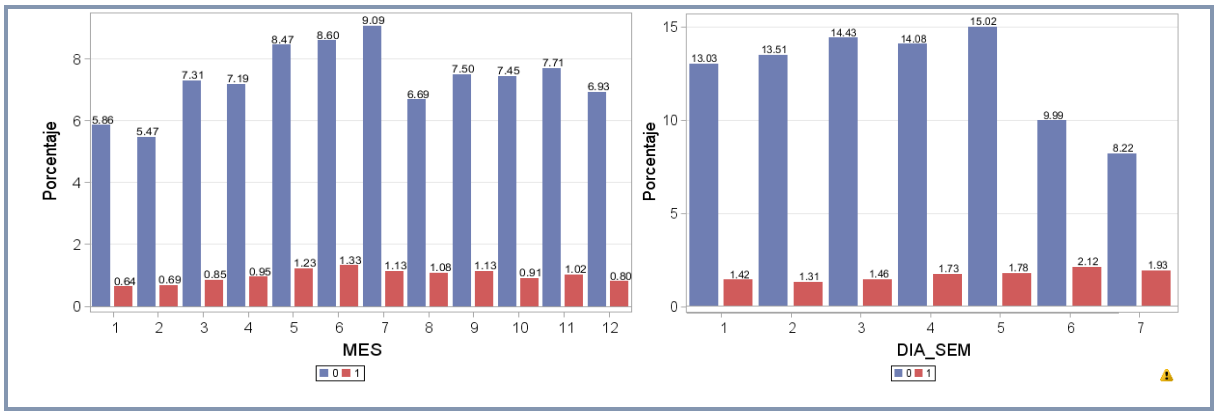


Ilustración 10. Variables Mes y Día de la semana

En la ilustración 10 se observa que los meses primaverales son los que presentan una mayor concentración de accidentes. Atendiendo a los días de la semana, se puede apreciar que de lunes a viernes existe una mayor proporción de accidentes leves, pero los accidentes graves suelen producirse en mayor cantidad los fines de semana.

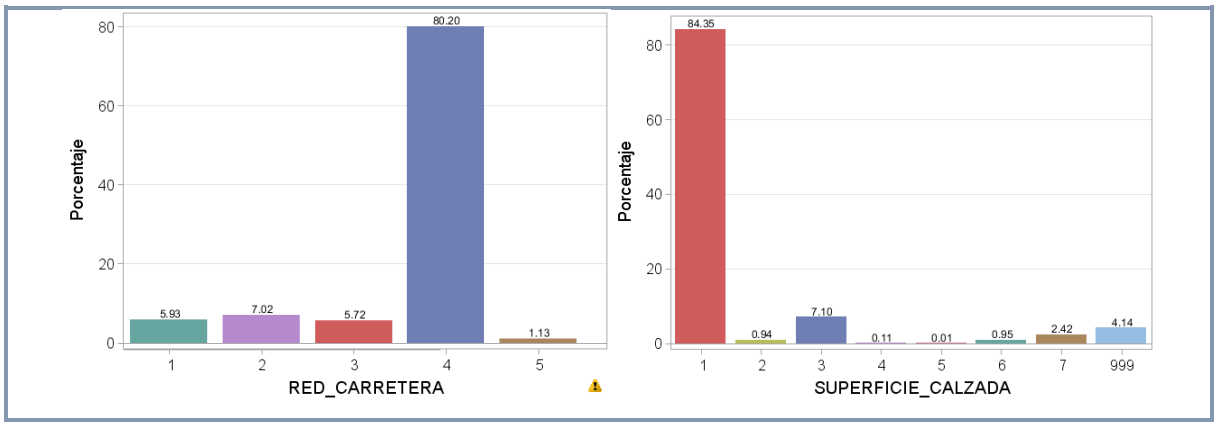


Ilustración 11. Variables Red de Carretera y Superficie de la Calzada

La ilustración 11 muestra las características del camino. Se aprecia que la gran mayoría de los accidentes se producen en vías de titularidad municipal (80.20%). Observando la superficie de la calzada, en el 84.35% de los accidentes de motocicletas la superficie estaba seca y limpia, aunque hay que destacar que las calzadas mojadas son responsables de 7.10% de este tipo de accidentes, quizá motivados por la menor adherencia de este tipo de vehículos.

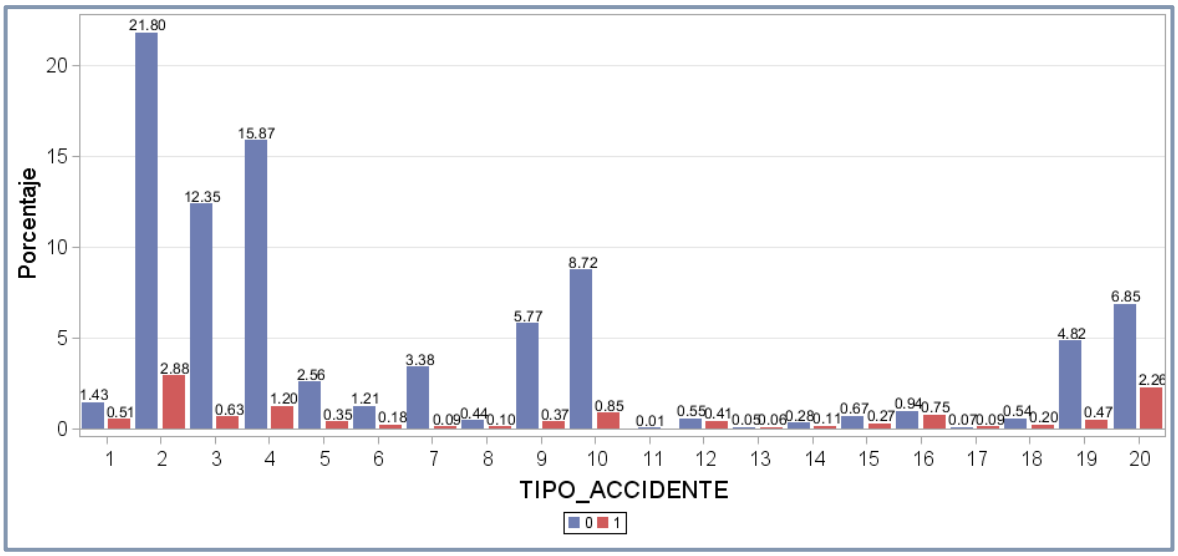


Ilustración 12. Variable Tipo de Accidente

En la ilustración 12 se puede apreciar cómo los tipos de accidentes más frecuentes son el accidente fronto-lateral, accidente por alcance y accidente lateral. Ahora bien, si se observa la proporción entre un resultado leve o grave, las categorías que más peligro entrañan para un motociclista son los accidentes con salida de vía y despeñamiento (categoría 17), salida de vía con colisión (categoría 16) o los accidentes frontales (categoría 1).

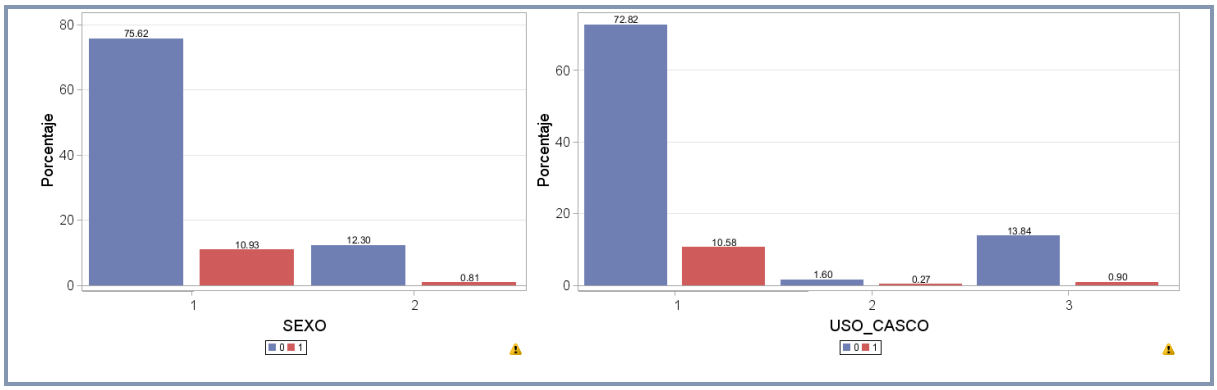


Ilustración 13. Variables Sexo y Uso del Casco

Ya en lo referente a las características de los motoristas accidentados, en la ilustración 13 se puede observar que la gran mayoría de accidentados son hombres y, además, protagonizan en mayor medida los accidentes graves (10.93%). En cuanto al uso del casco, destaca que la gran mayoría de los accidentados lo llevaban puesto, pero es algo que en un 10.58% de los accidentes no evitó que sufran un accidente grave o fatal. Sin embargo, en los accidentes en los que se constató que el accidentado no lo llevaba puesto, se aprecia una mayor probabilidad de sufrir un accidente grave.

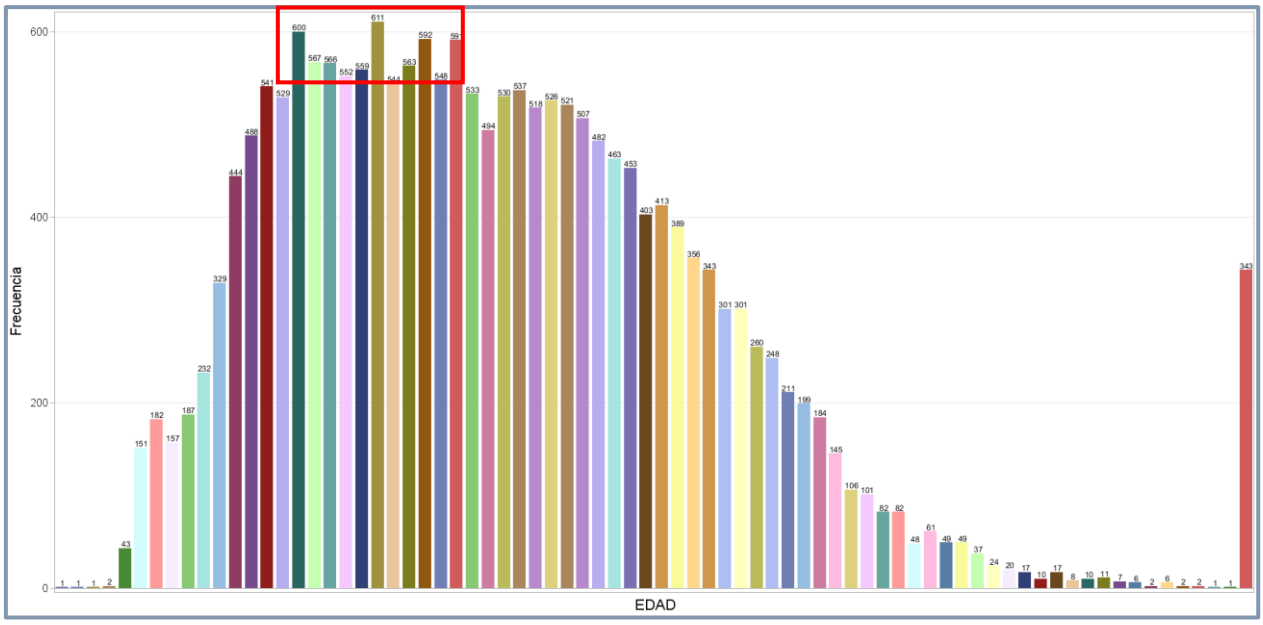


Ilustración 14. Variable Edad del Conductor

Por último, para terminar con esta breve explicación de la situación general de los accidentes de motocicleta, se puede observar la distribución de los accidentes de motociclistas con respecto a sus edades en la ilustración 14, donde destaca la gran cantidad de accidentes que ocurren entre los 27 y 37 años.

4.4. Depuración de los datos

El proceso de depuración comienza con 43 variables independientes más la variable "ACCDO_GRAVE", que es la variable objetivo de este trabajo. Para realizar la depuración se llevará a cabo un proceso secuencial que comenzará con la exploración inicial de las variables y terminará con una preselección de las variables más importantes.

4.4.1. Exploración inicial

Variables de clase

Variable	Número de niveles	Ausente	Tipo
ACDO_GRAVE	2	0	N
ANOMALIA	7	0	N
CARRETERA	1815	16746	C
CASCO_P	3	20336	N
COM_AUT	18	0	N
DIA_SEM	7	0	N
FACTORES_ATMOS	10	0	N
INFRACC_ALUMBRADO	3	0	N
INFRACC_CARGA_VEH	3	0	N
INFRACC_COND	10	0	N
INFRACC_RESUMEN	3	0	N
INFRACC_VELOCIDAD	4	0	N
LUMINOSIDAD	4	0	N
MES	12	0	N
MES_MATRICULA_VEH	13	0	N
MUNICIPIO	892	0	N
PASAJERO	1	20336	N
PRIORIDAD_CEDA	2	0	N
PRIORIDAD_MARCAS	2	0	N
PRIORIDAD_OTRA	2	0	N
PRIORIDAD_PASO	2	0	N
PRIORIDAD_SEMAFORO	2	0	N
PRIORIDAD_STOP	2	0	N
PROVINCIA	52	0	N
RED_CARRETERA	5	0	N
SEXO	3	0	N
SEXO_P	3	20336	N
SUPERFICIE_CALZADA	8	0	N
TIPO_ACCIDENTE	20	0	N
TIPO_INTERSEC	5	0	N
TIPO_VIA	9	0	N
TRAZADO_NO_INTERSEC	3	10901	N
USO_CASCO	3	0	N
USO_CINTURON	3	0	N
VISIBILIDAD_RESTRI	8	0	N
ZONA	3	0	N
ZONA_AGRUPADA	2	0	N

Ilustración 15. Exploración inicial variables de clase

Con respecto a las variables de clase (ilustración 15), lo primero que se realizó fue una revisión en busca de algún error en la codificación de sus categorías. Para ello, se efectuaron los cambios mostrados en la ilustración 16 utilizando el "nodo reemplazo".

Variable	Acciones		
Carretera	Sustituir "missing" por 0, que significa que el accidente no se produjo en una carretera.		
Casco_P	Sustituir "missing" por 0, que significa que en ese accidente no viajaba ningún pasajero.		
Factores_Atmos	Cambiar valor "999" por ausente. Se agrupa "2 - niebla intensa" con "3 - niebla ligera" en una misma categoría (2 - niebla); "4- lloviznando" con "5- lluvia fuerte" en una nueva categoría (3 -lluvia); "6-granizando", "7-nevando" "8-viento" y "9-otro" se incluyen una misma categoría (4-otro)	Nuevo nivel	Significado
		1	Buen tiempo
		2	Niebla
		3	Lluvia
		4	Otro
Infrac_Alumbrado	Cambiar valor "3-desconocido" por "missing".		
Infrac_Carga_Veh	Cambiar valor "3-desconocido" por "missing".		
Infrac_Cond	Cambiar valor "10-desconocido" por "missing".		
Infrac_Resumen	Cambiar valor "3-desconocido" por "missing".		
Infrac_Velocidad	Cambiar valor "4-desconocido" por "missing". Agrupar "1-infracción" y "2-marcha lenta" en una sola categoría. Se recodifica el resto	Nuevo nivel	Significado
		1	Infracción de velocidad
		2	No

		3	Se desconoce (missing)
Mes_Matricula_Veh	Cambiar valor “99” por “0” y asignarle el significado de no matriculado.		
Pasajero	Sustituir “missing” por 0, que significa que no había pasajero.		
Sexo	Cambiar valor “999” por ausente.		
Sexo_P	Sustituir “missing” por 0, que son los casos en los que no hay pasajero. Cambiar valor “999” por ausente.		
Superficie_Calzada	Cambiar valor “999” por ausente.		
Trazado_No_ Interseccion	Se cambia “missing” por 0, ya que significa que sí se trata de una intersección y “999” por ausente.		
Visibilidad_Resti	Cambiar valor “999” por “0”, ya que realmente es otra categoría que significa que la visibilidad no estaba restringida.		

Ilustración 16. Recodificación de variables

Variables de Intervalo

Variable	Etiqueta	Ausente	N	Mínimo	Máximo	Media	Desviación estándar	Asimetría	Curtosis
ANIO MATRICULA VEH	ANI...	0	22604	1901	9999	5260.705	3926.344	0.378222	-1.85711
ANIO PERMISO	ANI...	0	22604	1953	9999	3439.312	3073.57	1.665804	0.77506
EDAD	ED...	0	22604	5	999	55.49128	125.1121	7.338309	52.39282
EDAD P	ED...	20336	2268	2	999	67.53924	179.8529	4.95757	22.77204
HORA	HO...	0	22604	0	23	14.22863	4.992293	-0.38469	-0.15714
NUM OCUPANTES VEH	NU...	0	22604	1	999	20.57388	138.0205	6.948511	46.28616
TOT VEH IMPLICADOS	TO...	0	22604	1	10	1.757123	0.590094	1.223223	10.22725

Ilustración 17. Exploración inicial variables de intervalo

Respecto a las variables de intervalo mostradas en la ilustración 17, lo primero que destaca es que existen unos máximos que son indicativos que esos valores no son correctos. En estas variables los valores “9999” o “999” nos indican que no se dispone de ese dato. Para corregirlo se emplea el “nodo reemplazo”, estableciendo manualmente unos límites que permitan pasar esos valores máximos a datos ausentes.

Estadísticos descriptivos de la variable de intervalo

Variable	Etiqueta	Ausente	N	Mínimo	Máximo	Media	Desviación estándar	Asimetría	Curtosis
HORA	HORA	0	22604	0	23	14.23	4.9923	-0.38469	-0.1571
REP_ANIO_MATRICULA_VEH	Replacement: ANIO_MATRICULA_VEH	9202	13402	1901	2015	2007.33	6.4040	-2.52215	18.1481
REP_ANIO_PERMISO	Replacement: ANIO_PERMISO	4069	18535	1953	2015	1999.26	12.2064	-0.70950	-0.2995
REP_EDAD	Replacement: EDAD	387	22217	5	95	39.06	12.1764	0.40853	-0.3470
REP_EDAD_P	Replacement: EDAD_P	20417	2187	2	79	33.04	14.3598	0.39194	-0.7046
REP_NUM_OCUPANTES_VEH	Replacement: NUM_OCUPANTES_VEH	441	22163	1	4	1.11	0.3073	2.60293	4.9770
TOT_VEH_IMPLICADOS	TOT_VEH_IMPLICADOS	0	22604	1	10	1.76	0.5901	1.22322	10.2273

Ilustración 18. Corrección de variables de intervalo

Tras ejecutar este nodo, como se observa en la ilustración 18, las variables “Rep_Anio_Matricula_Veh”, “Rep_Anio_Permiso” Y “Rep_Edad_P” presentan un gran número de valores ausentes. Sin embargo, estos valores sí pueden tener un significado (vehículo no matriculado, conductor sin permiso y que no había pasajero, respectivamente) por lo que se decide discretizar estas variables, generando tramos y transformando ese gran número datos

ausentes en una categoría más, utilizando el "nodo transformar variables" mediante el método de "agrupamiento óptimo".

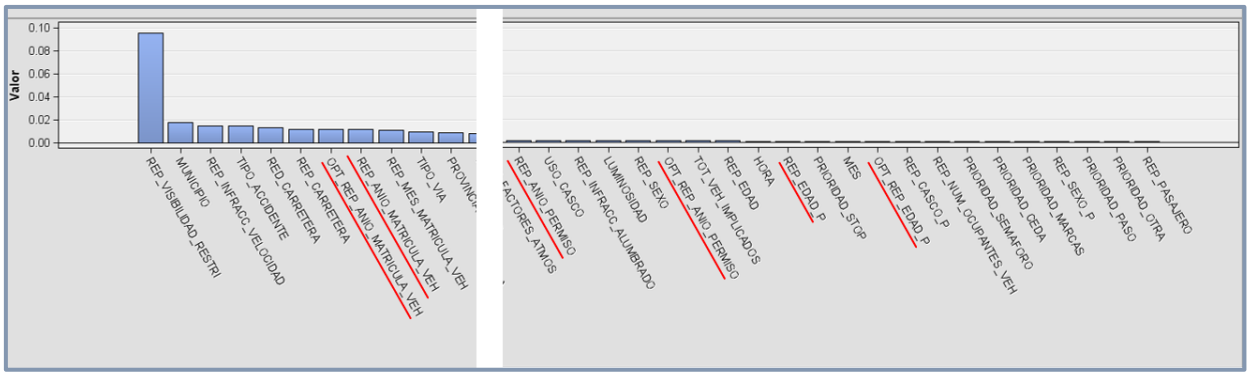


Ilustración 19. Discretización de las variables de intervalo

Como se puede apreciar en la ilustración 19, si se atiende a la relación con la variable objetivo, la transformación de la variable "Anio_Matricula_Veh" ayuda a explicar más de la variable objetivo. En el caso de las otras dos variables se pierde un poco de información en comparación con las originales, pero no en gran medida.

Tras haber corregido el problema anterior en las variables de intervalo, se buscan valores atípicos empleando la desviación típica para las variables con distribuciones simétricas y MAD (median absolute deviation) para variables con distribuciones asimétricas y mediana distinta de cero. Estos datos serán sustituidos por valores ausentes para su posterior imputación tal y como muestra la ilustración 20.

Variable	Rol	Etiqueta	Entrenamiento
HORA	INPUT	HORA	0
REP_EDAD	INPUT	Replacement: EDAD	62
REP_NUM_OCUPANTES_VEH	INPUT	Replacement: NUM_OCUPANTE...	0
TOT_VEH_IMPLICADOS	INPUT	TOT_VEH_IMPLICADOS	170

Ilustración 20. Detección datos atípicos

4.4.2. Tratamiento de datos ausentes

Variables de clase

Variable	Tipo	Número de niveles	Ausente
ACDO_GRAVE	N	2	0
ANOMALIA	N	7	0
COM_AUT	N	18	0
DIA_SEM	N	7	0
LUMINOSIDAD	N	4	0
MES	N	12	0
MUNICIPIO	N	892	0
OPT_REP_ANIO_MATRICULA_VEH	C	6	0
OPT_REP_ANIO_PERMISO	C	3	0
OPT_REP_EDAD_P	C	3	0
PRIORIDAD_CEDA	N	2	0
PRIORIDAD_MARCAS	N	2	0
PRIORIDAD_OTRA	N	2	0
PRIORIDAD_PASO	N	2	0
PRIORIDAD_SEMAFORO	N	2	0
PRIORIDAD_STOP	N	2	0
PROVINCIA	N	52	0
RED_CARRETERA	N	5	0
REP_CARRETERA	C	1816	0
REP_CASCO_P	N	4	0
REP_FACTORES_ATMOS	N	4	1164
REP_INFRACC_ALUMBRADO	N	3	0
REP_INFRACC_CARGA_VEH	N	3	0
REP_INFRACC_COND	N	10	0
REP_INFRACC_RESUMEN	N	3	0
REP_INFRACC_VELOCIDAD	N	3	0
REP_MES_MATRICULA_VEH	N	13	0
REP_PASAJERO	N	2	0
REP_SEXO	N	2	81
REP_SEXO_P	N	3	3
REP_SUPERFICIE_CALZADA	N	7	936
REP_TRAZADO_NO_INTERSEC	N	3	2922
REP_VISIBILIDAD_RESTRI	N	8	0
TIPO_ACCIDENTE	N	20	0
TIPO_INTERSEC	N	5	0
TIPO_VIA	N	9	0
USO_CASCO	N	3	0
USO_CINTURON	N	3	0
ZONA	N	3	0
ZONA_AGRUPADA	N	2	0

Ilustración 21. Tratamiento datos ausentes en variables de clase

En la ilustración 21 se pueden observar los distintos datos ausentes. En esta ocasión cada variable debe ser analizada por separado:

• En la variable “Factores_Atmos”, como los ausentes representan poco más del 5% y ya existe la categoría “otro” se procederá a imputar estos datos.

• En las variables “Sexo” y “Sexo_P” los ausentes serán imputados al ser muy pocos datos.

• En la variable “Superficie_Calzada”, al representar un 4% de las observaciones se decide imputarlos.

• En la variable “Trazado_No_Intersec”, al superar el 11% de las observaciones, la variable se recategorizará, asignando a esta categoría el valor 3 (otro).

Variables de intervalo

Variable	Ausente	N	Mínimo	Máximo	Media	Desviación estándar	Asimetría	Curtosis
REP_HORA	0	22604	0	23	14.2286	4.9923	-0.38469	-0.15714
REP_REP_EDAD	449	22155	5	75	38.9437	12.0047	0.34836	-0.53859
REP_REP_NUM_OCUPANTES_VEH	441	22163	1	4	1.1051	0.3073	2.60293	4.97698
REP_TOT_VEH_IMPLICADOS	170	22434	1	3	1.7351	0.5254	-0.18579	-0.42085

Ilustración 22. Tratamiento datos ausentes en variables de intervalo

Para realizar la imputación de los valores ausentes se emplea el "nodo imputar" empleando la moda para las variables de clase y distribución en las variables de intervalo. Los resultados se pueden observar en la ilustración 22.

4.4.3. Reducción de niveles

Como un paso previo a la selección de variables, se procedió a agrupar algunas categorías, ya que había muchas variables que tenían demasiados niveles. Este paso se consideró de gran importancia para este trabajo, ya que como se puede observar la mayoría de las variables son nominales y, para agilizar el trabajo computacional, resulta muy conveniente tratar de reducir los niveles de las variables. Para ello se empleó el "nodo selección de variables", que reduce los niveles de las variables basándose en la relación con la variable objetivo.

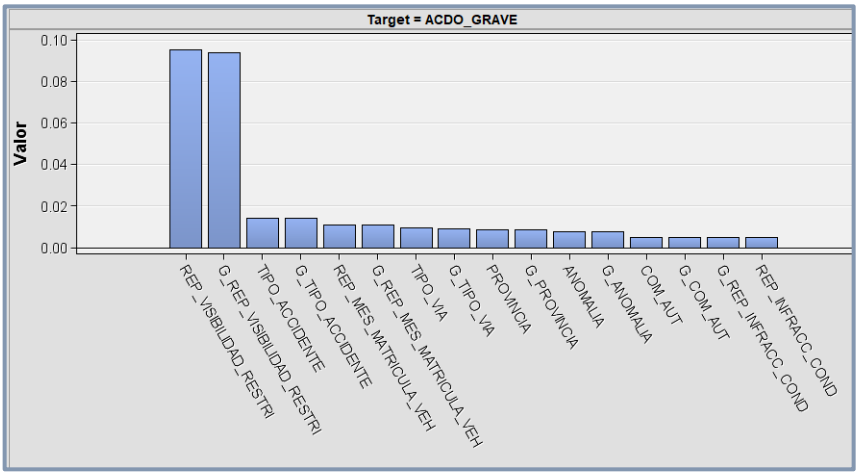


Ilustración 23. Agrupación de niveles

Para ver que la pérdida de información con respecto a las variables originales no suponía un problema se comparó la información que aportaban las variables con clases agrupadas y las variables originales y, como se observa en la ilustración 23, esta pérdida fue mínima y en el caso de la variable "Infrac_Cond" se ganó en información.

También se trató de agrupar los niveles de las variables "Carretera" (1815 niveles) y "Municipio" (892 niveles). Sin embargo, estas agrupaciones no se llevaron a cabo por parte de EM por no aportar la suficiente información a la variable objetivo. El nodo también rechazaba

a las variables originales precisamente por su gran cantidad de niveles que de manera predeterminada los limita a 100 niveles por variable.

Los niveles de las variables "Mes" y "Superficie_Calzada" también fueron agrupados, pero como muestra la ilustración 24 tanto las variables originales como las agrupadas tenían un R2 demasiado pequeño, por lo que el nodo las descartaba. Se decidió no revertir esta acción al ser variables con 12 y 7 niveles, respectivamente, que tendrían un peso más significativo en el trabajo computacional que en el entrenamiento de los modelos.

Class: IMP_REP_SUPERFICIE_CALZADA	6	0.004266	R2 < MINR2
Group: IMP_REP_SUPERFICIE_CALZADA	3	0.004238	R2 < MINR2
Class: MES	11	0.001375	R2 < MINR2
Group: MES	6	0.001356	R2 < MINR2

Ilustración 24. R2 de variables rechazadas

En la ilustración 25 se resume la reducción de niveles conseguida mediante la agrupación mencionada anteriormente.

Variable	N.º de niveles nuevos	N.º de niveles originales
Visibilidad_Restri	3	8
Tipo_Accidente	5	20
Mes_Matricula_Veh	3	13
Tipo_Via	3	9
Provincia	9	52
Infrac_Cond	5	10
Com_Aut	6	18
TOTAL	34	130

Ilustración 25. Reducción del número de niveles

4.4.4. Preselección de variables

Para facilitar la selección de las variables que pasarán a la etapa de modelización, se crea una variable aleatoria utilizando el "nodo transformar variables". Con esta variable se puede hacer una primera idea sobre la relación que tienen las variables input con la variable objetivo. Además, mantener esta variable aleatoria también será de utilidad durante el entrenamiento de los distintos modelos de machine learning al observar la importancia de las variables para cada algoritmo.

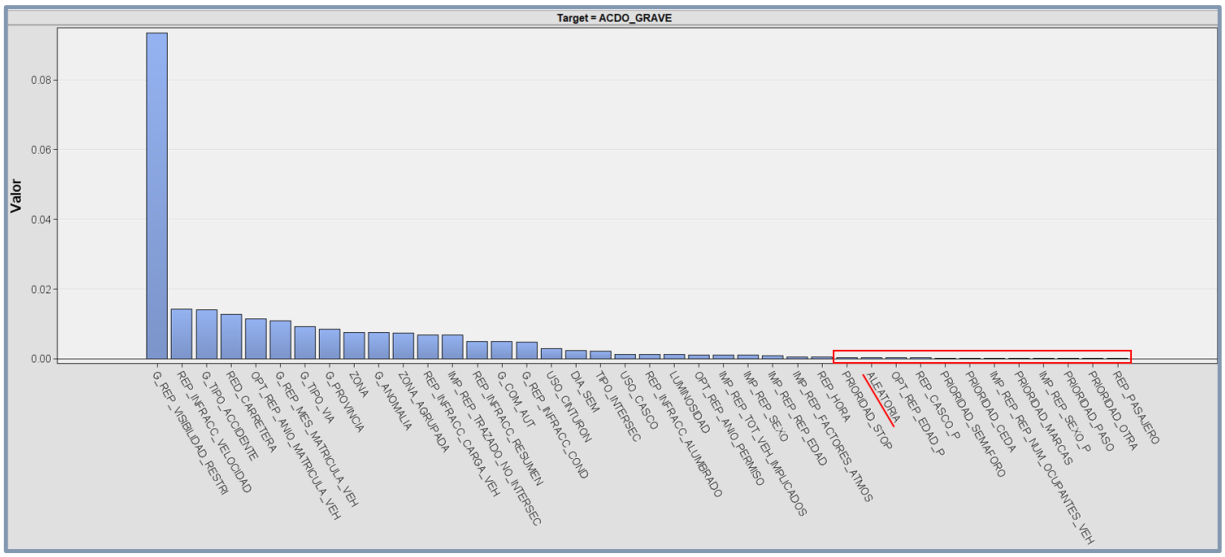


Ilustración 26. Selección de variables

Para llevar a cabo esta preselección, se emplea el "nodo explorador de estadísticos" observando el valor que tiene cada una de las variables como muestra la ilustración 26. Lo primero a destacar es que existe una variable que parece estar muy relacionada con la gravedad de los accidentes y esa es "Visilidad_Restri", algo que tiene sentido, ya que la incorporación a la circulación de este tipo de vehículos es uno de los momentos más críticos. También destacan las variables que indican si existía infracción de velocidad por parte del motociclista y el tipo de accidente que sufre. Por otro lado, se pueden apreciar aquellas variables que se relacionan muy poco con la variable objetivo. Las variables que se descartarán manualmente, por explicar lo mismo o menos que la variable aleatoria, son "Prioridad_Stop", "Edad_P", "Casco_P", "Prioridad_Semaforo", "Prioridad_Ceda", "Num_Ocupantes_Veh", "Prioridad_Marcas", "Sexo_P", "Prioridad_Paso", "Prioridad_Otra" y "Pasajero".

La preselección de variables se completa utilizando el "nodo metadatos" que permite rechazar aquellas variables que no son de interés tras la depuración. Antes de guardar los datos depurados se cambia el nombre de las variables tratadas con prefijos asignados por EM al ser transformadas o imputadas. Para ello se usa el "nodo código SAS".

Finalizado el proceso de depuración, son 28 las variables que se utilizarán para realizar los modelos de machine learning.

5. Modelización

Para modelizar los distintos algoritmos primero se obtendrá un set de variables mediante un proceso de selección. Los distintos modelos se irán afinando mediante un proceso de tuneado

de los hiper-parámetros que correspondan y, finalmente, para cada algoritmo se compararán los posibles modelos mediante validación cruzada repetida de cuatro grupos y cinco repeticiones.

5.1. Selección de variables

Para seleccionar las variables que formarán parte de los modelos se han realizado varias regresiones logísticas en SAS. Concretamente se han utilizado los métodos de selección stepwise, forward y backward mediante el uso de una macro que permita repetir el proceso con varias semillas y así obtener dos conjuntos de variables por cada método de selección como muestra la ilustración 27.

Método	Modelo	Variables
STEP1	1	Anomalia_ Com_Aut_ Dia_Sem Factores_Atmos_ Infracc_Alumbrado_ Infracc_Carga_Veh_ Infracc_Cond_ Infracc_Velocidad_ Mes_Matricula_Veh_ Provincia_ Red_Carretera Tipo_Accidente_ Tipo_Intersec Tot_Veh_Implicados_ Uso_Casco Uso_Cinturon Visibilidad_Restri_ Zona_Agrupada
STEP2	2	Anomalia_ Com_Aut_ Dia_Sem Factores_Atmos_ Infracc_Alumbrado_ Infracc_Carga_Veh_ Infracc_Cond_ Infracc_Resumen_ Infracc_Velocidad_ Mes_Matricula_Veh_ Provincia_ Red_Carretera Tipo_Accidente_ Tipo_Intersec Tipo_Via_ Tot_Veh_Implicados_ Uso_Casco Visibilidad_Restri_
FOR1	3	Anomalia_ Com_Aut_ Dia_Sem Factores_Atmos_ Infracc_Alumbrado_ Infracc_Carga_Veh_ Infracc_Cond_ Infracc_Velocidad_ Mes_Matricula_Veh_ Provincia_ Red_Carretera Tipo_Accidente_ Tipo_Intersec Tipo_Via_ Tot_Veh_Implicados_ Uso_Casco Uso_Cinturon Visibilidad_Restri_
FOR2		Mismo modelo que FOR1
BACK1	4	Anio_Permiso_ Anomalia_ Com_Aut_ Dia_Sem Factores_Atmos_ Infracc_Alumbrado_ Infracc_Carga_Veh_ Infracc_Cond_ Infracc_Velocidad_ Mes_Matricula_Veh_ Provincia_ Red_Carretera Tipo_Accidente_ Tipo_Intersec Tipo_Via_ Tot_Veh_Implicados_ Uso_Casco Uso_Cinturon Visibilidad_Restri_
BACK2	5	Anio_Permiso_ Anomalia_ Com_Aut_ Dia_Sem Infracc_Alumbrado_ Infracc_Carga_Veh_ Infracc_Cond_ Infracc_Velocidad_ Mes_Matricula_Veh_ Provincia_ Red_Carretera Tipo_Accidente_ Tipo_Intersec Tipo_Via_ Tot_Veh_Implicados_ Uso_Casco Uso_Cinturon Visibilidad_Restri_

Ilustración 27. Posibles conjuntos de variables

En un principio se debieron obtener 6 modelos, pero dos métodos de selección (STEPT2 y FOR2) llegaron a seleccionar el mismo conjunto de variables por lo que se procede a comparar 5 conjuntos mediante validación cruzada repetida.

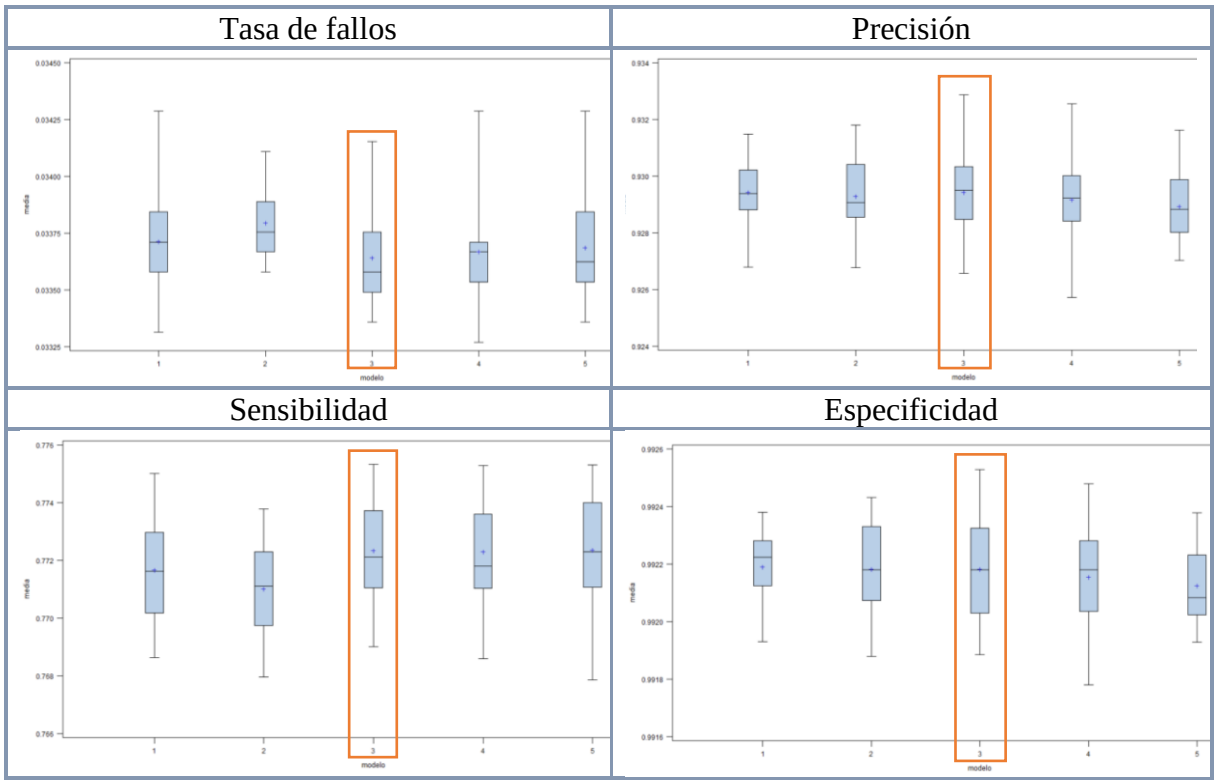


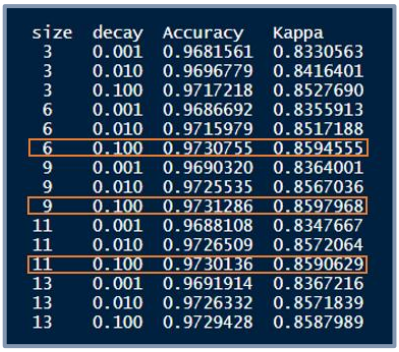
Ilustración 28. Métricas para la selección de variables

Teniendo en cuenta distintas métricas mostradas en la ilustración 28 se va a elegir el modelo 3 para tomar de él las variables y comenzar a modelizar en el resto de algoritmos de machine learning.

5.2. Redes Neuronales

En R se comienza realizando la normalización de las variables continuas y se transforman en dummies a las variables categóricas para poder realizar un correcto tuneo de hiper-parámetros. Además, es necesario realizar algunos cambios en los valores de la variable objetivo "Acdo_Grave" ya que se necesita pasar de "1" y "0" a valores "yes" y "no", respectivamente. Para el tuneo de los hiper-parámetros se aplicará validación cruzada repetida.

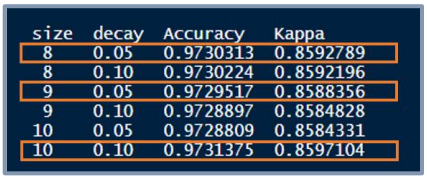
Antes de fijar la parrilla se estima el número de nodos máximo que se va a fijar. Para ello se debe considerar que el número de parámetros en una red es igual a $h(k+1)+h+1$, donde h es el número de nodos ocultos y k el número de variables. Teniendo en cuenta que es adecuado tener al menos 30 observaciones por parámetro se puede obtener una idea del máximo número de nodos a probar durante el tuneo de la red, el cual se fija en 13 nodos ocultos.



size	decay	Accuracy	Kappa
3	0.001	0.9681561	0.8330563
3	0.010	0.9696779	0.8416401
3	0.100	0.9717218	0.8527690
6	0.001	0.9686692	0.8355913
6	0.010	0.9715979	0.8517188
6	0.100	0.9730755	0.8594555
9	0.001	0.9690320	0.8364001
9	0.010	0.9725535	0.8567036
9	0.100	0.9731286	0.8597968
11	0.001	0.9688108	0.8347667
11	0.010	0.9726509	0.8572064
11	0.100	0.9730136	0.8590629
13	0.001	0.9691914	0.8367216
13	0.010	0.9726332	0.8571839
13	0.100	0.9729428	0.8587989

Ilustración 29. Resultados primer grid Avnnet

Primero se establece un grid o parrilla con varios números de nodos (3, 6, 9, 11 y 13) y con un *decay* o *learning rate* que contempla (0.001, 0.01, 0.1). Como se observa en la ilustración 29, se obtiene que el número de nodos puede estar cercano a los 9. Además, se encuentra un patrón claro respecto al *decay* ya que parece que valores próximos a 0.1 obtiene mejores resultados que para valores bajos. Por ello, se decide construir un nuevo grid con número de nodos (8, 9, 10) e incluir un nuevo valor para el *decay* (0.05 y 0.1).



size	decay	Accuracy	Kappa
8	0.05	0.9730313	0.8592789
8	0.10	0.9730224	0.8592196
9	0.05	0.9729517	0.8588356
9	0.10	0.9728897	0.8584828
10	0.05	0.9728809	0.8584331
10	0.10	0.9731375	0.8597104

Ilustración 30. Resultado segundo grid Avnnet

La ilustración 30 muestra algunas combinaciones de número de nodos y *decay* interesantes, por lo que se probarán varios modelos con validación cruzada repetida para poder seleccionar el mejor modelo. También se utiliza una regresión logística para poder compararlo con las redes neuronales de la ilustración 31.

Nombre	Tipo	*Nº de nodos*	*Decay*
Logística	Regresión Logística		
Avnnet1	Red Neuronal	6	0.1
Avnnet2	Red Neuronal	8	0.05
Avnnet3	Red Neuronal	9	0.05
Avnnet4	Red Neuronal	9	0.1
Avnnet5	Red Neuronal	10	0.1

Ilustración 31. Modelos candidatos de redes

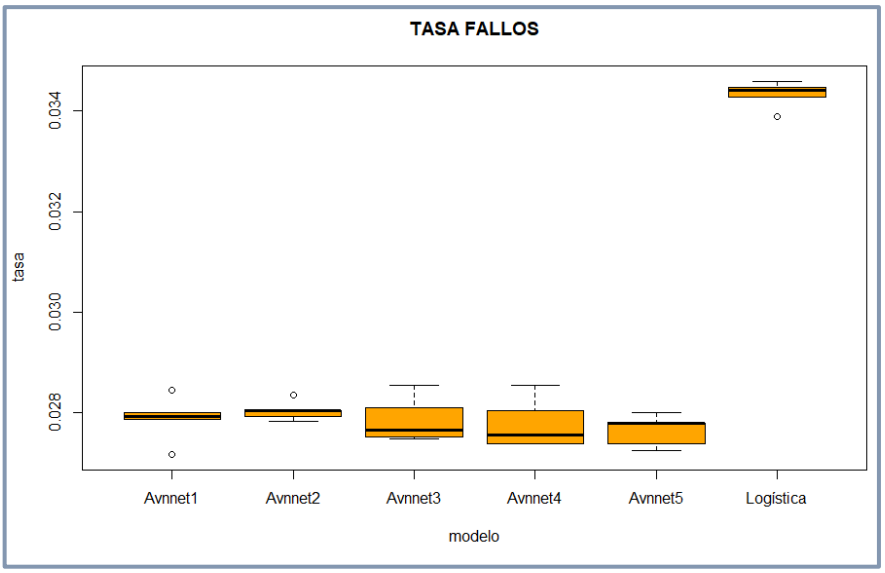


Ilustración 32. Boxplot tasa de fallos Avnnet

La ilustración 32 muestra que los modelos de redes mejoran al modelo de regresión logística calculado, si bien es una mejora de milésimas. Se observa que, en términos de tasa de fallos, los modelos muestran un bajo sesgo y varianza controlada, especialmente "Avnnet1" y "Avnnet2".

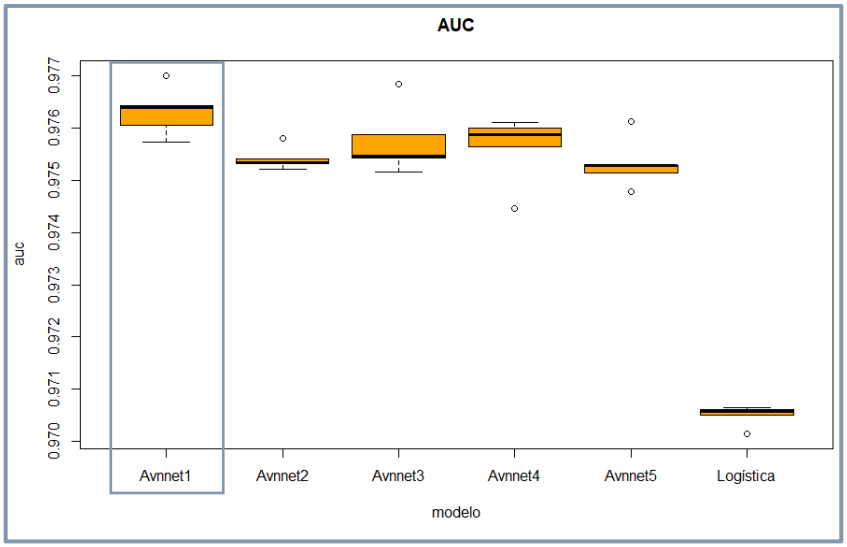


Ilustración 33. Boxplot AUC Avnnet

En el boxplot de la ilustración 33 se puede apreciar que, en términos de AUC, todos los modelos de redes presentan una performance muy buena. Se selecciona el modelo "Avnnet1" como la mejor red neuronal por presentar la mejor performance considerando el AUC, siendo además la red más sencilla, con 6 nodos ocultos.

Visualizando su matriz de confusión en la ilustración 34, se puede observar que el modelo falla muy poco a la hora de clasificar los accidentados leves, ya que estos son la gran mayoría. A la

hora de clasificar la clase positiva ya se observa una mayor carencia pues sólo es capaz de clasificarlos bien un 79.78% de las veces.

	Referencia				
Predicción		Sí	No		Exactitud	0.9724
	Sí	21181	868		Sensibilidad	0.79778
	No	5369	198622		Especificidad	0.99565
					Precisión	0.96063
					F1-Score	0.87166

Ilustración 34. Matriz de confusión y métricas de Avnnet1

5.3. Random Forest

Se realiza el tuneo de hiper-parámetros partiendo del set de 55 variables con el que se ha trabajado en las redes neuronales. Se fija una semilla y se procede a diseñar el grid con distintos valores para *mtry*, ya que este hiper-parámetro permite obtener el número óptimo de variables a sortear candidatas para abrir cada nodo del árbol.

mtry	Accuracy	Kappa
5	0.9079811	0.3371995
10	0.9586358	0.7723540
15	0.9661121	0.8199922
20	0.9675721	0.8292866
25	0.9676606	0.8299017
30	0.9681030	0.8323491
35	0.9685896	0.8353861
40	0.9674835	0.8300842
45	0.9677933	0.8319417
50	0.9679260	0.8325755
55	0.9668642	0.8274095

Ilustración 35. Resultado tuneo de mtry Random Forest

Se observa en la ilustración 35 que el accuracy es mayor para valores de *mtry* cercanos a 30-35 variables, por lo que, a continuación, se estudia el hiper-parámetro *ntree* referente número de iteraciones necesarias y, para ello, se observa el error OOB (out of bag).

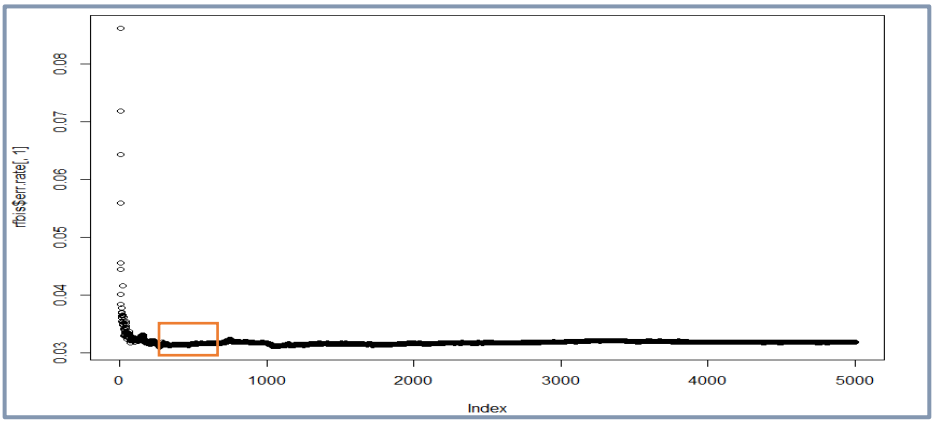


Ilustración 36. Error OOB según el número de árboles Random Forest

La ilustración 36 muestra que a partir de los 300-500 árboles el error ya está bastante estabilizado, por lo que calcular más árboles no van a mejorar o empeorar el modelo.

También se estudió el hiper-parámetro *sampsize*, referente al tamaño muestral. Para ello se probaron tamaños muestrales de 1000, 3000, 5000, 7000, 9000 y 11000. El error desciende y se estabiliza a partir de un tamaño muestral igual a 7000 como se observa en la ilustración 37.

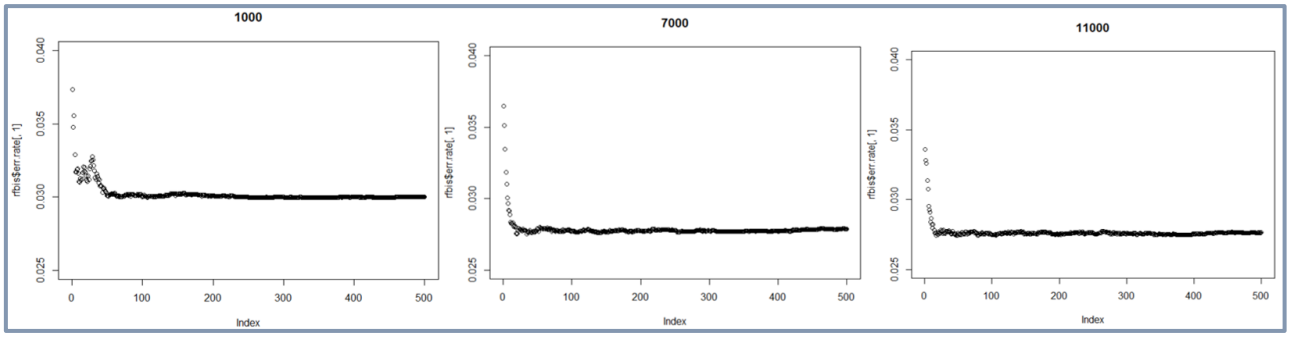


Ilustración 37. Error según el tamaño muestral Random Forest

Por último, se realiza un estudio del hiper-parámetro *nodesize*, referente al tamaño de hoja del modelo. Se probaron tamaños 5, 10, 15 y 20 mediante la observación del error del modelo para cada valor, manteniendo constantes el resto de hiper-parámetros. Como se observa en la ilustración 38, el tamaño de hoja que mejor funciona es cinco.

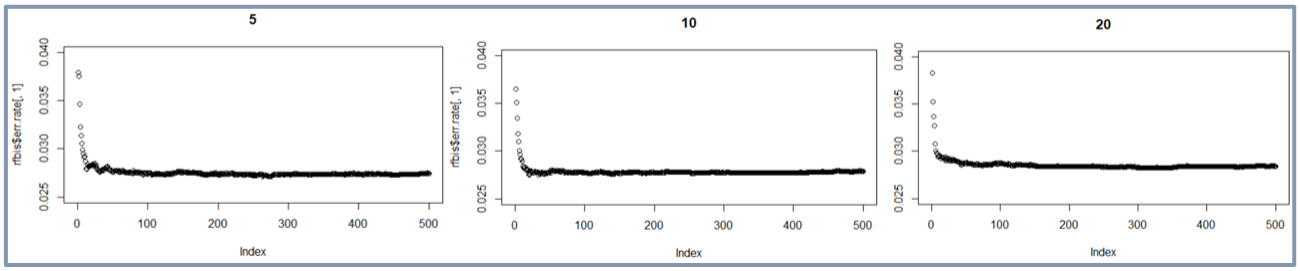


Ilustración 38. Error según el tamaño de la hoja Random Forest

Tras haber tuneado los hiper-parámetros que mejor funcionan, se vuelve a probar el número de variables candidatas que mejor pueden funcionar. Para ello se probaron valores de *mtry* desde 5 hasta 55 (bagging). También se probó un segundo grid con valores desde 25 hasta 30 al ser los valores con un accuracy más alto.

mtry	Accuracy	Kappa
5	0.9694744	0.8394816
10	0.9714210	0.8510247
15	0.9719960	0.8541087
20	0.9724827	0.8568288
25	0.9724826	0.8568594
30	0.9725269	0.8571362
35	0.9723499	0.8562594
40	0.9724384	0.8566798
45	0.9723942	0.8564895
50	0.9723500	0.8562218
55	0.9722172	0.8554956

mtry	Accuracy	Kappa
25	0.9725269	0.8572272
26	0.9724827	0.8569159
27	0.9724826	0.8568772
28	0.9725711	0.8573861
29	0.9727481	0.8583584
30	0.9723499	0.8563272

Ilustración 39. Resultado de pruebas de mtry

Como se puede observar en la ilustración 39, casi todos los posibles modelos presentan un accuracy muy parecido. Se decide probar los cuatro modelos marcados en recuadros naranja

mediante validación cruzada repetida para observar cuál de ellos presenta una mejor performance. Por tanto, los cuatro modelos tendrán parámetros comunes *ntree*=500, *sampsize*=7000, *nodesize*=5. Lo que se probará con validación cruzada repetida será el número de variables a sortear de cada modelo mostrado en la ilustración 40.

Modelo	RF1	RF2	RF3	RF4
mtry	20	25	29	30

Ilustración 40. Modelos candidatos de RF

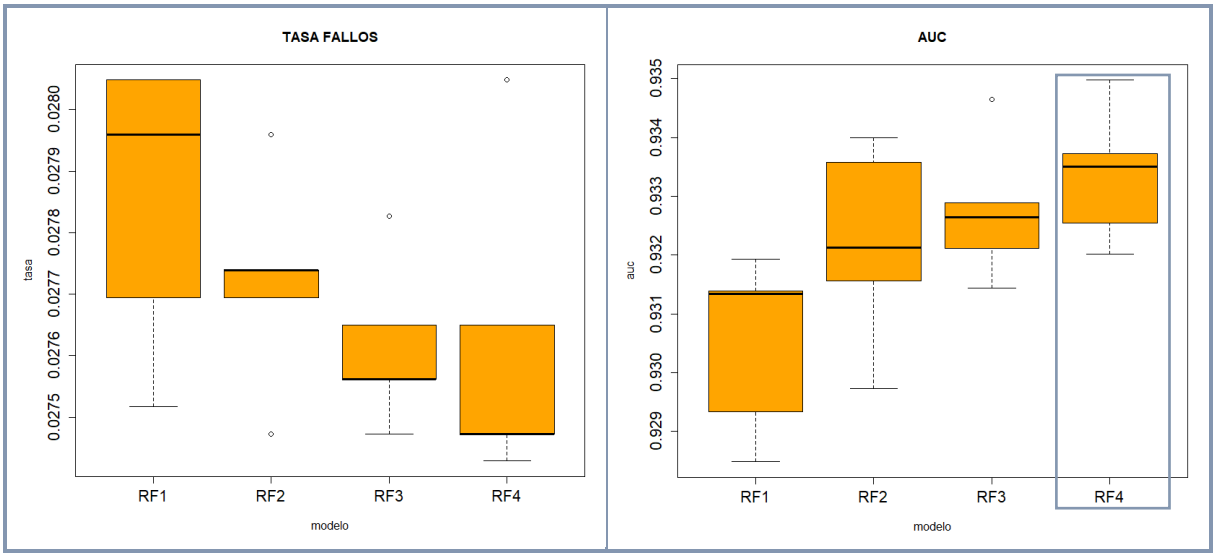


Ilustración 41. Boxplot tasa de fallos y AUC Random Forest

Si se atiende a los boxplot de la ilustración 41 podemos ver que a medida que aumenta el número de variables a sortear mejora la performance tanto en términos de AUC como de tasas de fallos. Priorizando el criterio del área bajo la curva ROC se elige al modelo "RF4" como el mejor de los modelos calculados empleando este algoritmo.

Analizando su matriz de confusión en la ilustración 42 se aprecia que el modelo de Random Forest lo hace igual de bien que el modelo de redes a la hora de clasificar la clase negativa y que mejora ligeramente con respecto a la clasificación de la clase positiva, haciéndolo correctamente en un 80.01% de los casos. Sin embargo, respecto al AUC sí se observa una pérdida considerable, pues hay que recordar que el AUC es invariable al punto de corte y las matrices de confusión presentadas en R toman por defecto un punto de corte de 0.5.

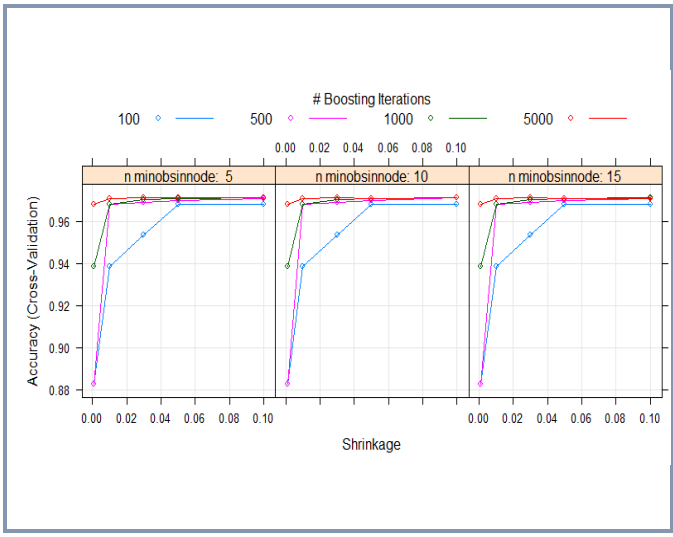
	Referencia		
Predicción		Sí	No
	Sí	21259	939
	No	5291	198551

Exactitud	0.9724
Sensibilidad	0.80072
Especificidad	0.99529
Precisión	0.95770
F1-Score	0.8722

Ilustración 42. Matriz de confusión y métricas de RF4

5.4. Gradient Boosting

Para el tuneo en Gradient Boosting se comienza diseñando un grid inicial con todos los hiper-parámetros a evaluar. El objetivo es evaluar la performance según la constante de regularización (*shrinkage*), probando entre 0.001 y 0.1; el número de observaciones en cada nodo final (*n.minobsinnode*), probando entre 5 y 15; así como el número de iteraciones (*n.trees*), probando valores entre 100 y 5000. Como se van a calcular únicamente árboles binarios el hiper-parámetro *interaction.depth* siempre será igual a 2. Hay que recordar que el punto de partida será el conjunto de 55 variables con el que se ha venido trabajando.



shrinkage	n.minobsinnode	n.trees	Accuracy	Kappa
0.030	15	100	0.9536364	0.7378772
0.030	15	500	0.9690320	0.8386910
0.030	15	1000	0.9703150	0.8457470
0.030	15	5000	0.9710227	0.8495574
0.050	5	100	0.9679260	0.8309773
0.050	5	500	0.9698726	0.8433961
0.050	5	1000	0.9708901	0.8488529
0.050	5	5000	0.9711112	0.8498926
0.050	10	100	0.9679260	0.8309773
0.050	10	500	0.9699611	0.8439189
0.050	10	1000	0.9708458	0.8486533
0.050	10	5000	0.9709342	0.8490069
0.050	15	100	0.9679260	0.8309773
0.050	15	500	0.9700495	0.8443797
0.050	15	1000	0.9708901	0.8487941
0.050	15	5000	0.9708458	0.8486688
0.100	5	100	0.9679702	0.8324888
0.100	5	500	0.9709343	0.8490046
0.100	5	1000	0.9711997	0.8504771
0.100	5	5000	0.9712439	0.8504416
0.100	10	100	0.9679702	0.8324888
0.100	10	500	0.9709786	0.8491996
0.100	10	1000	0.9710670	0.8497582
0.100	10	5000	0.9710670	0.8493635
0.100	15	100	0.9679702	0.8324888
0.100	15	500	0.9709343	0.8490595
0.100	15	1000	0.9711112	0.8500096
0.100	15	5000	0.9707573	0.8476535

Ilustración 43. Resultados tuneo de parámetros GBM

En la ilustración 43, se puede observar cómo la mejora la exactitud a medida que aumenta el número de iteraciones, si bien entre 1000 y 5000 iteraciones apenas hay mejora. Lo mismo ocurre si aumenta el *shrinkage*, siendo preferibles los valores 0.05 y 0.1 para este hiper-parámetro. En lo relativo al tamaño de hoja (*n.minobsinnode*), se observa que funciona mejor un tamaño de hoja pequeño en todos los modelos. Existen 4 modelos que son aquellos que presentan un mayor accuracy (resaltados en recuadros naranja). De estos destacan dos modelos que presentan mismo *shrinkage* (0.1) y número de observaciones por nodo (5) por lo que se fijarán estos hiper-parámetros para poder estudiar más detalladamente el número de iteraciones óptimo.

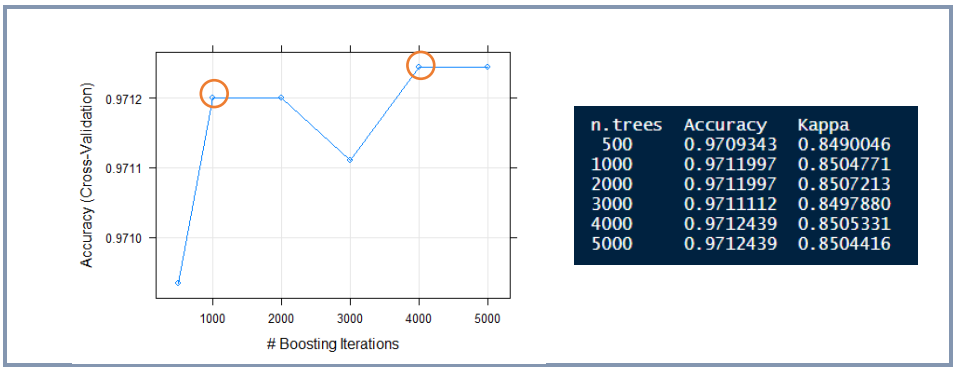


Ilustración 44. Prueba de Early Stopping GBM

Se observa, en la ilustración 44, que con 4000 iteraciones se puede conseguir el mismo accuracy que con 5000, por lo que se fija en este valor el número de iteraciones. Además, con 1000 iteraciones se obtienen resultados similares, por lo que ambos modelos serán probados mediante validación cruzada repetida junto con los dos modelos marcados anteriormente en la ilustración 43.

Una vez se han tuneado los hiper-parámetros, se han obtenido los cuatro posibles modelos de la ilustración 45 que deben ser comparados mediante validación cruzada repetida.

Modelo	*shrinkage*	*n.minobsinnode*	*n.trees*
GB1	0.1	5	1000
GB2	0.1	5	4000
GB3	0.05	5	5000
GB4	0.1	15	1000

Ilustración 45. Modelos candidatos GBM

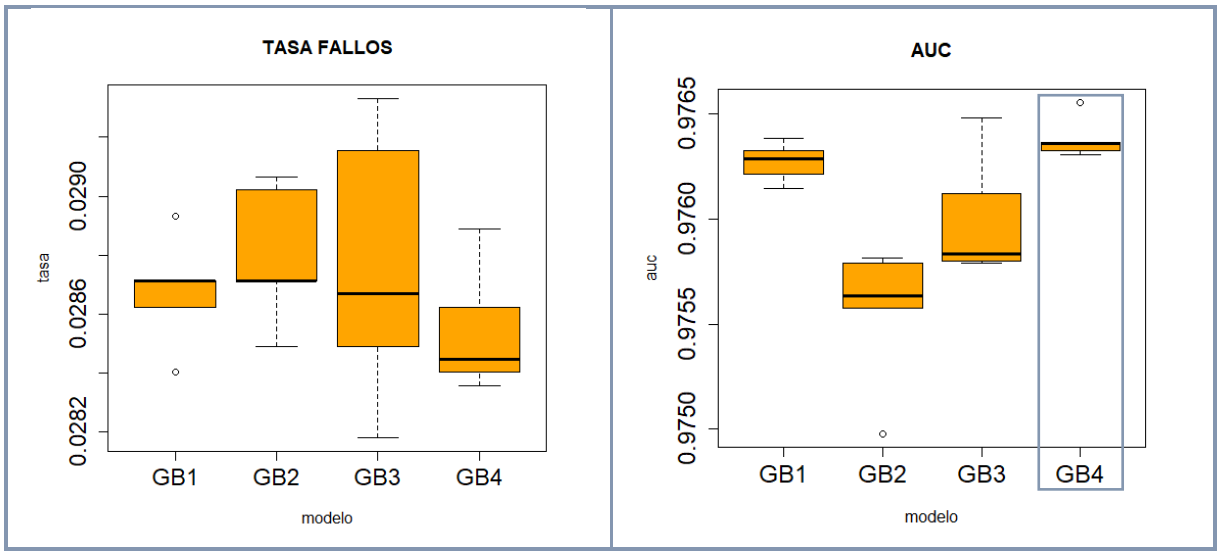


Ilustración 46. Boxplot tasa de fallos y AUC GBM

Tras aplicar validación cruzada repetida, se observa en la ilustración 46 que el modelo "GB4" es el que tiene una mejor performance tanto en términos de tasa de fallos como de área bajo la curva ROC. En esta métrica su performance destaca sobre todo su gran robustez.

Analizando su matriz de confusión de la ilustración 47, se aprecia que el mejor modelo de Gradient Boosting presenta métricas muy parecidas a Random Forest, si bien se observa una ligera pérdida en términos de sensibilidad. No obstante, en cuanto al AUC si se observa una mejora considerable con respecto a Random Forest.

	Referencia		
Predicción		Sí	No
	Sí	21151	1085
	No	5399	198405

Exactitud	0.9713
Sensibilidad	0.79665
Especificidad	0.99456
Precisión	0.95121
F1-Score	0.867093

Ilustración 47. Matriz de confusión y métricas de GB4

5.5. Extreme Gradient Boosting

Para realizar el tuneo del algoritmo Extreme Gradient Boosting (XGBoost) se establece un grid que nos permita estimar el valor de algunos hiper-parámetros como el número de iteraciones (*nrounds*), el tamaño de hoja (*min_child_weight*) y el *shrinkage* (*eta*).

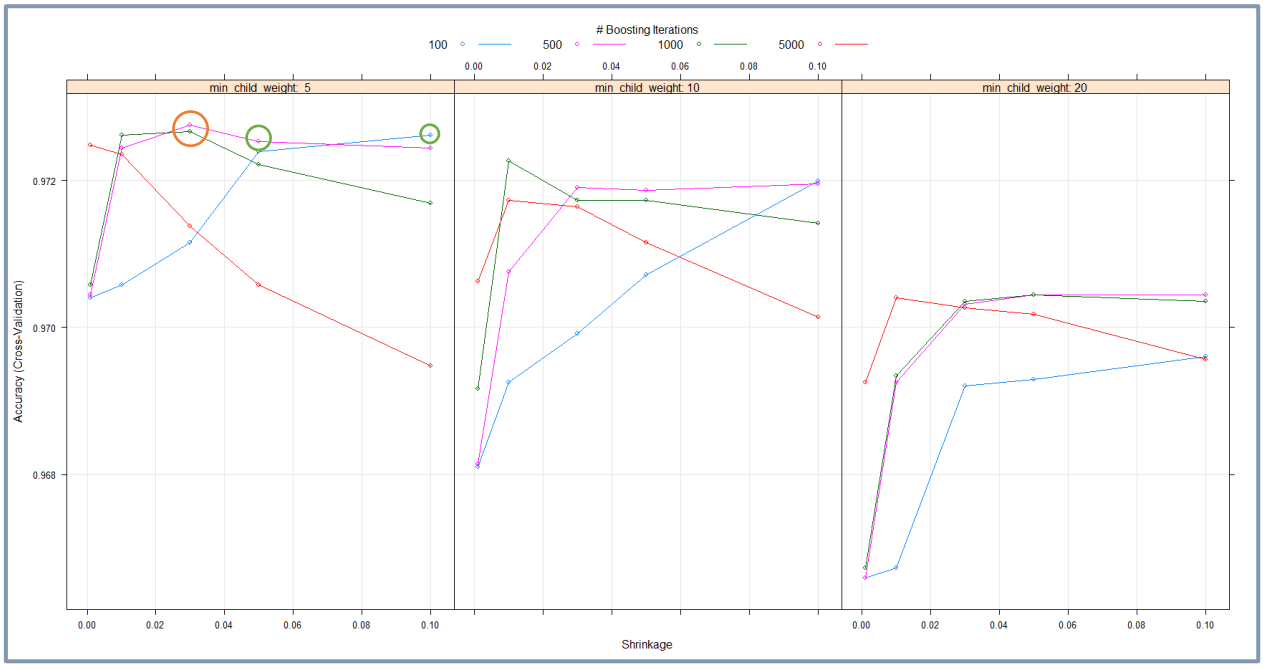


Ilustración 48. Resultados tuneo XGBoost

Lo primero que se observa en la ilustración 48 es la clara tendencia de los modelos a mejorar su performance a medida que disminuye el tamaño de hoja. Además, queda bastante claro cómo el *shrinkage* que mejor funciona es 0.03, tanto con 500 como con 1000 iteraciones (círculo

naranja). También se observa que un *shrinkage* alto funciona bastante bien con pocas iteraciones (círculos verdes).

Como se mencionó anteriormente, los modelos con *shrinkage* igual a 0.03 y tamaño de hoja igual a 5 presentan la mejor performance, por lo que se fijan estos dos parámetros y se estudiará un número de iteraciones más concreto con valores entre 300 y 1500.

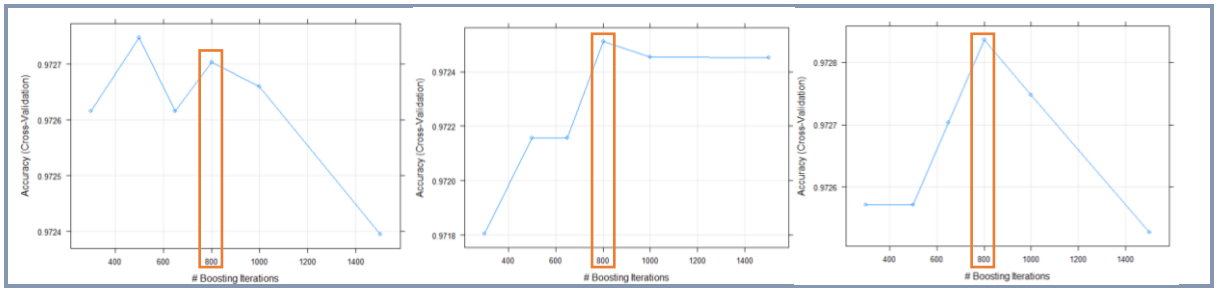


Ilustración 49. Early Stopping XGBoost

En la ilustración 49 se observa que, cambiando de semilla, el número de iteraciones necesarias parece ser 800, ya que a partir de ahí el algoritmo puede tender al sobreajuste.

También se estudia si se obtienen mejores resultados sorteando variables u observaciones. Para ello se crea una parrilla tanto para las variables como para las observaciones que permita comparar el accuracy de los modelos si se usan un 30%, 60% o 100% de observaciones y variables.

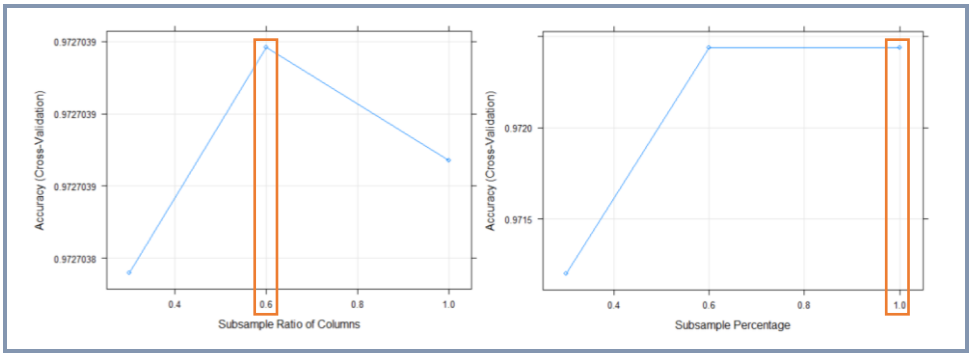


Ilustración 50. Pruebas de submuestreo en XGBoost

Como se puede apreciar en la ilustración 50 si se sortea un 60% de las variables se obtiene una mejor performance, mientras que si se sortean las observaciones no mejora la performance, por lo que se mantiene el parámetro por defecto en uno.

Por último, se evalúa el hiper-parámetro *gamma* probando valores entre 0 y 1, pues este hiper-parámetro penaliza el número de hojas del modelo.

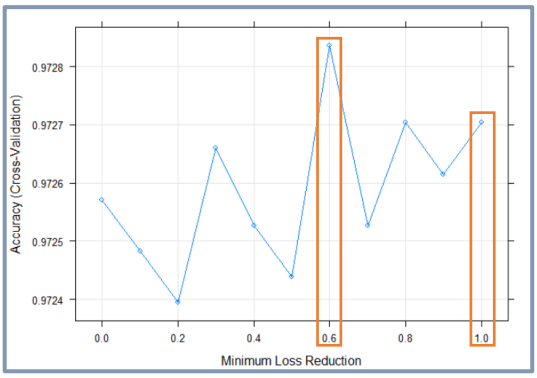


Ilustración 51. Tuneo valor de Gamma en XGBoost

Se aprecia en la ilustración 51 una clara tendencia creciente en el accuracy a medida que aumenta el valor de *gamma*. Por este motivo se probará el valor 0.6 con el que se consigue el accuracy más alto y el valor 1 que podría funcionar por esa tendencia creciente observada.

A la hora de comparar modelos mediante validación cruzada repetida, se tendrán en cuenta los 4 modelos de la ilustración 52. Dos de ellos provienen de todo el proceso de tuneado de hiper-parámetros (uno con *gamma*=0.6 y otro con *gamma*=1) y los otros dos corresponden a los modelos marcados en verde de la ilustración 36, que presentaban resultados similares con diferente número de iteraciones y *shrinkage*.

Modelo	*shrinkage*	*nrounds*	*min_child_weight*	*gamma*
Xgbm	0.03	800	5	0.6
Xgbm2	0.03	800	5	1
Xgbm3	0.05	500	5	0
Xgbm4	0.1	100	5	0

Ilustración 52. Modelos candidatos XGBM

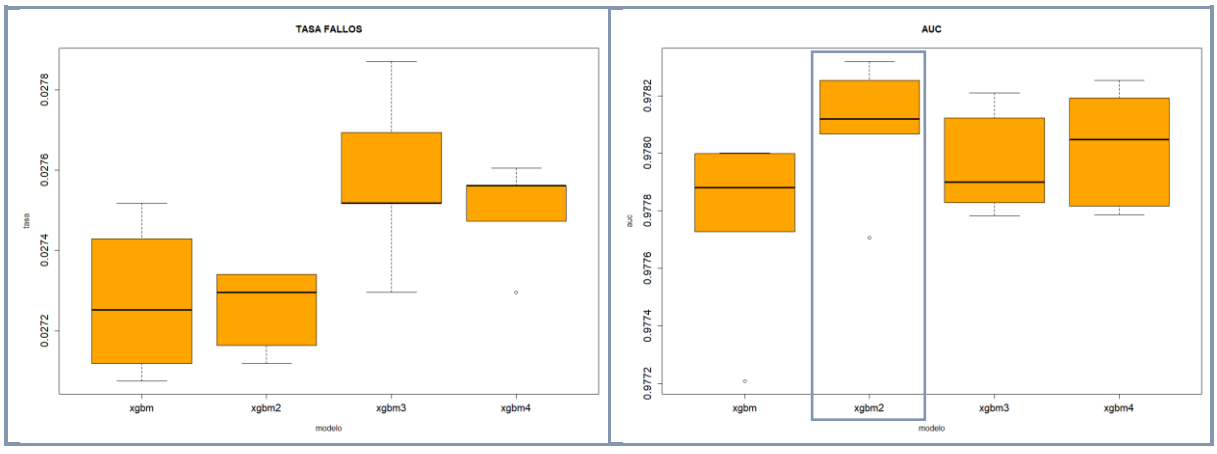


Ilustración 53. Boxplot tasa de fallos y AUC XGBoost

Analizando la ilustración 53 se observa que, aunque el modelo "Xgbm" es el que presenta un sesgo más bajo, el modelo "Xgbm2" presenta una performance en AUC realmente buena, por lo que se elige a este modelo como el mejor para este algoritmo.

Observando la matriz de confusión del mejor modelo XGBM en la ilustración 54 se aprecia cómo continúa en la línea de los anteriores dos modelos, aunque superando al modelo Gradient Boosting ligeramente en prácticamente todas las métricas expuestas.

	Referencia					
Predicción		Sí	No		Exactitud	0.9724
	Sí	21164	842		Sensibilidad	0.79714
	No	5386	198648		Especificidad	0.99578
					Precisión	0.96174
					F1-Score	0.871736

Ilustración 54. Matriz de confusión y métricas Xgbm2

5.6. Support Vector Machines

5.6.1. Support Vector Machines. Kernel Lineal

A la hora de trabajar con este algoritmo se debe tener en cuenta que sólo se puede tunear la constante de regularización *C*. El tuneo de este hiper-parámetro se realiza usando un grid que comprende valores entre 0.1 y 10.

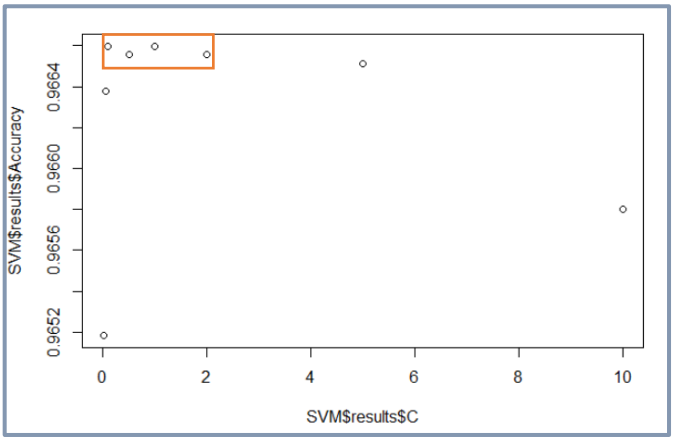


Ilustración 55. Tuneo valores C en SVM-L

En la ilustración 55 se puede observar cómo los mejores resultados en términos de accuracy se obtienen para valores de *C* entre 0.1 y 2, por lo que se prueba mediante validación cruzada repetida los cuatro modelos de la ilustración 56.

Modelo	*C*
SVM-L1	0.1
SVM-L2	0.5
SVM-L3	1
SVM-L4	2

Ilustración 56. Modelos candidatos SVM-L

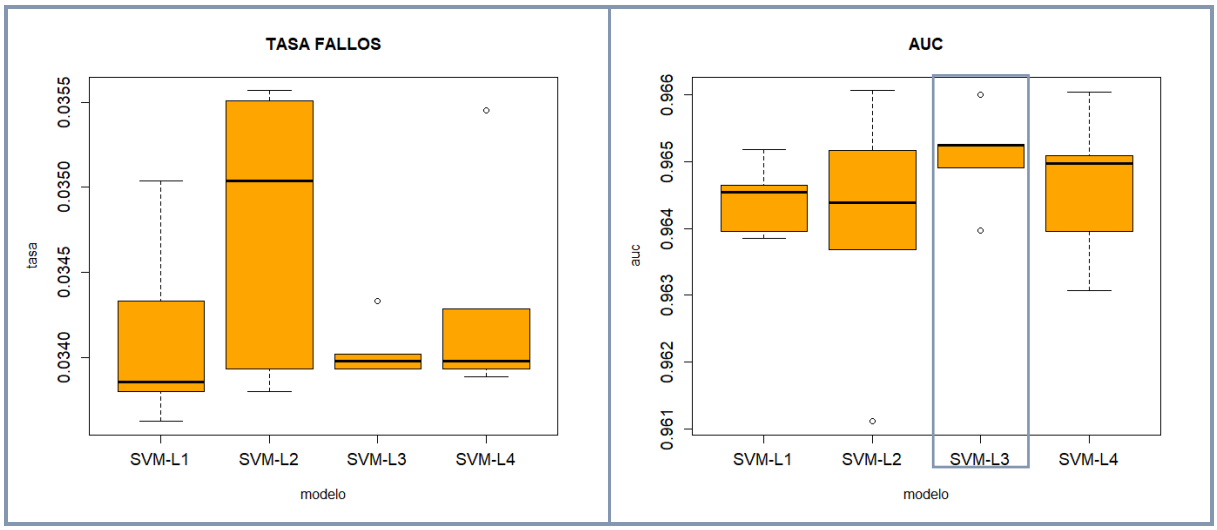


Ilustración 57. Boxplot tasa de fallos y AUC SVM-L

Si analizamos los boxplot de la ilustración 57 se puede apreciar cómo el modelo "SVM-L3" es el que presenta una mejor performance tanto en términos de tasa de fallos como en términos de área bajo la curva ROC. De este modelo destaca sobre todo su baja varianza en ambas métricas.

Observando su matriz de confusión de la ilustración 58 se puede ver una ligera pérdida en cuando a la sensibilidad, detectando un 77,86% de los accidentados graves frente al casi 80% de los modelos anteriores.

	Referencia					
Predicción		Sí	No		Exactitud	0.966
	Sí	20671	1806		Sensibilidad	0.77857
	No	5879	197684		Especificidad	0.99095
					Precisión	0.91965
					F1-Score	0.84325

Ilustración 58. Matriz de confusión y métricas de SVM-L3

5.6.2. Support Vector Machines. Kernel Polinomial

A la hora de tunear este algoritmo con kernel polinomial se tienen en cuenta tres hiper-parámetros: constante de regularización *C*, grado del polinomio (*degree*) y escala (*scale*). Para poder llevar a cabo este tuneo se prueban valores de *C* entre 0.01 y 10; grados del polinomio 2 y 3; y escala de 0.1 a 5. Los resultados se representan en un gráfico que nos permita medir el parámetro *C* como variable continua y los grados del polinomio y su escala como variables categóricas.

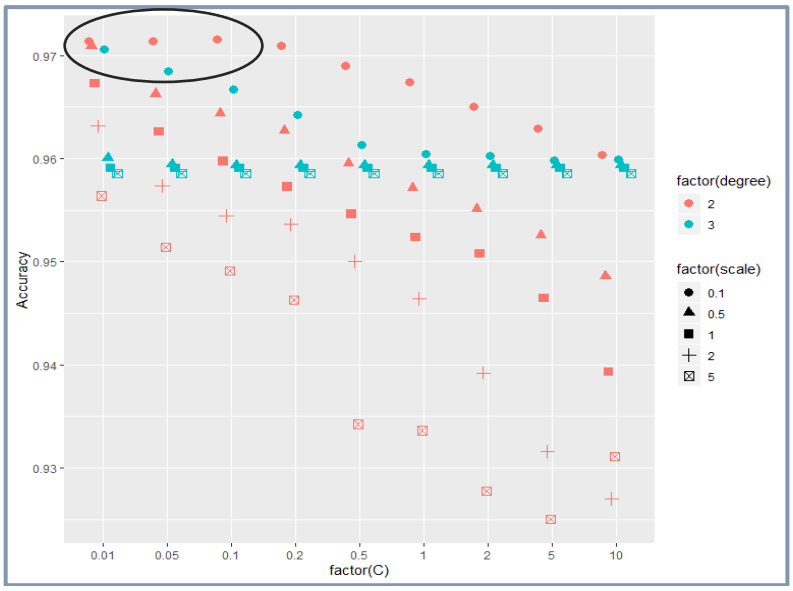


Ilustración 59. Tuneo parámetros SVM-Poly

En la ilustración 59 se aprecia cómo, en términos generales, el accuracy es mayor en los modelos polinómicos de grado 2 con un valor de escala igual a 0.1 y en un rango de *C* desde 0.01 hasta 0.1. También se observa cómo los modelos polinómicos de grado 3 presentan una performance mucho más estable salvo aquellos con valores de escala igual a 0.1 y valores de *C* muy pequeños, que presentan un accuracy más alto. Se prueban los mejores modelos mostrados en la ilustración 60 mediante validación cruzada repetida.

Modelo	*Degree*	*Scale*	*C*
SVM-Poly1	2	0.1	0.1
SVM-Poly2	2	0.1	0.05
SVM-Poly3	2	0.1	0.01
SVM-Poly4	3	0.05	0.01

Ilustración 60. Modelos candidatos SVM-Poly

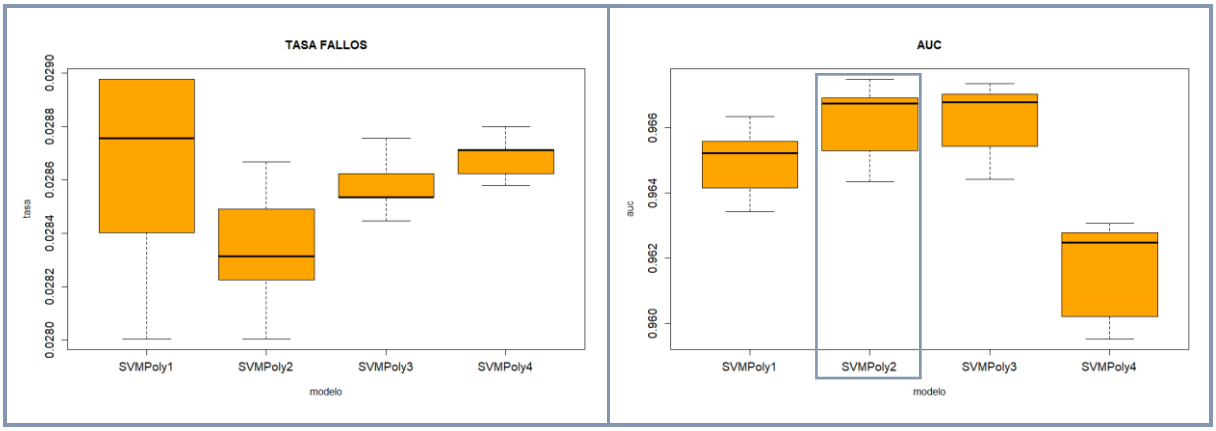


Ilustración 61. Boxplot tasa de fallos y AUC SVM-Poly

Observando los resultados de la ilustración 61, destaca como realmente presentan mejor performance los modelos con polinomio de grado 2 en términos de AUC. De todos los modelos

se seleccionará el modelo SVM-Poly2 por presentar una buena performance en AUC y el menor sesgo si se observa la tasa de fallos.

Observando su matriz de confusión en la ilustración 62 se observa que el modelo polinomial parece clasificar mejor que su variante lineal en prácticamente todas las métricas calculadas.

	Referencia		
Predicción		Sí	No
	Sí	21168	1041
	No	5382	198449

Exactitud	0.9716
Sensibilidad	0.79729
Especificidad	0.99478
Precisión	0.95313
F1-Score	0.86827

Ilustración 62. Matriz de confusión y métricas de SVM-Poly2

5.6.3. Support Vector Machines. Kernel RBF

Este kernel permite tunear el hiper-parámetro *sigma* junto con el valor de la constante de regularización. Se establece una parrilla con valores para *C* y *sigma* desde 0.01 hasta 30.

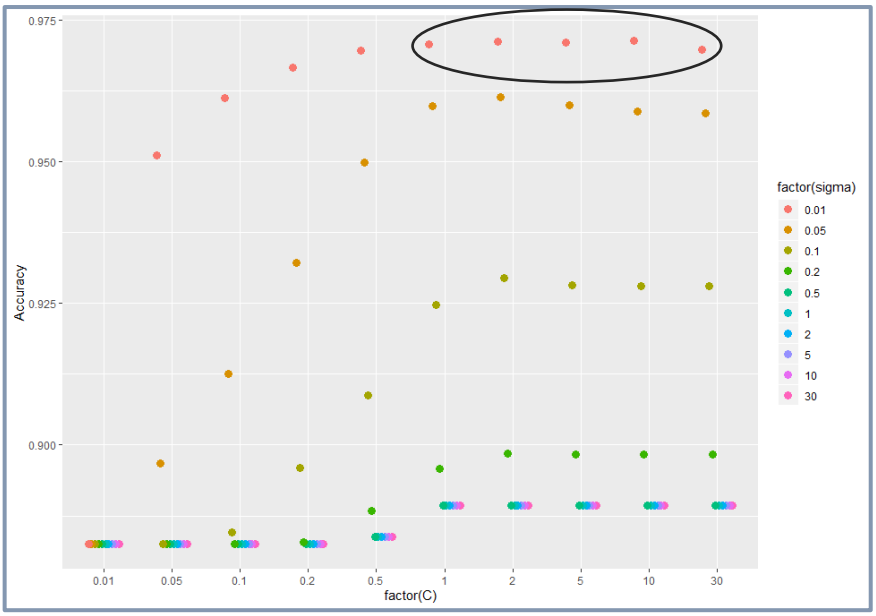


Ilustración 63. Tuneo parámetros SVM-RBF

En la ilustración 63 se puede apreciar que los mejores resultados se obtienen con *sigma* igual a 0.01. Además, se observa cómo el accuracy se estabiliza y se consiguen los mejores resultados para un *C* entre 1 y 30, por lo que estos modelos, mostrados en la ilustración 64, se prueban mediante validación cruzada repetida.

Modelo	Sigma	C
SVM-RBF1	0.01	1
SVM-RBF2	0.01	2
SVM-RBF3	0.01	5
SVM-RBF4	0.01	10

Ilustración 64. Modelos candidatos SVM-RBF

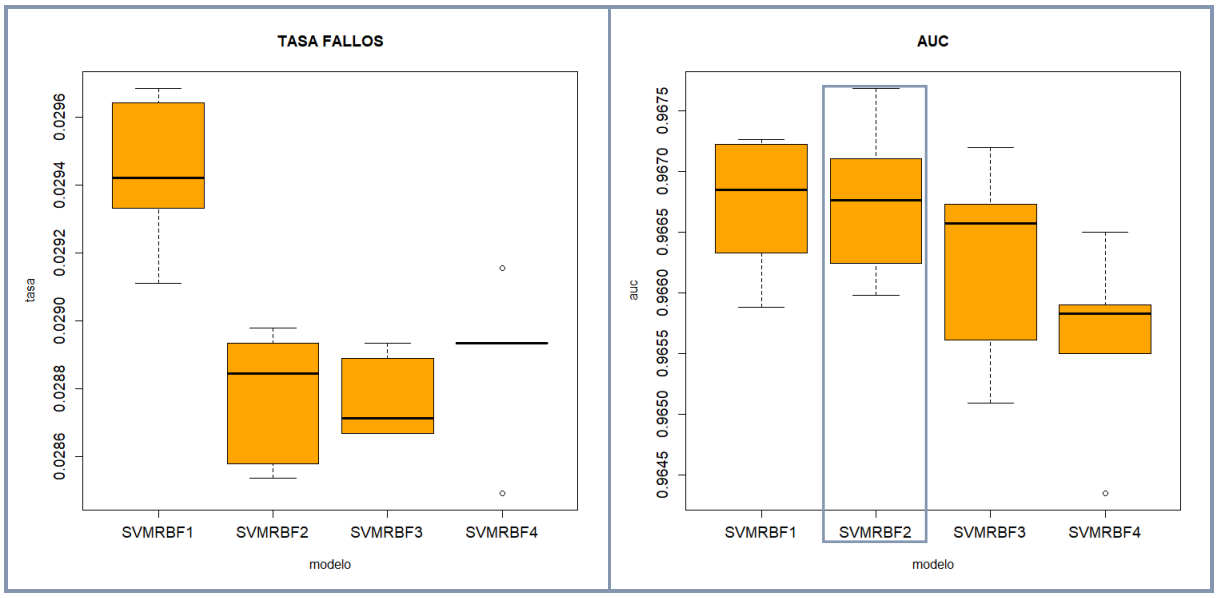


Ilustración 65. Boxplot tasa de fallos y AUC SVM-RBF

Atendiendo a la ilustración 65, se aprecia que, en términos de AUC, los mejores resultados los presentan los modelos "SVM-RBF1" y "SVM-RBF2". Sin embargo, el modelo "SVM-RBF2" presenta un sesgo mucho menor en tasa de fallos, por lo que se prefiere este modelo.

En su matriz de confusión de la ilustración 66 presenta prácticamente los mismos valores que los modelos comentados hasta ahora con una sensibilidad bastante correcta del 79.72%.

	Referencia		
Predicción		Sí	No
	Sí	21166	1110
	No	5384	198380

Exactitud	0.9713
Sensibilidad	0.79721
Especificidad	0.99444
Precisión	0.95017
F1-Score	0.866997

Ilustración 66. Matriz de confusión y métricas de SVM-RBF2

5.7. Ensamblado de modelos

Por último, se propone realizar una serie de modelos ensamblados con los mejores modelos de cada algoritmo. Para este ensamblado se procederá a promediar las probabilidades de varios modelos y observar si se puede conseguir una mejora sustancial con respecto a los modelos individuales calculados hasta ahora presentados en la ilustración 67.

Modelo	Métricas	
	Tasa de Fallos	AUC
Logística	0,03438	0,97052
Redes	0,02759	0,97644
Random Forest	0,02756	0,93303
GBM	0,02869	0,97634
XGBM	0,02755	0,97797

SVM-Lineal	0,03400	0,96504
SVM-Polinomial	0,02842	0,96635
SVM-Radial	0,02873	0,96691

Ilustración 67. Tasa de fallos y AUC de los mejores modelos

En total se calcularon 54 ensamblados combinando desde 2 hasta 8 modelos originales. De todos estos modelos calculados se muestran, en la ilustración 68, los 9 mejores ensamblados en términos de AUC, ordenados por su mediana y los modelos originales que los componen.

Orden	Ensamblado	Componente				
1	Predi10	XGBM	AVNNET			
2	Predi48	XGBM	AVNNET	GBM	SVM-RBF	
3	Predi32	XGBM	AVNNET	GBM		
4	Predi11	XGBM	GBM			
5	Predi34	XGBM	AVNNET	SVM-RBF		
6	Predi51	XGBM	AVNNET	GBM	RF	
7	Predi59	XGBM	AVNNET	GBM	LOGI	SVM-RBF
8	Predi39	AVNNET	GBM	SVM-RBF		
9	Predi47	XGBM	AVNNET	GBM	LOGI	

Ilustración 68. Mejores modelos ensamblados

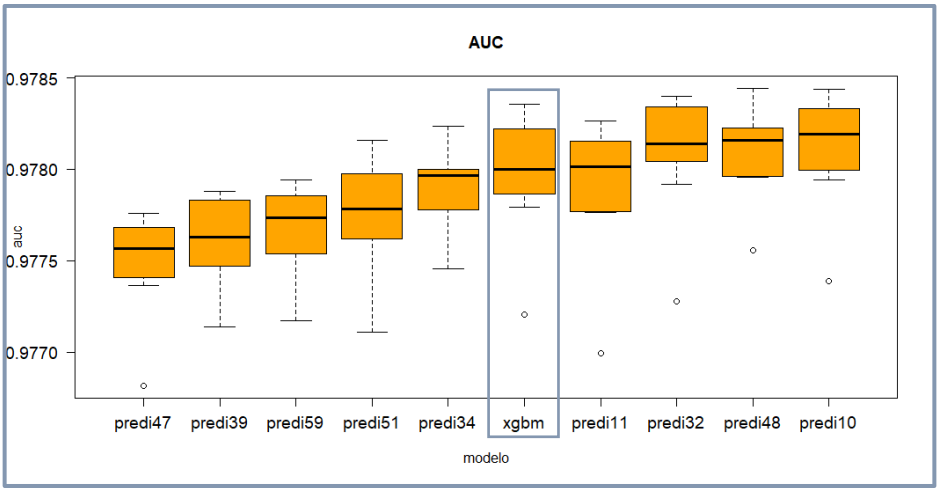


Ilustración 69. Boxplot AUC modelos ensamblados

En la ilustración 69 se observa que sólo son tres los ensamblados que superan al mejor algoritmo calculado hasta ahora, puesto que el modelo “predi11” muestra peor varianza. Estos mejores ensamblados claramente son el resultado de combinar el algoritmo XGBM con Redes Neuronales y eventualmente algún otro algoritmo. Como se aprecia en la imagen si bien existe una mejora, esta es mínima, por lo que se decide mantener el modelo XGBM como el mejor modelo por su buena performance y por su mayor simplicidad.

6. Comparación y evaluación de modelos

Tras haber observado que no tendría mucho sentido decantarse por un modelo de ensamblado ya que las mejoras son mínimas para la complejidad que agrega se decide presentar y comparar los modelos originales que resultaron ser los mejores modelos de cada técnica de machine learning. Los modelos originales y sus características son:

- Regresión logística.
- Red Neuronal: *size*=6, *decay*=0.1.
- Random Forest: mtry=30, ntrees=500, sampsize=7000 y *nodesize*=5.
- GBM: *shrinkage*=0.1, *n.minobsinnode*=15, *n.trees=1000*.
- XGBM: *shrinkage*=0.03, *min_child_weight*=5, *nrounds*=800 y *gamma*=1.
- SVM-Lineal: *C*=1.
- SVM-Polinomial: *C*=0.05, *grade*=2, *scale*=0.1.
- SVM-RBF: *C*= 2, *sigma*=0.01.

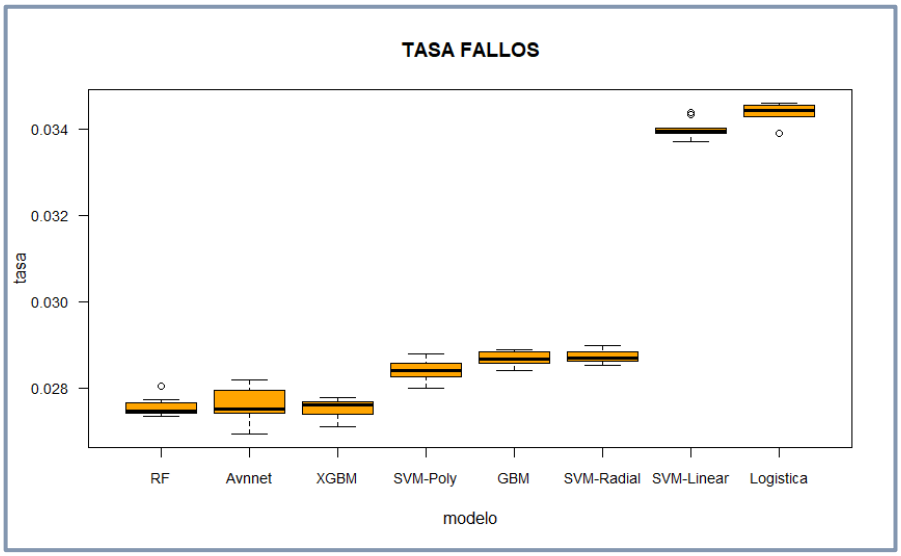


Ilustración 70. Boxplot tasa de fallos mejores algoritmos

Se puede observar cómo la mayoría de los modelos presentan una tasa de fallos bastante reducida atendiendo a los valores del eje Y de la ilustración 70. En general todos los modelos fallan bastante poco. Sin embargo, podemos ver que algunos algoritmos son aún un poco más fiables, destacando a Random Forest, Avnnet y XGBoost.

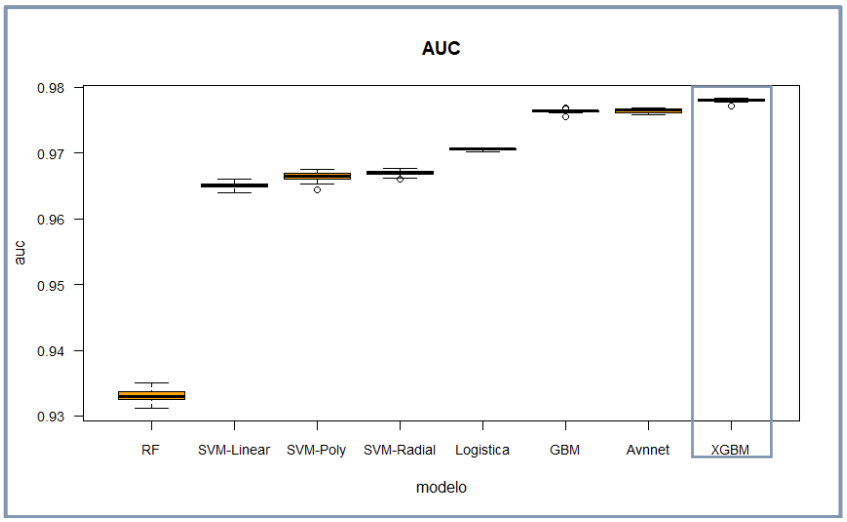


Ilustración 71. Boxplot AUC mejores algoritmos

Si se observa el área bajo la curva ROC, mostrado en la ilustración 71, destaca la baja performance en esta métrica del modelo de Random Forest, que había presentado la mejor tasa de fallos. Por el contrario, el algoritmo XGBoost es sin duda el modelo que mejor performance alcanza en esta medida. También es destacable cómo la regresión logística consigue una buena performance en términos de AUC.

Una vez elegido el mejor modelo, se exportan los datos de las probabilidades a Excel para poder observar la matriz de confusión y las métricas que se pueden extraer a partir de ésta para distintos valores de corte o umbrales de clasificación, algo necesario a tener en cuenta al existir desbalanceo de clases. Para elegir el punto de corte se ha considerado que para este tipo de problema convendría aumentar la sensibilidad que hasta el momento era del 79.71% mientras que la especificidad era del 99.58%.

Fijando la tasa de corte en 0.1175 (la proporción de desbalanceo) se obtuvo la siguiente matriz de confusión de la ilustración 72.

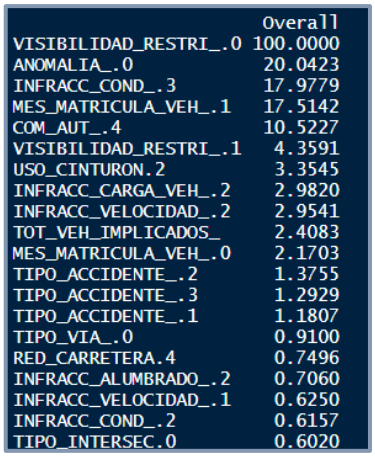
	Referencia					
Predicción		Sí	No		Exactitud	92,60%
	Sí	23889	14065		Sensibilidad	89,98%
	No	2661	185425		Especificidad	92,95%
					Precisión	62,94%
					F1-Score	74,07%

Ilustración 72. Matriz de confusión y métricas del mejor modelo (xgbm)

Se puede observar que fijando el punto de corte en 0.1175 mejora la sensibilidad del modelo ya que ahora se consigue un 13% más de verdaderos positivos y se reduce en un 51% los falsos negativos. Esto lo hace, evidentemente, a costa de reducir la especificidad, pero es algo lógico ya que, con clases desbalanceadas y un punto de corte de 0.5, el modelo tiende a clasificar

muchas más observaciones como negativas. Hay que destacar también que la precisión, es decir, la capacidad de acierto en las predicciones de la clase positiva ha bajado considerablemente de un 96.17% a un 62.94% debido a un aumento de los falsos positivos, el cual no ha sido proporcional al aumento de los verdaderos positivos.

Además, se procede a analizar las variables de mayor importancia para el modelo ganador Extreme Gradient Boosting.



	Overall
VISIBILIDAD_RESTRI_.0	100.0000
ANOMALIA_.0	20.0423
INFRACC_COND_.3	17.9779
MES_MATRICULA_VEH_.1	17.5142
COM_AUT_.4	10.5227
VISIBILIDAD_RESTRI_.1	4.3591
USO_CINTURON.2	3.3545
INFRACC_CARGA_VEH_.2	2.9820
INFRACC_VELOCIDAD_.2	2.9541
TOT_VEH_IMPLICADOS_	2.4083
MES_MATRICULA_VEH_.0	2.1703
TIPO_ACCIDENTE_.2	1.3755
TIPO_ACCIDENTE_.3	1.2929
TIPO_ACCIDENTE_.1	1.1807
TIPO_VIA_.0	0.9100
RED_CARRETERA.4	0.7496
INFRACC_ALUMBRADO_.2	0.7060
INFRACC_VELOCIDAD_.1	0.6250
INFRACC_COND_.2	0.6157
TIPO_INTERSEC.0	0.6020

Ilustración 73. Importancia de las variables XGBM

Si se observan en la ilustración 73 las variables más importantes para este algoritmo, se pueden realizar algunas interpretaciones relativas a su aparición y su significado:

- Sin duda, la variable visibilidad restringida (Visibilidad_Restri.0) es la variable que más le sirve al algoritmo para poder clasificar si un accidentado será una víctima mortal/grave o leve. Esta variable agrupada supone que es determinante para la gravedad del accidente tener restricciones como edificios, vegetación, factores atmosféricos o deslumbramiento.
- Se puede destacar también a la variable agrupada anomalía (Anomalia.0), que la componen aquellas motocicletas con neumáticos dañados o reventados o con problemas de frenos, algo que tiene sentido ya que ambas pueden afectar considerablemente a la capacidad para reducir la velocidad del vehículo.
- La variable agrupada infracción del conductor (Infrac_Cond.3) también es destacable, pues parece que las infracciones que más intervienen en la gravedad del motociclista accidentado son que éste no haya respetado regulaciones de prioridad (como ceda el paso, por ejemplo) o el no mantener un intervalo de seguridad adecuado.
- La variable agrupada Comunidad Autónoma (Com_Aut.4) también es una de las variables más importantes para este algoritmo y parece que los accidentes producidos

en Andalucía, Comunidad de Madrid, La Rioja o el País Vasco tienen incidencia en la gravedad del accidentado.

- Resalta que la variable uso del cinturón cuando no se lleva puesto (Uso_Cinturon.2) también sea determinante para clasificar la gravedad del herido. Quizá esto suponga que el chasis que tienen este tipo de motocicletas, que permite equipar un cinturón de seguridad, sirvan como una barrera entre el cuerpo del motociclista y el resto del entorno.

Por último, como se había mencionado, la regresión logística presentó unos buenos resultados a nivel general, por lo que, al presentar coeficientes, éstos se pueden interpretar y obtener una mayor información de las variables regresoras.

Para ello, es necesario calcular la exponencial de los parámetros y obtener el odds-ratio asociado a cada una de las variables. Como la gran mayoría de las variables del modelo son categóricas, su interpretación se debe realizar de acuerdo con la categoría de referencia.

Variables	Estimate	Pr(>|z|)	OR
VISIBILIDAD_RESTRI_.0	7,184519	< 2e-16	1.318,854095
ANOMALIA_.0	3,786561	< 2e-16	44,104459
INFRACC_COND_.3	1,388482	2.46e-15	4,008758
MES_MATRICULA_VEH_.1	2,674361	< 2e-16	14,503083
COM_AUT_.4	-1,975962	7.69e-14	0,138628
VISIBILIDAD_RESTRI_.1	4,491207	< 2e-16	89,229044
USO_CINTURON.2	0,549799	0.000945	1,732904

Ilustración 74. Odds-ratio de las variables más importantes

Como se puede apreciar en la ilustración 74, las variables más importantes en el modelo ganador sí son significativas en el modelo de regresión logística, lo que permite realizar nuevas observaciones relacionadas con las variables más importantes del modelo ganador:

- Vemos que la variable visibilidad restringida es fundamental a la hora de explicar las probabilidades de experimentar un accidente grave. El odds-ratio de la variable "Visibilidad_Restri.0", que engloba los tipos de restricciones visuales más comunes, nos indica que las posibilidades de sufrir un accidente grave se multiplican por más de 1300 que en accidentes en los que se desconoce o no hubo tal restricción. En el caso de la variable "visib_restri.1", que engloba a otro tipo de restricciones visuales, esta posibilidad se multiplica hasta por 89.

- El odds-ratio de la variable “anomalía.0”, que engloba anomalías en neumáticos, reventones y frenos, indica que si se sufre este tipo de anomalías la posibilidad de tener un accidente grave se multiplica hasta por 44 en comparación a accidentes con otro tipo de anomalía o en los que no hay anomalía (referencia “anomalía.1”).
- En lo relativo a las comunidades autónomas, se puede observar que en caso de sufrir un accidente en las comunidades de la variable agrupada “Com_Aut.4”, formadas por Andalucía, Comunidad de Madrid, La Rioja y País Vasco, las probabilidades de sufrir un accidente grave se reducen en un 79% en comparación a sufrirlo en la categoría de referencia “Com_Aut.5”, que agrupa a Cataluña, Ceuta y Melilla.

7. Conclusiones y trabajo futuro

Tras haber realizado un extenso trabajo de investigación relativo a la accidentalidad vial centrado en las motocicletas, se ha podido conseguir un modelo que permite clasificar correctamente a accidentados graves y leves. Para poder llegar a este resultado, se debió pasar por una compleja depuración de datos que permita entrenar una amplia variedad de algoritmos que se utilizan en la actualidad en machine learning. Con la obtención del modelo de Extreme Gradient Boosting se logró el objetivo principal fijado al inicio del trabajo.

Cabe destacar algunos aspectos importantes en cuanto al desarrollo de este trabajo:

- En primer lugar, hay que recalcar que la calidad de los datos disponibles es mejorable. La obtención de los datos es fácil, pero falta un mayor compromiso por parte de las instituciones públicas de actualizarlos anualmente. Además, se ha encontrado que muchas variables tenían significados distintos a los que indica la propia fuente de los microdatos en su diccionario de variables.
- En algunas variables, se han encontrado valores nulos y se desconoce su causa, ya que esta información proviene de los atestados policiales y puede intervenir el factor humano en la recolección de los datos.
- En cuanto a los modelos, hay que destacar la excelente capacidad de éstos para clasificar la gravedad de los accidentados. Destaca el algoritmo Extreme Gradient Boosting, que consiguió una sensibilidad del 89.98% y una especificidad del 92.95%. Siguiendo esta línea, quedó claro que para este tipo de datos con clases desbalanceadas es necesario cambiar el umbral de clasificación ya que, de lo contrario, el modelo tenderá a predecir

las observaciones como la clase mayoritaria, reduciendo su capacidad para detectar verdaderos positivos.

- Por otra parte, con respecto a la capacidad predictiva, se puede observar que al cambiar el punto de corte y obtener una mayor sensibilidad, el número de falsos positivos crece mucho más que el número de verdaderos positivos, por lo que la precisión del modelo para predecir al evento de interés baja hasta un 62.94%.
- En lo relativo al análisis de la importancia de las variables, se pudo observar cómo algunos aspectos de la visibilidad, determinadas anomalías o algunas infracciones cometidas por motociclistas son determinantes a la hora de sufrir accidentes leves o graves. Conocer esta información podría ayudar a actuar mejor en caso de aplicar este tipo de estudios a la atención de heridos en accidentes de motociclistas.

Por último, respecto a posibles trabajos futuros, convendría tener en cuenta algunos aspectos:

- Sería interesante poder ampliar el número de variables y su calidad. Con estas variables se tiene una imagen muy general del suceso, pero esta información es un tanto anticuada ya que no aprovecha los beneficios de la tecnología actual. Por ejemplo, se podría obtener en tiempo real (si los conductores lo permiten) información relativa a la velocidad, la desaceleración del cuerpo del conductor o muchas otras variables que puedan ayudar a predecir con mayor precisión la gravedad de los accidentados.
- Siguiendo la línea de la apreciación anterior, sería conveniente comenzar a trabajar con datos que, por ejemplo, se puedan recoger en vehículos autónomos, ya que tienen la capacidad de obtener datos basados en la telemetría, por lo que su pureza podría ser muy adecuada para predecir este tipo de eventos.
- Por último, considero que sería un acierto centrar futuros estudios en algún usuario concreto de la vía, ya que permite orientar la perspectiva del problema y ayudaría a interpretar mejor los resultados. Por ejemplo, se podría llevar a cabo estudios similares enfocados en los accidentes con peatones.

8. Bibliografía

Abellán, J., López, G., & de Oña, J. (2013). Analysis of traffic accident severity using Decision Rules via Decision Trees. *Expert Systems with Applications*, *40*(15), 6047-6054. https://doi.org/10.1016/j.eswa.2013.05.027

Calviño, A. (2019). *Material de la asignatura Técnicas y Metodología de la Minería de Datos (SEMMA)*.

Chong, M., Abraham, A., & Paprzycki, M. (2005). Traffic Accident Analysis Using Machine Learning Paradigms. *Informatica (Slovenia), 29*, 89-98.

de Oña, J., Mujalli, R. O., & Calvo, F. J. (2011). Analysis of traffic accident injury severity on Spanish rural highways using Bayesian networks. *Accident Analysis & Prevention, 43*(1), 402-411. https://doi.org/10.1016/j.aap.2010.09.010

DGT. (2015). *Anuario Estadístico de Accidentes*. http://www.dgt.es/es/seguridad-vial/estadisticas-e-indicadores/publicaciones/anuario-estadistico-accidentes/

Espinosa, J. C. (2020, enero 2). El número de motoristas muertos alcanza su máximo de la última década. *El País*. https://elpais.com/politica/2020/01/02/actualidad/1577990565_300638.html

European Commission. (2018). *Traffic Safety Basic Facts on Motorcycles & Mopeds*.

Expansión. (2019, abril 7). *España, el sexto país de la UE con las carreteras más seguras*. EXPANSION. https://www.expansion.com/empresas/transporte/2019/04/07/5caa0351e5fdea197d8b45cd.html

Gálvez, J. M. J. (2015, octubre 19). Los accidentes de tráfico le cuestan a España 9.600 millones de euros. *El País*. https://elpais.com/politica/2015/10/15/actualidad/1444908822_442694.html

Herceg, L., & Yaman, E. (2019). Analysis of Road Accidents Using Machine Learning Techniques. *International Conference on Electrical Engineering and Computer Science*, 14-18.

Lantz, B. (2015). *Machine learning with R: Discover how to build machine learning algorithms, prepare data, and dig deep into data prediction techniques with R* (Second edition). Packt Publishing.

Montt, C., Castro, F., & Rodríguez, N. (2011). Análisis de Accidentes de Tránsito con Máquinas de Soporte Vectorial LS-SVM. *Ingeniería de Transporte*, *15*(2), 7-14.

Montt, C., Rubio, J. M., & Lanata, S. (2013). Análisis de accidentes de tránsito con inteligencia computacional. *Congreso Chileno de Ingeniería de Transporte*, *16*. https://revistas.uchile.cl/index.php/CIT/article/view/28446

Portela, J. (2019). *Material de la asignatura Técnicas de Machine Learning*.

RACE. (2019, septiembre 12). Accidentes de motos. ¿Por qué los motoristas son vulnerables? *RACE*. https://www.race.es/accidentes-motos-conductores-vulnerables

Vila, D. (2020). *Predicción de la gravedad de los heridos en accidentes de tráfico en Barcelona* [Trabajo de Fin de Máster]. Universitat Oberta de Catalunya.

World Health Organization. (2015). *Global status report on road safety 2015*.

World Health Organization, issuing body, & ProQuest (Firm). (2018). *Global status report on road safety 2018*. https://ebookcentral.proquest.com/lib/qut/detail.action?docID=5910092

9. Anexos

9.1. Selección de variables en SAS

```
%include 'C:\Users\Mario\Desktop\P2_MachineLearning\Todas_macros_BIN.sas';

libname discoc 'C:\Users\Mario\Desktop\Sets_DEP';
data uno;set discoc.acc_full_train;run;

proc freq data=uno;run;
proc contents data=uno out=sal;run;quit;
data;set sal;put name @@;run;
options mprint=0;

/*Obtención de conjuntos de variables mediante stepwise, forward y
backward*/
/*Macro randomselect modificada*/
```

```
%macro
randomselectlogmodi(data=,listclass=,vardepen=,modelo=,sinicio=,sfinal=,fra
cciontrain=,directorio=,metodo=);
options nocenter linesize=256;
proc printto print="C:\Users\Mario\Desktop\myv\basura.txt";run;
data;file "C:\Users\Mario\Desktop\myv\cosa.txt" ;run;
%do semilla=&sinicio %to &sfinal;
proc surveyselect data=&data rate=&fracciontrain out=sal1234
seed=&semilla;run;

%if &listclass ne %then %do;
ods output type3=parametros;
proc logistic data=sal1234;
      class &listclass;
    model &vardepen= &modelo/ selection=backward;
run;
data parametros;length effect $20. modelo $ 20000;retain modelo " ";set
parametros end=fin;effect=cat(' ',effect);
if _n_ ne 1 then modelo=catt(modelo,' ',effect);if fin then
do;variable=modelo;output;end;
run;
%end;
%else %do;
ods output  Logistic.ParameterEstimates=parametros;
proc logistic data=sal1234;
        model &vardepen= &modelo/ selection= &metodo;
run;
%end;
ods graphics off;
ods html close;
data;file "C:\Users\Mario\Desktop\myv\cosa.txt" mod;set parametros;
%if &listclass ne %then %do; put variable @@;%end;
%else %do; if _n_ ne 1 then put variable @@;%end;
run;
%end;
proc printto ;run;
data todos;
infile "C:\Users\Mario\Desktop\myv\cosa.txt";
length efecto $ 400;
input efecto @@;
if efecto ne 'Intercept' then output;
run;
proc freq data=todos;tables efecto /out=sal;run;
proc sort data=sal;by descending count;
proc print data=sal;run;

data todos;
infile "C:\Users\Mario\Desktop\myv\cosa.txt";
length efecto $ 200;
input efecto $ &&;
run;
proc freq data=todos;tables efecto /out=sal;run;
proc sort data=sal;by descending count;
proc print data=sal;run;
data;set sal;put efecto;run;
%mend;

%randomselectlogmodi(data=uno,
listclass=ANIO_MATRICULA_VEH_ ANIO_PERMISO_ ANOMALIA_ COM_AUT_ DIA_SEM
FACTORES_ATMOS_ INFRACC_ALUMBRADO_ INFRACC_CARGA_VEH_
```

```
INFRACC_COND_ INFRACC_RESUMEN_ INFRACC_VELOCIDAD_ LUMINOSIDAD
MES_MATRICULA_VEH_ PROVINCIA_ RED_CARRETERA SEXO_
TIPO_ACCIDENTE_ TIPO_INTERSEC TIPO_VIA_ TOT_VEH_IMPLICADOS_
TRAZADO_NO_INTERSEC_ USO_CASCO USO_CINTURON VISIBILIDAD_RESTRI_
ZONA ZONA_AGRUPADA,
vardepen=ACDO_GRAVE,
modelo=ALEATORIA EDAD_ HORA_ ANIO_MATRICULA_VEH_ ANIO_PERMISO_ ANOMALIA_
COM_AUT_ DIA_SEM FACTORES_ATMOS_ INFRACC_ALUMBRADO_ INFRACC_CARGA_VEH_
INFRACC_COND_ INFRACC_RESUMEN_ INFRACC_VELOCIDAD_ LUMINOSIDAD
MES_MATRICULA_VEH_ PROVINCIA_ RED_CARRETERA SEXO_
TIPO_ACCIDENTE_ TIPO_INTERSEC TIPO_VIA_ TOT_VEH_IMPLICADOS_
TRAZADO_NO_INTERSEC_ USO_CASCO USO_CINTURON VISIBILIDAD_RESTRI_
ZONA ZONA_AGRUPADA,
sinicio=12345,sfinal=12380,fraccion train=0.8,metodo=forward)/* forward,
backward, stepwise*/

/*Probamos los mejores modelos con logística*/
%macro
cruzadalogistica(archivo=,vardepen=,conti=,categor=,ngrupos=,sinicio=,sfina
l=,objetivo=tasafallos);
title ' ';
data final;run;
/* Bucle semillas */
%do semilla=&sinicio %to &sfinal;
      data dos;set &archivo;u=ranuni(&semilla);
      proc sort data=dos;by u;run;
      data dos (drop=nume);
      retain grupo 1;
      set dos nobs=nume;
      if _n_>grupo*nume/&ngrupos then grupo=grupo+1;
      run;
      data fantasma;run;
      %do exclu=1 %to &ngrupos;
            data tres;set dos;if grupo ne &exclu then vardep=&vardepen;
            proc logistic data=tres noprint;/*<<<<<******SE PUEDE QUITAR EL
NOPRINT */
            %if (&categor ne) %then %do;class &categor;model vardep=&conti
&categor ;%end;
            %else %do;model vardep=&conti;%end;
            output out=sal p=predi;run;
            data sal2;set sal;pro=1-predi;if pro>0.5 then pre11=1; else
pre11=0;
            if grupo=&exclu then output;run;
            proc freq data=sal2;tables pre11*&vardepen/out=sal3;run;
            data estadisticos (drop=count percent pre11 &vardepen);
            retain vp vn fp fn suma 0;
            set sal3 nobs=nume;
            suma=suma+count;
            if pre11=0 and &vardepen=0 then vn=count;
            if pre11=0 and &vardepen=1 then fn=count;
            if pre11=1 and &vardepen=0 then fp=count;
            if pre11=1 and &vardepen=1 then vp=count;
            if _n_=nume then do;
            porcenVN=vn/suma;
            porcenFN=FN/suma;
            porcenVP=VP/suma;
            porcenFP=FP/suma;
            sensi=vp/(vp+fn);
            especif=vn/(vn+fp);
            tasafallos=1-(vp+vn)/suma;
            tasaciertos=1-tasafallos;
```

```
            precision=vp/(vp+fp);
            F_M=2*Sensi*Precision/(Sensi+Precision);
            output;
            end;
            run;

            data fantasma;set fantasma estadisticos;run;
      %end;
      proc means data=fantasma sum noprint;var &objetivo;
      output out=sumaresi sum=suma mean=media;
      run;
      data sumaresi;set sumaresi;semilla=&semilla;
      data final (keep=suma media semilla);set final sumaresi;if suma=.
then delete;run;
%end;
proc print data=final;run;
%mend;

/*OBJETIVO SENSIBILIDAD*/
/*cambiar objetivo=sensi, especif, precisión, tasafallos según necesidad*/
/*STEP1*/
%cruzadalogistica
(archivo=uno,vardepen=ACDO_GRAVE,
conti= TOT_VEH_IMPLICADOS_,
categor= ANOMALIA_ COM_AUT_ DIA_SEM FACTORES_ATMOS_ INFRACC_ALUMBRADO_
INFRACC_CARGA_VEH_ INFRACC_COND_ INFRACC_VELOCIDAD_ MES_MATRICULA_VEH_
PROVINCIA_ RED_CARRETERA TIPO_ACCIDENTE_ TIPO_INTERSEC USO_CASCO
USO_CINTURON VISIBILIDAD_RESTRI_ ZONA_AGRUPADA,
objetivo=sensi, ngrupos=4,sinicio=12345,sfinal=12365);
data final1;set final;modelo=1;
/*STEP2*/
%cruzadalogistica
(archivo=uno,vardepen=ACDO_GRAVE,
conti=TOT_VEH_IMPLICADOS_,
categor= ANOMALIA_ COM_AUT_ DIA_SEM FACTORES_ATMOS_ INFRACC_ALUMBRADO_
INFRACC_CARGA_VEH_ INFRACC_COND_ INFRACC_RESUMEN_ INFRACC_VELOCIDAD_
MES_MATRICULA_VEH_ PROVINCIA_ RED_CARRETERA TIPO_ACCIDENTE_ TIPO_INTERSEC
TIPO_VIA_ USO_CASCO VISIBILIDAD_RESTRI_,
objetivo=sensi, ngrupos=4,sinicio=12345,sfinal=12365);
data final2;set final;modelo=2;
/*FOR1*/
%cruzadalogistica
(archivo=uno,vardepen=ACDO_GRAVE,
conti=TOT_VEH_IMPLICADOS_,
categor= ANOMALIA_ COM_AUT_ DIA_SEM FACTORES_ATMOS_ INFRACC_ALUMBRADO_
INFRACC_CARGA_VEH_ INFRACC_COND_ INFRACC_VELOCIDAD_ MES_MATRICULA_VEH_
PROVINCIA_ RED_CARRETERA TIPO_ACCIDENTE_ TIPO_INTERSEC TIPO_VIA_ USO_CASCO
USO_CINTURON VISIBILIDAD_RESTRI_,
objetivo=sensi, ngrupos=4,sinicio=12345,sfinal=12365);
data final3;set final;modelo=3;
/*BACK1*/
%cruzadalogistica
(archivo=uno,vardepen=ACDO_GRAVE,
conti=TOT_VEH_IMPLICADOS_,
categor= ANIO_PERMISO_ ANOMALIA_ COM_AUT_ DIA_SEM FACTORES_ATMOS_
INFRACC_ALUMBRADO_ INFRACC_CARGA_VEH_ INFRACC_COND_ INFRACC_VELOCIDAD_
MES_MATRICULA_VEH_ PROVINCIA_ RED_CARRETERA TIPO_ACCIDENTE_ TIPO_INTERSEC
TIPO_VIA_ USO_CASCO USO_CINTURON VISIBILIDAD_RESTRI_,
objetivo=sensi, ngrupos=4,sinicio=12345,sfinal=12365);
data final4;set final;modelo=4;
/*BACK2*/
```

```
%cruzadalogistica
(archivo=uno,vardepen=ACDO_GRAVE,
conti= TOT_VEH_IMPLICADOS_,
categor= ANIO_PERMISO_ ANOMALIA_ COM_AUT_ DIA_SEM INFRACC_ALUMBRADO_
INFRACC_CARGA_VEH_ INFRACC_COND_ INFRACC_VELOCIDAD_ MES_MATRICULA_VEH_
PROVINCIA_ RED_CARRETERA TIPO_ACCIDENTE_ TIPO_INTERSEC TIPO_VIA_ USO_CASCO
USO_CINTURON VISIBILIDAD_RESTRI_,
objetivo=sensi, ngrupos=4,sinicio=12345,sfinal=12365);
data final5;set final;modelo=5;

data union;set final1 final2 final3 final4 final5;
proc boxplot data=union;plot media*modelo;run;
```

9.2. Código R Redes neuronales

```
#Preparación inicial
library(sas7bdat)
library(nnet)
library(h2o)
library(dummies)
library(MASS)
library(reshape)
library(caret)

motos<-read.sas7bdat("C:/Users/Mario/Desktop/Sets_DEP/acc_full_train.sas7bdat")
continuas<-c("HORA_","EDAD_","TOT_VEH_IMPLICADOS_","ALEATORIA")

categoricas<-c("DIA_SEM", "ZONA", "ZONA_AGRUPADA", "RED_CARRETERA", "TIPO_INTERSEC",
"LUMINOSIDAD","USO_CINTURON","USO_CASCO","INFRACC_ALUMBRADO_","INFRACC_CARGA_VEH_","INFRACC
_RESUMEN_","INFRACC_VELOCIDAD_","FACTORES_ATMOS_","SEXO_","TRAZADO_NO_INTERSEC_","VISIBILID
AD_RESTRI_","TIPO_ACCIDENTE_","MES_MATRICULA_VEH_","TIPO_VIA_","PROVINCIA_","ANOMALIA_","IN
FRACC_COND_","COM_AUT_","ANIO_MATRICULA_VEH_", "ANIO_PERMISO_")

# a)Eliminar las observaciones con missing en alguna variable
motos2<-na.omit(motos,(!is.na(motos)))

# b)pasar las categóricas a dummies
motos3<- dummy.data.frame(motos2, categoricas, sep = ".")

# c)estandarizar las variables continuas
# Calculo medias y dtipica de datos y estandarizo (solo las continuas)
means <-apply(motos3[,continuas],2,mean)
sds<-sapply(motos3[,continuas],sd)

# Estandarizo solo las continuas y uno con las categoricas
motosbis<-scale(motos3[,continuas], center = means, scale = sds)
numerocont<-which(colnames(motos3)%in%continuas)
motosbis<-cbind(motosbis,motos3[,-numerocont])

# Importante definir la variable de salida con valores alfanuméricos Yes, No
motosbis$ACDO_GRAVE<-ifelse(motosbis$ACDO_GRAVE==1,"Yes","No")

#--------TUNEO REDES NEURONALES--------#
library(doParallel)
registerDoParallel(cores = detectCores() - 1)

set.seed(12346)
control<-trainControl(method = "repeatedcv",number=4,repeats=5,
                      savePredictions = "all",classProbs=TRUE)

avnnetgrid <-expand.grid(size=c(3,6,9,11,13),decay=c(0.001,0.01,0.1),bag=FALSE)

redavnnet<-
train(ACDO_GRAVE~ANOMALIA_.0+COM_AUT_.0+COM_AUT_.1+COM_AUT_.2+COM_AUT_.3+COM_AUT_.4+DIA_SEM
.1+DIA_SEM.2+DIA_SEM.3+DIA_SEM.4+DIA_SEM.5+DIA_SEM.6+FACTORES_ATMOS_.1+FACTORES_ATMOS_.2+FA
```

```
CTORES_ATMOS_.3+INFRACC_ALUMBRADO_.1+INFRACC_ALUMBRADO_.2+INFRACC_CARGA_VEH_.1+INFRACC_CARG
A_VEH_.2+INFRACC_COND_.0+INFRACC_COND_.1+INFRACC_COND_.2+INFRACC_COND_.3+INFRACC_VELOCIDAD_
.1+INFRACC_VELOCIDAD_.2+MES_MATRICULA_VEH_.0+MES_MATRICULA_VEH_.1+PROVINCIA_.0+PROVINCIA_.1
+PROVINCIA_.2+PROVINCIA_.3+PROVINCIA_.4+PROVINCIA_.5+PROVINCIA_.6+PROVINCIA_.7+RED_CARRETER
A.1+RED_CARRETERA.2+RED_CARRETERA.3+RED_CARRETERA.4+TIPO_ACCIDENTE_.1+TIPO_ACCIDENTE_.2+TIP
O_ACCIDENTE_.3+TIPO_INTERSEC.0+TIPO_INTERSEC.1+TIPO_INTERSEC.2+TIPO_INTERSEC.3+TIPO_VIA_.0+
TIPO_VIA_.1+TOT_VEH_IMPLICADOS_+USO_CASCO.1+USO_CASCO.2+USO_CINTURON.1+USO_CINTURON.2+VISIB
ILIDAD_RESTRI_.0+VISIBILIDAD_RESTRI_.1,data=motosbis, method="avNNet",linout=
FALSE,maxit=100,trControl=control,tuneGrid=avnnetgrid, repeats=5)
redavnnet

#Realizamos una nueva rejilla incluyendo nuevos valores para el número de nodos y un decay
de 0.05
set.seed(12346)
control<-trainControl(method = "repeatedcv",number=4,repeats=5,
                      savePredictions = "all",classProbs=TRUE)

avnnetgrid <-expand.grid(size=c(8,9,10),decay=c(0.05,0.1),bag=FALSE)

redavnnet<- train(ACDO_GRAVE~
ANOMALIA_.0+COM_AUT_.0+COM_AUT_.1+COM_AUT_.2+COM_AUT_.3+COM_AUT_.4+DIA_SEM.1+DIA_SEM.2+DIA_
SEM.3+DIA_SEM.4+DIA_SEM.5+DIA_SEM.6+FACTORES_ATMOS_.1+FACTORES_ATMOS_.2+FACTORES_ATMOS_.3+I
NFRACC_ALUMBRADO_.1+INFRACC_ALUMBRADO_.2+INFRACC_CARGA_VEH_.1+INFRACC_CARGA_VEH_.2+INFRACC_
COND_.0+INFRACC_COND_.1+INFRACC_COND_.2+INFRACC_COND_.3+INFRACC_VELOCIDAD_.1+INFRACC_VELOCI
DAD_.2+MES_MATRICULA_VEH_.0+MES_MATRICULA_VEH_.1+PROVINCIA_.0+PROVINCIA_.1+PROVINCIA_.2+PRO
VINCIA_.3+PROVINCIA_.4+PROVINCIA_.5+PROVINCIA_.6+PROVINCIA_.7+RED_CARRETERA.1+RED_CARRETERA
.2+RED_CARRETERA.3+RED_CARRETERA.4+TIPO_ACCIDENTE_.1+TIPO_ACCIDENTE_.2+TIPO_ACCIDENTE_.3+TI
PO_INTERSEC.0+TIPO_INTERSEC.1+TIPO_INTERSEC.2+TIPO_INTERSEC.3+TIPO_VIA_.0+TIPO_VIA_.1+TOT_V
EH_IMPLICADOS_+USO_CASCO.1+USO_CASCO.2+USO_CINTURON.1+USO_CINTURON.2+VISIBILIDAD_RESTRI_.0+
VISIBILIDAD_RESTRI_.1,data=motosbis, method="avNNet",linout=
FALSE,maxit=100,trControl=control,tuneGrid=avnnetgrid, repeats=5)
redavnnet

#--------VALIDACIÓN CRUZADA REDES--------#
data<-motosbis

medias1<-cruzadalogistica(data=data,
vardep="ACDO_GRAVE",listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_A
UT_.3','COM_AUT_.4','DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6'
,'FACTORES_ATMOS_.1','FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRAC
C_ALUMBRADO_.2','INFRACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_CO
ND_.1','INFRACC_COND_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','M
ES_MATRICULA_VEH_.0','MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','P
ROVINCIA_.3','PROVINCIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1',
'RED_CARRETERA.2','RED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.
2','TIPO_ACCIDENTE_.3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC
.3','TIPO_VIA_.0','TIPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINT
URON.1','USO_CINTURON.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),
listclass=c(""), grupos=4,sinicio=1234,repe=5)
medias1$modelo="Logística"

medias2<-cruzadaavnnetbin(data=data,
vardep="ACDO_GRAVE",listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_A
UT_.3','COM_AUT_.4','DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6'
,'FACTORES_ATMOS_.1','FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRAC
C_ALUMBRADO_.2','INFRACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_CO
ND_.1','INFRACC_COND_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','M
ES_MATRICULA_VEH_.0','MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','P
ROVINCIA_.3','PROVINCIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1',
'RED_CARRETERA.2','RED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.
2','TIPO_ACCIDENTE_.3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC
.3','TIPO_VIA_.0','TIPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINT
URON.1','USO_CINTURON.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),
listclass=c(""),grupos=4,sinicio=1234,repe=5,
size=c(6),decay=c(0.1),repeticiones=5,itera=100)
medias2$modelo="Avnnet1"
```

```
medias3<-cruzadaavnnetbin(data=data,
vardep="ACDO_GRAVE",listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_A
UT_.3','COM_AUT_.4','DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6'
,'FACTORES_ATMOS_.1','FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRAC
C_ALUMBRADO_.2','INFRACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_CO
ND_.1','INFRACC_COND_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','M
ES_MATRICULA_VEH_.0','MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','P
ROVINCIA_.3','PROVINCIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1',
'RED_CARRETERA.2','RED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.
2','TIPO_ACCIDENTE_.3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC
.3','TIPO_VIA_.0','TIPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINT
URON.1','USO_CINTURON.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),
listclass=c(""),grupos=4,sinicio=1234,repe=5,
size=c(8),decay=c(0.05),repeticiones=5,itera=100)
medias3$modelo="Avnnet2"

medias4<-cruzadaavnnetbin(data=data,
vardep="ACDO_GRAVE",listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_A
UT_.3','COM_AUT_.4','DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6'
,'FACTORES_ATMOS_.1','FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRAC
C_ALUMBRADO_.2','INFRACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_CO
ND_.1','INFRACC_COND_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','M
ES_MATRICULA_VEH_.0','MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','P
ROVINCIA_.3','PROVINCIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1',
'RED_CARRETERA.2','RED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.
2','TIPO_ACCIDENTE_.3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC
.3','TIPO_VIA_.0','TIPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINT
URON.1','USO_CINTURON.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),
listclass=c(""),grupos=4,sinicio=1234,repe=5,
size=c(9),decay=c(0.05),repeticiones=5,itera=100)
medias4$modelo="Avnnet3"

medias5<-cruzadaavnnetbin(data=data,
vardep="ACDO_GRAVE",listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_A
UT_.3','COM_AUT_.4','DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6'
,'FACTORES_ATMOS_.1','FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRAC
C_ALUMBRADO_.2','INFRACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_CO
ND_.1','INFRACC_COND_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','M
ES_MATRICULA_VEH_.0','MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','P
ROVINCIA_.3','PROVINCIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1',
'RED_CARRETERA.2','RED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.
2','TIPO_ACCIDENTE_.3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC
.3','TIPO_VIA_.0','TIPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINT
URON.1','USO_CINTURON.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),
listclass=c(""),grupos=4,sinicio=1234,repe=5,
size=c(9),decay=c(0.1),repeticiones=5,itera=100)
medias5$modelo="Avnnet4"

medias6<-cruzadaavnnetbin(data=data,
vardep="ACDO_GRAVE",listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_A
UT_.3','COM_AUT_.4','DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6'
,'FACTORES_ATMOS_.1','FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRAC
C_ALUMBRADO_.2','INFRACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_CO
ND_.1','INFRACC_COND_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','M
ES_MATRICULA_VEH_.0','MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','P
ROVINCIA_.3','PROVINCIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1',
'RED_CARRETERA.2','RED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.
2','TIPO_ACCIDENTE_.3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC
.3','TIPO_VIA_.0','TIPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINT
URON.1','USO_CINTURON.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),
listclass=c(""),grupos=4,sinicio=1234,repe=5,
size=c(10),decay=c(0.1),repeticiones=5,itera=100)
medias6$modelo="Avnnet5"

union1<-rbind(medias1, medias2, medias3, medias4, medias5, medias6)
```

```
par(cex.axis=1)
boxplot(data=union1,tasa~modelo,main="TASA FALLOS",col='orange')
boxplot(data=union1,auc~modelo,main="AUC",col='orange')
```

9.3. Código R Random Forest

```
#--------TUNEO RANDOM FOREST--------#
library(dummies)
library(MASS)
library(reshape)
library(caret)
library(dplyr)
library(pROC)
library(randomForest)

data<-motosbis

# Use Parallel computing:
library(doParallel)
registerDoParallel(cores = detectCores() - 1)

#Primera aproximación al número de variables a sortear en RF (mtry):
set.seed(12345)
rfgrid<-expand.grid(mtry=c(5,10,15,20,25,30,35,40,45,50,55))
control<-trainControl(method = "cv",number=4,savePredictions = "all",classProbs=TRUE)

rf<-
train(factor(ACDO_GRAVE)~ANOMALIA_.0+COM_AUT_.0+COM_AUT_.1+COM_AUT_.2+COM_AUT_.3+COM_AUT_.4
+DIA_SEM.1+DIA_SEM.2+DIA_SEM.3+DIA_SEM.4+DIA_SEM.5+DIA_SEM.6+FACTORES_ATMOS_.1+FACTORES_ATM
OS_.2+FACTORES_ATMOS_.3+INFRACC_ALUMBRADO_.1+INFRACC_ALUMBRADO_.2+INFRACC_CARGA_VEH_.1+INFR
ACC_CARGA_VEH_.2+INFRACC_COND_.0+INFRACC_COND_.1+INFRACC_COND_.2+INFRACC_COND_.3+INFRACC_VE
LOCIDAD_.1+INFRACC_VELOCIDAD_.2+MES_MATRICULA_VEH_.0+MES_MATRICULA_VEH_.1+PROVINCIA_.0+PROV
INCIA_.1+PROVINCIA_.2+PROVINCIA_.3+PROVINCIA_.4+PROVINCIA_.5+PROVINCIA_.6+PROVINCIA_.7+
RED_CARRETERA.1+RED_CARRETERA.2+RED_CARRETERA.3+RED_CARRETERA.4+TIPO_ACCIDENTE_.1+TIPO_ACCI
DENTE_.2+TIPO_ACCIDENTE_.3+TIPO_INTERSEC.0+TIPO_INTERSEC.1+TIPO_INTERSEC.2+TIPO_INTERSEC.3+
TIPO_VIA_.0+TIPO_VIA_.1+TOT_VEH_IMPLICADOS_+USO_CASCO.1+USO_CASCO.2+USO_CINTURON.1+USO_CINT
URON.2+VISIBILIDAD_RESTRI_.0+VISIBILIDAD_RESTRI_.1,data=data,method="rf",trControl=control,
tuneGrid=rfgrid,linout = FALSE,ntree=1000,sampsize=200,nodesize=10,replace=TRUE)
rf

# Estudiamos el número de árboles a sortear (ntree):
library(randomForest)
set.seed(12345)

rfbis<-randomForest(factor(ACDO_GRAVE)~
ANOMALIA_.0+COM_AUT_.0+COM_AUT_.1+COM_AUT_.2+COM_AUT_.3+COM_AUT_.4+DIA_SEM.1+DIA_SEM.2+DIA_
SEM.3+DIA_SEM.4+DIA_SEM.5+DIA_SEM.6+FACTORES_ATMOS_.1+FACTORES_ATMOS_.2+FACTORES_ATMOS_.3+I
NFRACC_ALUMBRADO_.1+INFRACC_ALUMBRADO_.2+INFRACC_CARGA_VEH_.1+INFRACC_CARGA_VEH_.2+INFRACC_
COND_.0+INFRACC_COND_.1+INFRACC_COND_.2+INFRACC_COND_.3+INFRACC_VELOCIDAD_.1+INFRACC_VELOCI
DAD_.2+MES_MATRICULA_VEH_.0+MES_MATRICULA_VEH_.1+PROVINCIA_.0+PROVINCIA_.1+PROVINCIA_.2+PRO
VINCIA_.3+PROVINCIA_.4+PROVINCIA_.5+PROVINCIA_.6+PROVINCIA_.7+
RED_CARRETERA.1+RED_CARRETERA.2+RED_CARRETERA.3+RED_CARRETERA.4+TIPO_ACCIDENTE_.1+TIPO_ACCI
DENTE_.2+TIPO_ACCIDENTE_.3+TIPO_INTERSEC.0+TIPO_INTERSEC.1+TIPO_INTERSEC.2+TIPO_INTERSEC.3+
TIPO_VIA_.0+TIPO_VIA_.1+TOT_VEH_IMPLICADOS_+USO_CASCO.1+USO_CASCO.2+USO_CINTURON.1+USO_CINT
URON.2+VISIBILIDAD_RESTRI_.0+VISIBILIDAD_RESTRI_.1,
data=data,mtry=35,ntree=5000,sampsize=200,nodesize=10,replace=TRUE)

plot(rfbis$err.rate[,1])

# Probamos diferentes tamaños muestrales (sampsize):
for (muestra in seq(1000,11000,2000))
{
  # controlamos la semilla pues bagging depende de ella
set.seed(12345)
```

```
rfbis<-randomForest(factor(ACDO_GRAVE)~
ANOMALIA_.0+COM_AUT_.0+COM_AUT_.1+COM_AUT_.2+COM_AUT_.3+COM_AUT_.4+DIA_SEM.1+DIA_SEM.2+DIA_
SEM.3+DIA_SEM.4+DIA_SEM.5+DIA_SEM.6+FACTORES_ATMOS_.1+FACTORES_ATMOS_.2+FACTORES_ATMOS_.3+I
NFRACC_ALUMBRADO_.1+INFRACC_ALUMBRADO_.2+INFRACC_CARGA_VEH_.1+INFRACC_CARGA_VEH_.2+INFRACC_
COND_.0+INFRACC_COND_.1+INFRACC_COND_.2+INFRACC_COND_.3+INFRACC_VELOCIDAD_.1+INFRACC_VELOCI
DAD_.2+MES_MATRICULA_VEH_.0+MES_MATRICULA_VEH_.1+PROVINCIA_.0+PROVINCIA_.1+PROVINCIA_.2+PRO
VINCIA_.3+PROVINCIA_.4+PROVINCIA_.5+PROVINCIA_.6+PROVINCIA_.7+
RED_CARRETERA.1+RED_CARRETERA.2+RED_CARRETERA.3+RED_CARRETERA.4+TIPO_ACCIDENTE_.1+TIPO_ACCI
DENTE_.2+TIPO_ACCIDENTE_.3+TIPO_INTERSEC.0+TIPO_INTERSEC.1+TIPO_INTERSEC.2+TIPO_INTERSEC.3+
TIPO_VIA_.0+TIPO_VIA_.1+TOT_VEH_IMPLICADOS_+USO_CASCO.1+USO_CASCO.2+USO_CINTURON.1+USO_CINT
URON.2+VISIBILIDAD_RESTRI_.0+VISIBILIDAD_RESTRI_.1,
data=data,mtry=35,ntree=500,sampsize=muestra,nodesize=10,replace=TRUE)

plot(rfbis$err.rate[,1],main=muestra,ylim=c(0.025,0.04))
}

# Probamos diferentes números de nodos (nodesize):
for (muestra in seq(5,20,5))
{
# controlamos la semilla pues bagging depende de ella
set.seed(12345)
rfbis<-randomForest(factor(ACDO_GRAVE)~
ANOMALIA_.0+COM_AUT_.0+COM_AUT_.1+COM_AUT_.2+COM_AUT_.3+COM_AUT_.4+DIA_SEM.1+DIA_SEM.2+DIA_
SEM.3+DIA_SEM.4+DIA_SEM.5+DIA_SEM.6+FACTORES_ATMOS_.1+FACTORES_ATMOS_.2+FACTORES_ATMOS_.3+I
NFRACC_ALUMBRADO_.1+INFRACC_ALUMBRADO_.2+INFRACC_CARGA_VEH_.1+INFRACC_CARGA_VEH_.2+INFRACC_
COND_.0+INFRACC_COND_.1+INFRACC_COND_.2+INFRACC_COND_.3+INFRACC_VELOCIDAD_.1+INFRACC_VELOCI
DAD_.2+MES_MATRICULA_VEH_.0+MES_MATRICULA_VEH_.1+PROVINCIA_.0+PROVINCIA_.1+PROVINCIA_.2+PRO
VINCIA_.3+PROVINCIA_.4+PROVINCIA_.5+PROVINCIA_.6+PROVINCIA_.7+
RED_CARRETERA.1+RED_CARRETERA.2+RED_CARRETERA.3+RED_CARRETERA.4+TIPO_ACCIDENTE_.1+TIPO_ACCI
DENTE_.2+TIPO_ACCIDENTE_.3+TIPO_INTERSEC.0+TIPO_INTERSEC.1+TIPO_INTERSEC.2+TIPO_INTERSEC.3+
TIPO_VIA_.0+TIPO_VIA_.1+TOT_VEH_IMPLICADOS_+USO_CASCO.1+USO_CASCO.2+USO_CINTURON.1+USO_CINT
URON.2+VISIBILIDAD_RESTRI_.0+VISIBILIDAD_RESTRI_.1,
data=data, mtry=35,ntree=500,sampsize=7000,nodesize=muestra,replace=TRUE)

plot(rfbis$err.rate[,1],main=muestra,ylim=c(0.025,0.04))
}

#Probamos mtry de 5 a 55 (bagging) manteniendo el resto de parámetros fijos
set.seed(12345)
rfgrid<-expand.grid(mtry=c(5,10,15,20,25,30,35,40,45,50,55))
control<-trainControl(method = "cv",number=4,savePredictions = "all",classProbs=TRUE)

rf<- train(factor(ACDO_GRAVE)~
ANOMALIA_.0+COM_AUT_.0+COM_AUT_.1+COM_AUT_.2+COM_AUT_.3+COM_AUT_.4+DIA_SEM.1+DIA_SEM.2+DIA_
SEM.3+DIA_SEM.4+DIA_SEM.5+DIA_SEM.6+FACTORES_ATMOS_.1+FACTORES_ATMOS_.2+FACTORES_ATMOS_.3+I
NFRACC_ALUMBRADO_.1+INFRACC_ALUMBRADO_.2+INFRACC_CARGA_VEH_.1+INFRACC_CARGA_VEH_.2+INFRACC_
COND_.0+INFRACC_COND_.1+INFRACC_COND_.2+INFRACC_COND_.3+INFRACC_VELOCIDAD_.1+INFRACC_VELOCI
DAD_.2+MES_MATRICULA_VEH_.0+MES_MATRICULA_VEH_.1+PROVINCIA_.0+PROVINCIA_.1+PROVINCIA_.2+PRO
VINCIA_.3+PROVINCIA_.4+PROVINCIA_.5+PROVINCIA_.6+PROVINCIA_.7+
RED_CARRETERA.1+RED_CARRETERA.2+RED_CARRETERA.3+RED_CARRETERA.4+TIPO_ACCIDENTE_.1+TIPO_ACCI
DENTE_.2+TIPO_ACCIDENTE_.3+TIPO_INTERSEC.0+TIPO_INTERSEC.1+TIPO_INTERSEC.2+TIPO_INTERSEC.3+
TIPO_VIA_.0+TIPO_VIA_.1+TOT_VEH_IMPLICADOS_+USO_CASCO.1+USO_CASCO.2+USO_CINTURON.1+USO_CINT
URON.2+VISIBILIDAD_RESTRI_.0+VISIBILIDAD_RESTRI_.1,data=data,method="rf",trControl=control,
tuneGrid=rfgrid,linout = FALSE,ntree=500,sampsize=7000,nodesize=5,replace=TRUE)
rf

#Probamos mtry de 25 a 30:
set.seed(12345)
rfgrid<-expand.grid(mtry=c(25,26,27,28,29,30))

#--------CRUZADA RANDOM FOREST--------#

# Use Parallel computing:
registerDoParallel(cores = detectCores() - 1)
set.seed(12345)
```

```
medias1<-cruzadarfbinb(data=data, vardep="ACDO_GRAVE",
listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_AUT_.3','COM_AUT_.4',
'DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6','FACTORES_ATMOS_.1'
,'FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRACC_ALUMBRADO_.2','INF
RACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_COND_.1','INFRACC_COND
_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','MES_MATRICULA_VEH_.0'
,'MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','PROVINCIA_.3','PROVIN
CIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1','RED_CARRETERA.2','R
ED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.2','TIPO_ACCIDENTE_.
3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC.3','TIPO_VIA_.0','T
IPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINTURON.1','USO_CINTURO
N.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),listclass=c(""),
grupos=4,sinicio=1234,repe=5,nodesize=5,mtry=20,ntree=500,replace=TRUE,sampsize=7000)

medias1$modelo="RF1"

medias2<-cruzadarfbinb(data=data, vardep="ACDO_GRAVE",
listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_AUT_.3','COM_AUT_.4',
'DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6','FACTORES_ATMOS_.1'
,'FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRACC_ALUMBRADO_.2','INF
RACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_COND_.1','INFRACC_COND
_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','MES_MATRICULA_VEH_.0'
,'MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','PROVINCIA_.3','PROVIN
CIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1','RED_CARRETERA.2','R
ED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.2','TIPO_ACCIDENTE_.
3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC.3','TIPO_VIA_.0','T
IPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINTURON.1','USO_CINTURO
N.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),listclass=c(""),
grupos=4,sinicio=1234,repe=5,nodesize=5,mtry=25,ntree=500,replace=TRUE,sampsize=7000)

medias2$modelo="RF2"

medias3<-cruzadarfbinb(data=data, vardep="ACDO_GRAVE",
listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_AUT_.3','COM_AUT_.4',
'DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6','FACTORES_ATMOS_.1'
,'FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRACC_ALUMBRADO_.2','INF
RACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_COND_.1','INFRACC_COND
_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','MES_MATRICULA_VEH_.0'
,'MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','PROVINCIA_.3','PROVIN
CIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1','RED_CARRETERA.2','R
ED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.2','TIPO_ACCIDENTE_.
3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC.3','TIPO_VIA_.0','T
IPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINTURON.1','USO_CINTURO
N.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),listclass=c(""),
grupos=4,sinicio=1234,repe=5,nodesize=5,mtry=29,ntree=500,replace=TRUE,sampsize=7000)

medias3$modelo="RF3"

medias4<-cruzadarfbinb(data=data, vardep="ACDO_GRAVE",
listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_AUT_.3','COM_AUT_.4',
'DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6','FACTORES_ATMOS_.1'
,'FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRACC_ALUMBRADO_.2','INF
RACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_COND_.1','INFRACC_COND
_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','MES_MATRICULA_VEH_.0'
,'MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','PROVINCIA_.3','PROVIN
CIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1','RED_CARRETERA.2','R
ED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.2','TIPO_ACCIDENTE_.
3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC.3','TIPO_VIA_.0','T
IPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINTURON.1','USO_CINTURO
N.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),listclass=c(""),
grupos=4,sinicio=1234,repe=5,nodesize=5,mtry=30,ntree=500,replace=TRUE,sampsize=7000)

medias4$modelo="RF4"

union1<-rbind(medias1,medias2,medias3,medias4)
```

```
par(cex.axis=1.4)
boxplot(data=union1,tasa~modelo,main="TASA FALLOS",col="orange")
boxplot(data=union1,auc~modelo,main="AUC",col="orange")
```

9.4. Código R Gradient Boosting

```
#--------TUNEO GBM--------#
library(dummies)
library(MASS)
library(reshape)
library(caret)
library(dplyr)
library(pROC)
library(randomForest)

data<-motosbis
# Use Parallel computing:
library(doParallel)
registerDoParallel(cores = detectCores() - 1)

# Realizo el primer tuneo de los parámetros de forma general:
set.seed(12345)
gbmgrid<-expand.grid(shrinkage=c(0.001,0.01,0.03,0.05,0.1), n.minobsinnode=c(5,10,15),
                     n.trees=c(100,500,1000,5000), interaction.depth=c(2))

control<-trainControl(method = "cv",number=4,savePredictions = "all",
                      classProbs=TRUE)

gbm<- train(factor(ACDO_GRAVE)~
ANOMALIA_.0+COM_AUT_.0+COM_AUT_.1+COM_AUT_.2+COM_AUT_.3+COM_AUT_.4+DIA_SEM.1+DIA_SEM.2+DIA_
SEM.3+DIA_SEM.4+DIA_SEM.5+DIA_SEM.6+FACTORES_ATMOS_.1+FACTORES_ATMOS_.2+FACTORES_ATMOS_.3+I
NFRACC_ALUMBRADO_.1+INFRACC_ALUMBRADO_.2+INFRACC_CARGA_VEH_.1+INFRACC_CARGA_VEH_.2+INFRACC_
COND_.0+INFRACC_COND_.1+INFRACC_COND_.2+INFRACC_COND_.3+INFRACC_VELOCIDAD_.1+INFRACC_VELOCI
DAD_.2+MES_MATRICULA_VEH_.0+MES_MATRICULA_VEH_.1+PROVINCIA_.0+PROVINCIA_.1+PROVINCIA_.2+PRO
VINCIA_.3+PROVINCIA_.4+PROVINCIA_.5+PROVINCIA_.6+PROVINCIA_.7+
RED_CARRETERA.1+RED_CARRETERA.2+RED_CARRETERA.3+RED_CARRETERA.4+TIPO_ACCIDENTE_.1+TIPO_ACCI
DENTE_.2+TIPO_ACCIDENTE_.3+TIPO_INTERSEC.0+TIPO_INTERSEC.1+TIPO_INTERSEC.2+TIPO_INTERSEC.3+
TIPO_VIA_.0+TIPO_VIA_.1+TOT_VEH_IMPLICADOS_+USO_CASCO.1+USO_CASCO.2+USO_CINTURON.1+USO_CINT
URON.2+VISIBILIDAD_RESTRI_.0+VISIBILIDAD_RESTRI_.1,data=data,
method="gbm",trControl=control,tuneGrid=gbmgrid,
distribution="bernoulli", bag.fraction=1,verbose=FALSE)
gbm
plot(gbm)

# Fijamos shrinkage y n.minobsinnode para estudiar Early Stopping:
set.seed(12345)
gbmgrid<-expand.grid(shrinkage=c(0.1), n.minobsinnode=c(5),
                     n.trees=c(500,1000,2000,3000,4000,5000), interaction.depth=c(2))

control<-trainControl(method = "cv",number=4,savePredictions = "all",
                      classProbs=TRUE)

gbm<- train(factor(ACDO_GRAVE)~
ANOMALIA_.0+COM_AUT_.0+COM_AUT_.1+COM_AUT_.2+COM_AUT_.3+COM_AUT_.4+DIA_SEM.1+DIA_SEM.2+DIA_
SEM.3+DIA_SEM.4+DIA_SEM.5+DIA_SEM.6+FACTORES_ATMOS_.1+FACTORES_ATMOS_.2+FACTORES_ATMOS_.3+I
NFRACC_ALUMBRADO_.1+INFRACC_ALUMBRADO_.2+INFRACC_CARGA_VEH_.1+INFRACC_CARGA_VEH_.2+INFRACC_
COND_.0+INFRACC_COND_.1+INFRACC_COND_.2+INFRACC_COND_.3+INFRACC_VELOCIDAD_.1+INFRACC_VELOCI
DAD_.2+MES_MATRICULA_VEH_.0+MES_MATRICULA_VEH_.1+PROVINCIA_.0+PROVINCIA_.1+PROVINCIA_.2+PRO
VINCIA_.3+PROVINCIA_.4+PROVINCIA_.5+PROVINCIA_.6+PROVINCIA_.7+
RED_CARRETERA.1+RED_CARRETERA.2+RED_CARRETERA.3+RED_CARRETERA.4+TIPO_ACCIDENTE_.1+TIPO_ACCI
DENTE_.2+TIPO_ACCIDENTE_.3+TIPO_INTERSEC.0+TIPO_INTERSEC.1+TIPO_INTERSEC.2+TIPO_INTERSEC.3+
TIPO_VIA_.0+TIPO_VIA_.1+TOT_VEH_IMPLICADOS_+USO_CASCO.1+USO_CASCO.2+USO_CINTURON.1+USO_CINT
URON.2+VISIBILIDAD_RESTRI_.0+VISIBILIDAD_RESTRI_.1,data=data,
method="gbm",trControl=control,tuneGrid=gbmgrid,
distribution="bernoulli", bag.fraction=1,verbose=FALSE)
```

```
gbm
plot(gbm)

#--------CRUZADA GBM--------#
# Use Parallel computing:
library(doParallel)
registerDoParallel(cores = detectCores() - 1)

medias1<-cruzadagbmbin(data=data,

vardep="ACDO_GRAVE",listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_A
UT_.3','COM_AUT_.4','DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6'
,'FACTORES_ATMOS_.1','FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRAC
C_ALUMBRADO_.2','INFRACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_CO
ND_.1','INFRACC_COND_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','M
ES_MATRICULA_VEH_.0','MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','P
ROVINCIA_.3','PROVINCIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1',
'RED_CARRETERA.2','RED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.
2','TIPO_ACCIDENTE_.3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC
.3','TIPO_VIA_.0','TIPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINT
URON.1','USO_CINTURON.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),
listclass=c(""),grupos=4,sinicio=1234,repe=5,
n.minobsinnode=5,shrinkage=0.1,n.trees=1000,interaction.depth=2)
medias1$modelo="GB1"

medias2<-cruzadagbmbin(data=data,

vardep="ACDO_GRAVE",listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_A
UT_.3','COM_AUT_.4','DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6'
,'FACTORES_ATMOS_.1','FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRAC
C_ALUMBRADO_.2','INFRACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_CO
ND_.1','INFRACC_COND_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','M
ES_MATRICULA_VEH_.0','MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','P
ROVINCIA_.3','PROVINCIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1',
'RED_CARRETERA.2','RED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.
2','TIPO_ACCIDENTE_.3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC
.3','TIPO_VIA_.0','TIPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINT
URON.1','USO_CINTURON.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),
listclass=c(""),grupos=4,sinicio=1234,repe=5,
n.minobsinnode=5,shrinkage=0.1,n.trees=4000,interaction.depth=2)
medias2$modelo="GB2"

medias3<-cruzadagbmbin(data=data,

vardep="ACDO_GRAVE",listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_A
UT_.3','COM_AUT_.4','DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6'
,'FACTORES_ATMOS_.1','FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRAC
C_ALUMBRADO_.2','INFRACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_CO
ND_.1','INFRACC_COND_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','M
ES_MATRICULA_VEH_.0','MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','P
ROVINCIA_.3','PROVINCIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1',
'RED_CARRETERA.2','RED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.
2','TIPO_ACCIDENTE_.3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC
.3','TIPO_VIA_.0','TIPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINT
URON.1','USO_CINTURON.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),
listclass=c(""),grupos=4,sinicio=1234,repe=5,
n.minobsinnode=5,shrinkage=0.05,n.trees=5000,interaction.depth=2)
medias3$modelo="GB3"

medias4<-cruzadagbmbin(data=data,

vardep="ACDO_GRAVE",listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_A
UT_.3','COM_AUT_.4','DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6'
,'FACTORES_ATMOS_.1','FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRAC
C_ALUMBRADO_.2','INFRACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_CO
ND_.1','INFRACC_COND_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','M
```

```
ES_MATRICULA_VEH_.0','MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','P
ROVINCIA_.3','PROVINCIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1',
'RED_CARRETERA.2','RED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.
2','TIPO_ACCIDENTE_.3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC
.3','TIPO_VIA_.0','TIPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINT
URON.1','USO_CINTURON.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),
listclass=c(""),grupos=4,sinicio=1234,repe=5,
n.minobsinnode=15,shrinkage=0.05,n.trees=1000,interaction.depth=2)
medias4$modelo="GB4"

union1<-rbind(medias1,medias2,medias3,medias4)
par(cex.axis=1.5)
boxplot(data=union1,tasa~modelo,main="TASA FALLOS",col="orange")
boxplot(data=union1,auc~modelo,main="AUC",col="orange")
```

9.5. Código R Extreme Gradient Boosting

```
#--------TUNEO XGBM--------#
library(dummies)
library(MASS)
library(reshape)
library(caret)
library(dplyr)
library(pROC)
library(randomForest)

data<-motosbis

# Use Parallel computing:
library(doParallel)
registerDoParallel(cores = detectCores() - 1)

set.seed(12345)

xgbmgrid<-expand.grid(min_child_weight=c(5,10,20),eta=c(0.1,0.05,0.03,0.01,0.001),
nrounds=c(100,500,1000,5000),max_depth=6,gamma=0,colsample_bytree=1,subsample=1)

control<-trainControl(method = "cv",number=4,savePredictions = "all",classProbs=TRUE)

xgbm<-
train(factor(ACDO_GRAVE)~ANOMALIA_.0+COM_AUT_.0+COM_AUT_.1+COM_AUT_.2+COM_AUT_.3+COM_AUT_.4
+DIA_SEM.1+DIA_SEM.2+DIA_SEM.3+DIA_SEM.4+DIA_SEM.5+DIA_SEM.6+FACTORES_ATMOS_.1+FACTORES_ATM
OS_.2+FACTORES_ATMOS_.3+INFRACC_ALUMBRADO_.1+INFRACC_ALUMBRADO_.2+INFRACC_CARGA_VEH_.1+INFR
ACC_CARGA_VEH_.2+INFRACC_COND_.0+INFRACC_COND_.1+INFRACC_COND_.2+INFRACC_COND_.3+INFRACC_VE
LOCIDAD_.1+INFRACC_VELOCIDAD_.2+MES_MATRICULA_VEH_.0+MES_MATRICULA_VEH_.1+PROVINCIA_.0+PROV
INCIA_.1+PROVINCIA_.2+PROVINCIA_.3+PROVINCIA_.4+PROVINCIA_.5+PROVINCIA_.6+PROVINCIA_.7+
RED_CARRETERA.1+RED_CARRETERA.2+RED_CARRETERA.3+RED_CARRETERA.4+TIPO_ACCIDENTE_.1+TIPO_ACCI
DENTE_.2+TIPO_ACCIDENTE_.3+TIPO_INTERSEC.0+TIPO_INTERSEC.1+TIPO_INTERSEC.2+TIPO_INTERSEC.3+
TIPO_VIA_.0+TIPO_VIA_.1+TOT_VEH_IMPLICADOS_+USO_CASCO.1+USO_CASCO.2+USO_CINTURON.1+USO_CINT
URON.2+VISIBILIDAD_RESTRI_.0+VISIBILIDAD_RESTRI_.1,
data=data,method="xgbTree",trControl=control,tuneGrid=xgbmgrid,verbose=FALSE)
xgbm
plot(xgbm)

#Estudiamos early stoping:
xgbmgrid<-expand.grid(eta=c(0.03),min_child_weight=c(5),
nrounds=c(300,500,650,800,1000,1500),max_depth=6,gamma=0,colsample_bytree=1,subsample=1)

set.seed(12345)
#set.seed(12498)
#set.seed(12555)
control<-trainControl(method = "cv",number=4,savePredictions = "all",classProbs=TRUE)

xgbm<- train(factor(ACDO_GRAVE)~
ANOMALIA_.0+COM_AUT_.0+COM_AUT_.1+COM_AUT_.2+COM_AUT_.3+COM_AUT_.4+DIA_SEM.1+DIA_SEM.2+DIA_
SEM.3+DIA_SEM.4+DIA_SEM.5+DIA_SEM.6+FACTORES_ATMOS_.1+FACTORES_ATMOS_.2+FACTORES_ATMOS_.3+I
```

```
NFRACC_ALUMBRADO_.1+INFRACC_ALUMBRADO_.2+INFRACC_CARGA_VEH_.1+INFRACC_CARGA_VEH_.2+INFRACC_
COND_.0+INFRACC_COND_.1+INFRACC_COND_.2+INFRACC_COND_.3+INFRACC_VELOCIDAD_.1+INFRACC_VELOCI
DAD_.2+MES_MATRICULA_VEH_.0+MES_MATRICULA_VEH_.1+PROVINCIA_.0+PROVINCIA_.1+PROVINCIA_.2+PRO
VINCIA_.3+PROVINCIA_.4+PROVINCIA_.5+PROVINCIA_.6+PROVINCIA_.7+
RED_CARRETERA.1+RED_CARRETERA.2+RED_CARRETERA.3+RED_CARRETERA.4+TIPO_ACCIDENTE_.1+TIPO_ACCI
DENTE_.2+TIPO_ACCIDENTE_.3+TIPO_INTERSEC.0+TIPO_INTERSEC.1+TIPO_INTERSEC.2+TIPO_INTERSEC.3+
TIPO_VIA_.0+TIPO_VIA_.1+TOT_VEH_IMPLICADOS_+USO_CASCO.1+USO_CASCO.2+USO_CINTURON.1+USO_CINT
URON.2+VISIBILIDAD_RESTRI_.0+VISIBILIDAD_RESTRI_.1,
data=data,method="xgbTree",trControl=control,tuneGrid=xgbmgrid,verbose=FALSE)
xgbm
plot(xgbm)

# IMPORTANCIA DE VARIABLES
varImp(xgbm)
plot(varImp(xgbm))

# Se prueba a ver si mejora sorteando variables:
set.seed(12345)

xgbmgrid<-expand.grid(min_child_weight=c(5),eta=c(0.03),
nrounds=c(800),max_depth=6,gamma=0,colsample_bytree=c(0.3,0.6,1),subsample=1)

control<-trainControl(method = "cv",number=4,savePredictions = "all",classProbs=TRUE)

xgbm<- train(factor(ACDO_GRAVE)~
ANOMALIA_.0+COM_AUT_.0+COM_AUT_.1+COM_AUT_.2+COM_AUT_.3+COM_AUT_.4+DIA_SEM.1+DIA_SEM.2+DIA_
SEM.3+DIA_SEM.4+DIA_SEM.5+DIA_SEM.6+FACTORES_ATMOS_.1+FACTORES_ATMOS_.2+FACTORES_ATMOS_.3+I
NFRACC_ALUMBRADO_.1+INFRACC_ALUMBRADO_.2+INFRACC_CARGA_VEH_.1+INFRACC_CARGA_VEH_.2+INFRACC_
COND_.0+INFRACC_COND_.1+INFRACC_COND_.2+INFRACC_COND_.3+INFRACC_VELOCIDAD_.1+INFRACC_VELOCI
DAD_.2+MES_MATRICULA_VEH_.0+MES_MATRICULA_VEH_.1+PROVINCIA_.0+PROVINCIA_.1+PROVINCIA_.2+PRO
VINCIA_.3+PROVINCIA_.4+PROVINCIA_.5+PROVINCIA_.6+PROVINCIA_.7+
RED_CARRETERA.1+RED_CARRETERA.2+RED_CARRETERA.3+RED_CARRETERA.4+TIPO_ACCIDENTE_.1+TIPO_ACCI
DENTE_.2+TIPO_ACCIDENTE_.3+TIPO_INTERSEC.0+TIPO_INTERSEC.1+TIPO_INTERSEC.2+TIPO_INTERSEC.3+
TIPO_VIA_.0+TIPO_VIA_.1+TOT_VEH_IMPLICADOS_+USO_CASCO.1+USO_CASCO.2+USO_CINTURON.1+USO_CINT
URON.2+VISIBILIDAD_RESTRI_.0+VISIBILIDAD_RESTRI_.1,
data=data,method="xgbTree",trControl=control,tuneGrid=xgbmgrid,verbose=FALSE)
xgbm
plot(xgbm)

# Se prueba a ver si mejora sorteando observaciones:
set.seed(12345)

xgbmgrid<-expand.grid(min_child_weight=c(5),eta=c(0.03),
nrounds=c(800),max_depth=6,gamma=0,colsample_bytree=(0.6),subsample=c(0.3,0.6,1))

control<-trainControl(method = "cv",number=4,savePredictions = "all",classProbs=TRUE)

xgbm<- train(factor(ACDO_GRAVE)~
ANOMALIA_.0+COM_AUT_.0+COM_AUT_.1+COM_AUT_.2+COM_AUT_.3+COM_AUT_.4+DIA_SEM.1+DIA_SEM.2+DIA_
SEM.3+DIA_SEM.4+DIA_SEM.5+DIA_SEM.6+FACTORES_ATMOS_.1+FACTORES_ATMOS_.2+FACTORES_ATMOS_.3+I
NFRACC_ALUMBRADO_.1+INFRACC_ALUMBRADO_.2+INFRACC_CARGA_VEH_.1+INFRACC_CARGA_VEH_.2+INFRACC_
COND_.0+INFRACC_COND_.1+INFRACC_COND_.2+INFRACC_COND_.3+INFRACC_VELOCIDAD_.1+INFRACC_VELOCI
DAD_.2+MES_MATRICULA_VEH_.0+MES_MATRICULA_VEH_.1+PROVINCIA_.0+PROVINCIA_.1+PROVINCIA_.2+PRO
VINCIA_.3+PROVINCIA_.4+PROVINCIA_.5+PROVINCIA_.6+PROVINCIA_.7+
RED_CARRETERA.1+RED_CARRETERA.2+RED_CARRETERA.3+RED_CARRETERA.4+TIPO_ACCIDENTE_.1+TIPO_ACCI
DENTE_.2+TIPO_ACCIDENTE_.3+TIPO_INTERSEC.0+TIPO_INTERSEC.1+TIPO_INTERSEC.2+TIPO_INTERSEC.3+
TIPO_VIA_.0+TIPO_VIA_.1+TOT_VEH_IMPLICADOS_+USO_CASCO.1+USO_CASCO.2+USO_CINTURON.1+USO_CINT
URON.2+VISIBILIDAD_RESTRI_.0+VISIBILIDAD_RESTRI_.1,
data=data,method="xgbTree",trControl=control,tuneGrid=xgbmgrid,verbose=FALSE)
xgbm
plot(xgbm)

# Se prueban valores de gamma:
set.seed(12345)

xgbmgrid<-expand.grid(min_child_weight=c(5),eta=c(0.03),nrounds=c(800),
```

```
max_depth=6,gamma=c(0,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,1),colsample_bytree=(0.6),subsamp
le=(1))

control<-trainControl(method = "cv",number=4,savePredictions = "all",classProbs=TRUE)

xgbm<- train(factor(ACDO_GRAVE)~
ANOMALIA_.0+COM_AUT_.0+COM_AUT_.1+COM_AUT_.2+COM_AUT_.3+COM_AUT_.4+DIA_SEM.1+DIA_SEM.2+DIA_
SEM.3+DIA_SEM.4+DIA_SEM.5+DIA_SEM.6+FACTORES_ATMOS_.1+FACTORES_ATMOS_.2+FACTORES_ATMOS_.3+I
NFRACC_ALUMBRADO_.1+INFRACC_ALUMBRADO_.2+INFRACC_CARGA_VEH_.1+INFRACC_CARGA_VEH_.2+INFRACC_
COND_.0+INFRACC_COND_.1+INFRACC_COND_.2+INFRACC_COND_.3+INFRACC_VELOCIDAD_.1+INFRACC_VELOCI
DAD_.2+MES_MATRICULA_VEH_.0+MES_MATRICULA_VEH_.1+PROVINCIA_.0+PROVINCIA_.1+PROVINCIA_.2+PRO
VINCIA_.3+PROVINCIA_.4+PROVINCIA_.5+PROVINCIA_.6+PROVINCIA_.7+
RED_CARRETERA.1+RED_CARRETERA.2+RED_CARRETERA.3+RED_CARRETERA.4+TIPO_ACCIDENTE_.1+TIPO_ACCI
DENTE_.2+TIPO_ACCIDENTE_.3+TIPO_INTERSEC.0+TIPO_INTERSEC.1+TIPO_INTERSEC.2+TIPO_INTERSEC.3+
TIPO_VIA_.0+TIPO_VIA_.1+TOT_VEH_IMPLICADOS_+USO_CASCO.1+USO_CASCO.2+USO_CINTURON.1+USO_CINT
URON.2+VISIBILIDAD_RESTRI_.0+VISIBILIDAD_RESTRI_.1,
data=data,method="xgbTree",trControl=control,tuneGrid=xgbmgrid,verbose=FALSE)
xgbm
plot(xgbm)

#--------CRUZADA XGBM--------#
# Use Parallel computing:
library(doParallel)
registerDoParallel(cores = detectCores() - 1)

medias1<-cruzadaxgbmbin(data=data, vardep="ACDO_GRAVE",
listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_AUT_.3','COM_AUT_.4',
'DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6','FACTORES_ATMOS_.1'
,'FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRACC_ALUMBRADO_.2','INF
RACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_COND_.1','INFRACC_COND
_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','MES_MATRICULA_VEH_.0'
,'MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','PROVINCIA_.3','PROVIN
CIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1','RED_CARRETERA.2','R
ED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.2','TIPO_ACCIDENTE_.
3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC.3','TIPO_VIA_.0','T
IPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINTURON.1','USO_CINTURO
N.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),listclass=c(""),
grupos=4,sinicio=1234,repe=5,min_child_weight=5,eta=0.03,nrounds=800,max_depth=6,
gamma=0.6,colsample_bytree=0.6,subsample=1)

medias1$modelo="xgbm"

medias2<-cruzadaxgbmbin(data=data, vardep="ACDO_GRAVE",
listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_AUT_.3','COM_AUT_.4',
'DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6','FACTORES_ATMOS_.1'
,'FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRACC_ALUMBRADO_.2','INF
RACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_COND_.1','INFRACC_COND
_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','MES_MATRICULA_VEH_.0'
,'MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','PROVINCIA_.3','PROVIN
CIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1','RED_CARRETERA.2','R
ED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.2','TIPO_ACCIDENTE_.
3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC.3','TIPO_VIA_.0','T
IPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINTURON.1','USO_CINTURO
N.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),listclass=c(""),
grupos=4,sinicio=1234,repe=5,min_child_weight=5,eta=0.03,nrounds=800,max_depth=6,
gamma=1,colsample_bytree=0.6,subsample=1)

medias2$modelo="xgbm2"

medias3<-cruzadaxgbmbin(data=data, vardep="ACDO_GRAVE",
listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_AUT_.3','COM_AUT_.4',
'DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6','FACTORES_ATMOS_.1'
,'FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRACC_ALUMBRADO_.2','INF
RACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_COND_.1','INFRACC_COND
_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','MES_MATRICULA_VEH_.0'
,'MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','PROVINCIA_.3','PROVIN
```

```
CIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1','RED_CARRETERA.2','R
ED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.2','TIPO_ACCIDENTE_.
3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC.3','TIPO_VIA_.0','T
IPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINTURON.1','USO_CINTURO
N.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),listclass=c(""),
grupos=4,sinicio=1234,repe=5,min_child_weight=5,eta=0.1,nrounds=100,max_depth=6,
gamma=1,colsample_bytree=1,subsample=1)

medias3$modelo="xgbm3"

medias4<-cruzadaxgbmbin(data=data, vardep="ACDO_GRAVE",
listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_AUT_.3','COM_AUT_.4',
'DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6','FACTORES_ATMOS_.1'
,'FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRACC_ALUMBRADO_.2','INF
RACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_COND_.1','INFRACC_COND
_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','MES_MATRICULA_VEH_.0'
,'MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','PROVINCIA_.3','PROVIN
CIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1','RED_CARRETERA.2','R
ED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.2','TIPO_ACCIDENTE_.
3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC.3','TIPO_VIA_.0','T
IPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINTURON.1','USO_CINTURO
N.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),listclass=c(""),
grupos=4,sinicio=1234,repe=5,min_child_weight=5,eta=0.05,nrounds=500,max_depth=6,
gamma=1,colsample_bytree=1,subsample=1)

medias4$modelo="xgbm4"

union1<-rbind(medias1,medias2,medias3,medias4)
par(cex.axis=1.3,cex=1)
boxplot(data=union1,tasa~modelo,main="TASA FALLOS",col="orange")
boxplot(data=union1,auc~modelo,main="AUC",col="orange")
```

9.6. Código R Support Vector Machines

```
#--------TUNEO SVM-L--------#
library(dummies)
library(MASS)
library(reshape)
library(caret)
library(dplyr)
library(pROC)
library(randomForest)

# Use Parallel computing:
library(doParallel)
registerDoParallel(cores = detectCores() - 1)

# Se tunea el parámetro C:
set.seed(12345)
SVMgrid<-expand.grid(C=c(0.01,0.05,0.1,0.5,1,2,5,10))

control<-trainControl(method = "cv",number=4,savePredictions = "all")

SVM<-
train(data=data,factor(ACDO_GRAVE)~ANOMALIA_.0+COM_AUT_.0+COM_AUT_.1+COM_AUT_.2+COM_AUT_.3+
COM_AUT_.4+DIA_SEM.1+DIA_SEM.2+DIA_SEM.3+DIA_SEM.4+DIA_SEM.5+DIA_SEM.6+FACTORES_ATMOS_.1+FA
CTORES_ATMOS_.2+FACTORES_ATMOS_.3+INFRACC_ALUMBRADO_.1+INFRACC_ALUMBRADO_.2+INFRACC_CARGA_V
EH_.1+INFRACC_CARGA_VEH_.2+INFRACC_COND_.0+INFRACC_COND_.1+INFRACC_COND_.2+INFRACC_COND_.3+
INFRACC_VELOCIDAD_.1+INFRACC_VELOCIDAD_.2+MES_MATRICULA_VEH_.0+MES_MATRICULA_VEH_.1+PROVINC
IA_.0+PROVINCIA_.1+PROVINCIA_.2+PROVINCIA_.3+PROVINCIA_.4+PROVINCIA_.5+PROVINCIA_.6+PROVINC
IA_.7+RED_CARRETERA.1+RED_CARRETERA.2+RED_CARRETERA.3+RED_CARRETERA.4+TIPO_ACCIDENTE_.1+TIP
O_ACCIDENTE_.2+TIPO_ACCIDENTE_.3+TIPO_INTERSEC.0+TIPO_INTERSEC.1+TIPO_INTERSEC.2+TIPO_INTER
SEC.3+TIPO_VIA_.0+TIPO_VIA_.1+TOT_VEH_IMPLICADOS_+USO_CASCO.1+USO_CASCO.2+USO_CINTURON.1+US
O_CINTURON.2+VISIBILIDAD_RESTRI_.0+VISIBILIDAD_RESTRI_.1,
method="svmLinear",trControl=control,
tuneGrid=SVMgrid,verbose=FALSE)
```

```
SVM$results
plot(SVM$results$C,SVM$results$Accuracy)

#--------CRUZADA SVM-L--------#
# Use Parallel computing:
registerDoParallel(cores = detectCores() - 1)

medias1<-cruzadaSVMbin(data=data, vardep="ACDO_GRAVE",
listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_AUT_.3','COM_AUT_.4',
'DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6','FACTORES_ATMOS_.1'
,'FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRACC_ALUMBRADO_.2','INF
RACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_COND_.1','INFRACC_COND
_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','MES_MATRICULA_VEH_.0'
,'MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','PROVINCIA_.3','PROVIN
CIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1','RED_CARRETERA.2','R
ED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.2','TIPO_ACCIDENTE_.
3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC.3','TIPO_VIA_.0','T
IPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINTURON.1','USO_CINTURO
N.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),listclass=c(""),
grupos=4,sinicio=1234,repe=5,C=0.1)

medias1$modelo="SVM-L1"

medias2<-cruzadaSVMbin(data=data, vardep="ACDO_GRAVE",

listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_AUT_.3','COM_AUT_.4',
'DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6','FACTORES_ATMOS_.1'
,'FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRACC_ALUMBRADO_.2','INF
RACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_COND_.1','INFRACC_COND
_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','MES_MATRICULA_VEH_.0'
,'MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','PROVINCIA_.3','PROVIN
CIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1','RED_CARRETERA.2','R
ED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.2','TIPO_ACCIDENTE_.
3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC.3','TIPO_VIA_.0','T
IPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINTURON.1','USO_CINTURO
N.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),listclass=c(""),
grupos=4,sinicio=1234,repe=5,C=0.5)

medias2$modelo="SVM-L2"

medias3<-cruzadaSVMbin(data=data, vardep="ACDO_GRAVE",
listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_AUT_.3','COM_AUT_.4',
'DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6','FACTORES_ATMOS_.1'
,'FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRACC_ALUMBRADO_.2','INF
RACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_COND_.1','INFRACC_COND
_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','MES_MATRICULA_VEH_.0'
,'MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','PROVINCIA_.3','PROVIN
CIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1','RED_CARRETERA.2','R
ED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.2','TIPO_ACCIDENTE_.
3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC.3','TIPO_VIA_.0','T
IPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINTURON.1','USO_CINTURO
N.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),listclass=c(""),
grupos=4,sinicio=1234,repe=5,C=1)

medias3$modelo="SVM-L3"

medias4<-cruzadaSVMbin(data=data, vardep="ACDO_GRAVE",

listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_AUT_.3','COM_AUT_.4',
'DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6','FACTORES_ATMOS_.1'
,'FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRACC_ALUMBRADO_.2','INF
RACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_COND_.1','INFRACC_COND
_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','MES_MATRICULA_VEH_.0'
,'MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','PROVINCIA_.3','PROVIN
CIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1','RED_CARRETERA.2','R
```

```
ED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.2','TIPO_ACCIDENTE_.
3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC.3','TIPO_VIA_.0','T
IPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINTURON.1','USO_CINTURO
N.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),listclass=c(""),
grupos=4,sinicio=1234,repe=5,C=2)

medias4$modelo="SVM-L4"

union1<-rbind(medias1,medias2,medias3,medias4,medias5)

par(cex.axis=1.2)
boxplot(data=union1,tasa~modelo,main="TASA FALLOS",col="orange")
boxplot(data=union1,auc~modelo,main="AUC",col="orange")


#--------TUNEO SVM-P--------#
library(doParallel)
registerDoParallel(cores = detectCores() - 1)

#Se puede tunear C, degree y scale:
SVMgrid<-expand.grid(C=c(0.01,0.05,0.1,0.2,0.5,1,2,5,10),
                     degree=c(2,3),scale=c(0.1,0.5,1,2,5))

control<-trainControl(method = "cv",
                      number=4,savePredictions = "all")

SVM<- train(data=data,factor(ACDO_GRAVE)~
ANOMALIA_.0+COM_AUT_.0+COM_AUT_.1+COM_AUT_.2+COM_AUT_.3+COM_AUT_.4+DIA_SEM.1+DIA_SEM.2+DIA_
SEM.3+DIA_SEM.4+DIA_SEM.5+DIA_SEM.6+FACTORES_ATMOS_.1+FACTORES_ATMOS_.2+FACTORES_ATMOS_.3+I
NFRACC_ALUMBRADO_.1+INFRACC_ALUMBRADO_.2+INFRACC_CARGA_VEH_.1+INFRACC_CARGA_VEH_.2+INFRACC_
COND_.0+INFRACC_COND_.1+INFRACC_COND_.2+INFRACC_COND_.3+INFRACC_VELOCIDAD_.1+INFRACC_VELOCI
DAD_.2+MES_MATRICULA_VEH_.0+MES_MATRICULA_VEH_.1+PROVINCIA_.0+PROVINCIA_.1+PROVINCIA_.2+PRO
VINCIA_.3+PROVINCIA_.4+PROVINCIA_.5+PROVINCIA_.6+PROVINCIA_.7+RED_CARRETERA.1+RED_CARRETERA
.2+RED_CARRETERA.3+RED_CARRETERA.4+TIPO_ACCIDENTE_.1+TIPO_ACCIDENTE_.2+TIPO_ACCIDENTE_.3+TI
PO_INTERSEC.0+TIPO_INTERSEC.1+TIPO_INTERSEC.2+TIPO_INTERSEC.3+TIPO_VIA_.0+TIPO_VIA_.1+TOT_V
EH_IMPLICADOS_+USO_CASCO.1+USO_CASCO.2+USO_CINTURON.1+USO_CINTURON.2+VISIBILIDAD_RESTRI_.0+
VISIBILIDAD_RESTRI_.1,method="svmPoly",trControl=control,tuneGrid=SVMgrid,verbose=FALSE)

SVM
SVM$results

dat<-as.data.frame(SVM$results)
library(ggplot2)

# PLOT DE DOS VARIABLES CATEGÓRICAS, UNA CONTINUA
ggplot(dat, aes(x=factor(C), y=Accuracy,
color=factor(degree),pch=factor(scale))) +
geom_point(position=position_dodge(width=0.5),size=3)

#--------CRUZADA SVM-P--------#
registerDoParallel(cores = detectCores() - 1)

medias20<-cruzadaSVMbinPoly(data=data, vardep="ACDO_GRAVE",
listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_AUT_.3','COM_AUT_.4',
'DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6','FACTORES_ATMOS_.1'
,'FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRACC_ALUMBRADO_.2','INF
RACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_COND_.1','INFRACC_COND
_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','MES_MATRICULA_VEH_.0'
,'MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','PROVINCIA_.3','PROVIN
CIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1','RED_CARRETERA.2','R
ED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.2','TIPO_ACCIDENTE_.
3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC.3','TIPO_VIA_.0','T
IPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINTURON.1','USO_CINTURO
N.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),
listclass=c(""),grupos=4,sinicio=1234,repe=5,C=0.1,degree=2,scale=0.1)
```

```
medias20$modelo="SVMPoly1"

medias21<-cruzadaSVMbinPoly(data=data, vardep="ACDO_GRAVE",
listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_AUT_.3','COM_AUT_.4',
'DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6','FACTORES_ATMOS_.1'
,'FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRACC_ALUMBRADO_.2','INF
RACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_COND_.1','INFRACC_COND
_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','MES_MATRICULA_VEH_.0'
,'MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','PROVINCIA_.3','PROVIN
CIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1','RED_CARRETERA.2','R
ED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.2','TIPO_ACCIDENTE_.
3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC.3','TIPO_VIA_.0','T
IPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINTURON.1','USO_CINTURO
N.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),
listclass=c(""),grupos=4,sinicio=1234,repe=5,C=0.05,degree=2,scale=0.1)

medias21$modelo="SVMPoly2"

medias22<-cruzadaSVMbinPoly(data=data, vardep="ACDO_GRAVE",
listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_AUT_.3','COM_AUT_.4',
'DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6','FACTORES_ATMOS_.1'
,'FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRACC_ALUMBRADO_.2','INF
RACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_COND_.1','INFRACC_COND
_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','MES_MATRICULA_VEH_.0'
,'MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','PROVINCIA_.3','PROVIN
CIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1','RED_CARRETERA.2','R
ED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.2','TIPO_ACCIDENTE_.
3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC.3','TIPO_VIA_.0','T
IPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINTURON.1','USO_CINTURO
N.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),
listclass=c(""),grupos=4,sinicio=1234,repe=5,C=0.01,degree=2,scale=0.1)

medias22$modelo="SVMPoly3"

medias23<-cruzadaSVMbinPoly(data=data, vardep="ACDO_GRAVE",
listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_AUT_.3','COM_AUT_.4',
'DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6','FACTORES_ATMOS_.1'
,'FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRACC_ALUMBRADO_.2','INF
RACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_COND_.1','INFRACC_COND
_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','MES_MATRICULA_VEH_.0'
,'MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','PROVINCIA_.3','PROVIN
CIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1','RED_CARRETERA.2','R
ED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.2','TIPO_ACCIDENTE_.
3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC.3','TIPO_VIA_.0','T
IPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINTURON.1','USO_CINTURO
N.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),
listclass=c(""),grupos=4,sinicio=1234,repe=5,C=0.01,degree=3,scale=0.05)

medias23$modelo="SVMPoly4"

union1<-rbind(medias20,medias21,medias22,medias23)

par(cex.axis=1.2)
boxplot(data=union1,tasa~modelo,main="TASA FALLOS",col="orange")
boxplot(data=union1,auc~modelo,main="AUC",col="orange")

#--------TUNEO SVM-RBF--------#
library(doParallel)
registerDoParallel(cores = detectCores() - 1)

#Se puede tunear los parámetros C y Sigma:
set.seed(12345)

SVMgrid<-expand.grid(C=c(0.01,0.05,0.1,0.2,0.5,1,2,5,10,30),
                     sigma=c(0.01,0.05,0.1,0.2,0.5,1,2,5,10,30))
```

```
control<-trainControl(method = "cv",
                      number=4,savePredictions = "all")

SVM<- train(data=data,factor(ACDO_GRAVE)~
ANOMALIA_.0+COM_AUT_.0+COM_AUT_.1+COM_AUT_.2+COM_AUT_.3+COM_AUT_.4+DIA_SEM.1+DIA_SEM.2+DIA_
SEM.3+DIA_SEM.4+DIA_SEM.5+DIA_SEM.6+FACTORES_ATMOS_.1+FACTORES_ATMOS_.2+FACTORES_ATMOS_.3+I
NFRACC_ALUMBRADO_.1+INFRACC_ALUMBRADO_.2+INFRACC_CARGA_VEH_.1+INFRACC_CARGA_VEH_.2+INFRACC_
COND_.0+INFRACC_COND_.1+INFRACC_COND_.2+INFRACC_COND_.3+INFRACC_VELOCIDAD_.1+INFRACC_VELOCI
DAD_.2+MES_MATRICULA_VEH_.0+MES_MATRICULA_VEH_.1+PROVINCIA_.0+PROVINCIA_.1+PROVINCIA_.2+PRO
VINCIA_.3+PROVINCIA_.4+PROVINCIA_.5+PROVINCIA_.6+PROVINCIA_.7+RED_CARRETERA.1+RED_CARRETERA
.2+RED_CARRETERA.3+RED_CARRETERA.4+TIPO_ACCIDENTE_.1+TIPO_ACCIDENTE_.2+TIPO_ACCIDENTE_.3+TI
PO_INTERSEC.0+TIPO_INTERSEC.1+TIPO_INTERSEC.2+TIPO_INTERSEC.3+TIPO_VIA_.0+TIPO_VIA_.1+TOT_V
EH_IMPLICADOS_+USO_CASCO.1+USO_CASCO.2+USO_CINTURON.1+USO_CINTURON.2+VISIBILIDAD_RESTRI_.0+
VISIBILIDAD_RESTRI_.1,method="svmRadial",trControl=control,tuneGrid=SVMgrid,verbose=FALSE)

SVM

#Realizamos un plot para ver la relación entre C y sigma:
dat<-as.data.frame(SVM$results)

ggplot(dat, aes(x=factor(C), y=Accuracy,
color=factor(sigma)))+
geom_point(position=position_dodge(width=0.5),size=3)

#--------CRUZADA SVM-RBF--------#
registerDoParallel(cores = detectCores() - 1)

medias30<-cruzadaSVMbinRBF(data=data, vardep="ACDO_GRAVE",
listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_AUT_.3','COM_AUT_.4',
'DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6','FACTORES_ATMOS_.1'
,'FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRACC_ALUMBRADO_.2','INF
RACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_COND_.1','INFRACC_COND
_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','MES_MATRICULA_VEH_.0'
,'MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','PROVINCIA_.3','PROVIN
CIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1','RED_CARRETERA.2','R
ED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.2','TIPO_ACCIDENTE_.
3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC.3','TIPO_VIA_.0','T
IPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINTURON.1','USO_CINTURO
N.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),
listclass=c(""),grupos=4,sinicio=1234,repe=5,C=1,sigma=0.01)

medias30$modelo="SVMRBF1"

medias31<-cruzadaSVMbinRBF(data=data, vardep="ACDO_GRAVE",
listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_AUT_.3','COM_AUT_.4',
'DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6','FACTORES_ATMOS_.1'
,'FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRACC_ALUMBRADO_.2','INF
RACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_COND_.1','INFRACC_COND
_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','MES_MATRICULA_VEH_.0'
,'MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','PROVINCIA_.3','PROVIN
CIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1','RED_CARRETERA.2','R
ED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.2','TIPO_ACCIDENTE_.
3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC.3','TIPO_VIA_.0','T
IPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINTURON.1','USO_CINTURO
N.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),
listclass=c(""),grupos=4,sinicio=1234,repe=5,C=2,sigma=0.01)

medias31$modelo="SVMRBF2"

medias32<-cruzadaSVMbinRBF(data=data, vardep="ACDO_GRAVE",
listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_AUT_.3','COM_AUT_.4',
'DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6','FACTORES_ATMOS_.1'
,'FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRACC_ALUMBRADO_.2','INF
RACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_COND_.1','INFRACC_COND
_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','MES_MATRICULA_VEH_.0'
,'MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','PROVINCIA_.3','PROVIN
```

```
CIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1','RED_CARRETERA.2','R
ED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.2','TIPO_ACCIDENTE_.
3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC.3','TIPO_VIA_.0','T
IPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINTURON.1','USO_CINTURO
N.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),
listclass=c(""),grupos=4,sinicio=1234,repe=5,C=5,sigma=0.01)

medias32$modelo="SVMRBF3"

medias33<-cruzadaSVMbinRBF(data=data, vardep="ACDO_GRAVE",
listconti=c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_AUT_.3','COM_AUT_.4',
'DIA_SEM.1','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6','FACTORES_ATMOS_.1'
,'FACTORES_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRACC_ALUMBRADO_.2','INF
RACC_CARGA_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_COND_.1','INFRACC_COND
_.2','INFRACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','MES_MATRICULA_VEH_.0'
,'MES_MATRICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','PROVINCIA_.3','PROVIN
CIA_.4','PROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1','RED_CARRETERA.2','R
ED_CARRETERA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.2','TIPO_ACCIDENTE_.
3','TIPO_INTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC.3','TIPO_VIA_.0','T
IPO_VIA_.1','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINTURON.1','USO_CINTURO
N.2','VISIBILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1'),
listclass=c(""),grupos=4,sinicio=1234,repe=5,C=10,sigma=0.01)

medias33$modelo="SVMRBF4"

union<-rbind(medias30,medias31,medias32,medias33)

par(cex.axis=1.2)
boxplot(data=union,tasa~modelo,main="TASA FALLOS",col="orange")
boxplot(data=union,auc~modelo,main="AUC",col="orange")
```

9.7. Código R comparación de modelos

```
#--------COMPARACIÓN DE MODELOS Y ENSAMBLADO--------#
library(dummies)
library(MASS)
library(reshape)
library(caret)
library(dplyr)
library(pROC)
library(randomForest)
library(doParallel)

vardep<-"ACDO_GRAVE"
listconti<-
c('ANOMALIA_.0','COM_AUT_.0','COM_AUT_.1','COM_AUT_.2','COM_AUT_.3','COM_AUT_.4','DIA_SEM.1
','DIA_SEM.2','DIA_SEM.3','DIA_SEM.4','DIA_SEM.5','DIA_SEM.6','FACTORES_ATMOS_.1','FACTORES
_ATMOS_.2','FACTORES_ATMOS_.3','INFRACC_ALUMBRADO_.1','INFRACC_ALUMBRADO_.2','INFRACC_CARGA
_VEH_.1','INFRACC_CARGA_VEH_.2','INFRACC_COND_.0','INFRACC_COND_.1','INFRACC_COND_.2','INFR
ACC_COND_.3','INFRACC_VELOCIDAD_.1','INFRACC_VELOCIDAD_.2','MES_MATRICULA_VEH_.0','MES_MATR
ICULA_VEH_.1','PROVINCIA_.0','PROVINCIA_.1','PROVINCIA_.2','PROVINCIA_.3','PROVINCIA_.4','P
ROVINCIA_.5','PROVINCIA_.6','PROVINCIA_.7','RED_CARRETERA.1','RED_CARRETERA.2','RED_CARRETE
RA.3','RED_CARRETERA.4','TIPO_ACCIDENTE_.1','TIPO_ACCIDENTE_.2','TIPO_ACCIDENTE_.3','TIPO_I
NTERSEC.0','TIPO_INTERSEC.1','TIPO_INTERSEC.2','TIPO_INTERSEC.3','TIPO_VIA_.0','TIPO_VIA_.1
','TOT_VEH_IMPLICADOS_','USO_CASCO.1','USO_CASCO.2','USO_CINTURON.1','USO_CINTURON.2','VISI
BILIDAD_RESTRI_.0','VISIBILIDAD_RESTRI_.1')
listclass<-c("")
grupos<-4 #Número de grupos de CV
sinicio<-1234
repe<-10 #Número de repeiciones de CV

#Regresión Logística:
medias1<-cruzadalogistica(data=data,
                          vardep=vardep,listconti=listconti,
                          listclass=listclass,grupos=grupos,sinicio=sinicio,repe=repe)
```

```
medias1bis<-as.data.frame(medias1[1])
medias1bis$modelo<-"Logistica"
predi1<-as.data.frame(medias1[2])
predi1$logi<-predi1$Yes
#Matriz de confusión:
confusionMatrix(medias1[[2]][["pred"]],medias1[[2]][["obs"]],"Yes")

#Red Neuronal:
registerDoParallel(cores = detectCores() - 1)
medias2<-cruzadaavnnetbin(data=data,
                          vardep=vardep,listconti=listconti,
                          listclass=listclass,grupos=grupos,sinicio=sinicio,repe=repe,
                          size=c(6),decay=c(0.1),repeticiones=repe,itera=200)

medias2bis<-as.data.frame(medias2[1])
medias2bis$modelo<-"Avnnet"
predi2<-as.data.frame(medias2[2])
predi2$avnnet<-predi2$Yes
#Matriz de confusión:
confusionMatrix(medias2[[2]][["pred"]],medias2[[2]][["obs"]],"Yes")


#Random Forest:
registerDoParallel(cores = detectCores() - 1)
medias3<-cruzadarfbin(data=data,
                      vardep=vardep,listconti=listconti,
                      listclass=listclass,grupos=grupos,sinicio=sinicio,repe=repe,
                      mtry=30,ntree=500,nodesize=5,sampsize=7000,replace=TRUE)

medias3bis<-as.data.frame(medias3[1])
medias3bis$modelo<-"RF"
predi3<-as.data.frame(medias3[2])
predi3$rf<-predi3$Yes
#Matriz de confusión:
confusionMatrix(medias3[[2]][["pred"]],medias3[[2]][["obs"]],"Yes")

#Gradient Boosting:
registerDoParallel(cores = detectCores() - 1)

medias4<-cruzadagbmbin(data=data,
                       vardep=vardep,listconti=listconti,
                       listclass=listclass,grupos=grupos,sinicio=sinicio,repe=repe,
                       n.minobsinnode=15,shrinkage=0.1,n.trees=1000,interaction.depth=2)

medias4bis<-as.data.frame(medias4[1])
medias4bis$modelo<-"GBM"
predi4<-as.data.frame(medias4[2])
predi4$gbm<-predi4$Yes
#Matriz de confusión:
confusionMatrix(medias4[[2]][["pred"]],medias4[[2]][["obs"]],"Yes")

#XGBoost:
registerDoParallel(cores = detectCores() - 1)
medias5<-cruzadaxgbmbin(data=data,
                        vardep=vardep,listconti=listconti,
                        listclass=listclass,grupos=grupos,sinicio=sinicio,repe=repe,
                        min_child_weight=5,eta=0.03,nrounds=800,max_depth=6,
                        gamma=1,colsample_bytree=1,subsample=1,
                        alpha=0,lambda=0,lambda_bias=0)

medias5bis<-as.data.frame(medias5[1])
medias5bis$modelo<-"XGBM"
predi5<-as.data.frame(medias5[2])
predi5$xgbm<-predi5$Yes
#Matriz de confusión:
confusionMatrix(medias5[[2]][["pred"]],medias5[[2]][["obs"]],"Yes")
```

```
#SVM-Lineal:
registerDoParallel(cores = detectCores() - 1)
medias6<-cruzadaSVMbin(data=data,
                       vardep=vardep,listconti=listconti,
                       listclass=listclass,grupos=grupos,
                       sinicio=sinicio,repe=repe,C=1)

medias6bis<-as.data.frame(medias6[1])
medias6bis$modelo<-"SVM-Linear"
predi6<-as.data.frame(medias6[2])
predi6$svmLinear<-predi6$Yes
#Matriz de confusión:
confusionMatrix(medias6[[2]][["pred"]],medias6[[2]][["obs"]],"Yes")

#SVM-Polinomial:
registerDoParallel(cores = detectCores() - 1)

medias7<-cruzadaSVMbinPoly(data=data,
                           vardep=vardep,listconti=listconti,
                           listclass=listclass,grupos=grupos,sinicio=sinicio,repe=repe,
                           C=0.05,degree=2,scale=0.1)

medias7bis<-as.data.frame(medias7[1])
medias7bis$modelo<-"SVM-Poly"
predi7<-as.data.frame(medias7[2])
predi7$svmPoly<-predi7$Yes
#Matriz de confusión:
confusionMatrix(medias7[[2]][["pred"]],medias7[[2]][["obs"]],"Yes")

#SVM-RBF:
registerDoParallel(cores = detectCores() - 1)
medias8<-cruzadaSVMbinRBF(data=data,
                          vardep=vardep,listconti=listconti,
                          listclass=listclass,grupos=grupos,
                          sinicio=sinicio,repe=repe,
                          C=2,sigma=0.01)

medias8bis<-as.data.frame(medias8[1])
medias8bis$modelo<-"SVM-Radial"
predi8<-as.data.frame(medias8[2])
predi8$svmRadial<-predi8$Yes
#Matriz de confusión:
confusionMatrix(medias8[[2]][["pred"]],medias8[[2]][["obs"]],"Yes")

union1<-rbind(medias1bis,medias2bis,medias3bis,medias4bis,medias5bis,medias6bis,medias7bis,
medias8bis)

par(cex.axis=0.9)
boxplot(data=union1,tasa~modelo,col="orange",main='TASA FALLOS')
boxplot(data=union1,auc~modelo,col="orange",main='AUC')

uni<-union1
uni$modelo <- with(uni,reorder(modelo,tasa, median))
par(cex.axis=0.9,las=1)
boxplot(data=uni,tasa~modelo,col="orange",main="TASA FALLOS")

uni<-union1
uni$modelo <- with(uni,reorder(modelo,auc, median))
par(cex.axis=0.9,las=1)
boxplot(data=uni,auc~modelo,col="orange",main="AUC")
```

9.8. Código R ensamblado de modelos

```
# Ensamblado de modelos:
# Es necesario unir por columnas todos los archivos predi para hacer el ensamblado:
```

```
unipredi<-cbind(predi1,predi2,predi3,predi4,predi5,predi6,predi7,predi8)

# Se eliminan columnas duplicadas
unipredi<- unipredi[, !duplicated(colnames(unipredi))]

# Construccion de ensamblados:
unipredi$predi10<-(unipredi$xgbm+unipredi$avnnet)/2
unipredi$predi11<-(unipredi$xgbm+unipredi$gbm)/2
unipredi$predi12<-(unipredi$xgbm+unipredi$logi)/2
unipredi$predi13<-(unipredi$xgbm+unipredi$svmRadial)/2
unipredi$predi14<-(unipredi$xgbm+unipredi$svmPoly)/2
unipredi$predi15<-(unipredi$xgbm+unipredi$svmLinear)/2
unipredi$predi16<-(unipredi$xgbm+unipredi$rf)/2
unipredi$predi17<-(unipredi$avnnet+unipredi$gbm)/2
unipredi$predi18<-(unipredi$avnnet+unipredi$logi)/2
unipredi$predi19<-(unipredi$avnnet+unipredi$svmRadial)/2
unipredi$predi20<-(unipredi$avnnet+unipredi$svmPoly)/2
unipredi$predi21<-(unipredi$avnnet+unipredi$svmLinear)/2
unipredi$predi22<-(unipredi$avnnet+unipredi$rf)/2
unipredi$predi23<-(unipredi$gbm+unipredi$logi)/2
unipredi$predi24<-(unipredi$gbm+unipredi$svmRadial)/2
unipredi$predi25<-(unipredi$gbm+unipredi$svmPoly)/2
unipredi$predi26<-(unipredi$gbm+unipredi$svmLinear)/2
unipredi$predi27<-(unipredi$gbm+unipredi$rf)/2
unipredi$predi28<-(unipredi$logi+unipredi$svmRadial)/2
unipredi$predi29<-(unipredi$logi+unipredi$svmPoly)/2
unipredi$predi30<-(unipredi$logi+unipredi$svmLinear)/2
unipredi$predi31<-(unipredi$logi+unipredi$rf)/2

unipredi$predi32<-(unipredi$xgbm+unipredi$avnnet+unipredi$gbm)/3
unipredi$predi33<-(unipredi$xgbm+unipredi$avnnet+unipredi$logi)/3
unipredi$predi34<-(unipredi$xgbm+unipredi$avnnet+unipredi$svmRadial)/3
unipredi$predi35<-(unipredi$xgbm+unipredi$avnnet+unipredi$svmPoly)/3
unipredi$predi36<-(unipredi$xgbm+unipredi$avnnet+unipredi$svmLinear)/3
unipredi$predi37<-(unipredi$xgbm+unipredi$avnnet+unipredi$rf)/3
unipredi$predi38<-(unipredi$avnnet+unipredi$gbm+unipredi$logi)/3
unipredi$predi39<-(unipredi$avnnet+unipredi$gbm+unipredi$svmRadial)/3
unipredi$predi40<-(unipredi$avnnet+unipredi$gbm+unipredi$svmPoly)/3
unipredi$predi41<-(unipredi$avnnet+unipredi$gbm+unipredi$svmLinear)/3
unipredi$predi42<-(unipredi$avnnet+unipredi$gbm+unipredi$rf)/3
unipredi$predi43<-(unipredi$gbm+unipredi$logi+unipredi$svmRadial)/3
unipredi$predi44<-(unipredi$gbm+unipredi$logi+unipredi$svmPoly)/3
unipredi$predi45<-(unipredi$gbm+unipredi$logi+unipredi$svmLinear)/3
unipredi$predi46<-(unipredi$gbm+unipredi$logi+unipredi$rf)/3

unipredi$predi47<-(unipredi$xgbm+unipredi$avnnet+unipredi$gbm+unipredi$logi)/4
unipredi$predi48<-(unipredi$xgbm+unipredi$avnnet+unipredi$gbm+unipredi$svmRadial)/4
unipredi$predi49<-(unipredi$xgbm+unipredi$avnnet+unipredi$gbm+unipredi$svmPoly)/4
unipredi$predi50<-(unipredi$xgbm+unipredi$avnnet+unipredi$gbm+unipredi$svmLinear)/4
unipredi$predi51<-(unipredi$xgbm+unipredi$avnnet+unipredi$gbm+unipredi$rf)/4
unipredi$predi52<-(unipredi$avnnet+unipredi$gbm+unipredi$logi+unipredi$svmRadial)/4
unipredi$predi53<-(unipredi$avnnet+unipredi$gbm+unipredi$logi+unipredi$svmPoly)/4
unipredi$predi54<-(unipredi$avnnet+unipredi$gbm+unipredi$logi+unipredi$svmLinear)/4
unipredi$predi55<-(unipredi$avnnet+unipredi$gbm+unipredi$logi+unipredi$rf)/4
unipredi$predi56<-(unipredi$gbm+unipredi$logi+unipredi$svmRadial+unipredi$svmPoly)/4
unipredi$predi57<-(unipredi$gbm+unipredi$logi+unipredi$svmRadial+unipredi$svmLinear)/4
unipredi$predi58<-(unipredi$gbm+unipredi$logi+unipredi$svmRadial+unipredi$rf)/4

unipredi$predi59<-
(unipredi$xgbm+unipredi$avnnet+unipredi$gbm+unipredi$logi+unipredi$svmRadial)/5
unipredi$predi60<-
(unipredi$xgbm+unipredi$avnnet+unipredi$gbm+unipredi$logi+unipredi$svmPoly)/5
unipredi$predi61<-
(unipredi$xgbm+unipredi$avnnet+unipredi$gbm+unipredi$logi+unipredi$svmLinear)/5
unipredi$predi62<-(unipredi$xgbm+unipredi$avnnet+unipredi$gbm+unipredi$logi+unipredi$rf)/5
```

```
unipredi$predi63<-
(unipredi$xgbm+unipredi$avnnet+unipredi$gbm+unipredi$logi+unipredi$svmRadial+unipredi$svmPo
ly+unipredi$svmLinear+unipredi$rf)/8

listado<-c(  "logi", "avnnet", "rf",  "gbm", "xgbm", "svmLinear", "svmPoly",
             "svmRadial", "predi10", "predi11",
             "predi12", "predi13", "predi14", "predi15", "predi16", "predi17",
             "predi18", "predi19", "predi20", "predi21", "predi22", "predi23",
             "predi24", "predi25", "predi26", "predi27", "predi28", "predi29",
             "predi30", "predi31", "predi32", "predi33", "predi34", "predi35",
             "predi36", "predi37", "predi38", "predi39", "predi40", "predi41",
             "predi42", "predi43", "predi44", "predi45", "predi46", "predi47",
             "predi48", "predi49", "predi50", "predi51", "predi52", "predi53",
             "predi54", "predi55", "predi56", "predi57", "predi58", "predi59",
             "predi60", "predi61", "predi62", "predi63")

tasafallos<-function(x,y) {
  confu<-confusionMatrix(x,y)
  tasa<-confu[[3]][1]
  return(tasa)
}

auc<-function(x,y) {
  curvaroc<-roc(response=x,predictor=y)
  auc<-curvaroc$auc
  return(auc)
}

# Se obtiene el numero de repeticiones CV y se calculan las medias por repetición:
repeticiones<-nlevels(factor(unipredi$Rep))
unipredi$Rep<-as.factor(unipredi$Rep)
unipredi$Rep<-as.numeric(unipredi$Rep)

medias0<-data.frame(c())
for (prediccion in listado)
{
  unipredi$proba<-unipredi[,prediccion]
  unipredi[,prediccion]<-ifelse(unipredi[,prediccion]>0.5,"Yes","No")
  for (repe in 1:repeticiones)
  {
    paso <- unipredi[(unipredi$Rep==repe),]
    pre<-factor(paso[,prediccion])
    archi<-paso[,c("proba","obs")]
    archi<-archi[order(archi$proba),]
    obs<-paso[,c("obs")]
    tasa=1-tasafallos(pre,obs)
    t<-as.data.frame(tasa)
    t$modelo<-prediccion
    auc<-auc(archi$obs,archi$proba)
    t$auc<-auc
    medias0<-rbind(medias0,t)
  }
}
#El objeto medias0 contiene las medias para los diferentes modelos

#Boxplot tasa de fallos:

par(cex.axis=0.5,las=2)
boxplot(data=medias0,tasa~modelo,col="orange",main="TASA FALLOS")

#Boxplot tasa de fallos:

boxplot(data=medias0,auc~modelo,col="orange",main="AUC")

#Se ordena por la tasa de fallo:
```

```
tablamedias<-medias0 %>%
  group_by(modelo) %>%
  summarize(tasa=mean(tasa))

tablamedias<-tablamedias[order(tablamedias$tasa),]

# PARA EL GRAFICO
medias0$modelo <- with(medias0,
                       reorder(modelo,tasa, mean))
par(cex.axis=1,las=2)
boxplot(data=medias0,tasa~modelo,col="orange", main='TASA FALLOS')

#Se ordena por AUC:
tablamedias2<-medias0 %>%
  group_by(modelo) %>%
  summarize(auc=mean(auc))

tablamedias2<-tablamedias2[order(-tablamedias2$auc),]

# PARA EL GRAFICO
medias0$modelo <- with(medias0,
                       reorder(modelo,auc, mean))
par(cex.axis=1,las=2)
boxplot(data=medias0,auc~modelo,col="orange", main='AUC')

# Se seleccionan los mejores modelos:

listadobis<-c("xgbm", "predi10", "predi48",  "predi32","predi11", "predi34",
              "predi51", "predi59", "predi39", "predi47")

medias0$modelo<-as.character(medias0$modelo)

mediasver<-medias0[medias0$modelo %in% listadobis,]

mediasver$modelo <- with(mediasver,
                         reorder(modelo,auc, median))

par(cex.axis=1.2,las=1)
boxplot(data=mediasver,auc~modelo,col="orange",main='AUC')

mediasver$modelo <- with(mediasver,
                         reorder(modelo,tasa, median))
boxplot(data=mediasver,tasa~modelo,col="orange", main='TASA FALLOS')
```