Trabajo Fin de Máster

Máster en Ciencia de Datos e Ingeniería de Datos en la Nube

Aplicación de Ciencia de Datos en la Optimización de modelos predictivos para la estimación de mortalidad en Ictus Isquémico

Autor: Luis Téllez Ramírez

Tutor: Jesús Martínez Gómez

Cotutor: Juan Manuel García Torrecillas

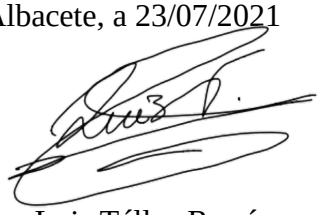
Julio, 2021

Dedicado a mi familia.

Declaración de Autoría

Yo, LUIS TELLEZ RAMIREZ con DNI 77440644N, declaro que soy el único autor del trabajo fin de máster titulado "**Aplicación de Ciencia de Datos en la Optimización de modelos predictivos para la estimación de mortalidad en Ictus Isquémico**" y que el citado trabajo no infringe las leyes en vigor sobre propiedad intelectual y que todo el material no original contenido en dicho trabajo está apropiadamente atribuido a sus legítimos autores.

Albacete, a 23/07/2021



Fdo: Luis Téllez Ramírez

Este trabajo se ha desarrollado en el seno del proyecto de investigación AP-0013-2020-C1-F2, que lleva por título "Optimización de un modelo predictivo para la mortalidad en ictus isquémico a través de una cohorte de Real World Data". Dicho proyecto se enmarca en la Convocatoria 2020 de Proyectos de Investigación de Innovación en el Ambito de la Atención Primaria, Hospitales Comarcales y Centros de Alta Resolución de Andalucía, en el seno de la Consejería de Salud y Familias, y bajo el PAIDI 2020.

Resumen

El propósito de este trabajo no es otro que la aplicación de algunas de las técnicas vistas a lo largo del máster de cara a trabajar en un ámbito donde puede haber una gran conexión, como es la medicina, y en particular, el desarrollo de modelos para estimar la gravedad de un paciente.

A su vez, se trabaja con una base de datos extraída del CMBD, completamente anonimizada, donde los pacientes hayan sufrido un ictus isquémico en alguno de los diagnósticos. Se pretende entonces desarrollar modelos alternativos tanto de Machine Learning como Deep Learning, aplicar algoritmos de aprendizaje supervisado y no supervisado, técnicas de resumen de información, de selección de variables y también de explicabilidad e interpretabilidad de modelos para comprobar si estos realizan predicciones de manera coherente con el criterio experto.

Finalmente, se pretende implementar estos modelos desarrollados en la plataforma AWS para poder utilizarlos y validarlos con nuevos datos externos de cara a su recalibración y mejora en un futuro.

Índice general

Índice de figuras

Capítulo 1

Introducción

1.1 Situación

En este capítulo se realiza una recopilación de información relacionada con el objetivo de situarnos. En este trabajo hay bastantes cosas ya hechas por parte del grupo investigador en el que actualmente me incluyo, y, por tanto, va a jugar un papel importante el hecho de contextualizarse correctamente.

Este grupo está formado por: María del Carmen Lea Pereira (Medicina Interna), Patricia Martínez Sánchez (Neurología), María Isabel Álvarez Moreno (Medicina Interna), José Galván Espinosa (Gestión), María del Mar Iglesias Espinosa (Neurología), Juan José López Ramos (Informática), Juan Manuel García Torrecillas (Medicina de Familia y Epidemiología), Fernando Reche Lorite (Estadística e investigación operativa), María del Mar Rodríguez (Enfermería). Se trata de un proyecto de investigación e innovación que tiene lugar dentro del **Sistema Sanitario Público de Andalucía**, y tiene por título: "*Optimización de un modelo predictivo para la mortalidad en ictus isquémico a través de una cohorte de "Real World Data*" " [0].

Uno de los objetivos del proyecto inicial, es la realización de una validación externa de un modelo (regresión multivariante) actualmente implementado. Este objetivo llevará un tiempo considerable (reunir casos para una muestra) y, mientras tanto, se ha visto oportuno hacer una revisión del trabajo realizado hasta ahora aportando una nueva metodología y técnicas innovadoras de Ciencia de Datos. Muchas de estas técnicas han sido contempladas a lo largo del máster, y, aunque no sea el objetivo primordial del

proyecto, el de este TFM sí será hacer un análisis exploratorio de los datos, tratar el preprocesamiento de estos, aplicar distintas técnicas de modelado predictivo, hablar de la selección de variables, interpretar/explicar los modelos utilizados, aplicar modelos más avanzados como el uso de redes neuronales, ensambles, … y, en definitiva, tratar de poner en marcha este proceso en producción mediante el uso de servicios AWS.

1.2 Fundamentos

El ictus isquémico supone la segunda causa de mortalidad en nuestro país en la población general y la primera causa de mortalidad en la mujer [1]. A nivel mundial el ictus es la segunda causa de mortalidad y la tercera más común en los países industrializados [2]. Por tanto, nos enfrentamos a un auténtico problema de salud pública.

La mortalidad hospitalaria según fuentes españolas procedentes de registros de carácter clínico se sitúa en torno al 12.9% [3]. Un porcentaje elevadísimo de la misma son ictus isquémicos no lisables, esto es, no subsidiarios de tratamiento fibrinolítico (un tipo de fármaco) por no cumplir los estrictos criterios para la aplicación de este tratamiento [0].

Hasta la fecha se han detectado una serie de factores de riesgo tanto para el desarrollo de un evento isquémico como para estimar la probabilidad de fallecer o presentar secuelas [3]. Los trabajos de Smith et al [4] permitieron la obtención de modelos predictivos para la mortalidad hospitalaria, tanto por ictus isquémico como hemorrágico, y con un escaso número de variables, alcanzando una excelente capacidad discriminativa estimada mediante el C-Statistic de 0.85. Existen otros trabajos de alto interés que han mostrado una metodología acertada en la elaboración de modelos predictivos para ictus [5].

Estos factores han sido obtenidos fundamentalmente a partir de registros hospitalarios y en algún caso desde el Conjunto Mínimo Básico de Datos, si bien no hemos encontrado en la literatura un modelo predictivo basado en CMBD* que nos permita una adecuada estimación de la probabilidad de fallecer durante la hospitalización por un ictus agudo no lisable (GRD 14).

1.3 Justificación

Se ha demostrado que la existencia de un plan integral de actuaciones que, desde la llegada del paciente al hospital maximice y optimice su atención repercute de manera

beneficiosa en los pacientes que han padecido un ictus agudo, aumentando las probabilidades de recuperación [2]. A lo largo de los últimos veinte años no sólo el cambio en las actividades preventivas, sino la actuación precoz, reglada y acorde a los estándares de calidad más avanzados, ha demostrado conseguir una disminución importante tanto de la mortalidad como de las secuelas del ictus [0].

Una vez se produce la admisión del paciente con ictus, no sólo el tiempo y las medidas adoptadas, sino también la formación y el equipo que presta la atención ha mostrado ser un factor determinante en la reducción de la tasa de mortalidad intrahospitalaria.

El conocimiento de las variables subsidiarias de ser recogidas a pie de cama al ingreso o en los primeros días de estancia del paciente y que, integradas en un modelo predictivo nos permitan estimar la probabilidad de fallecer durante el ingreso, tiene importantes repercusiones en la atención al paciente:

i). Permitiría modular la intensidad y velocidad del esfuerzo diagnóstico terapéutico en aquellos pacientes con scores de riesgo de puntaje más elevado.

ii). Facilitaría la integración en circuitos asistenciales específicos de los pacientes de alto riesgo

iii). Sería un elemento clave para la optimización de la costo-eficiencia en la atención a pacientes con ictus no lisable.

CMBD: El conjunto mínimo básico de datos (CMBD) es una base clínico-administrativa de obligado cumplimiento para los hospitales de nuestro Sistema Nacional de Salud. A partir de los informes de alta que realizamos, las unidades de codificación de cada centro hospitalario se encargan de recoger un conjunto mínimo básico de datos.

Cada vez que un paciente llega al hospital, recibe una serie de diagnósticos, procedimientos, toma de datos, … Todo ello queda registrado en una instancia de la base de datos CMDB, es decir, un mismo paciente puede tener varias instancias de dicha base de datos asociadas a su ID, pero si el paciente vuelve al hospital por el mismo motivo (mismo diagnóstico) que se trató la vez anterior en menos de 30 días, entonces se continuará sobre la misma fila, y la variable REINGRESO (booleana) pasará a ser 1.

Es importante entender esto porque los datos con los que vamos a trabajar han sido obtenidos de esta base de datos. Además, si tenemos en cuenta que es una base de datos a nivel Nacional, no es difícil imaginar la cantidad de “ruido” que podemos encontrar en ella. Esta misma fuente de ruido será un “verificador” y a la vez un factor limitante en la calidad de los modelos (se verá más adelante).

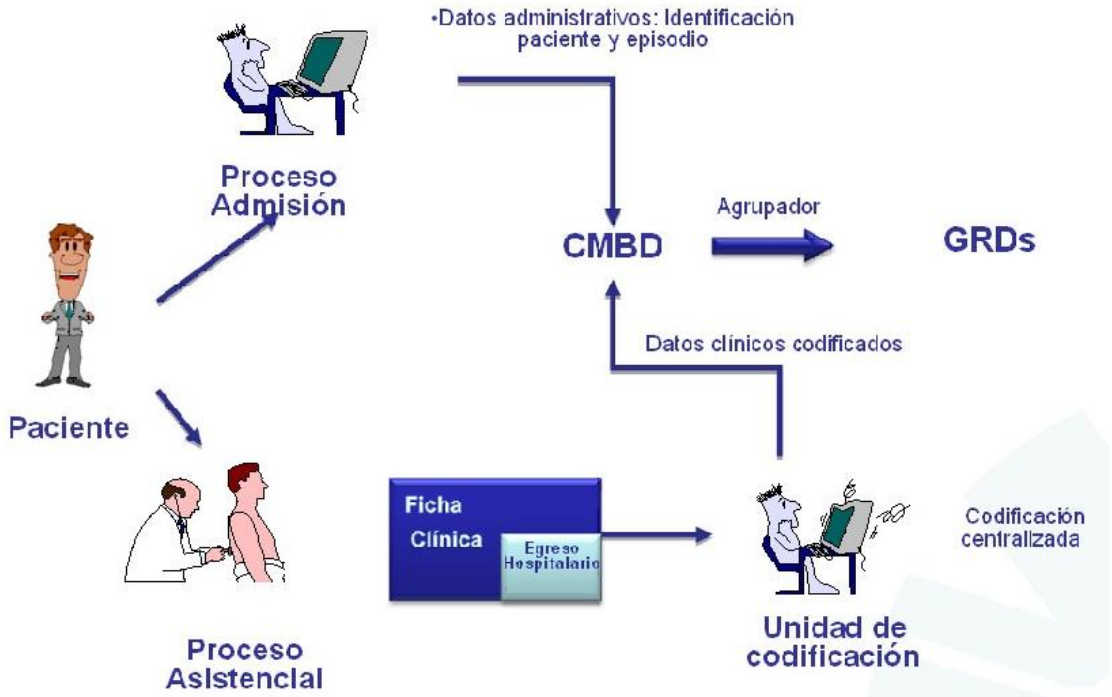


Figura 1.1. Esquema de uso de CMBD.

El utilizar estas bases de datos para temas médicos tiene su inicio en EE. UU. en 1973, con "Uniform Hospital Discharge Data Set" (UHDDS), y tiene por objetivo la recogida de datos básicos que sirvan tanto para tema clínico como para tema administrativo. En 1982, la Comunidad Europea trata de imitarla, con "European Minimum Basic Data Set" (MDBS) con una finalidad de gestión, planificación, evaluación e investigación. Y posteriormente se implanta en España el Conjunto Mínimo Básico de Datos en 1987, siendo de obligado cumplimiento desde 1992. Consta de un sistema de codificación propio GRD (Grupos relacionados por el diagnóstico).

1.4 Hipótesis del trabajo

El estudio ya realizado parte de las siguientes hipótesis de trabajo:

i). Es posible identificar determinadas variables clínico-administrativas contenidas en el CMBD asociadas a la mortalidad intrahospitalaria de los pacientes con ictus no lisado.

ii). Es posible elaborar un modelo predictivo de mortalidad intrahospitalaria basado en dichas variables que anticipe un mayor riesgo de mortalidad por ictus durante la hospitalización, facilitando el establecimiento de los recursos precisos para minimizar dicho desenlace.

La hipótesis del presente trabajo postula que, los pacientes ingresados con diagnóstico principal correspondiente a ictus isquémico agudo no lisado comparten características detectables, aislables y mensurables que permitirían la elaboración de un modelo logístico predictivo para la probabilidad de presentar mortalidad durante la hospitalización. Dicho modelo tendrá una capacidad discriminativa estimada mediante el área bajo la curva (estadístico C) no inferior a 0.70 y una calibración aceptable desde el punto de vista gráfico aún con valores de $\chi 2$ significativos para la prueba de Hosmer -Lemeshow [0].

1.5 Datos Utilizados

Para el desarrollo del modelo predictivo se estudiaron todos los episodios de hospitalización por el GRD 14 (ictus isquémico no lisado) a partir del Conjunto Mínimo Básico de Datos al alta (CMBD) de España desde 2008 hasta 2012. La información fue facilitada por el Instituto de Información Sanitaria del Ministerio de Sanidad y Consumo (IIS-MSC).

Para la codificación de los diagnósticos y procedimientos se ha utilizado la Clasificación Internacional de Enfermedades, 9ª edición Modificación Clínica, y para la agrupación previa de altas, el sistema de clasificación de pacientes de los Grupos Relacionados por el Diagnóstico -AP-GRD-, versión 21.09. La información demográfica manejada procede del Instituto Nacional de Estadística (INE) [8].

1.6 Diseño

Se construyó una cohorte histórica a partir de todos los pacientes admitidos en el ingreso bajo el GRD 14, la cual fue desagregada posteriormente en dos grupos, de tal modo que se confrontaron los pacientes que no fallecieron durante el ingreso con aquellos que sí lo hicieron. Por tanto, se realizó un estudio de cohorte histórica con un grupo de comparación interna [0].

1.7 Objetivos del Proyecto

Principal:

- **Validación externa** mediante la determinación de la capacidad de discriminación y calibración de un modelo predictivo de mortalidad en ictus isquémico (*modelo matemático presentado como patente y App como registro de propiedad intelectual*) en una cohorte de pacientes reales hospitalizados.

i). **Específico 1: Recalibración** de los coeficientes del modelo teórico a partir de los datos de la cohorte real, determinando el punto de corte óptimo para la sensibilidad y especificidad en la predicción de mortalidad del modelo estadístico con los nuevos coeficientes.

ii). **Específico 2: Trasladar y transferir** a todos los niveles asistenciales implicados en la atención al ictus una aplicación móvil y web recalibrada y optimizada.

1.8 Objetivos del TFM

Actualmente, el modelo de regresión logística se encuentra patentado y desplegado en una web, incluso su puesta en funcionamiento a nivel sanitario es inminente.
Sin embargo, para reunir datos con cierta calidad que sirvan para realizar una validación y calibración del modelo, hace falta tiempo. Durante ese "tiempo" lo que nos proponemos es:

- Hacer una revisión de la metodología utilizada. Implementar nuevas funciones que permitan ampliar el horizonte de opciones de cara a la investigación, esta vez de la mano de Ciencia de Datos.
- Traer nuevas técnicas utilizadas en Ciencia de Datos y proporcionadas por Librerías en Python como Sklearn, StatsModels, Tensorflow, Shap, XGBoost tratando de incorporar nuevas perspectivas en cuanto a la complejidad de modelos, explicabilidad de resultados, técnicas de resumen de información …
- Desarrollo de nuevos modelos de estimación de mortalidad con los que se espera igualar o mejorar los resultados, tanto para **pre-ingreso** del paciente, como para **post-ingreso**, lo que se corresponderá con los **escenarios A y B.**

- Una vez desarrollado y discutido el punto anterior, se propondrá a una puesta en funcionamiento en AWS mediante API Gateway y otros servicios, de manera que se puedan realizar estimaciones.

En definitiva, se pretende hacer un recorrido sobre muchos de los contenidos vistos en el máster de cara a ponerlos en funcionamiento en un caso real con posibilidad de realizar alguna publicación en este ámbito en un futuro.

Capítulo 2

Análisis Exploratorio

2.1 Preprocesado y descripción de los Datos

La base de datos de la que disponemos es de SPSS (viene en formado .sav), debido a que los datos han formado parte de un proceso de estudio previo. Los datos gozan de cierta calidad, algunas variables han sido mapeadas para su uso, aunque al guardarse como archivo .sav pierden parte del procesamiento y hay que volver a realizarlo. Con *pandas* podemos leer archivos de este tipo con **read_spss**. Procederemos a detallar algunos de los pasos.

En dicha base de datos contamos con 186245 filas y 37 variables. Se trata de un problema en el que buscamos estimar una probabilidad, pero, en definitiva, es un problema de clasificación, donde la variable objetivo es 'EXITUS' (fallecidos según la jerga médica).

En la base de datos inicial hay más de 37 variables, pero dichas variables se corresponden con temas administrativos y después de la decodificación y recodificación no tienen sentido.

De ahora en adelante, hablaremos de **dos escenarios**:

- **Escenario A:** Se desarrollarán todas las pruebas y modelos pertinentes, pero únicamente sobre variables que han sido seleccionadas por criterio experto. Estas variables son aquellas que pueden obtenerse de manera

relativamente sencilla una vez ingresa el paciente. Se tendrían 10 variables predictoras y 1 variable objetivo.

- **Escenario B:** Puesto que disponemos de 37 variables, nos preguntamos si somos capaces de mejorar los resultados obtenidos en el **Escenario A** empleando únicamente técnicas de ciencia de datos.

Independientemente del escenario, se **normalizan los nombres de las columnas** y se ponen todos los nombres de las variables en mayúscula, evitando los acentos, caracteres especiales y cambio de 'ñ' por 'ny'.
Para en el **Escenario A** encontramos las siguientes variables (que serán comunes al B):

Edad: Se considera como variable cuantitativa continua expresada en años completos.
Sexo: Cualitativa booleana. Categorías Mujer vs Varón.
Exitus: Cualitativa booleana. Indica la muerte del paciente.
Reingreso: Cualitativa booleana. Indica si el paciente ha vuelto a ingresar por el mismo motivo en los 30 días posteriores al primer diagnóstico.
HTA: Cualitativa booleana. Indica la presencia de hipertensión.
DM: Cualitativa booleana. Indica la presencia de Diabetes mellitus.
ARR: Cualitativa booleana. Indica la presencia de Arritmias.
DISLIPEM: Cualitativa booleana. Indica la presencia de Dislipemia.
ICC: Cualitativa booleana. Indica la presencia de insuficiencia cardíaca crónica.
ESTBASILAR: Cualitativa booleana. Indica la presencia de estenosis de la arteria basilar.

Notas:

1. Estas son las variables con las que primero se desarrolló el modelo patentado con una regresión logística multivariante.
2. La idea inicial era usar la variable **FA** (fibrosis auricular en lugar de ARR), pero la calidad de la variable **FA** es deficiente, y, por tanto, se sustituye por **ARR** (que engloba a la variable FA), ya que está mejor codificada.
3. En el análisis que tendrá lugar en los siguientes puntos, se realizará una descripción global de la muestra.

Si estudiamos las variables que tienen más valores faltantes, obtenemos lo siguiente:

Variable	Valores faltantes
ARR	4740
VALV	2285
DM	2010
TNO_GI	1934
CISQ	1096
DISLIPEM	1043
FA	882
HTA	788
ICC	667
IRA	322
ANEMIA	291
TNO_HDROELEC	195

Tabla 1. Valores faltantes por variable.

Pero si atendemos únicamente a aquellas observaciones que tienen al menos un valor faltante en una de las 37 variables, obtenemos un total de 14742 registros, lo que corresponde a un 7.92% del total.

De esta manera, quedan 171503 registros donde los fallecidos ocupan el 6.76% de la muestra, por ello, no tiene sentido utilizar métricas como accuracy. Abordaremos esta cuestión más adelante.

Consideramos que al ser un tema delicado como es una enfermedad, realizar una técnica de imputación de valores faltantes puede ser arriesgado, y procedemos a la eliminación de esos registros.

Se transforman las variables binarias haciendo los siguientes cambios:

1. No -> 0
2. Sí -> 1
3. Vivo -> 0
4. Fallecido -> 1
5. Hombre -> 0
6. Mujer -> 1

Como hemos visto, salvo la edad, el resto son todas variables booleanas. A primera vista, podríamos pensar que esto es algo bueno, ya que si las clases **fallecer padeciendo Ictus** y **no fallecer padeciendo ictus** estuviesen “bien separadas” (esto es, si existiera una clara diferencia entre las características necesarias para pertenecer a una o a otra), se podría definir una frontera de decisión que separase las clases.
Pero ahora veremos que esto no es así exactamente, y que, aunque los predictores elegidos ayudan en las predicciones, no son suficientes como para discriminar correctamente.

Algunas variables más que tendremos en cuenta en el **Escenario B**:

EDAD_R: Edad estratificada transformada a cualitativa:

'24-34':1, '35-44':2, … '85 o más':7

NDA: Número de diagnósticos del paciente, variable categórica donde se ha hecho la siguiente transformación: '0-3 diag':1, '4-7 diag':2, … '>=12 diag':4

NPA: Número de procedimientos del paciente, variable categórica con la siguiente transformación: '0-3 proc':1, '4-7 proc':2, …. '>= 16 proc':5

Tipo de Ingreso: Nos dice si el ingreso estaba programado o se debe a una urgencia: 'Urgente':1, 'Programado':2

ESTANCIA: Tiempo transcurrido en días completos desde fecha de ingreso hasta fecha de alta.

El resto de las variables se corresponden con otras enfermedades como Arritmias, congénitas, obesidad … siendo todas estas variables booleanas que indican la presencia.

2.2 Edad en los grupos

El predictor Edad va a ser el más importante como veremos más adelante, y es que hay una clara diferencia entre los grupos de fallecidos y los que no fallecen.

Diferencia de medias observada: 8.7311641818239

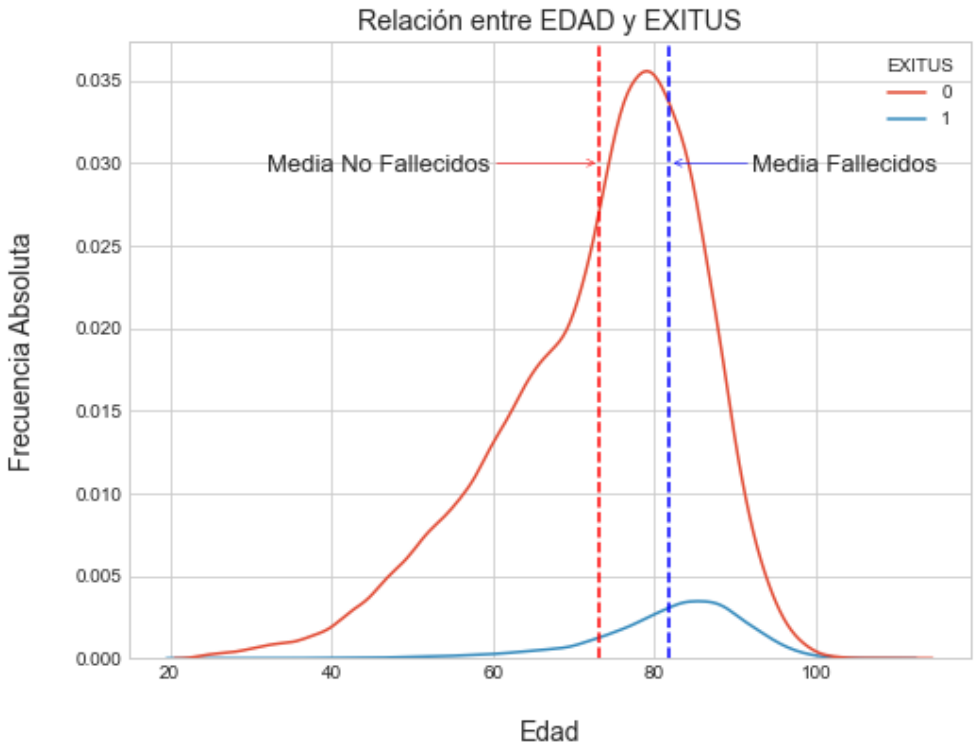


Figura 2.1 Diferencia de Edad entre el grupo de fallecidos y no fallecidos.

Podemos hacer un contraste de hipótesis mediante distribuciones Bootstraping, obteniendo lo siguiente:

- Intervalo al 95% de confianza: [-8.91621768, -8.54262454]

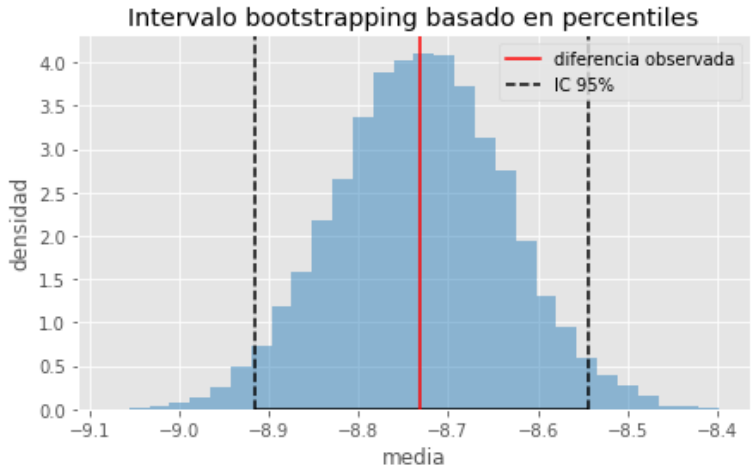


Figura 2.2 Bootstrapping para la diferencia de Edad.

Por tanto podríamos afirmar que sí existen evidencias para afirmar que la edad entre los dos grupos es distinta. El intervalo de confianza al 95% para la diferencia de medias obtenido por bootstrapping indica que, en promedio, la edad del grupo de fallecidos está entre 8.54 y 8.91 años más.

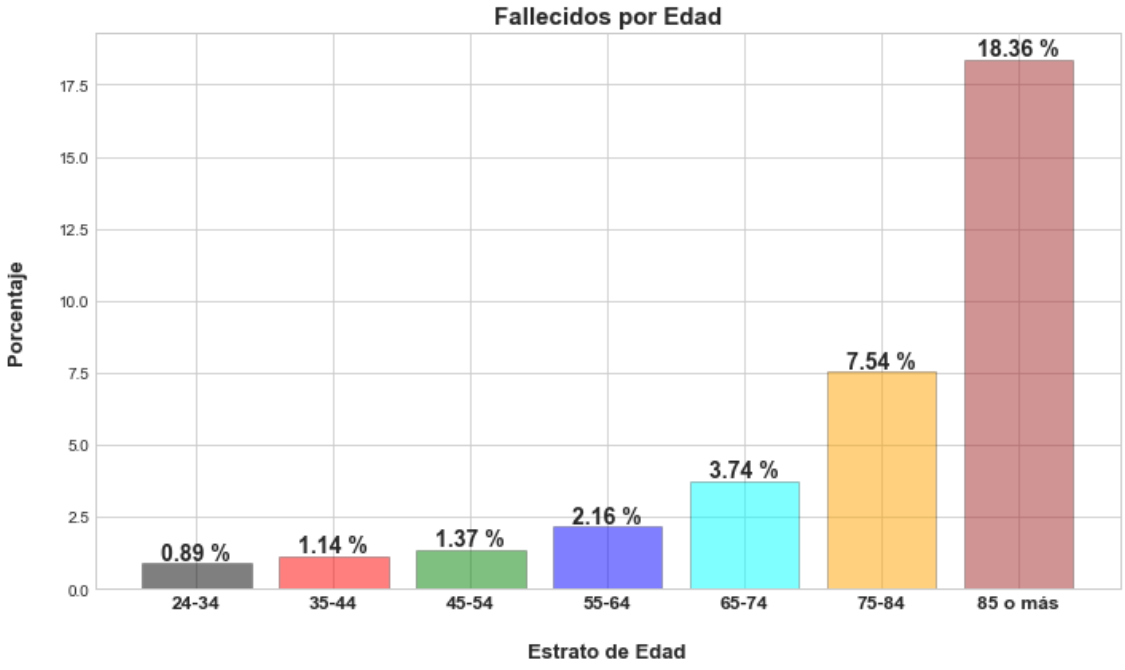


Figura 2.3 Fallecidos por estrato de Edad.

Una vez más, vemos como la Edad es un predictor clave, ya que aumenta el número de fallecidos de manera considerable con edades avanzadas.

2.3 Sexo

Como hemos comentado en el primer apartado, el fallecimiento por Ictus isquémico es más prevalente en las mujeres que en los hombres. Veámoslo en nuestros datos:

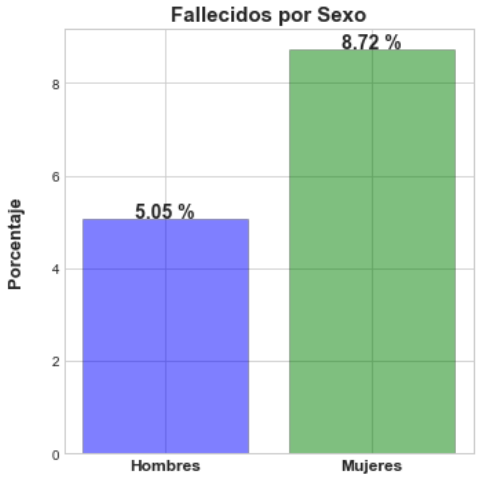


Figura 2.4 Fallecidos – Sexo en el escenario global (tanto A como B).

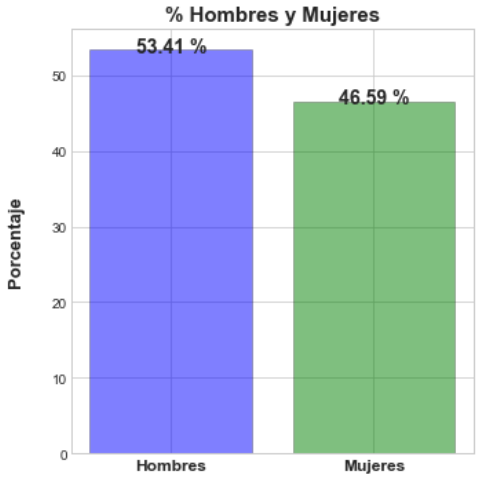


Figura 2.5 % Mujeres y Hombres en la muestra global (tanto A como B).

Claramente, fallecen más las mujeres que los hombres, y será determinante en la estimación de la probabilidad.

2.4 Fallecidos por el resto de predictores

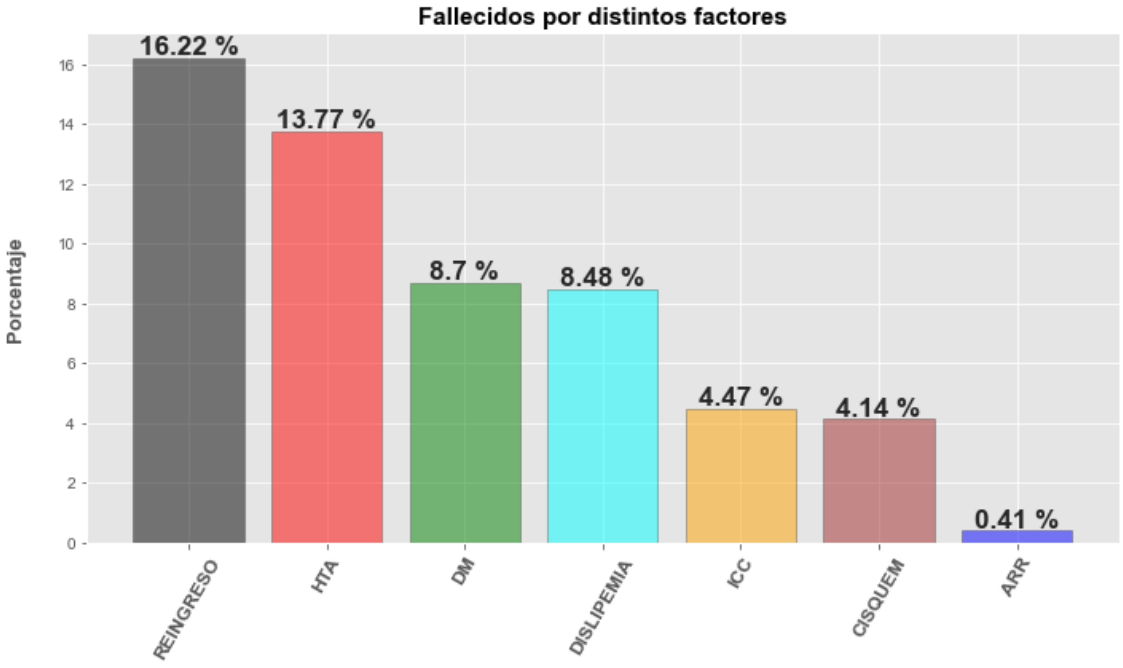


Figura 2.6 Fallecidos por diversos factores.

Si además echamos un vistazo a la prevalencia de los factores en la población:

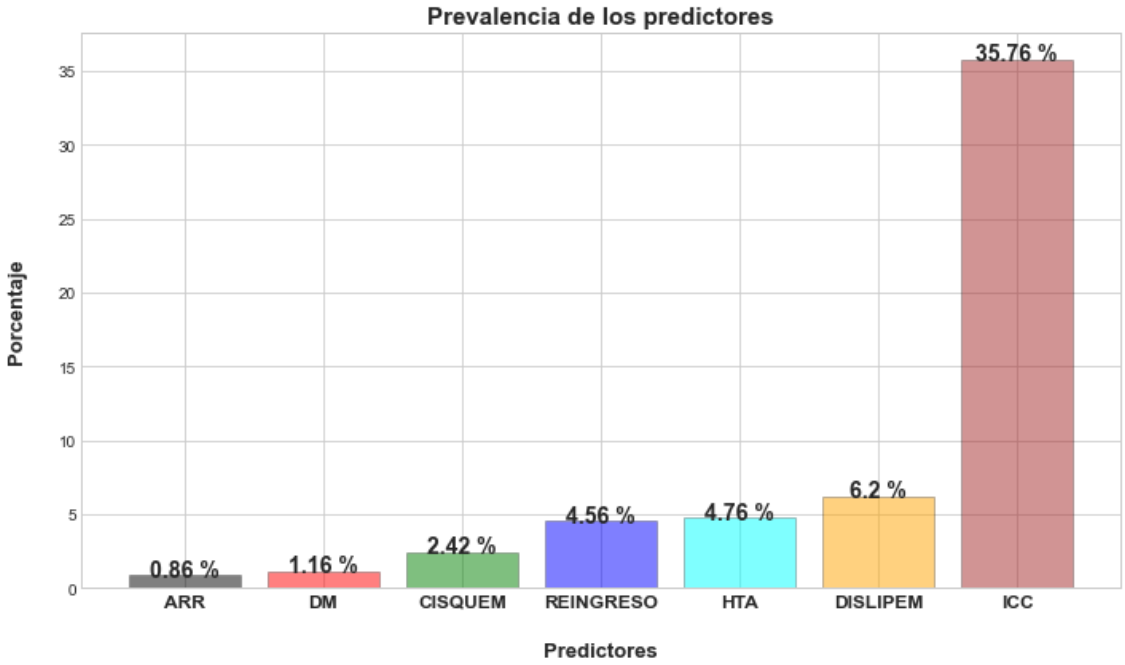


Figura 2.7 Prevalencia de los factores en la población.

Cuando se estaban visualizando estas variables, ha salido una anomalía en cuanto a la codificación de la variable FA.

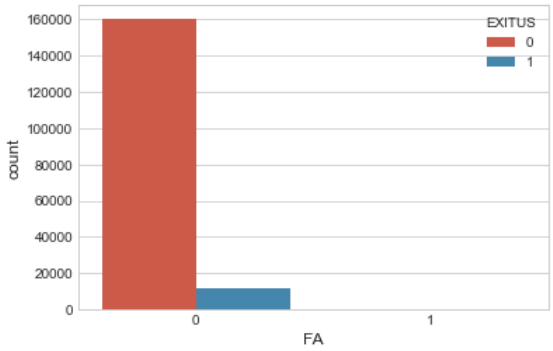


Figura 2.8 Anomalía en la variable FA.

Todas estas observaciones están codificadas como **No**, salvo un único caso que fallece. Es una variable que no nos proporciona ninguna información y que vamos a cambiar por la variable ARR, que, en particular, generaliza a la FA (existe multicolinealidad), y está mejor codificada.

2.5 Discusión de la métrica

Como se ha comentado previamente, no tiene sentido utilizar la métrica **accuracy** por estar ante un problema desbalanceado.

Por un lado, queremos evitar los falsos negativos a toda costa, ya que es bastante grave decirle a un paciente que no tiene nada si este se encuentra muy grave. Por ello, **recall** podría ser una buena métrica.

Sin embargo, también es importante la **precisión**, que mide la habilidad de clasificar un verdadero positivo como positivo. Si maximizamos el recall pero muchos casos se nos van a falsos positivos, entendemos que el clasificador no discrimina correctamente.

Una buena métrica podría ser el **área bajo la curva precisión-recall** (AUPRC), métrica recomendada para casos de datos desbalanceados.

F1-Score será la métrica base que utilizaremos junto con precisión y recall a la hora de estudiar nuestros resultados. También tendremos en cuenta el comportamiento de la curva **ROC** (que mide la habilidad de distinguir entre las clases) y la métrica **Balanced-Accuracy.**

$$Balanced\ Acc = \frac{TPR + TNR}{2}$$

$$TPR\ (True\ positive\ rate) = Recall = \frac{\#\ True\ positive}{\#\ Positives} = \frac{TP}{TP + FN}$$

$$TNR\ (True\ negative\ rate) = \frac{\#True\ Negative}{\#Negatives} = \frac{TN}{TN + FP}$$

Como clasificador base para estudiar qué está ocurriendo, utilizaremos el modelo RandomForest, aunque entrenaremos también una regresión logística y probaremos algún otro modelo del estilo K-NN, Redes Neuronales, …

2.6 División en train-test

Quitando la variable FA, nos quedan 36 variables con las que procedemos a realizar una división en train (85%) / test (15%). Para asegurarnos que la muestra es siempre la misma, establecemos una semilla (random_state = 44). Dividiremos la muestra con 36 variables y posteriormente filtraremos las variables para crear la división en el Escenario A.

Como método de entrenamiento y evaluación se utilizará la **validación cruzada + Holdout**, ofrecida por el paquete Scikit-learn en el objeto **GridSearchCV,** el cual hace una búsqueda exhaustiva probando todas las combinaciones posibles de parámetros y quedándose con la mejor, a la vez que realiza una validación cruzada para validar el modelo internamente, no haciendo falta dividir a su vez el conjunto de entrenamiento en entrenamiento / validación.

Para mostrar los resultados de los distintos modelos, mostraremos las métricas comentadas en el apartado (2.5) de dos maneras distintas, train vs train y train vs test.

2.7 Algoritmo de reducción de dimensionalidad, t-SNE

Procedemos a utilizar un algoritmo de visualización de datos en alta dimensionalidad, como es el algoritmo t – SNE, el cual tampoco hay que interpretar de manera literal en cuanto a la interpretación de separación de clases, ya que es un algoritmo de tipo “Embedding”, pero puede ayudarnos a ver cómo de separadas se encuentran las clases.

En él podemos ver que hay dos clases, pero que no están claramente diferenciadas, algunas instancias rojas sí, incluso algunas azules, pero no podemos asegurar la buena separación de las clases a priori.

Podemos ver como en el escenario A, los estratos de edad se ven marcados más claramente, donde va destacando el color azul conforme aumenta la edad. Esto lo vemos por la leve separación de zonas en la figura 2.9. Mientras que en el escenario B, a pesar de existir dos zonas con mayor densidad de observaciones de la misma clase, no parece existir una frontera de decisión clara.

Esto puede interpretarse como que no tenemos suficientes predictores como para separar las dos clases de una mejor manera.

Escenario A train

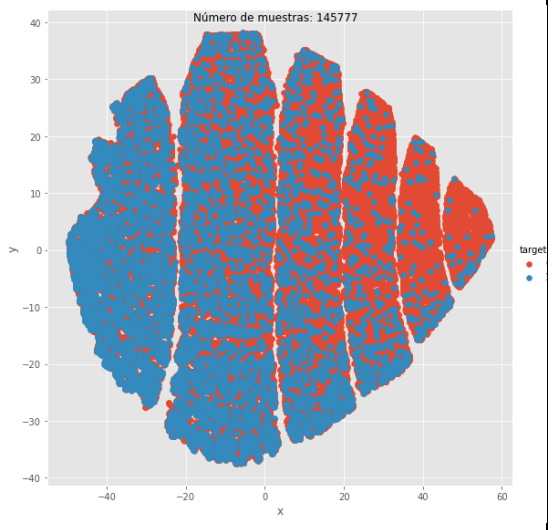


Figura 2.9 t-SNE Escenario A.

Escenario B train

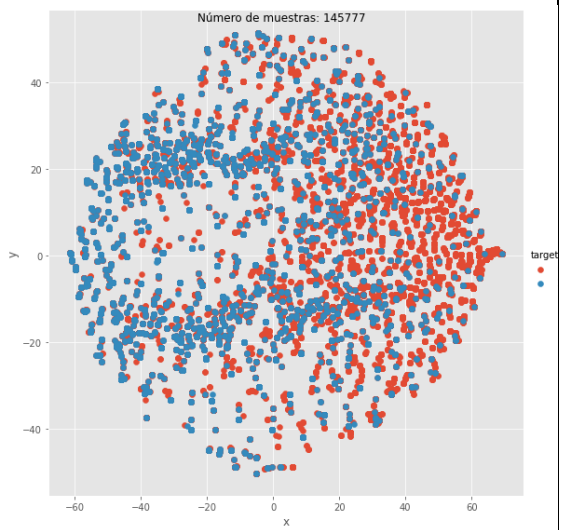


Figura 2.10 t-SNE Escenario B.

Capítulo 3

Metodología

3.1 Introducción

Hasta ahora, hemos explorado los datos y hemos dejado marcadas algunas pautas del camino a seguir, en cuanto a la métrica que utilizar para poder comparar los modelos. Sin embargo, aún no hemos explorado las opciones que tenemos con un conjunto de datos desbalanceado a la hora de entrenar un modelo, y es lo que se procede a discutir.

3.2 Datos desbalanceados

Cuando nos enfrentamos a un problema desbalanceado, principalmente tenemos tres estrategias de abordaje:

- Ignorar el problema.
- Submuestrear la clase mayoritaria.
- Sobremuestrear la clase minoritaria.

Ninguna de estas formas de abordar el problema es perfecta, todas y cada una de ellas tiene ventajas e inconvenientes. Comentaremos brevemente cada una de estas estrategias y más adelante las terminaremos de reseñar al aplicarlas de diversas maneras usando el paquete **imb-learn.**

En primer lugar, **ignorar el problema.** Si construimos un clasificador usando los datos tal y como se presentan, la predicción del modelo siempre devolverá la clase mayoritaria, es decir, acabará volviéndose un clasificador sesgado.

Submuestrear la clase mayoritaria. Esta es una de las estrategias más sencillas para tratar los datos desequilibrados. Existen diversas técnicas de submuestreo, pero no se ha probado que dichas técnicas aporten una mejora notable respecto a una simple selección aleatoria [27].

Sobremuestrear la clase minoritaria. puede llevar a problemas de sobreajuste si no se hace de la manera correcta, por ejemplo, si duplicamos alguna de las entradas de la clase minoritaria y procedemos a hacer la validación cruzada, estaremos incorporando el mismo registro a parte de la muestra test y a parte de la muestra de ajuste, por tanto, acabaremos sobreestimando el funcionamiento de nuestro modelo.

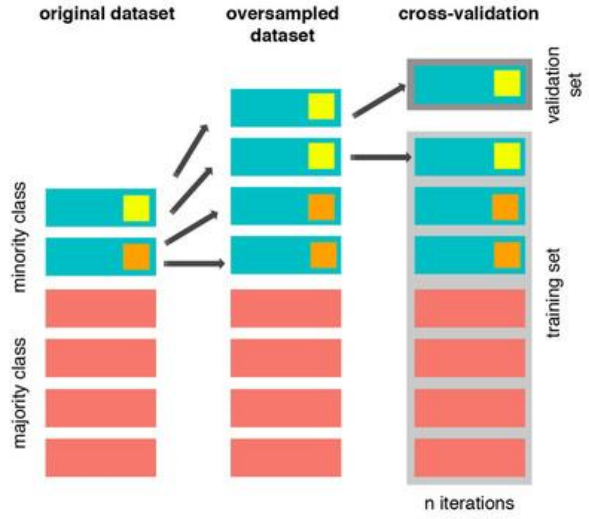


Figura 3.1 Oversampling - Cross Validation Erróneo.

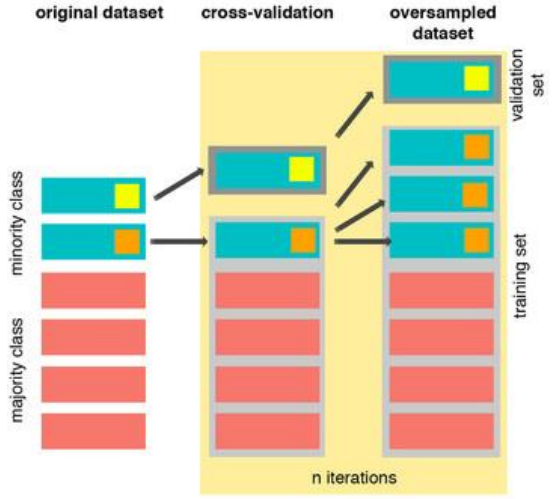


Figura 3.2 Oversampling - Cross Validation Correcto.

Otras técnicas que mejoran muchos de los resultados del sobremuestreo pueden ser las técnicas de SMOTE, que se basan en interpolaciones de la clase minoritaria.

Los modelos que entrenemos se guardarán en formato pickle (.pkl) formado por el string 'nombre_modelo' + '.pkl'.

Finalmente, nosotros vamos a utilizar únicamente las 2 primeras. Por un lado, ignoraremos el problema y vemos qué ocurre con los modelos, para después hacer un pipeline formado por un submuestreo aleatorio y un modelo.

3.3 Calibración del modelo [22]

Recordemos que el objetivo del trabajo era estimar la probabilidad que tiene una persona que llega a un centro médico diagnosticado por ictus isquémico de fallecer (en primera instancia).

Estos modelos, tanto en el **escenario A** como en el **B** son capaces de predecir si una persona va a sobrevivir o fallecer teniendo en cuenta que tiene un ictus isquémico, atendiendo a una serie de variables.

Pero la manera en la que lo hace no es otra que estimar la probabilidad de que la observación correspondiente pertenezca a una de las clases y comprobar si sobrepasa un cierto umbral.

Sin embargo, para este problema, no nos interesa tanto si una persona sobrevive o muere (si la probabilidad estimada sobrepasa o no el umbral), sino si la estimación de probabilidad realizada por el modelo se ajusta a la realidad.

¿Hasta qué punto debemos confiar en las predicciones del modelo? Aquí es donde entra el proceso de calibración del modelo.

> "Un modelo calibrado es aquel en el que, el valor estimado de probabilidad puede interpretarse directamente como la confianza que se tiene de que la clasificación predicha es correcta. Por ejemplo, si para un modelo de clasificación binaria (perfectamente calibrado) se seleccionan las predicciones cuya probabilidad estimada es de 0.8, en torno al 80% estarán bien clasificadas."
>
> "Todo lo que sabe un modelo es lo que ha podido aprender de los datos de entrenamiento y, por lo tanto, tiene una "visión" limitada." [21]

Definición: "La calibración de un modelo de clasificación consiste en reajustar las probabilidades predichas para que correspondan con la proporción de casos reales observados. En otras palabras, corregir las probabilidades predichas por un modelo cuando este las subestima o sobrestima."

Un modelo está perfectamente calibrado cuando, para cualquier valor p, la clasificación predicha con una confianza (probabilidad) de p, es correcta el $p * 100$ por ciento de las veces.

$$P\left(\hat{Y} = Y \middle| \hat{P} = p\right) = p, \qquad p \in [0,1]$$

Por ejemplo, si se seleccionan las observaciones cuya probabilidad predicha es $\hat{p} = 0.8$, es de esperar que el porcentaje de esas observaciones bien clasificadas sea del 80%.

Esto debería ser una curva de calibrado perfecta:

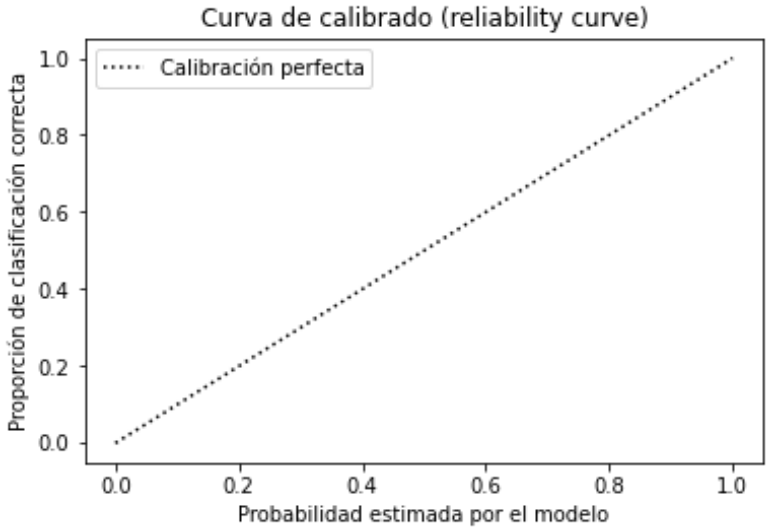


Figura 3.3 Curva de calibrado perfecta.

> "El proceso de calibración es importante cuando se quieren emplear las probabilidades asociadas a las predicciones. Si únicamente son de interés las clasificaciones finales, la calibración no aporta valor. Esto último es importante tenerlo en cuenta ya que, calibrar un modelo, puede implicar reducir su porcentaje de clasificaciones correctas (*accuracy*)."

> "Cuanto mejor calibrado esté el modelo, más próximos serán los valores de proporción empírica y de confianza, es decir, más se aproxima la curva obtenida a la diagonal. La curva de calibrado queda por encima de la diagonal si el modelo tiende a infravalorar las probabilidades y por debajo si las sobrevalora."

Aunque las curvas de calibración aportan información detallada, es interesante disponer de una métrica que permita cuantificar con un único valor la calidad de calibración del modelo. *Brier score* es la diferencia cuadrática media (*mean squared difference*) entre la probabilidad estimada por el modelo y la probabilidad real (1 para la clase positiva y 0 para la negativa). Cuanto menor es su valor, mejor calibrado está el modelo. Esta métrica es adecuada solo para clasificaciones binarias.

Para nosotros, va a ser fundamental no solo el desarrollo del modelo, sino también su proceso de calibrado, con el que veremos cómo se comporta realmente.

3.4 Algoritmo genético de selección de variables

Recordemos que la base de datos inicial tiene 37 variables. Los primeros modelos, desarrollados sobre el **escenario A**, están *basados en variables seleccionadas únicamente por criterio experto y además son variables que pueden tomarse de forma directa cuando el paciente llegue al centro de salud.*

¿Sería posible mejorar dichos resultados obtenidos por estos modelos usando únicamente técnicas de Ciencia de Datos? Como podría ser; Introduciendo otras variables que requieran el ingreso del paciente, algunas pruebas específicas, … (teniendo también cuidado de no introducir multicolinealidad en el modelo).

Lo que haremos será aplicar un algoritmo para selección genética de predictores propuesto por Joaquín Amat [20] y ver cómo se comportan los modelos propuestos hasta ahora. Los algoritmos genéticos son solo una de las muchas estrategias que existen para seleccionar los predictores más relevantes, y no tiene por qué ser la más adecuada en todos los escenarios. Existen estrategias iterativas Stepwise selection, modelos como Random Forest, Boosting y Lasso capaces de excluir automáticamente predictores, y técnicas de reducción de dimensión como PCA y t-SNE.

> "Los algoritmos genéticos son métodos de optimización heurística que, entre otras aplicaciones, pueden emplearse para encontrar la combinación de variables que consigue maximizar la capacidad predictiva de un modelo. Su funcionamiento está inspirado en la teoría evolutiva de selección natural propuesta por Darwin y Alfred Russel: los individuos de una población se reproducen generando nuevos descendientes, cuyas características, son combinación de las características de los progenitores (más ciertas mutaciones). De todos ellos, únicamente los mejores individuos sobreviven y pueden reproducirse de nuevo, transmitiendo así sus características a las siguientes generaciones."

Objetivo: Encontrar al mejor individuo, es decir, la combinación de predictores que da lugar al mejor modelo posible.

Configuración: En nuestro caso, estamos ante un problema de clasificación, optimizaremos el modelo randomforest mediante la métrica f1, usando como máximo 50 generaciones, probabilidad de mutación de 0.1, método de selección "ruleta" (la probabilidad de que un individuo sea seleccionado es proporcional a su *fitness* relativo) y método de cruce "uniforme" (el valor que toma cada posición del nuevo individuo se obtiene de uno de los dos parentales), con una validación cruzada interna.

Funcionamiento del algoritmo:

1) En primer lugar, se crea una población inicial aleatoria de *P* individuos. En este caso, cada individuo representa una combinación de predictores.
2) Calcular el fitness de cada individuo de la población (calcular con una métrica de calidad del modelo). Mayor fitness cuanto mayor sea la métrica.
3) Crear una población vacía y repetir los pasos hasta que se hayan creado *P* nuevos individuos.

    a. Seleccionar dos individuos de la población existente, donde la probabilidad de selección es proporcional al *fitness* de los individuos.

    b. Cruzar los dos individuos seleccionados para generar un nuevo descendiente (*crossover*).

    c. Aplicar un proceso de mutación aleatorio sobre el nuevo individuo.

    d. Añadir el nuevo individuo a la nueva población.

4) Reemplazar la antigua población por la nueva.
5) Si no se cumple el criterio de parada, volver al paso 2.

El término individuo hace referencia a una de las posibles soluciones del problema que se quiere optimizar, en nuestro caso, la selección de los mejores predictores.

Tras generar cada nuevo individuo de la descendencia, este se somete a un proceso de mutación en el que, cada una de sus posiciones, puede verse modificada con una probabilidad *P*. Este paso es importante para añadir diversidad al proceso y evitar que el algoritmo caiga en mínimos locales porque todos los individuos sean demasiado parecidos de una generación a otra.

Resultados:

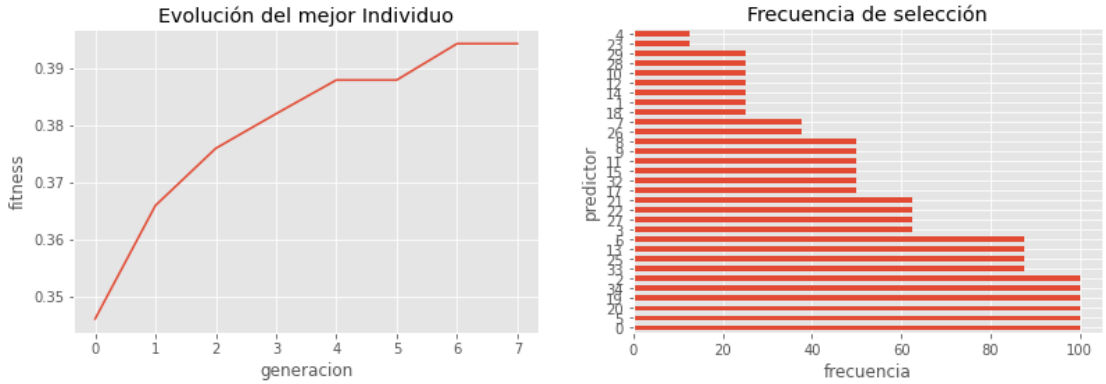


Figura 3.4 Algoritmo genético selección de predictores.

```
Algoritmo detenido en la generación 7 por falta cambio absoluto mínimo de
0.01 durante 5 generaciones consecutivas.
------------------------------------------
Optimización finalizada 2021-06-29 10:30:40
------------------------------------------
Duración optimización: 2513.0435683727264
Número de generaciones: 8
Predictores óptimos: [ 0 1 2 5 6 9 11 12 13 14 15 17 18 19 20 22 25 27 28
33 34]
Valor métrica óptimo: 0.39430238978439613
```

3.5 Conclusión

En el escenario B, partiremos directamente del uso de estos 20 predictores que son óptimos para el randomforest entrenado con la métrica f1, en lugar de usar las 37 variables de las que partimos.

Además, probaremos a hacer filtrados según el criterio de importancia asignado por el modelo randomforest y también una evaluación de modelo a la vez que se hace una eliminación recursiva de predictores.

Tanto en el escenario A como en el escenario B, probaremos un randomforest y una regresión logística, tanto balanceados en primer lugar, como en un pipeline con un submuestreo aleatorio previo. Para entrenar y ver cómo se comportan los modelos usaremos las métricas que han sido discutidas.

Capítulo 4

Escenario A

4.1 Introducción

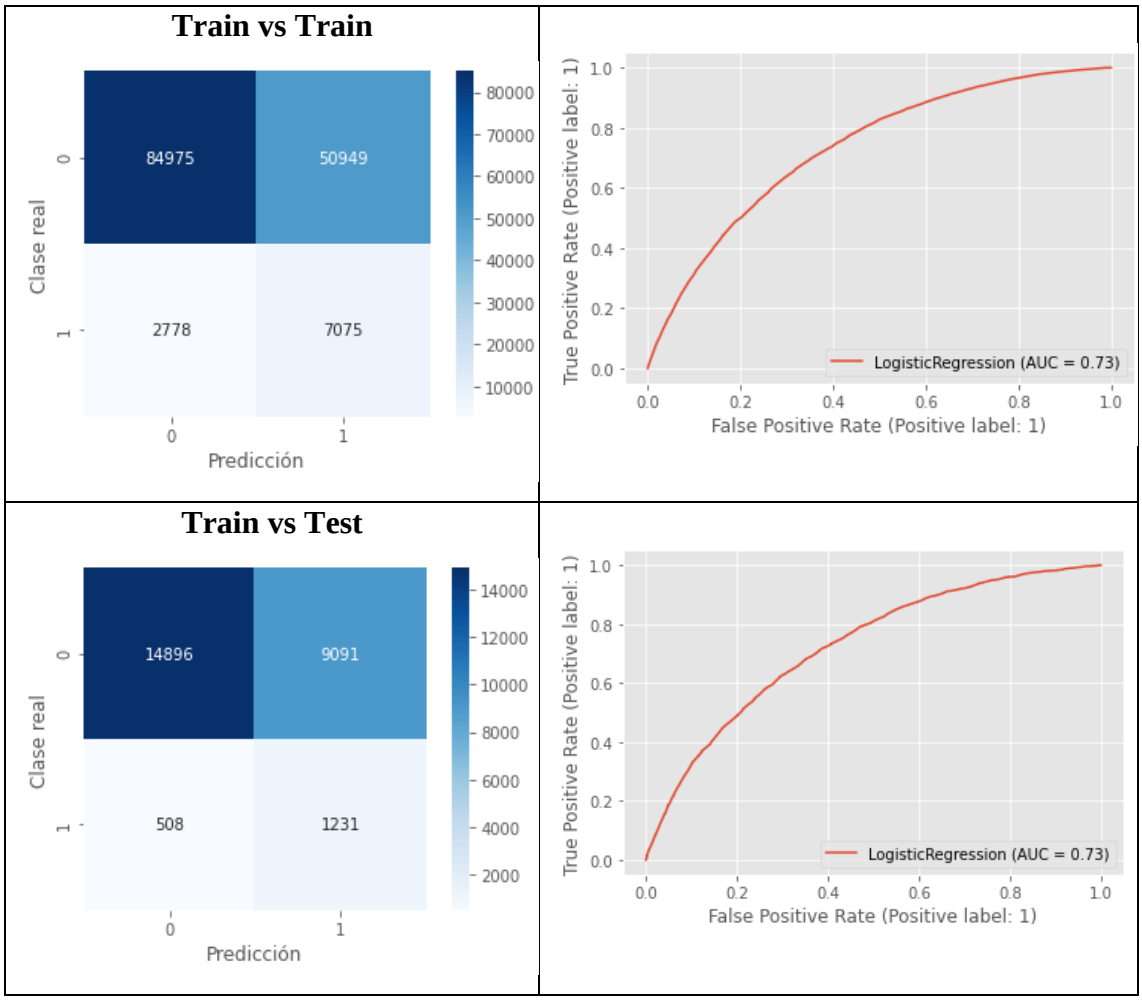
Este es el primer escenario que vamos a desarrollar, principalmente porque es más sencillo, y segundo porque el otro modelo englobará estas variables y parte del razonamiento aplicado en este caso podrá extrapolarse al escenario B.
Procedemos entonces a evaluar los modelos, a compararlos, calibrarlos, y tratar de estudiar la explicabilidad / interpretabilidad de estos.

Nota:
Recordemos que todos los modelos se van a entrenar usando la misma metodología, validación cruzada interna con una búsqueda exhaustiva del espacio de hiperparámetros, tratando de optimizar la métrica F1-Score.
Y, aunque pueda variar un poco la muestra de training (debido a que algún modelo estará compuesto por un pipeline, donde la primera parte sea hacer un submuestreo de la clase mayoritaria para entrenar con un 50-50), todos los modelos usarán la misma muestra test para poder comparar los resultados.

4.2 Regresión logística

4.2.1 Regresión logística balanceada

Se ha utilizado una penalización inversamente proporcional al número de casos de la clase minoritaria.



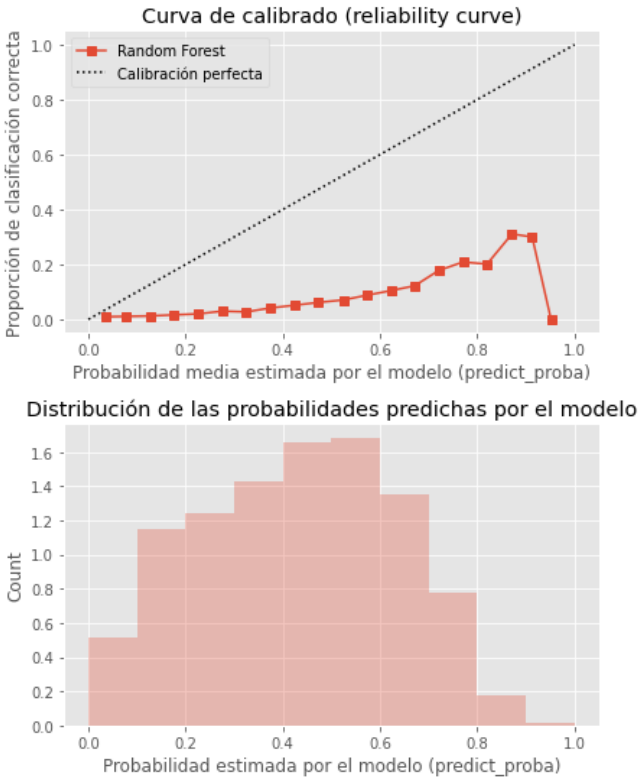


Figura 4.1 Curva de calibrado de la Regresión Logística Balanceada.

Por lo estudiado en la sección de calibración, sabemos que el modelo está sobreestimando las probabilidades, de ahí la baja precisión que estamos obteniendo y el gran número de *falsos positivos*. Recordemos que esto es preferible a la clasificación de *falso negativo*, pero básicamente es una cuestión de umbral, y lo que nos interesa es cuánto se ajusta dicha curva a la "perfecta".

```
Brier score = 0.21518750972103237
```

Figura 4.2 Modelo calibrado.

```
Brier score = 0.06001531521008679
```

Aunque la métrica de Brier sea menor, podemos observar que la calibración de este modelo es bastante deficiente debido a la gran cantidad de casos negativos del modelo, y acaba siendo un clasificador trivial.

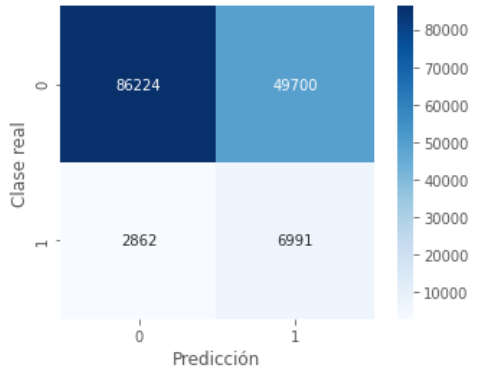
4.2.2 Pipeline Submuestrear + Regresión Logística

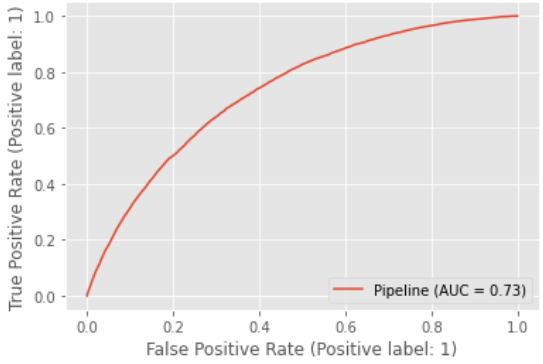
A pesar de haber realizado un submuestreo aleatorio sobre la clase mayoritaria, el modelo no consigue aprender a discriminar mejor que el visto en el punto anterior.

Vemos entonces que algo extraño está ocurriendo, probemos con un random forest usando las mismas técnicas comentadas y estudiemos la explicabilidad e interpretabilidad.

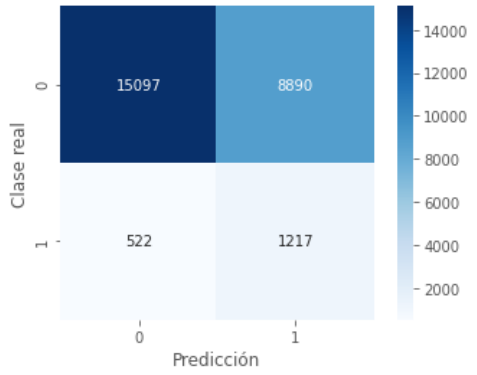
4.3 Random Forest balanceado

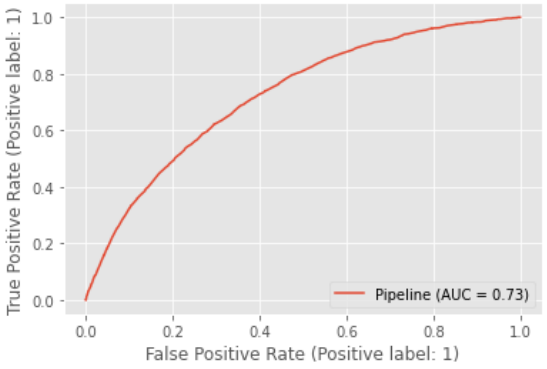
Train vs Train





Train vs Test





	Accuracy	Balanced accuracy	F1-Score	AUROC	Precision	Recall
Balanced Logistic Regression	0.633756	0.671532	0.208851	0.732837	0.122282	0.715214
Random Undersample + Logistic Regression	0.637426	0.671994	0.209760	0.732947	0.123003	0.711966
Balanced random forest	0.651886	0.673818	0.213537	0.733811	0.126013	0.699178
Under-sampling + Random forest	0.639648	0.661089	0.204715	0.712233	0.120337	0.685882

Tabla 2. Resultados de los modelos.

Como podemos observar, tenemos unos resultados similares tanto en el random forest como en la regresión logística, a pesar ser un modelo bastante más complejo que el otro. Tratemos de echar un vistazo a lo que está ocurriendo.

4.4 Explicabilidad e Interpretabilidad

Vamos a intentar echar un vistazo a algunos casos donde el modelo clasifique como fallecido, cuando realmente no fallece, tratando de entender por qué el modelo está clasificando de esa manera.

En primer lugar, veamos la asignación de importancia a los predictores por parte de nuestro modelo:

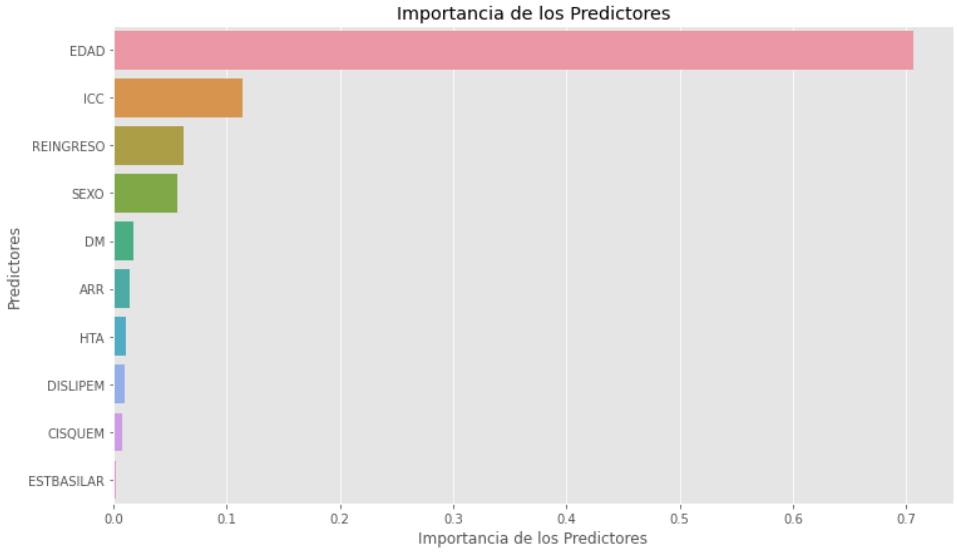


Figura 4.3 Importancia de los predictores según RandomForest.

Vemos que existe una gran diferencia entre el predictor Edad y el resto, y probablemente debido a ello se produzca la gran cantidad de falsos positivos. Es sabido que la importancia asignada por el modelo RandomForest tiende a sobreestimar las variables cuantitativas, y a modo de corrección se suele aplicar la importancia por permutaciones:

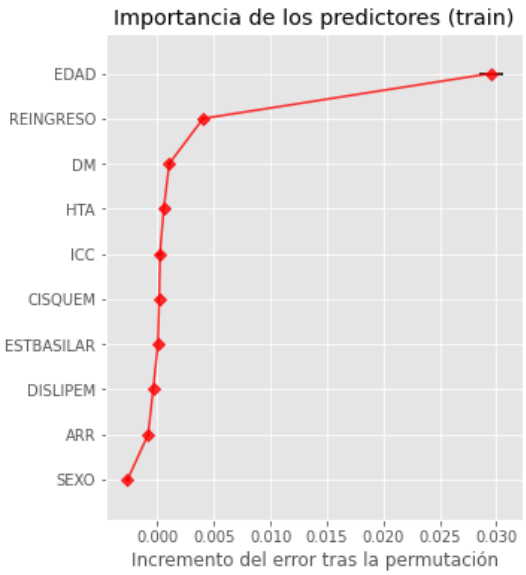


Figura 4.4 Test de importancia por permutaciones.

Si echamos un vistazo a algunas de las probabilidades estimadas por el modelo y nos vamos a los percentiles [0, 25, 50, 75, 100] podemos ver algunas diferencias:

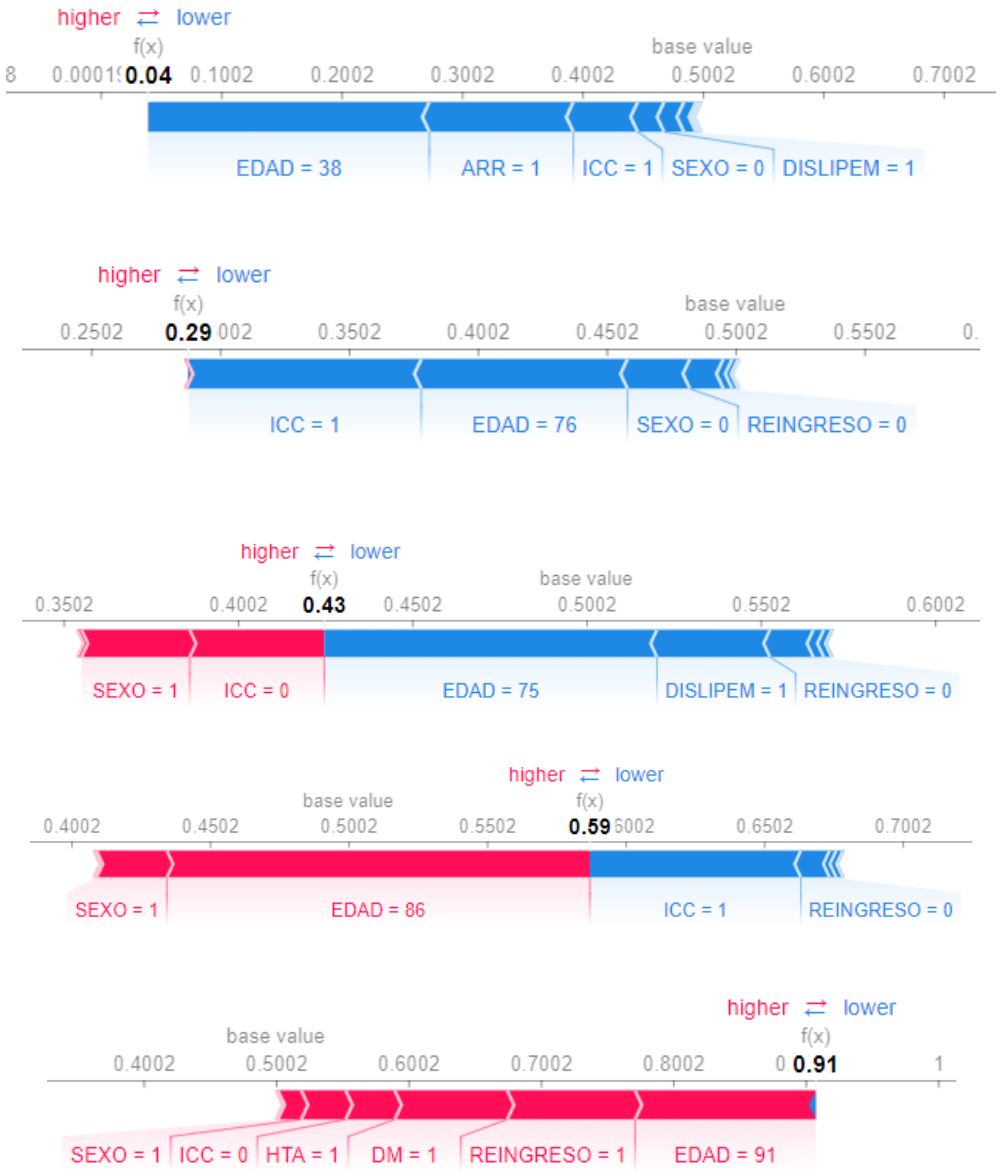


Figura 4.5 Valores SHAP.

Se ha demostrado que la HTA (hipertensión) es un factor protector frente al ictus isquémico, pero el modelo no parece detectarla, al menos en estos ejemplos, o si la detecta la tasa negativamente (puede deberse a la edad avanzada). Vemos también como la variable ICC la toma como positiva, disminuyendo la probabilidad. Además, el sexo también es un factor bastante marcado junto a la Edad.

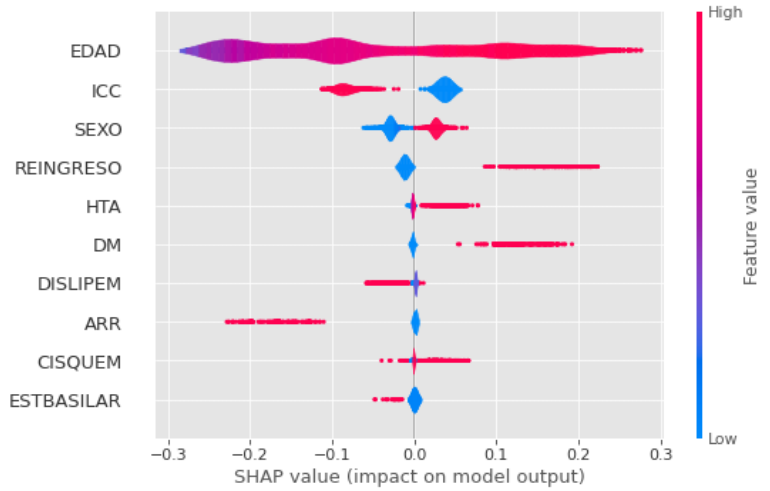


Figura 4.6 Influencia de los predictores según valores SHAP.

Aquí vemos cómo se comporta el modelo respecto a las observaciones de la muestra test y a los valores SHAP. En esta imagen vemos el azul como la disminución de la probabilidad de fallecer (aporte negativo) y la parte roja como un aumento de la probabilidad (aporte positivo). Así vemos cómo la Edad es mayoritariamente para negativo. En ICC hay cierto balance, aunque cuando afecta de manera negativa lo hace de forma superior a cuando lo hace de manera positiva, el reingreso mayoritariamente afecta si es de manera negativa…

Si vemos qué ocurre con los *falsos positivos*:

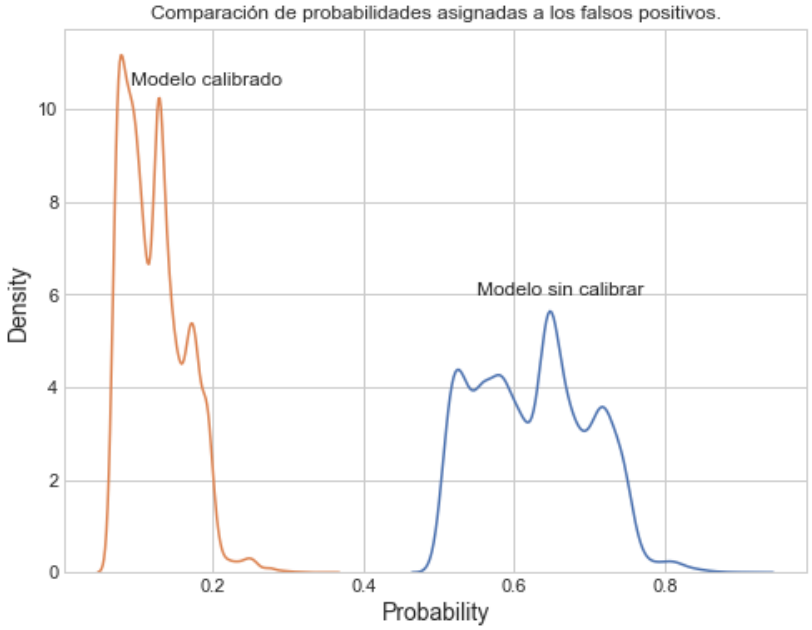


Figura 4.7 Distribución de probabilidad de falsos positivos.

Aquí vemos que, si miramos únicamente a los *falsos positivos*, el modelo calibrado es obvio que consigue neutralizar esos casos. Sin embargo, esta imagen está sesgada, ya que únicamente se está mostrando la probabilidad sobre los *falsos positivos*. Si vemos cómo se comportan estos modelos frente a la muestra completa:

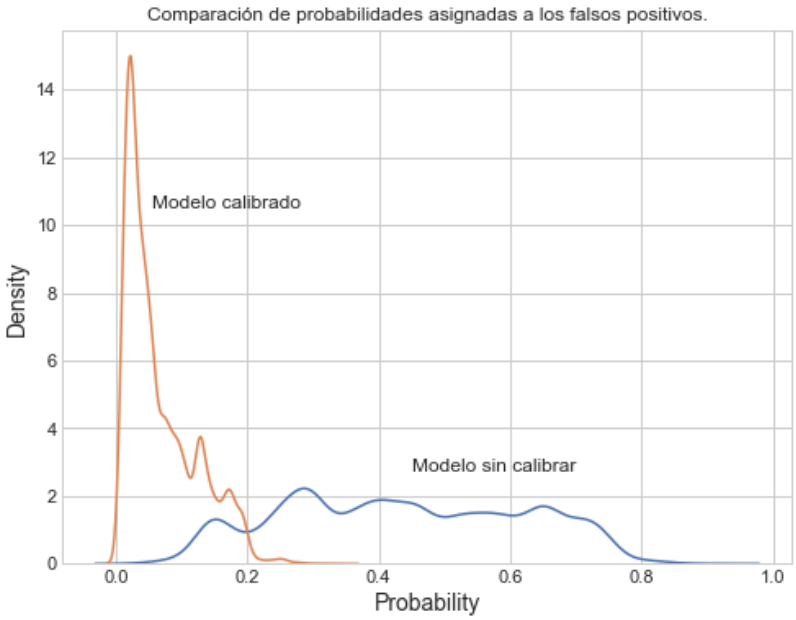
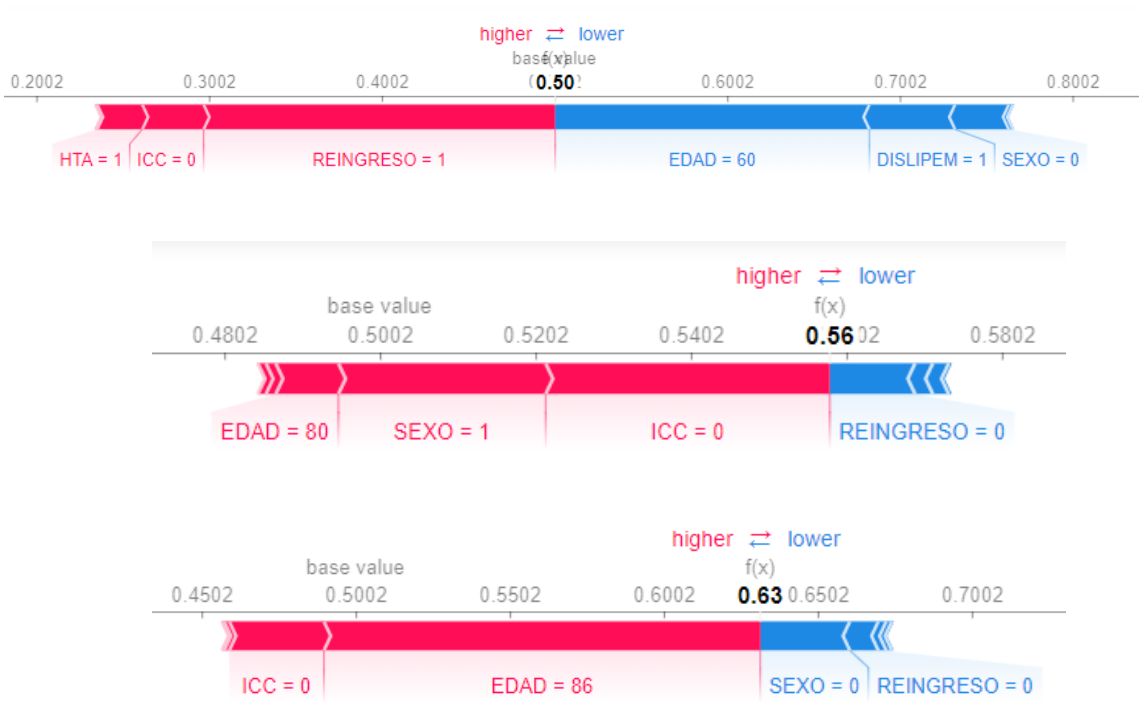


Figura 4.8 Distribución de probabilidades de modelo calibrado y no calibrado.

Podemos observar entonces que el modelo calibrado está sesgado de alguna manera por la muestra desbalanceada, y acaba siendo de alguna manera un clasificador trivial, el modelo sin calibrar ha captado un poco mejor la sensibilidad.

Es interesante ver cómo la mayoría de estos casos presentan una probabilidad entre 0.5 y 0.76, por tanto, se podría corregir subiendo un poco el umbral de clasificación. Aunque lo que nos incumbe es si la probabilidad asignada tiene sentido, veamos algún ejemplo.



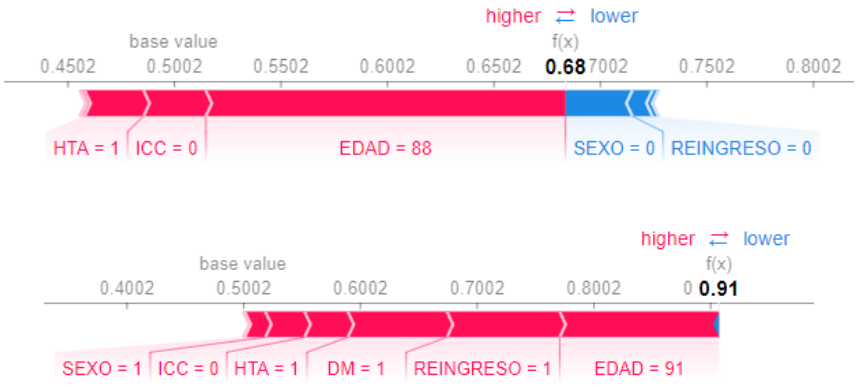


Figura 4.9 Shap values de los falsos positivos.

A primera vista no parece que estas probabilidades estén sobreestimadas, lo que ocurre es que tenemos muchos casos, que, aun siendo pacientes donde muchos con esas mismas características han fallecido, pueden no fallecer, y eso causa que aumente la cantidad de falsos positivos. Esto es debido en su mayor parte por la Edad.

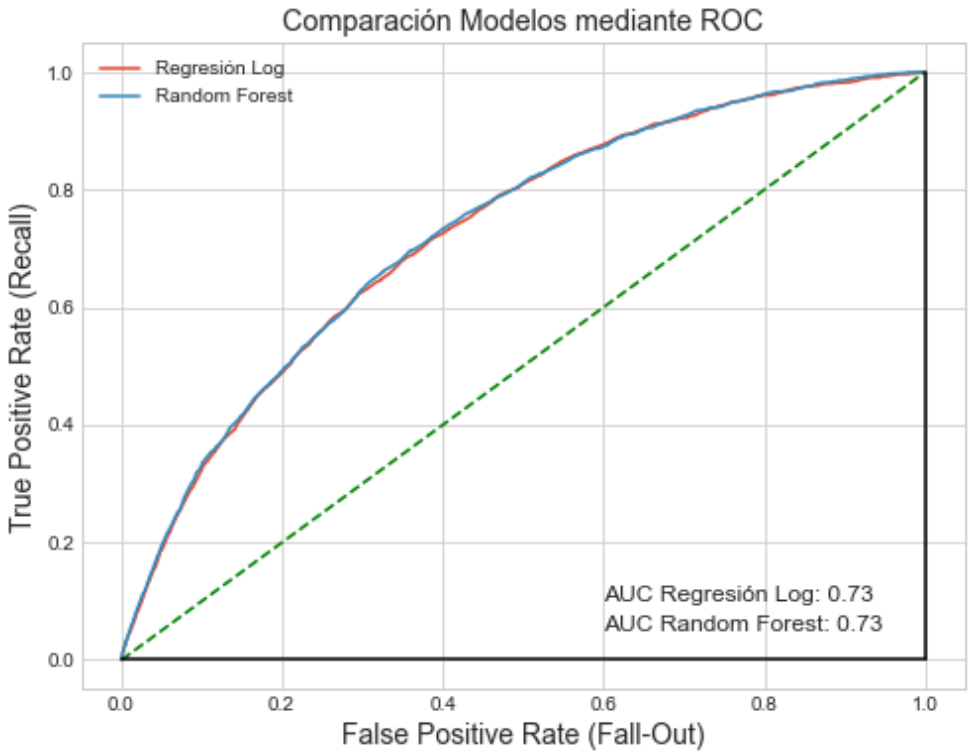


Figura 4.10 Comparación de Modelos por ROC.

4.5 Selección recursiva de predictores

Una pregunta que podríamos hacernos es la siguiente; A lo largo de esta sección hemos visto cómo los primeros predictores, EDAD, ICC, REINGRESO, SEXO tienen una importancia muy por encima del resto de predictores. ¿Cómo se vería afectado el modelo si empezamos a eliminar algunos de los predictores que tienen menor importancia?

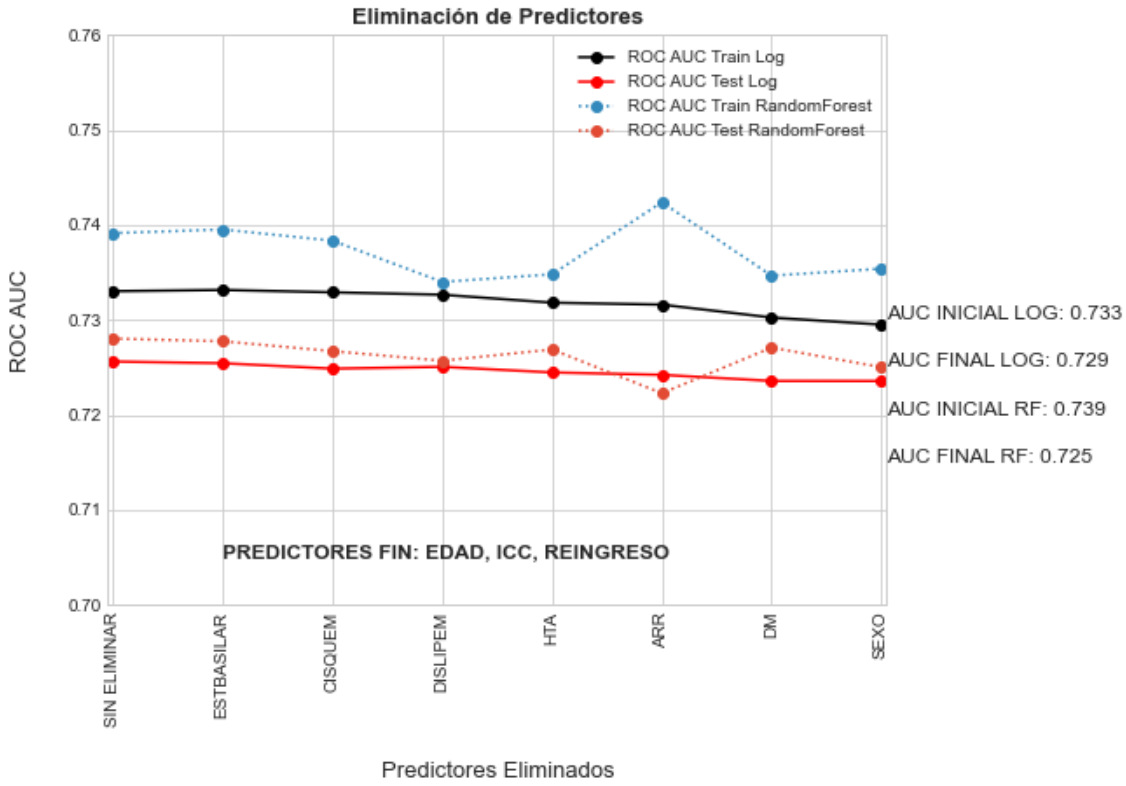


Figura 4.11 Poder ROC mediante eliminación recursiva de predictores.

4.6 Conclusiones del capítulo

Como hemos visto hasta ahora, con los dos modelos iniciales, conseguimos un poder ROC medianamente considerable, pero qué ocurre con los resultados:

- La medicina no es una ciencia exacta, y según hemos visto, los falsos positivos tienen bastante sentido en cuanto a la asignación de probabilidad estimada, por ejemplo, en el último caso, persona de 91 años, mujer, con DM, REINGRESO, HTA, … y, sin embargo, no fallece. Eso puede ser debido a que en el momento en que se tomaron los datos no había fallecido aún o que se salvan casos incluso en las situaciones más complicadas debido a los avances científicos. Por eso tenemos tal grado de "error" y en parte no es malo que el modelo sobreestime un poco las probabilidades,

que hemos visto que tienen cierto sentido, pero que, para calibrarlo, tendríamos que reducir de alguna manera las probabilidades estimadas por el modelo.

- Con estos dos primeros modelos, la tasa de falsos positivos es muy alta, debido a que es preferible este caso al falso negativo en este problema concreto.
- Se han probado otros modelos como VotingClassifiers, k-NN, redes neuronales y XGBoost, y no se obtienen mejores resultados. De hecho, la cantidad de *falsos negativos* aumenta considerablemente.
  La manera de mejorar estos clasificadores que suelen ser más potentes en cuanto a complejidad y coste computacional suele ser balancear la muestra hasta un 50-50, y aunque mejoran los resultados en cuanto a la clasificación, el poder ROC no acaba de subir por encima de 0.75.

Nota:

Tampoco esperamos que la ROC suba mucho más de 0.75 debido a la cantidad de "ruido" que tiene la base de datos, pero veremos si podemos mejorar este primer resultado, sin hacer uso del clasificador "perfecto".

También usaremos toda la base de datos en la cual aplicaremos un algoritmo genético para selección de predictores y desarrollo de un modelo post-ingreso, donde las variables que incluiremos no serán escogidas por criterio experto.

Además, podríamos plantearnos eliminar algunas de las variables que tienen menor importancia, y el modelo no se vería afectado en cuanto a capacidad de discriminación.

Capítulo 5

Escenario B

5.1 Introducción

Al final del capítulo 3, obtuvimos los resultados del lanzamiento del algoritmo genético de selección de variables. Alcanzamos un individuo óptimo formado por 20 variables que optimiza los resultados de un randomforest entrenado con F1-Score.
Para empezar, nos quedaremos con dichas variables y procedemos a lanzar los modelos del escenario A.

5.2 Resultado de los modelos

	Accuracy	**Bal-Acc**	**F1-Score**	**AUROC**	**Precis.**	**Recall**
Balanced Logistic Regression	0.680313	0.703465	0.235936	0.770848	0.140699	0.730236
Random Undersample + Logistic Regression	0.683414	0.703810	0.236999	0.770645	0.141567	0.727394
Balanced random forest	0.688579	0.716228	0.245163	0.788986	0.146609	0.748200
Under-sampling + Random forest	0.671052	0.679860	0.220933	0.737112	0.131527	0.690045

Importancia de predictores por RandomForest:

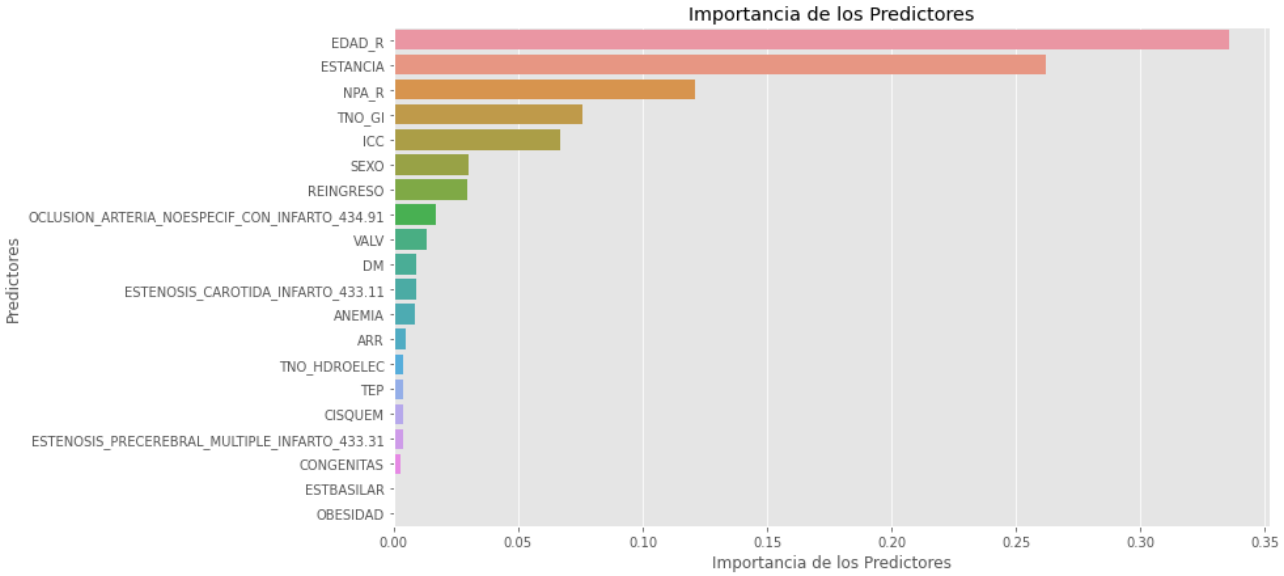


Figura 5.1 Importancia predictores escenario B.

La importancia de la permutación de la característica se define como la disminución de la puntuación de un modelo cuando se baraja aleatoriamente un único valor de la característica. Este procedimiento rompe la relación entre la característica y el objetivo, por lo que la disminución de la puntuación del modelo es indicativa de cuánto depende el modelo de la característica. Esta técnica se beneficia de ser agnóstica al modelo y puede calcularse muchas veces con diferentes permutaciones de la característica [28].

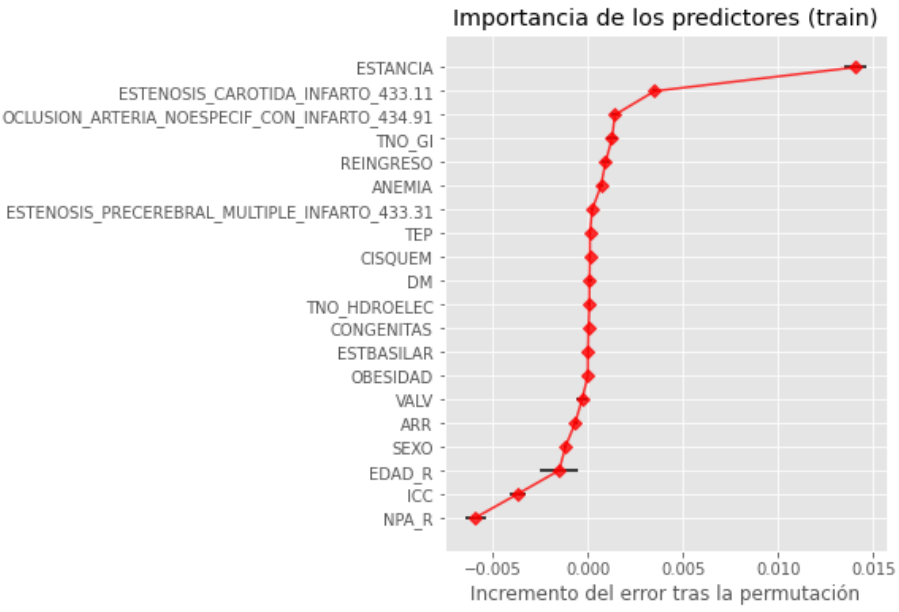


Figura 5.2 Importancia por permutación.

Si comparamos entonces los modelos mediante la curva ROC en el conjunto de Test:

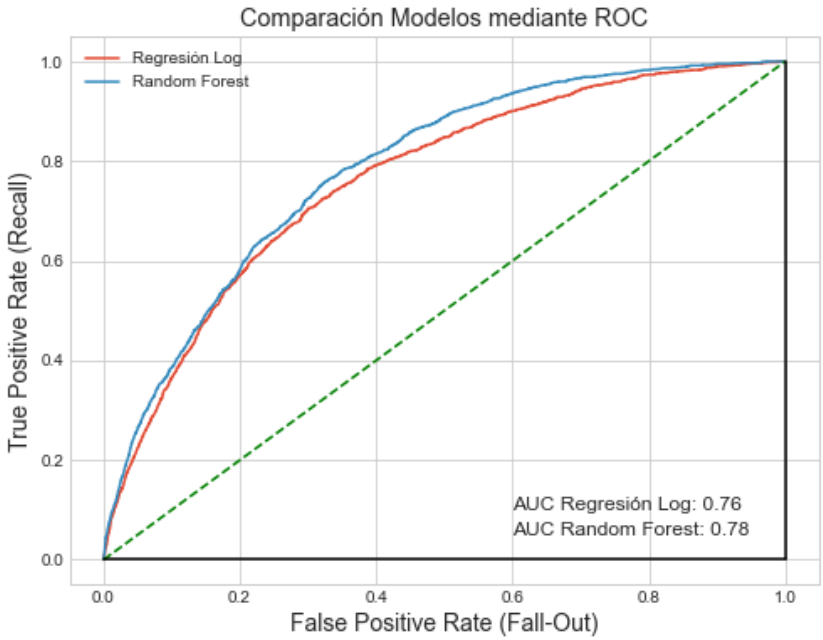


Figura 5.3 Comparación modelos escenario B.

Hemos obtenido un modelo post-ingreso (muchas de las variables requieren del ingreso del paciente para poder ser obtenidas) que es capaz de discriminar mejor que el que teníamos de partida.

Además, es posible que se puedan reducir la cantidad de predictores del modelo sin que ello conlleve un empeoramiento de los resultados **(Figura 5.4)**.

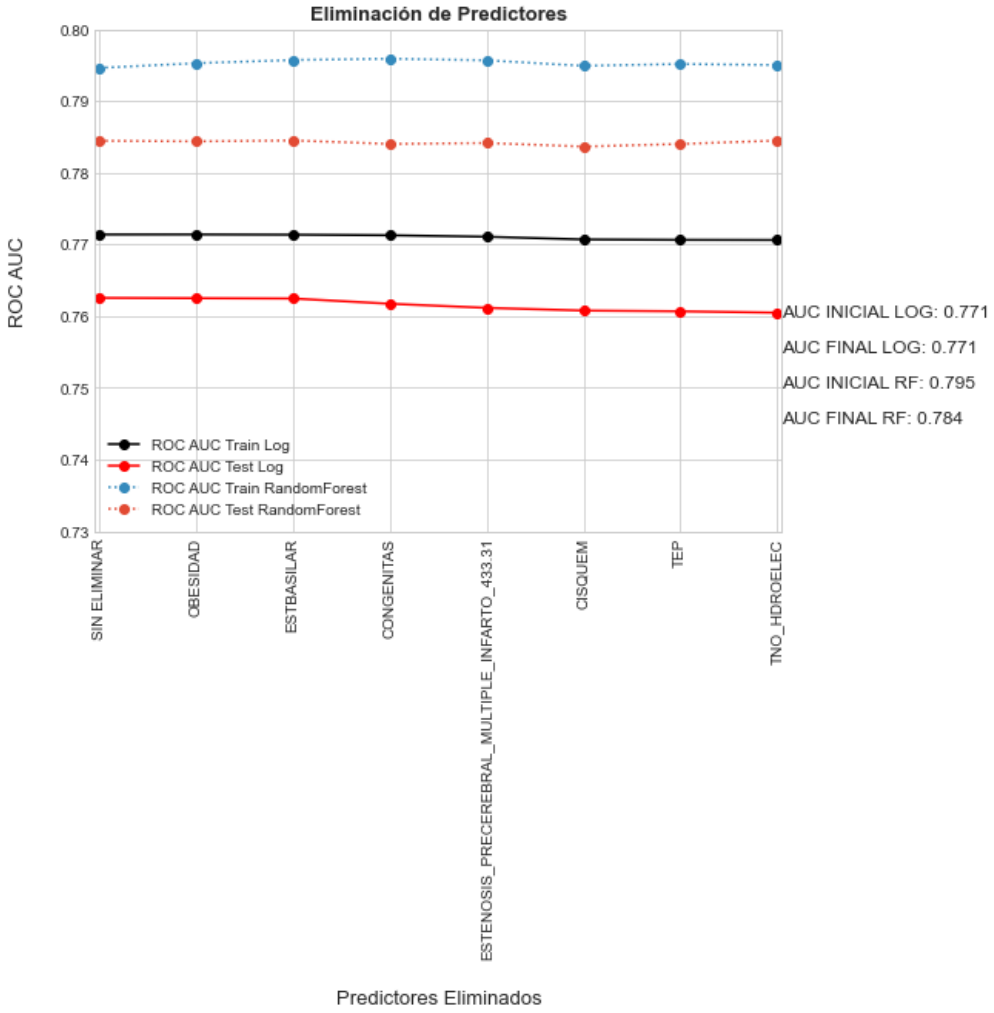


Figura 5.4 Eliminación recursiva de predictores sin importancia.

Si ahora relanzamos el modelo partiendo de selección de predictores cuya importancia es superior a 0.01, se obtiene la siguiente gráfica (**Figura 5.5)**.

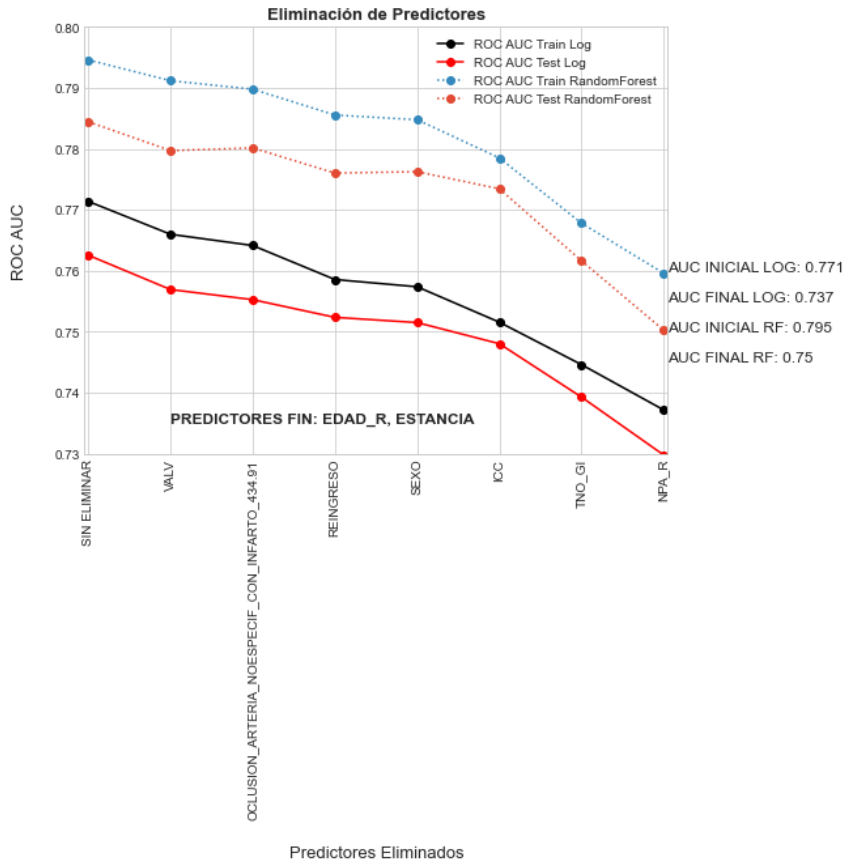


Figura 5.5 Eliminación recursiva de predictores importantes.

Vemos como ahora sí, el modelo empieza a sufrir mucho la eliminación de variables, por tanto, podría ser un buen criterio el seleccionar predictores cuya importancia asignada por el randomforest superase cierto umbral.

5.3 Conclusiones del capítulo

A lo largo de este capítulo, se ha abordado la base de datos en su completo, y aplicando únicamente técnicas de Ciencia de datos (al contrario que en el otro escenario donde los predictores han sido escogidas por criterio experto), se han obtenido unos resultados que *mejoran lo que habíamos logrado usando únicamente las variables del escenario A*.

Posteriormente, se ha utilizado una eliminación recursiva de predictores basándose en la importancia asignada por el RandomForest, y hemos visto que se pueden eliminar aquellas que tienen importancia menor a 0.01, sin que esto afecte demasiado al modelo.

Por tanto, podemos obtener un modelo en el escenario B, que teniendo aproximadamente la misma cantidad de variables (pero diferentes), consigue una mejor discriminación de las dos clases.

Capítulo 6

Conclusiones y trabajo futuro

6.1 Introducción

A lo largo de este capítulo se entenderá por aplicación, la arquitectura integrada por un sistema de peticiones (API), un modelo que haga inferencia sobre dichas peticiones, y una base de datos que acabe registrando todo el proceso.

El desarrollo futuro de este proyecto pasa por la puesta en producción, monitorización y recalibrado de los modelos. Para ello, en primer lugar, necesitaremos montar una estructura de soporte de peticiones, donde a su vez pueda desplegarse el modelo. Una vez montada dicha estructura (aplicación), será de vital importancia la colaboración por parte de los sanitarios que utilicen dicha aplicación para recoger nuevas observaciones, usarla para producir inferencia sobre éstas y, en un futuro, contrastar las estimaciones realizadas con las observadas.

6.2 ¿Qué modelo poner en producción?

A lo largo de los dos capítulos anteriores hemos visto las diferencias entre los distintos modelos (regresión logística y random forest), pero además, hemos observado también sus distintas versiones (sin calibrar y calibrado).

Por un lado, hemos observado que los clasificadores calibrados tienden a acertar más, pero no era el accuracy la métrica por la que estábamos escogiendo los modelos.

Si atendemos a las estimaciones realizadas por estos modelos, obtendremos muchos casos negativos, debido a que tiende a estandarizar más las probabilidades según la verdadera distribución de clases. Pero tengamos en cuenta que la distribución está claramente desbalanceada. Por tanto, si usamos este tipo de modelos, tendremos más "acierto" en cuanto a accuracy se refiere, pero disminuiremos la asignación de riesgo a los diversos pacientes. Lo que hemos visto, que sí que es asignado de una manera más correcta con el random forest sin calibrar.

Una asignación de riesgo considerable a un paciente recaerá en una atención más exhaustiva y precoz, que finalmente se inclinará por un *falso positivo*, es decir, se le habrá salvado la vida al paciente.

¿Con ello queremos decir que a todos los pacientes haya que tratarlos sobreestimando el riesgo? Pues claramente no, recordemos que con esto se prevé optimizar costo-eficiencia, por tanto, podría ser interesante hacer alguna ligera ponderación de las probabilidades estimadas por el modelo sin calibrar.

6.3 Servicios AWS

Todo el trabajo se ha realizado usando un entorno interactivo como es Jupyter Notebooks, y en un futuro, el objetivo sería trasladarlo a los servicios de AWS.

Podríamos montar la aplicación, en primer lugar, usando SageMaker, ya que facilita mucho el proceso de puesta en producción del modelo. Ya que permite desarrollar, entrenar y desplegar bajo un control de versiones y un hosting gestionado y autoescalable.

Por tanto, únicamente necesitaríamos hacer uso de un bucket de S3 donde poder almacenar los datos con los que entrenaremos el modelo, el modelo, y las inferencias realizadas por éste.

Aunque la idea es usar alguna base de datos, como pueda ser DynamoDB de cara a una mayor escala de la puesta en producción del modelo.

Desde otro punto, se podrían implementar Lambdas para que, cada cierto tiempo, se fuese reentrenando el modelo con datos nuevos, montando una arquitectura que no dependiera de SageMaker, haciendo únicamente uso de S3, API Gateway, Lambdas y DynamoDB o RDS.

6.4 Conclusiones del TFM

A lo largo de este TFM, se ha hecho un recorrido por distintas técnicas de Ciencia de Datos para abordar un problema de estas características. Es la primera vez que realizo un trabajo de estas características de manera un poco más profesional, que se espera que tenga cierta aplicación y, además, sobre un problema que está abierto.

Con ello me refiero a que no existe aún ningún modelo que de forma completamente determinista sea capaz de estimar un riesgo de mortalidad de manera perfecta.

La idea también ha sido acercar a la Medicina algunas técnicas de Ciencia de Datos que no son tan comunes aún, como es el uso de modelos más complejos que los modelos lineales, el estudio de explicabilidad/interpretabilidad de estos, la selección de predictores usando diversas técnicas…

De cara al futuro esperamos reunir nuevos datos que hayan sido tomados de manera más rigurosa (datos obtenidos por médicos que colaboran en el proyecto), y poder realizar un estudio similar al realizado con esos nuevos datos. También que sirva como una validación externa de los modelos propuestos hasta ahora.

Bibliografía

0. Memoria de Registro del Modelo a Validar, Modelo predictivo para la mortalidad hospitalaria en el ictus isquémico no lisado. Juan Manuel García Torrecillas.
1. Estrategia en Ictus del Sistema Nacional de Salud: Ministerio de Sanidad y Política Social; 2009. 163 p.
2. Feigin VL, Lawes CM, Bennett DA, Anderson CS. Stroke epidemiology: a review of population-based studies of incidence, prevalence, and case-fatality in the late 20th century. The Lancet Neurology. 2003;2(1):43-53.
3. Arboix A, García-Eroles L, Comes E, Oliveres M, Targa C, Balcells M, et al. Importancia del perfil cardiovascular en la mortalidad hospitalaria de los infartos cerebrales. Revista espanola de cardiologia. 2008;61(10):1020-9.
4. Smith EE, Shobha N, Dai D, Olson DM, Reeves MJ, Saver JL, et al. A risk score for in-hospital death in patients admitted with ischemic or hemorrhagic stroke. Journal of the American Heart Association. 2013;2(1): e005207.
5. Whisnant JP. Modeling of risk factors for ischemic stroke. The Willis Lecture. Stroke; a journal of cerebral circulation. 1997;28(9):1840-4.
6. Anon. n.d. “El Conjunto Mínimo Básico de Datos (CMBD) Como Fuente Para La Investigación En Medicina Interna. - Por Una Medicina Interna de Alto Valor.” (https://medicinainternaaltovalor.fesemi.org/instrumentos-y-agrupadores-

necesarios-en-una-medicina-moderna/el-conjunto-minimo-basico-de-datos-cmbd-como-fuente-para-la-investigacion-en-medicina-interna/).

7. López González, Ramiro A. n.d. *XXIX CONGRESO NACIONAL DE LA SEMI. A CORUÑA, CMBD*.(https://www.fesemi.org/sites/default/files/documentos/ponencias/xxix-congreso-semi/Dr.%20Lopez%20GOnzalez.pdf)
8. "Tablas de mortalidad - Instituto Nacional de Estadística. (National ..." 2015. 26 Jun. 2016 <http://www.ine.es/jaxi/menu.do?type=pcaxis&path=/t20/p319a/&file=inebase>.
9. Van Der Maaten, Laurens, and Geoffrey Hinton. 2008. *Visualizing Data Using T-SNE*. Vol.9.(https://jmlr.csail.mit.edu/papers/volume9/vandermaaten08a/vandermaaten08a.pdf)
10. Anon. n.d. "An Illustrated Introduction to the T-SNE Algorithm – O'Reilly." Retrieved June 23, 2021 (https://www.oreilly.com/content/an-illustrated-introduction-to-the-t-sne-algorithm/).
11. Anon. n.d. "T-SNE – Laurens van Der Maaten." Retrieved June 23, 2021 (https://lvdmaaten.github.io/tsne/).
12. Wattenberg, Martin, Fernanda Viégas, and Ian Johnson. 2017. "How to Use T-SNE Effectively."*Distill*1(10):e2.doi:10.23915/distill.00002. (https://distill.pub/2016/misread-tsne/)
14. Altmann, André, Laura Toloşi, Oliver Sander, and Thomas Lengauer. 2010. "Permutation Importance: A Corrected Feature Importance Measure." *Bioinformatics* 26(10):1340–47.doi:10.1093/bioinformatics/btq134. (https://academic.oup.com/bioinformatics/article/26/10/1340/193348)
15. Anon. n.d. "Permutation Importance vs Random Forest Feature Importance (MDI) — Scikit-Learn 0.24.2 Documentation." Retrieved June 25, 2021 (https://scikit-learn.org/stable/auto_examples/inspection/plot_permutation_importance.html#sphx-glr-auto-examples-inspection-plot-permutation-importance-py).
16. Breiman, Leo. 2001. "Random Forests." Machine Learning 45(1):5–32. doi: 10.1023/A:1010933404324. (https://doi.org/10.1023/A:1010933404324)
17. Anon. n.d. "3.2. Tuning the Hyper-Parameters of an Estimator — Scikit-Learn 0.24.2 Documentation."Retrieved June 25, 2021 (https://scikit-learn.org/stable/modules/grid_search.html).

18. Anon. n.d. “Sklearn.Model_selection.GridSearchCV — Scikit-Learn 0.24.2 Documentation.” Retrieved June 25, 2021 (https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html).

19. Anon. n.d. “Welcome to the SHAP Documentation — SHAP Latest Documentation.” Retrieved June 28, 2021 (https://shap.readthedocs.io/en/latest/index.html).

20. Anon. n.d. “Seleccion de Predictores Mediante Algoritmo Genetico.” Retrieved June 29, 2021 (https://www.cienciadedatos.net/documentos/py03_seleccion_predictores_ga).

22. “Calibrar Modelos de Machine Learning.” (https://www.cienciadedatos.net/documentos/py11-calibrar-modelos-machine-learning.html).

23. Obtaining calibrated probability estimates from decision trees and naive Bayesian classifiers, B. Zadrozny & C. Elkan, ICML 2001

24. Transforming Classifier Scores into Accurate Multiclass Probability Estimates, B. Zadrozny & C. Elkan, (KDD 2002)

25. Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods, J. Platt, (1999)

26. Predicting Good Probabilities with Supervised Learning, A. Niculescu-Mizil & R. Caruana, ICML 2005

27. Japkowicz, N. (2000). The Class Imbalance Problem: Significance and Strategies. In Proceedings of the 200 International Conference on Artificial Intelligence (IC-AI’2000): Special Track on Inductive Learning Las Vegas, Nevada.

28. (*4.2. Permutation Feature Importance — Scikit-Learn 0.24.2 Documentation*, n.d.)