

Memoria Simplificada

Modelo Prevision Ventas Productos Financieros_

Tabla de Contenidos

1. INTRODUCCION
2. PROLOGO
3. OBJETIVOS
4. FINALIDAD
5. IMPORTANCIA DEL PROBLEMA A RESOLVER
6. INFORMACION DEL REPOSITORIO
7. DATOS UTILIZADOS
8. REQUISITOS TECNICOS
9. GUIA EJECUCION Y CARGA BASE DE DATOS
10. CONSTRUCCION DE LA BASE DE DATOS FINAL
11. PREPROCESADO
12. CONTEXTO GENERAL
13. NUESTRA BASE DE DATOS
14. CORRELACIONES ENTRE LAS VARIABLES
15. ESTUDIO DEL TARGET

16. MODELO

17. CONCLUSIONES

```
In [1]: # CAMBIAR SEGUN RUTA LOCAL DONDE SE HAYA UBICADO EL REPOSITORIO CLONADO.  
ruta = '/Users/manue/TFM'
```

```
In [2]: from IPython.display import Image
```

1 Motivacion personal

Desde 1998 hasta la actualidad, he desarrollado mi vida profesional en una entidad financiera, principalmente en Banca de particulares. Durante estos 23 años he pasado por todas las categorías laborales posibles dentro de una oficina comercial abierta al consumidor. Desde comercial de caja y de mesa, a subdirector y director de oficina.

En 2018, motivado por la búsqueda de nuevas habilidades, reciclaje laboral y personal, la adaptación a la nueva realidad de transformación digital y la necesidad de construir un plan alternativo debido a las inciertas perspectivas laborales, huyendo de mi zona de confort decido cursar un Master en Bussines Analytics con la intención de aprender nuevas formas de análisis de negocio y poder ponerlas en práctica.

Durante el curso me doy cuenta que aun sin ninguna base de programación o informática, estadística o matemáticas, procediendo de una licenciatura de letras, había encontrado una motivación, una nueva parcela de estudio y un nuevo reto.

Decido continuar la formación con el Master en Data Science de K-School, recomendado por un antiguo profesor y siempre avisado de la dificultad técnica del mismo.

El resultado lo puedo definir en una frase. Intenso pero entusiasmado y con ganas de continuar mi formación.

2 Prologo.

Los bancos son entidades importantes. Canalizan la riqueza entre los distintos actores de la sociedad. Ese ha sido siempre su principal objetivo y razón de ser. Captar recursos de aquellos que los tienen, remunerarles por esa captación y posteriormente prestarlos recibiendo a cambio un tipo de interés mayor.

Se trata de una transacción sencilla y fácil. Las entidades financieras pagan por captar dinero y cobran por prestarlo. De esta forma todas las partes salen beneficiadas. Quien pone a disposición del banco un dinero que no necesita, recibe un pago por ello, el banco por otro lado cobra intereses al prestarlo y aquellos que reciben el préstamo pueden destinar el dinero para aquello que necesitan. Una empresa, una compra de un vehículo...la compra de una casa... etc.

Resumiendo, el negocio tradicional de las entidades financieras ha sido captar, prestar y cobrar por el dinero que ha captado.

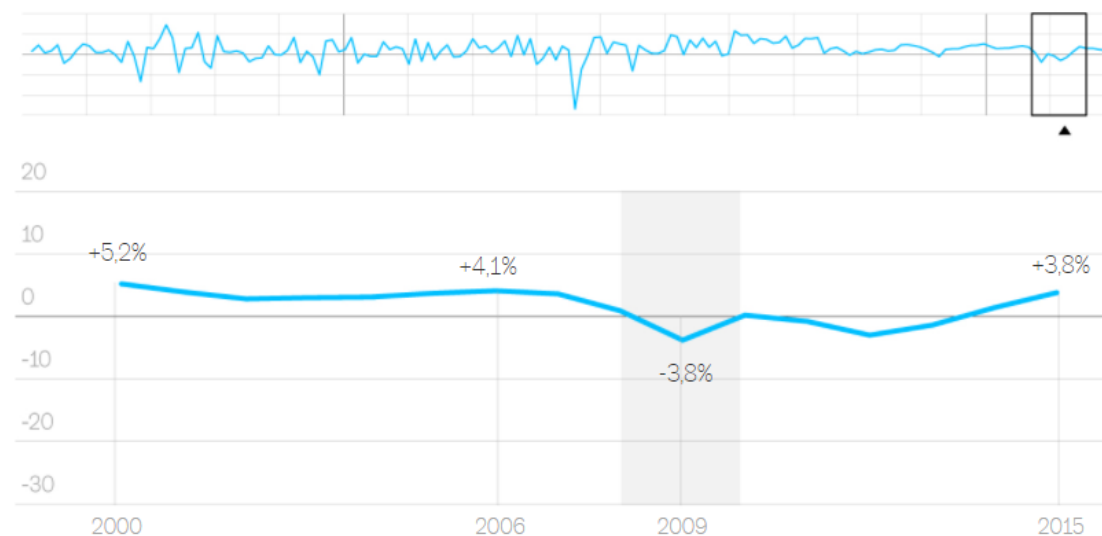
En España, hasta 2008 vivimos en un gran periodo de crecimiento económico impulsado principalmente por un modelo basado en el mercado inmobiliario. Este crecimiento se vio bruscamente interrumpido por la crisis de las hipotecas subprime. Se inicio un largo periodo de 6 años de crisis profunda y recesión en la que los principales indicadores económicos de la sociedad sufrieron importantes caídas. Las entidades financieras no fueron ajenas a este sufrimiento viendo sus cuentas de resultados mermadas y reducidas considerablemente por este efecto y por las bruscas caídas de tipo de interés. El negocio tradicional bancario se había paralizado ,se había acabado.

```
In [3]: Image(filename=(ruta + '/Financial-Product-Sales-Forecast-Model/Images/RECESION2008.png'), width = 600)
```

Out[3]:

La Gran Recesión (2008-2013)

Variación anual del PIB en %



Desde entonces, la venta de otros productos financieros como los Fondos de Inversión, Planes de Pensión y los seguros de riesgo entre otros, han

conseguido acaparar toda la importancia. Hoy en día, la generación de comisiones adheridas a la comercialización de estos productos ha supuesto un vuelco en la estrategia comercial, especializándose y poniendo el foco en su venta mediante el asesoramiento especializado a los clientes por parte de los empleados de las sucursales.

3 Caso de Uso. Importancia del problema a resolver

Dentro del sector financiero existen múltiples modelos de clasificación relacionados con las distintas necesidades.

- Modelos de Riesgo
- Modelos de Morosidad
- Modelos de RRHH
- Modelos de Segmentación Clientes
- Modelos de clientes digitales
- Modelos de ventas FFII – PP – Seguros
- **Modelos de ventas de Seguros**

El seguro de hogar está muy arraigado en la sociedad española. La cantidad del presupuesto del hogar destinada a los seguros , depende de muchos factores estructurales y contextuales de cada hogar, como el lugar de residencia, la situación laboral de la familia, la escala salarial de sus integrantes, etc.

Modelo. Con los datos y variables disponibles intentaremos entrenar una solución comercial que facilite la venta de este producto a los asesores de las entidades financieras.

La generación de comisiones es fundamental para la cuenta de resultados de la oficina y del banco. Los seguros de hogar se quedan en cartera durante un periodo medio de 5 años. Cada seguro de hogar contratado deja una comisión directa del 15 %. Esto sobre un seguro de hogar de prima media de 300€ supone 45€ de comisión anual. Nuestra base de datos correspondiente únicamente a 162 oficinas y 450.000 clientes podría llegar a generar unas comisiones anuales de más de 17mm€ en caso de que todos los clientes que no disponen de seguro de hogar lo contratasen. Solo con esta cifra y extrapolándola a un colectivo de clientes totales de 3 – 4 millones de clientes, queda más que explicado la evidente y clara la necesidad de identificar potenciales clientes que sean susceptibles de contratar el seguro de hogar.

4 Objetivos.

A través de un conjunto de datos pertenecientes a 450.000 clientes he querido desarrollar un modelo predictivo de potencial compra de estos productos financieros, concretando en el Seguro de Hogar. Un modelo predictivo de clasificación que ayude a toda la fuerza comercial de las sucursales a orientar la comercialización, a optimizar los tiempos, metodologías y sistemáticas utilizadas. Todo ello en búsqueda de un mayor éxito de ventas, satisfacción de los clientes y generación de margen para la entidad financiera.

Mi objetivo final será implementar una aplicación (CallorNot) donde mediante un cuestionario, incorporando una serie de características de un cliente podamos informar al gestor si el cliente es susceptible de contratar o no un seguro de hogar, sugiriendole finalmente si llamar o no llamar al cliente.”

5 Finalidad.

Busqueda de mayores ventas, mejores oportunidades, mayores márgenes y satisfacción final de los clientes

6 Información del Repositorio

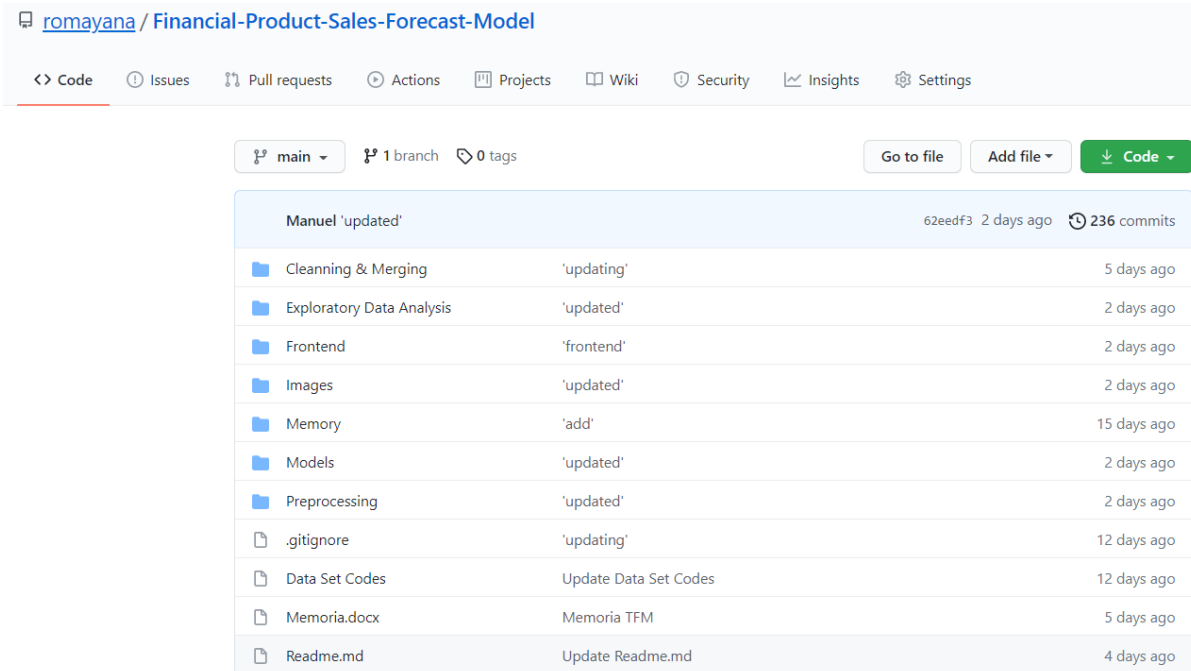
Toda la información de este TFM ha quedado recogida en un repositorio de GitHub al cual se accede a través de la siguiente dirección.

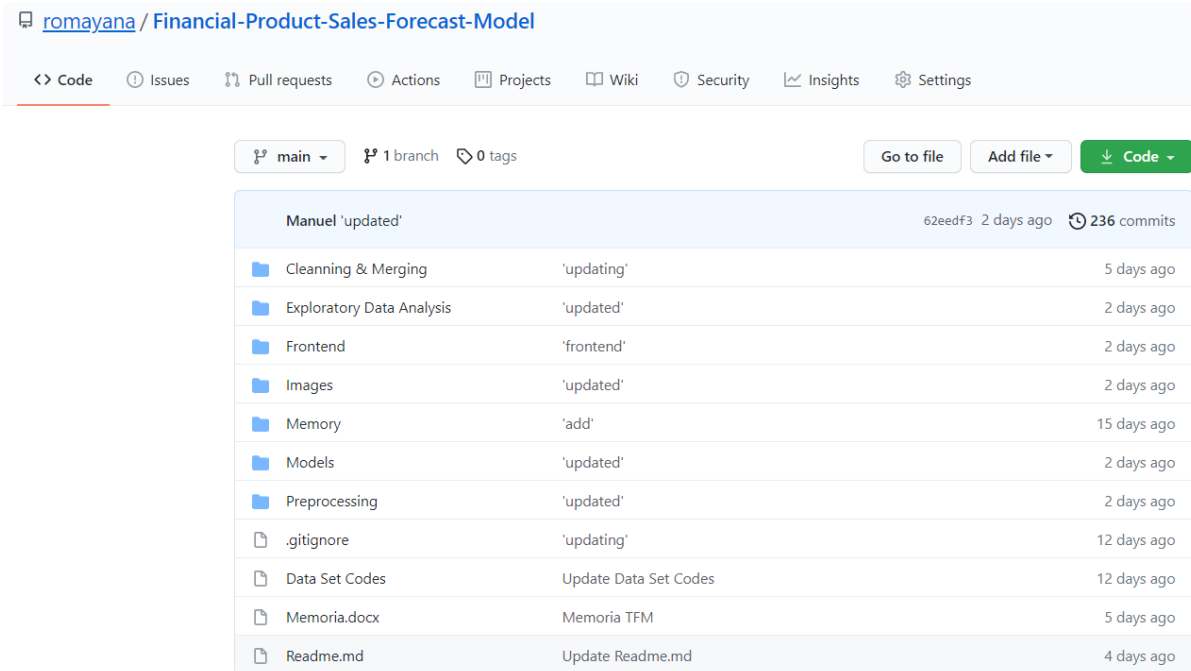
<https://github.com/romayana/Financial-Product-Sales-Forecast-Model.git> (<https://github.com/romayana/Financial-Product-Sales-Forecast-Model.git>)

El repositorio se estructura en 7 carpetas y 3 archivos. (Según posición en repositorio)

- Carpeta 1 – Códigos Python Limpieza y Unión
- Carpeta 2 – Códigos Python EDA Análisis Exploratorio
- Carpeta 3 – Frontend. Aplicación creada para nuestro modelo (APP CallorNot.)
- Carpeta 4 – Imágenes .png guardadas de cada una de las gráficas construidas
- Carpeta 5 – Memoria. Documentos y notebooks memoria TFM.
- Carpeta 6 – Códigos Python Modelos clasificación utilizados.
- Carpeta 7 – Códigos Python Preprocesado
- Archivo 1 – archivo .gitignore. Archivos descartados en las actualizaciones del repositorio
- Archivo 2 – Diccionario e información del significado de las variables
- Archivo 3 – Readme con primera información del Trabajo y comunicación de expectativas

```
In [4]: Image(filename=(ruta + '/Financial-Product-Sales-Forecast-Model/Images/IMAGENREPOSITORIO.png'),width = 600)
```

```
Out[4]: 
```



The screenshot shows the GitHub repository interface for 'romayana / Financial-Product-Sales-Forecast-Model'. At the top, there are navigation links: Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. Below these, there are buttons for 'Go to file', 'Add file', and 'Code'. The repository is currently on the 'main' branch, with 1 branch and 0 tags. The commit history shows a commit by 'Manuel' updated 2 days ago, with 236 commits in total. The file list includes folders for 'Cleaning & Merging', 'Exploratory Data Analysis', 'Frontend', 'Images', 'Memory', 'Models', and 'Preprocessing', as well as files like '.gitignore', 'Data Set Codes', 'Memoria.docx', and 'Readme.md'.

File/Folder	Commit Message	Time Ago
Manuel 'updated'	62eedf3	2 days ago
Cleaning & Merging	'updating'	5 days ago
Exploratory Data Analysis	'updated'	2 days ago
Frontend	'frontend'	2 days ago
Images	'updated'	2 days ago
Memory	'add'	15 days ago
Models	'updated'	2 days ago
Preprocessing	'updated'	2 days ago
.gitignore	'updating'	12 days ago
Data Set Codes	Update Data Set Codes	12 days ago
Memoria.docx	Memoria TFM	5 days ago
Readme.md	Update Readme.md	4 days ago

7 Datos Utilizados.

La información y explicación detallada de las variables que conforman la base de datos se encuentra dentro de una de los archivos de este repositorio con el nombre de Data-Set-Codes.

La extracción de los datos se ha realizado desde un sistema de información de gestión de una entidad financiera. Sistema de información que guarda millones de datos de tipo financiero y económicos.

Toda la información y datos necesarios para el estudio del modelo de clasificación se han obtenido de forma directa y con permisos limitados de una entidad financiera real. Permisos limitados ya que no se ha podido disponer de mucha información que hubiese mejorado el modelo. Información como por ejemplo género, estado civil, hijos, clase económica, renta disponible, importe de nómina, detalle de compras realizadas, detalle de llamadas comerciales realizadas y otras muchas variables.

Para la construcción de la base de datos final se han ido descargando de este sistema de información de gestión y de forma manual, archivos individuales extensión xlsx, relacionados con distintos epígrafes como saldos en cuenta, saldos en fondos de inversión o planes de pensión, líneas de producto, tarjetas, seguros o tipo de segmentación. En total han sido 99 archivos Excel descargados. 9 archivos por cada una de las 11 Direcciones de Zona disponibles lo que ha generado finalmente una base de datos de 15 millones de datos de 450.000 clientes.

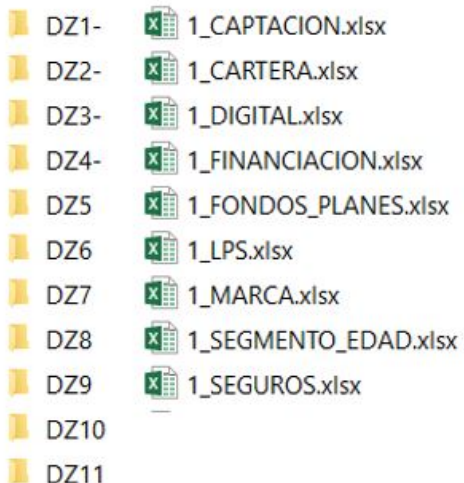
El peso total de los datos originales es de 112 MB (117.706.752 bytes) los cuales después de haber sido limpiados se han quedado en 70,3 MB (73.742.213 bytes).

La carpeta origin_data está compuesta por las distintas carpetas y archivos Excel descargados ya anonimizados. Los datos se han anonimizado previamente al guardado en esta carpeta. Numero de identificador de cliente ha sido cambiado por una secuencia desde 1 a 450.000. Números y códigos distintivos de las Direcciones de Zona han sido cambiados por una secuencia del 1 al 11. Nombres , direcciones y números de identificación fiscal de los clientes han sido eliminados.

Detalle Carpeta con Datos Originales. 11 carpetas x 9 archivos.

```
In [5]: Image(filename=(ruta + '/Financial-Product-Sales-Forecast-Model/Images/IMAGENCONSTRUCCIONBD.png'),width = 250)
```

```
Out[5]:
```



- DZ1- 1_CAPTACION.xlsx
- DZ2- 1_CARTERA.xlsx
- DZ3- 1_DIGITAL.xlsx
- DZ4- 1_FINANCIACION.xlsx
- DZ5- 1_FONDOS_PLANES.xlsx
- DZ6- 1_LPS.xlsx
- DZ7- 1_MARCA.xlsx
- DZ8- 1_SEGMENTO_EDAD.xlsx
- DZ9- 1_SEGUROS.xlsx
- DZ10
- DZ11

8. Requisitos Tecnicos

Para ejecutar los códigos es necesario tener instalado Python versión 3.8 así como distintos paquetes o librerías. Se recomienda tener instalada la Suite Anaconda donde se encontrarán preinstalados la mayoría de los paquetes y librerías que son necesarias.

Librerías utilizadas. La mayoría ya precargadas en Suite Anaconda

- Librerías manejo y análisis de estructuras de datos.

- `import pandas as pd`

- Librerías especializada en el cálculo numérico y el análisis de datos

- `import numpy as np`

- Librerías de Métricas

- `from sklearn.metrics import f1_score, recall_score, precision_score, accuracy_score`
 - `from sklearn.metrics import roc_auc_score, roc_curve`
 - `from sklearn.metrics import confusion_matrix`
 - `from sklearn.metrics import classification_report`
 - `from sklearn.metrics import auc`
 - `from sklearn.model_selection import cross_val_score`

- Librerías de Visualización

- `import matplotlib.pyplot as plt`
 - `import pylab as pl`
 - `import seaborn as sns`
 - `from pylab import rcParams`
 - `from matplotlib import pyplot`

- Librerías de Modelos


```
- from sklearn.linear_model import LogisticRegression
- from sklearn.neighbors import KNeighborsClassifier
- from sklearn.neighbors import KNeighborsRegressor
- from sklearn.tree import DecisionTreeClassifier
- from sklearn.ensemble import RandomForestClassifier
- from sklearn.model_selection import train_test_split
- from sklearn.model_selection import cross_val_score
- from collections import Counter
- from imblearn.over_sampling import SMOTE
- from imblearn.under_sampling import NearMiss
```

• Adicionalmente será necesaria la instalación de las siguientes librerías.

- Imbalanced learn - proporciona herramientas cuando se trata de la clasificación con clases desequilibradas. (Instalación mediante consola - `pip install imbalanced learn`)

- pydotplus - Visualización de árboles de decisión en Python con PyDotPlus. (instalación mediante consola - `pip install pydotplus`)

- streamlit - creación e intercambio de aplicaciones web personalizadas para el aprendizaje automático y la ciencia de datos. (instalación mediante consola - `pip install streamlit`)

9 Guia Ejecucion y Carga de Base de Datos

Paso 1. Clonar repositorio GitHub <https://github.com/romayana/Financial-Product-Sales-Forecast-Model.git>
(<https://github.com/romayana/Financial-Product-Sales-Forecast-Model.git>) en carpeta local elegida.

Paso 2. Descargar base de datos:

A pesar de haberse Anonimizado toda la base de datos, se ha decidido que la misma no estará disponible en el repositorio de GitHub. Para acceder a la base de datos ubicada en el Google Drive del propietario del TFM, se tendrá que solicitar permiso y acceso a la misma dirigiendo correo electrónico a manuelgonzalezprados@gmail.com el cual previa valoración de los fines y objetivos perseguidos podrá compartir el enlace con la persona solicitante.

Una vez compartido el acceso, descargar y ubicar la carpeta entera llamada Origin_Data dentro de la carpeta carpeta local donde se ha clonado el repositorio junto con el resto de carpetas.

Paso 3. Ejecutar código con la siguiente secuencia y orden. Los archivos csv se irán guardando en cada una de las carpetas.

1º Carpeta Leasing & Merging

- `_merging_data.ipynb`
- `_cleanning_data.ipynb`

2º Carpeta Exploratory Data Analysis

- `EDA.ipynb`

3º Carpeta Preprocessing

- `Preprocessing.ipynb`

4º Carpeta Models

- Ejecutar los modelos.

5º Carpeta Frontend

- Aplicación Callornot.

10 Construcción de la Base de Datos Final

11 carpetas correspondientes a 11 Direcciones de Zona y 9 archivos Excel cada uno se fusionan en un solo Data Frame. De forma secuencial se han ido leyendo los archivos de cada una de las Direcciones de Zona creando una única lista agregada por DZ y finalmente uniéndolos en una sola base de datos la totalidad de los 99 archivos Excel originales individuales. Todo el proceso de unión está ubicado en el notebook `_merging_data.ipynb` al cual se puede acceder dentro de la carpeta `Cleaning & Merging` del repositorio.

11 Preprocesado y limpieza de los datos

La tarea de preprocesado de los datos o sencillamente de preparación de los datos se puede encontrar inicialmente en la carpeta de `cleaning & merging` y concretamente en el notebook `_cleanning_data.ipynb`. Adicionalmente, dentro de la carpeta `Preprocessing` de este repositorio se encontrará el resto de este proceso dentro del notebook `preprocessing.ipynb`.

Detectaremos Nans o valores nulos, se corregirán, se buscarán posibles outliers en variables, se buscarán las mejores o más importantes variables mediante técnicas de feature selection, se estudiará la posible reducción de dimensionalidad, convertiremos variables categóricas en numéricas y normalizaremos las variables a una escala común.

Finalmente Todo ello con el fin de construcción un data set de calidad y así poder trabajar con la mejor información de datos para la construcción definitiva del modelo.

12 Contexto de nuestra base de datos

Para poder entender la importancia de nuestro modelo, el impacto económico y repercusión que puede llegar a tener el hecho de saber diferenciar a los clientes susceptibles de contratar el seguro de hogar vamos a hacer un retrato piramidal descendente del organigrama de como esta estructurada la entidad financiera.

- Dirección Banca Particulares. Área de Dirección de la que descuelgan las siguientes unidades. Destinada a gestión y organización de la atención de clientes particulares / personas físicas.
- Direcciones Territoriales. Banca de Particulares esta dividida en un número determinado de Direcciones Territoriales según distribución nacional.
- Direcciones de Zona. Cada una de las Direcciones Territoriales esta dividida en Direcciones de Zona dando cobertura a cada una de las zonas geográficas de ese territorio.
- Oficinas. Cada Dirección de Zona esta dividida en un número concreto de Oficinas atendiendo a situación geográfica vinculada con esa Dirección de Zona.
- Asesores Financieros- Cada una de estas oficinas dispone de un número determinado de Asesores Financieros. Entre 1 y 4. Dependiendo el volumen de clientes.
- Clientes de cada una de las Oficinas.

Nuestra base de datos hace referencia a una de esas territoriales y su estructura interna es la siguiente:

- Número Clientes 451.350
- Número Oficinas 162
- Número DZs 11
- Número Gestores 458

Teniendo en cuenta este volumen de negocio de clientes, el impacto y ganancia económica de una entidad financiera con 2000 - 3000 oficinas puede ser enorme.

```
In [6]: Image(filename=(ruta + '/Financial-Product-Sales-Forecast-Model/Images/ESTRUCTURAPIRAMIDAL.png'),width = 400)
```

Out[6]:



13. Nuestra base de Datos

Mostramos las variables de nuestro data set y visualizamos un ejemplo de 5 líneas del mismo. Disponemos inicialmente de 30 variables.

```
In [7]: import pandas as pd
df = pd.read_csv(ruta + '/Financial-Product-Sales-Forecast-Model/Memory/df_cleaned.csv')
print(df.columns)
df.sample(5)
```

```
Index(['cliente', 'saldo_captacion', 'esta_carterizado', 'cliente_bbp',
      'tipo_gestor', 'gestor', 'cartera_patron', 'codigo_cartera',
      'digital_3_meses', 'camino_digital', 'saldo_financiacion', 'saldo_ffii',
      'saldo_plp', 'lp_dom_ingresos', 'lp_tjta_cto', 'lp_tjt_rev', 'lp_rbos',
      'lp_of_int', 'marca_bp', 'marca_ccte', 'edad', 'seg_valor',
      'seg_recorrido', 'dz', 'oficina', 'lp_seg_vida', 'lp_seg_acc',
      'lp_seg_salud', 'lp_seg_hogar', 'lp_seg_auto'],
      dtype='object')
```

Out[7]:

	cliente	saldo_captacion	esta_carterizado	cliente_bbp	tipo_gestor	gestor	cartera_patron	codigo_cartera	digital_3_meses	camino
226470	226471	47274.89	SI	NO	ASESOR FINANCIERO	12109.0	TUTELA	27755.0	NO	SE
217761	217762	17782.71	SI	NO	ASESOR FINANCIERO	22866.0	ASESORAMIENTO FINANCIERO	15253.0	NO	SE
160696	160697	42.58	SI	NO	ASESOR FINANCIERO	36152.0	ASESORAMIENTO FINANCIERO	14061.0	SI	TRANSAC
122310	122311	304.59	SI	NO	ASESOR FINANCIERO	41083.0	ASESORAMIENTO FINANCIERO	14820.0	SI	CONS
238425	238426	89857.88	SI	NO	ASESOR FINANCIERO	7798.0	ASESORAMIENTO FINANCIERO	14111.0	SI	CONS

5 rows × 30 columns

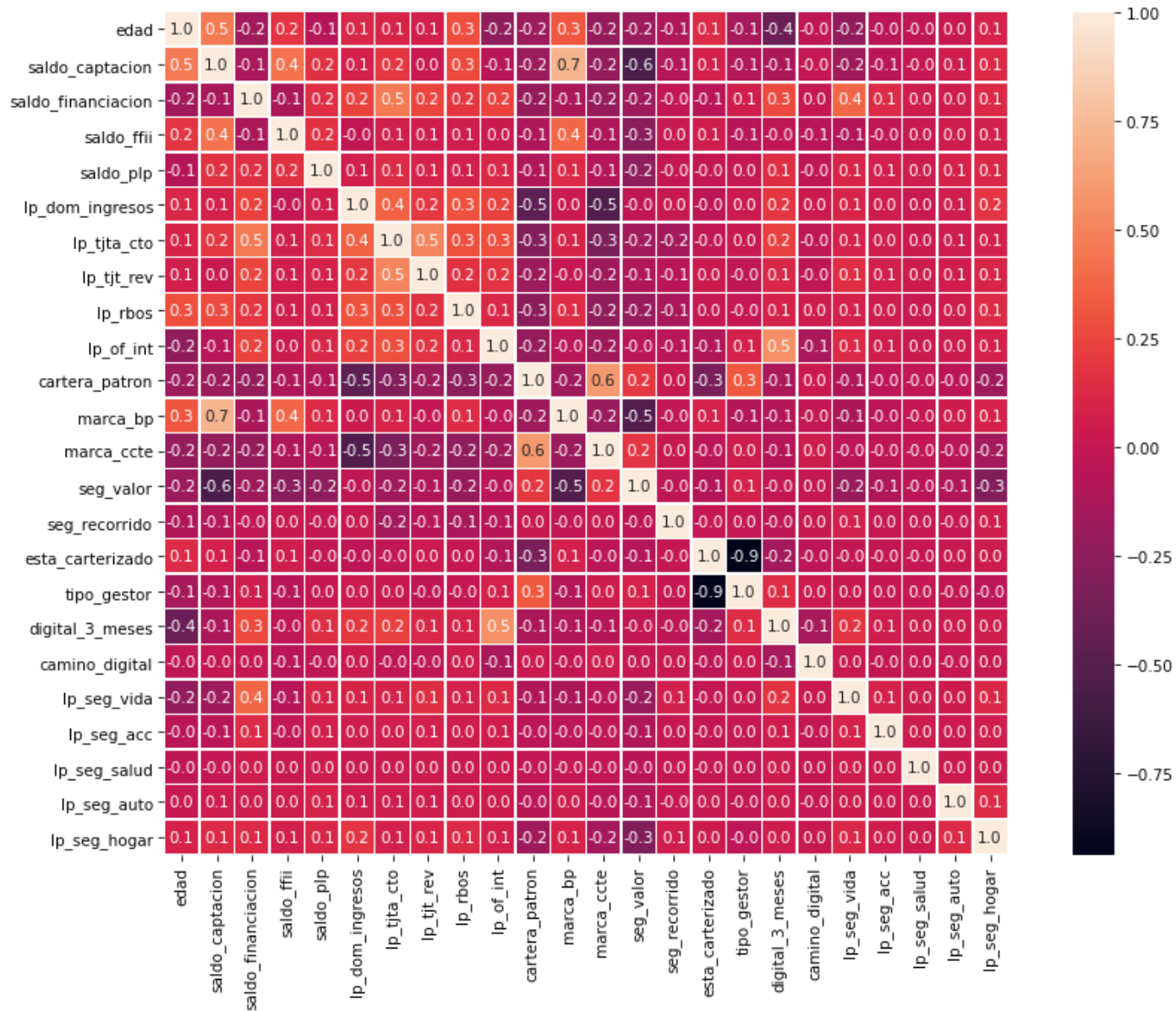


14. Correlaciones e importancia de las variables

Mostramos la grafica de correlaciones. Pretendemos visualizar si existen correlaciones altas que impliquen relaciones entre las propias variables. La correlación entre las variables es generalmente baja. Esto significa que las variables no dependen unas de otras por lo que inicialmente se pueden utilizar todas ellas en el modelo.

```
In [8]: Image(filename=(ruta + '/Financial-Product-Sales-Forecast-Model/Images/EDA- Correlaciones Todas.png'),width = 700)
```

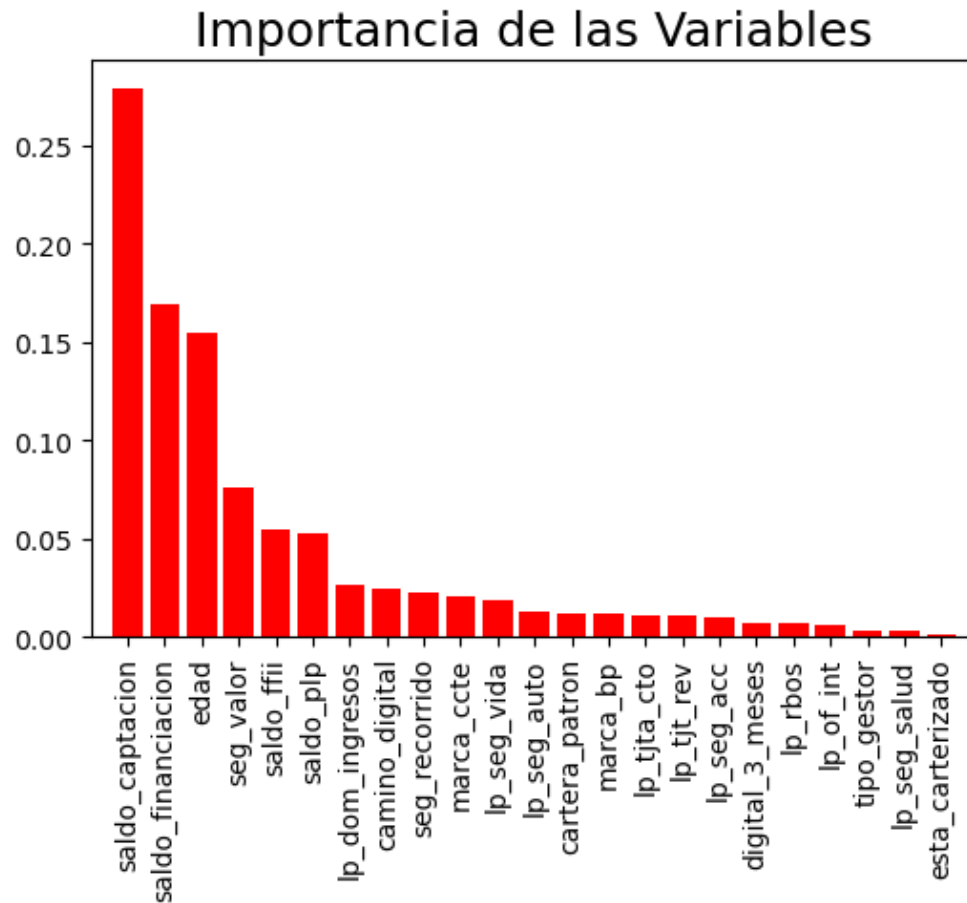
Out[8]:



En cuanto a la importancia de las variables, el estudio realizado nos informa que la variable que mas peso tiene dentro del data set es la variable de ahorro captacion o saldo en cuenta del cliente. De forma visual podemos ver que 3 variables tienen mas del 65% del peso de toda la base de datos.

```
In [9]: Image(filename=(ruta + '/Financial-Product-Sales-Forecast-Model/Images/EDA-Importancia de las Variables.png'),width = 500)
```

Out[9]:



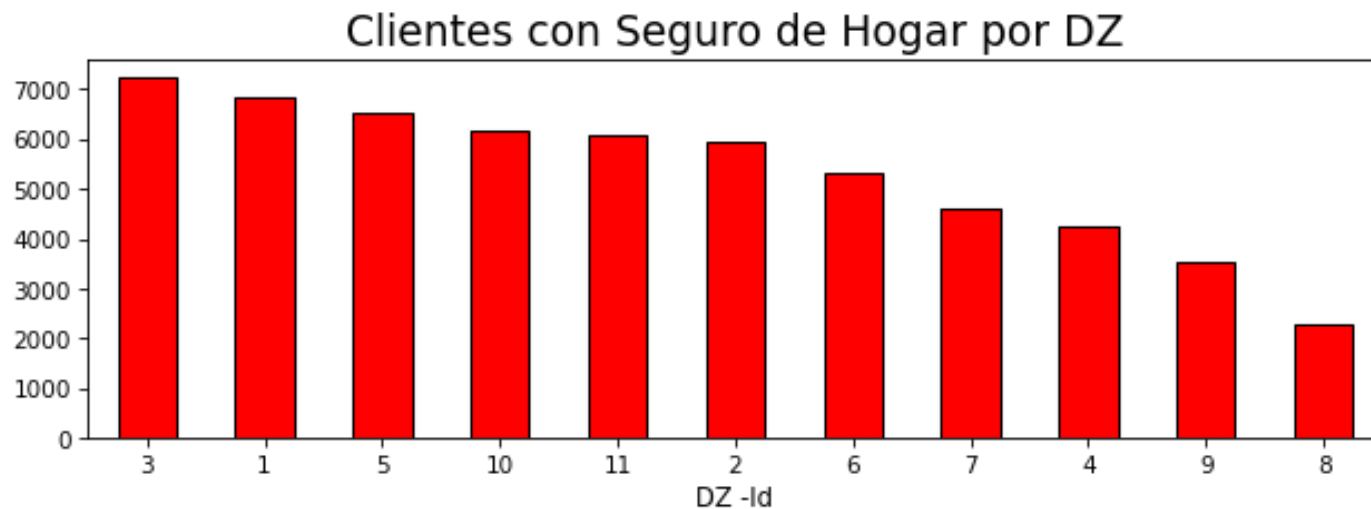
15. Estudio del Target

- ¿Cuál es su estructura con respecto a nuestro marco general

Solo 58.811 clientes de los 450.000, disponen de Seguro de Hogar. La DZ que más clientes tiene con seguro de hogar contratado es la DZ 3 con 7233 clientes mientras que la que menos tiene es la DZ 8 con 2.274 clientes. En todo caso, las cifras son bajas y la media se posiciona en 5.346 clientes. Estos datos reflejan un altísimo potencial de comercialización.

In [10]: `Image(filename=(ruta + '/Financial-Product-Sales-Forecast-Model/Images/EDA-Clientes con Seguro de Hogar por DZ.png'),wid`

Out[10]:

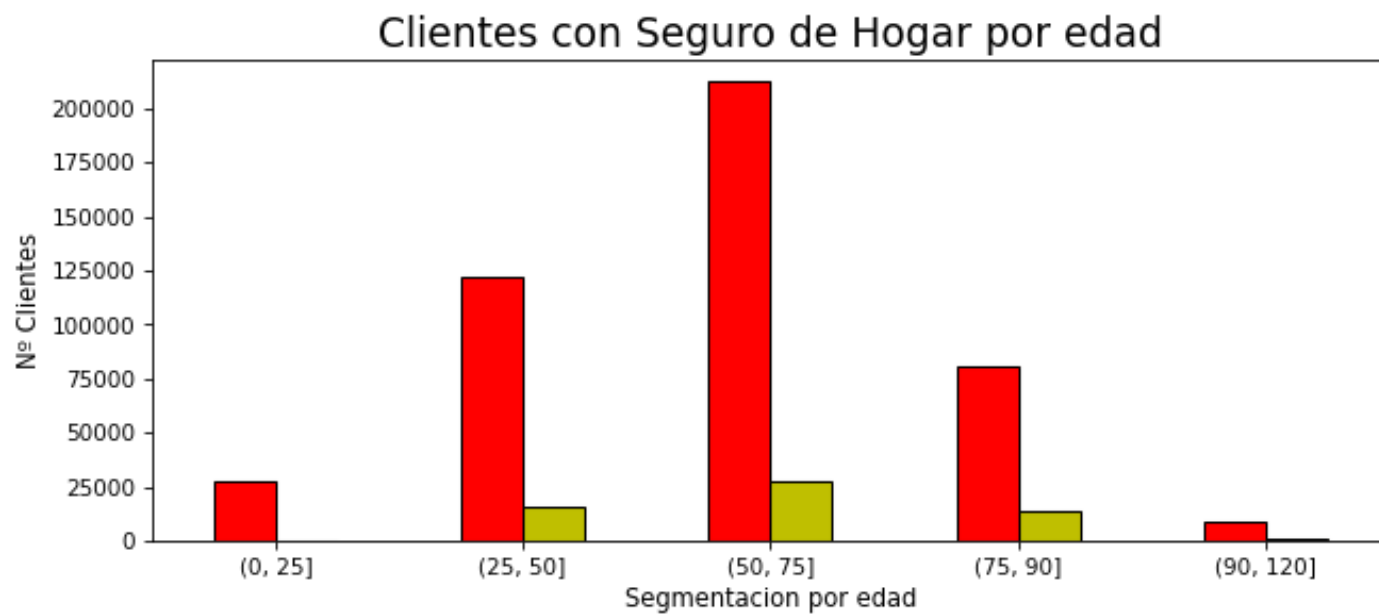


Solo el 13 % de los clientes tiene este seguro. El segmento de clientes que mas contrataciones de seguros de hogar son entre 25 y 90. Ninguno de los extremos tienen contratacion de seguros relevantes.


```
In [11]: print(df['lp_seg_hogar'].value_counts(normalize = True))  
Image(filename=(ruta + '/Financial-Product-Sales-Forecast-Model/Images/EDA-Clientes con Seguro de Hogar por edad.png'), w
```

```
0    0.869707  
1    0.130293  
Name: lp_seg_hogar, dtype: float64
```

Out[11]:



Desbalanceo. Las cifras anteriores reflejan otra circunstancia importante y es el gran desbalanceo que tiene el Target. Esta circunstancia se tendrá en cuenta en la elaboración del modelo.

```
In [12]: Image(filename=(ruta + '/Financial-Product-Sales-Forecast-Model/Images/EDA-TARGET Desbalanceo.png'),width = 700)
```

Out[12]:



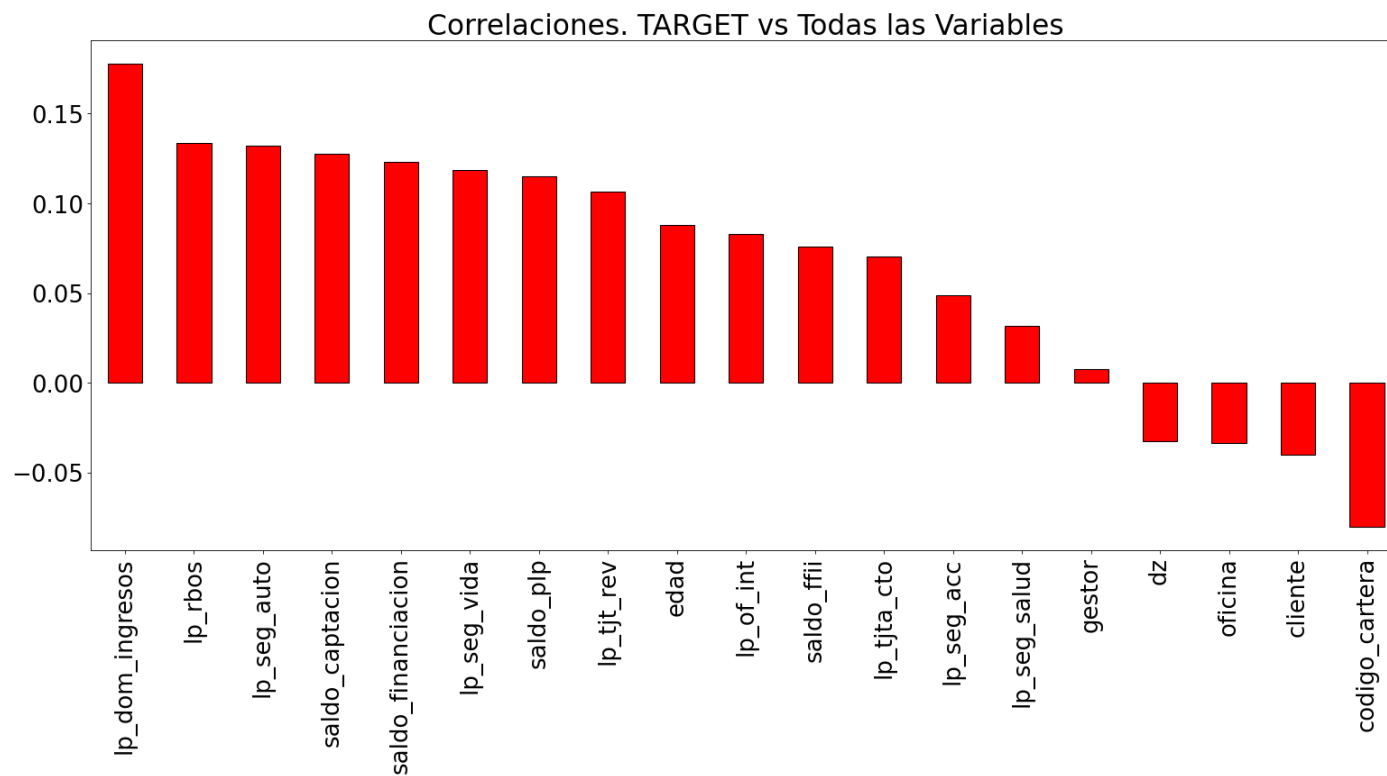
- ¿Correlaciones con otras variables.?

En general las correlaciones son bajas . Vemos como aquellos que tienen nomina domiciliada y recibos domiciliados son los clientes que más relación tienen con el seguro de hogar. Aun así, la mayor correlación es de 0.17. Cuantos mas nominas domiciliadas mas seguros de hogar.

Esto suele ser una realidad comercial ya que aquellos que tienen nomina domiciliada obtienen un descuento por su seguro de hogar.

```
In [13]: Image(filename=(ruta + '/Financial-Product-Sales-Forecast-Model/Images/EDA-Correlaciones TARGET vs Todas.png'),width = 700)
```

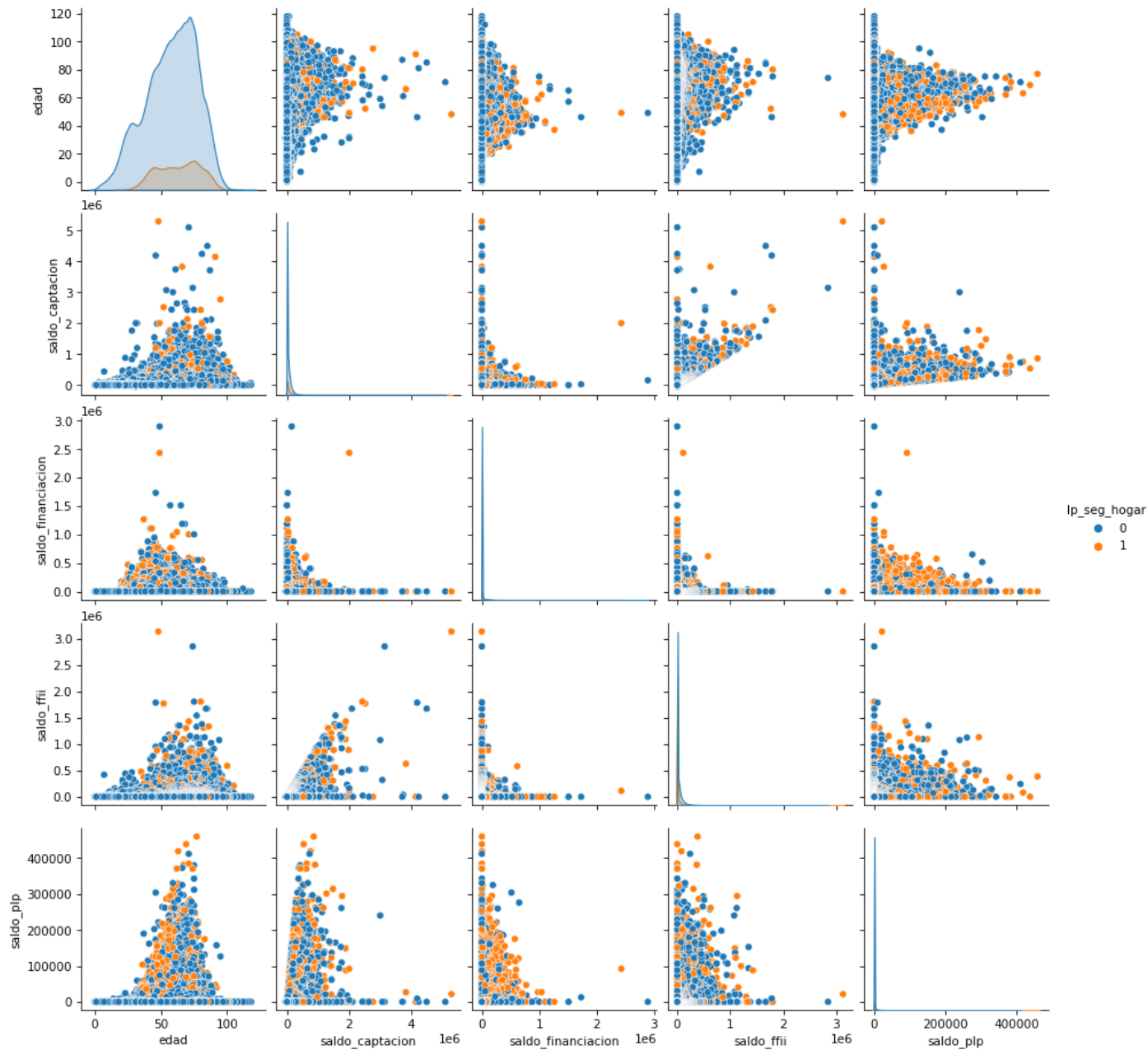
Out[13]:



Pairplot. Relacionamos las variables monetarias con el seguro de hogar para ver posibles diferencias y relaciones entre si.

```
In [14]: Image(filename=(ruta + '/Financial-Product-Sales-Forecast-Model/Images/PAIRPLOT.png'),width = 700)
```

Out[14]:



16. Construcción del Modelo

Elección del Modelo

Queremos saber si un cliente es susceptible de contratar o no un seguro de hogar.

Buscamos una sistemática comercial que simplifique el estudio de los clientes antes de llamarles.

Buscamos una optimización del tiempo para llegar a cuantos mas clientes mejor. Nos interesa finalmente filtrar aquellos clientes a los que tenemos que llamar.

Este filtro se puede obtener desde dos perspectivas. Y lo buscamos dentro de una matriz de confusión. Buscando verdaderos positivos para cargarlos en el objetivo de contactos diarios y que los comerciales puedan llamarles o identificando verdaderos negativos para desecharlos y finalmente coger los verdaderos positivos y llamarles. En ambos casos llegamos a la misma conclusión y obtenemos lo que realmente queremos. Llamar a los que nos interesan.

Se han desarrollado 5 modelos distintos de clasificación con el fin de buscar aquel que mejor métricas y mejores comportamientos tenga según nuestras necesidades concretas. Las principales métricas que utilizaremos como evaluadores de nuestros modelos son Recall y Roc Auc.

Modelos Machine Learning Entrenados.

- Modelo Clasificación Regresión Logística
- Modelo Clasificación K-Nearest Neighbor
- Modelo Clasificación TREE
- Modelo Clasificación Xgboost
- Modelo Clasificación Random Forest - **MODELO FINAL ELEGIDO**

Resumen de Resultados Obtenidos de los modelos entrenados

In [15]: Image(filename=(ruta + '/Financial-Product-Sales-Forecast-Model/Images/RESUMEN_RESULTADOS_FINALES.png'),width = 700)

Out[15]:

Modelo	accuracy	precision	recall	f1_score	roc_auc
Regresión Logística	0.6793	0.2818	0.8873	0.4277	0.8282
K Nearest Kneighbour	0.8767	0.5930	0.2785	0.3790	0.8283
Tree	0.7502	0.3327	0.8446	0.4774	0.8694
Random Forest	0.7773	0.3664	0.8543	0.5128	0.8863
XGBoost	0.8911	0.7173	0.3413	0.4625	0.8923

Desarrollo y construcción del Modelo

• Primera Aproximación

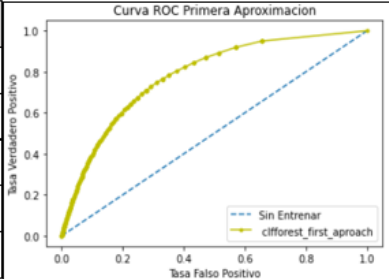
Mi primera intencion era saber que resultados obtendriamos sin hacer nada, simplemente utilizando la base de datos en crudo. Utilizamos en esta primera aproximacion al modelo la primera base de datos construida en nuestra fase de merging & cleanning. Se trata de una base de datos cuyo unico preprocesamiento ha sido la identificacion de los Nans y posterior asignacion de valores acordes a las particularidades de cada una de las variables. Dentro de esta base de datos utilizamos unicamente valores numericos, dejando fuera de la ecuacion las variables categoricas.

Los resultados obtenidos fueron las siguientes. Claramente mejorables.

In [16]: Image(filename=(ruta + '/Financial-Product-Sales-Forecast-Model/Images/RESULTADO_RANDOM_FOREST_PRIMERA_APROX.png'),width

Out[16]:

Resultados 1ª Aproximacion	Random Forest
accuracy	0.8564
precision	0.4089
recall	0.2239
f1_score	0.2893
roc_auc	0.7831



• Evolucion del modelo.

Comenzamos el estudio real del modelo. Durante la fase del preprocesado y atendiendo a las conclusiones obtenidas en el EDA, se generaron hasta 3 bases de datos distintas. Toda la secuencia se puede ver en el notebook preprocessing.ipynb dentro de la carpeta Preprocessing.

1. Base de datos eliminando variables categóricas, cambiando a booleano variables de FFII y PP, eliminando clientes de ciertos rangos de edad y adaptando las variables categóricas con Label Encoder. Esta base de Datos fue la finalmente utilizada.
2. Base de datos anterior a la que se le une la normalización a través de minmaxscaler para que todas las variables estuviesen en una misma escala.
3. Base de datos anterior implementando PCA reducción de dimensionalidad.

El modelo ha sido entrenado con las tres bases de datos y los resultados obtenidos fueron muy parecidos. Finalmente, con el fin de hacer el modelo mas sencillo, la base de datos utilizada fue la primera.

No vimos necesario reflejar en el modelo la reduccion de dimensionalidad con PCA implementada ya que despues de estudiarse la posible reduccion de dimensionalidad vimos que lo único que hacíamos era reducir dos variables con respecto a la base de datos principal.

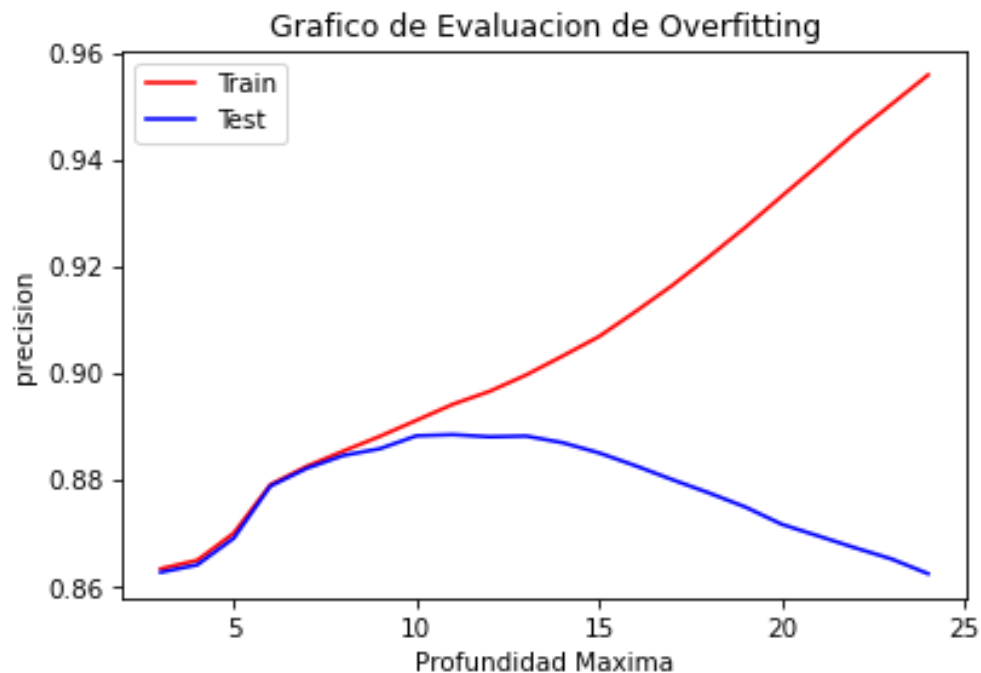
Tampoco la normalizacion de las variables ya que observamos que el hecho de haber normalizado los datos no habia sido un mejor condicionante para obtener mejores resultados. La explicación de esto es que en un modelo de RandomForest, la normalización puede no influir ya que no se comparan magnitudes. Se trata de dividir rangos y no de compararlos. Por otro lado, durante el preprocesado hemos visto como la importancia de las variables no difería mucho.

• Primeros Problemas detectados

- Conocimiento del desbalanceo del Target. 87%-13%. La primera decisión que tomamos en esta primera evolución del modelo es utilizar un estimador que intente poner remedio al gran desbalanceo de nuestro target. Para ello utilizamos el estimador de "Class Weigth = Balanced" para intentar corregirlo.
- Evidencias de Overfitting. Durante el entrenamiento se evidencia claramente la existencia de overfitting. Los resultados recall obtenidos de Train (0.99) y Test (0.37) demuestran que hay una grave situacion de sobreajuste confirmada posteriormente mediante tecnicas de CrossValidation. Lo demostramos visualmente mediante grafica de ajuste donde se observa como a partir de 8 "profundidades" empieza a no generalizar bien, separandose las curvas de Train y Test.

```
In [17]: Image(filename=(ruta + '/Financial-Product-Sales-Forecast-Model/Images/RANDOMFOREST Model - Evaluacion de Overfitting.png'))
```

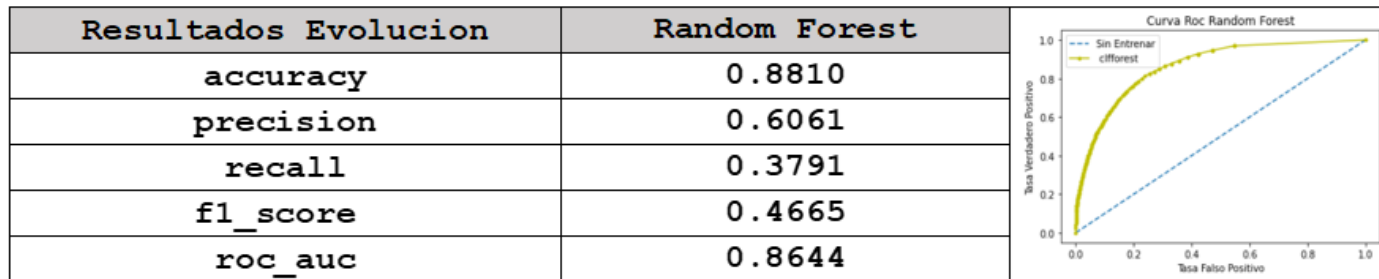
Out[17]:



Los resultados y metricas finales de esta primera evolucion solo mejoran algo nuestra primera aproximación. Necesitamos perfeccionar el modelo buscando mejores implementaciones que nos lleven a conseguir nuestro objetivo.

In [18]: `Image(filename=(ruta + '/Financial-Product-Sales-Forecast-Model/Images/RESULTADO_RANDOM_FOREST_1_EVOLUCION.png'),width =`

Out[18]:



• Soluciones planteadas a los problemas detectados.

El problema del desbalanceo del Target ha sido solucionado mediante la utilizacion del estimador “Class Weigth = Balanced”.

El problema del Overfitting queremos solucionarlo mediante tecnicas de **GridSearch** el cual permite evaluar y seleccionar de forma sistemática los parámetros de un modelo. Indicándole un modelo y los parámetros a probar, puede evaluar el rendimiento del primero en función de los segundos mediante validación cruzada.

Se proponen los siguientes parametros a probar:

```
'n_estimators' : [100,125],
'max_features' : ["auto", "log2"],
'criterion' : ['gini', 'entropy'],
'max_depth' : [2,4,6,8,10],
'min_samples_split' : [2,4],
'min_samples_leaf' : [15,20]
```

Resultando elegidos los siguientes:

```
'criterion': 'gini',
'max_depth': 10,
'max_features': 'auto',
'min_samples_leaf': 15,
'min_samples_split': 2,
'n_estimators': 100
```

• **Adaptación del Modelo con los Mejores Parametros propuestos mediante GridSearch.**

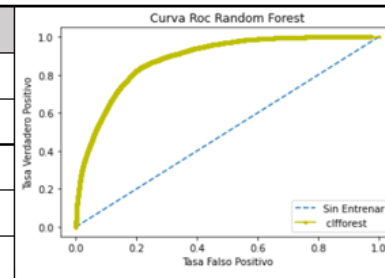
Implementamos los mejores parámetros recomendados recogiendo los siguientes resultados.

- Consolidamos la solución del desbalanceo mediante Class Weight = Balanced
- Conseguimos solucionar el overfitting igualando las métricas de Train y Test a 0.86 y 0.85
- Obtenemos unos resultados finales con los que quedamos satisfechos. Son acordes a nuestra búsqueda y necesidad.

In [19]: Image(filename=(ruta + '/Financial-Product-Sales-Forecast-Model/Images/RESULTADO_RANDOM_FOREST_FINAL.png'),width = 700)

Out[19]:

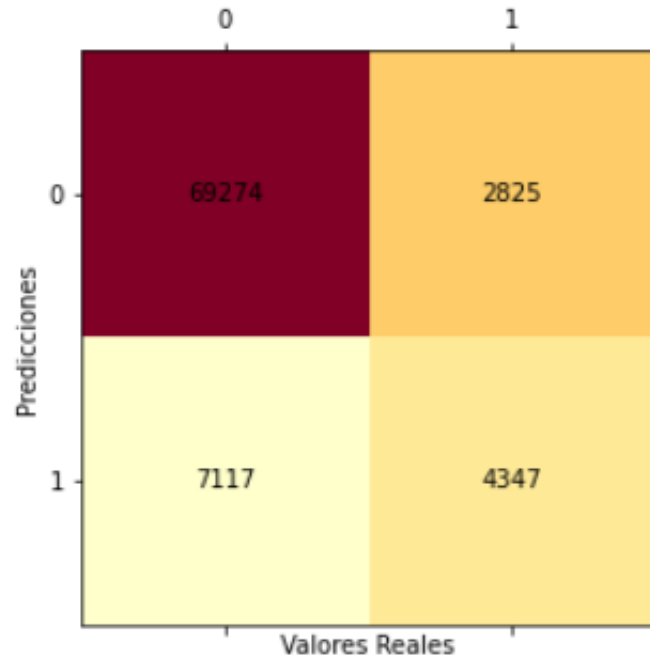
Resultados Finales	Random Forest
accuracy	0.7773
precision	0.3664
recall	0.8543
f1_score	0.5128
roc_auc	0.8863



• **Estudio de la Matriz de Confusion final obtenida**

```
In [20]: Image(filename=(ruta + '/Financial-Product-Sales-Forecast-Model/Images/RANDOMFOREST_MATRIZ_DE_CONFUSION.png'),width = 400)
```

Out[20]: Matriz de Confusion Random Forest con Gridsearch



En la diagonal de aciertos tenemos 69274 verdaderos positivos y 4347 verdaderos negativos, todo ello con el objetivo de optimizar el tiempo y la eficiencia comercial.

En nuestro modelo buscamos la detección de los clientes que sí quieren contratar el seguro. Necesitamos obtener una buena tasa de verdaderos positivos o verdaderos negativos.

Si detectamos a los clientes que probablemente vayan a contratar un seguro de hogar o que no lo vayan a hacer, podremos trabajar de forma más directa.

La realidad comercial es que nos da lo mismo trabajar con verdaderos positivos o negativos. Ambos sirven de filtro para poder llamar o no llamar. Finalmente, ¿qué proporción de positivos reales se identificó correctamente?

El resultado de Recall obtenido es óptimo, con un 85%.

17. Conclusiones

• Comerciales

La primera conclusión a la que llego es que el modelo podría mejorarse sustancialmente si se dispusiesen de otras muchas variables económicas, sociales, financieras, etc. Como he comentado en el estado del arte, existen muchísimos modelos ya implementados dentro de una entidad financiera. Modelos de riesgo, de comercialización, de morosidad, de recursos humanos... En concreto este modelo que he querido replicar, tengo conocimiento de la existencia de más de 1000 variables distintas para llegar a la decisión de si habilitarlo en una parrilla de llamadas o no.

Es importante que los comerciales de ventas de una entidad financiera sepan y puedan entender como se ha desarrollado, de donde han salido las variables utilizadas y de porqué la decisión de llamar o no a un cliente. Esta transmisión de conocimiento puede ser muy útil para toda la fuerza de ventas de las oficinas bancarias. Dado mi reciente incorporación al mundo del Data Science, creo que una de mis labores puede ser la de servir de nexo de unión entre el mundo senior del Data Science y la realidad comercial de asesoramiento y venta directa a los clientes. Explicar bajo estos parámetros, los motivos y ventajas de utilizar los modelos.

• Base de datos

Es importante contar con una buena base de datos donde encontremos distintas variables que reflejen la realidad de un cliente. Como he comentado anteriormente, los modelos actuales se nutren de cientos incluso miles de variables distintas lo que hacen que los resultados obtenidos sea bastantes óptimos y con porcentajes finales de conversión importantes. Se llegan a tasas de conversión de 1 sobre 4 clientes llamados, esto es un 25%. Porcentaje que, si bien siempre quiere ser aumentado, supone realmente un gran éxito.

Para ello, las técnicas de limpieza y preprocesado se hacen muy necesarias. En primer lugar la detección de campos vacíos y la decisión de qué hacer con ellos. Eliminarlos o rellenarlos y con qué datos hacerlo. Se pueden implementar distintas formas de preprocesador como hemos hecho. Técnicas de reducción de dimensionalidad, técnicas de normalización y aun así no ser necesarias. La utilización final de estas técnicas depende no solo de la base de datos, su dimensión y tipología sino también del modelo final de Machine Learning utilizado.

• Modelo utilizado

La búsqueda de la verdad es difícil y complicada. De cara a buscar unas buenas métricas, unos buenos resultados y en general buscar un punto final con el que estuviésemos cómodos y satisfechos, he entrenado hasta 5 modelos distintos con el fin de compararlos. Mi decisión final estaba entre un algoritmo de regresión y un algoritmo de random forest. Ambos ofrecían buenas métricas, aunque finalmente me he decidido por el ultimo al obtener un AUC superior.

La utilización de un solo árbol de decisión con una base de datos grande como la que tenemos corría el riesgo de no generalizar bien y de obtener resultados débiles. La utilización del Random Forest obedece no solo a una mejores métricas obtenidas sino al aprovechamiento de la baja correlación entre las variables, a una simplificación del estudio al no ser necesarias técnicas de normalización o de reducción de dimensionalidad. Se trata de dividir rangos y no de compararlos. Random forest es la suma de muchos Árboles de decisión individuales lo que lo hace mucho más fuerte. Adicionalmente suele tener mejores rendimientos en algoritmos de clasificación como es nuestro caso. Por último, su fácil interpretación y explicación basada en divisiones.

• Problemas detectados

A lo largo de la construcción del modelo he tenido problemas de convergencia en los modelos, he ido probando caminos, la mayoría de ellos sin llegar a ninguna parte, he ido probando y entrenando los modelos con más y menos variables, solo por probar esos escenarios, llegando la mayoría de las veces a puntos muertos.

Los principales problemas detectados en el estudio del modelo fueron el gran desbalanceo del data set 87%-13% y la falta de generalización evidenciada en el overfitting, motivado por la dimensión de la base de datos. Ambos problemas se han podido resolver mediante la utilización de hiperparametros sugeridos por técnicas de Gridsearch y por la utilización de estimadores de desbalanceo dentro del propio algoritmo.

• Métricas

Para la elección de las métricas que iban a ser utilizadas como explicativas del modelo, inicialmente estudié y entendí la teoría de la matriz de confusión. Finalmente, las métricas elegidas para evaluar los modelos de clasificación fueron Recall y AUC - Roc Curve. La métrica de Recall nos va a informar sobre la cantidad que el modelo de machine learning es capaz de identificar. ¿Qué porcentaje de los clientes están interesados somos capaces de identificar? Por último, la Curva ROC-AUC que nos informa y verifica el rendimiento del modelo.

• Matriz de Confusión

Finalmente, en la diagonal de aciertos tenemos 69274 Verdaderos positivos y 4347 verdaderos negativos. En nuestro modelo prima la detección de los clientes que sí son susceptibles de contratar el seguro. Necesitamos obtener una buena tasa de verdaderos positivos o verdaderos negativos. Recordemos que nuestras métricas objetivo son Recall y AUC. Si detectamos a los clientes que probablemente vayan a contratar un seguro de hogar o que no lo vayan a hacer, podremos trabajar de forma más directa. Por lo tanto, podemos obtener los mismos resultados por ambas vías. Todo ello con el objetivo de optimizar el tiempo y la eficiencia comercial. El resultado de Recall obtenido es óptimo, con un 85% y un 88% en AUC.

In []: