Resumen

En este trabajo se propone la aplicación de técnicas del aprendizaje estadístico supervisado para generar un modelo que permita predecir qué clientes de un banco se suscribirán a una campaña de marketing. Para ello se partirá de una base de datos con una cartera de clientes del banco y variables que muestran algunas características de estos en la entidad financiera.

Se construyeron dos modelos basados sobre Random Forest. Ambos presentan resultados muy similares reconociendo a los clientes del banco que se suscribirán a la campaña de marketing con una exactitud de 89%.

Palabras clave: CIENCIA DE DATOS - PREDICTOR DE SUBSCRIPCIÓN - SVM – LOGISTIC REGRESSION - APRENDIZAJE SUPERVISADO - CLASIFICACIÓN

Introducción y objetivos

Con el objetivo de generar un modelo que permita predecir la subscripción o no de un cliente bancario a una campaña de marketing, primeramente se desarrolla un EDA, es decir, un análisis exploratorio sobre el dataset en cuestión.

Luego de haber preprocesado el set de datos, se procede a desarrollar un pipeline de Machine Learning para predecir la variable dependiente, en este caso, ”Subscription”.

Además, se aplica el método Análisis de Componentes Principales, conocido como PCA por sus siglas en inglés, para reducir la dimensionalidad de los datos y, posteriormente, se construye otro modelo. Con este se vuelve a predecir el problema en cuestión.

Descripción del dataset

El conjunto de datos recolectados y disponibles está constituido por 45.211 muestras de clientes, las cuales están caracterizadas por 17 variables, entregando información como:

- Edad de los clientes
- Actividad laboral
- Estado Civil
- Nivel máximo educativo
- Promedio de saldo anual
- Tipo de contacto con el cliente
- Último día de contacto con el cliente
- Último mes de contacto con el cliente
- Duración del último contacto
- Cantidad de contactos con el cliente durante la campaña
- Cantidad de días que pasaron desde el último contacto
- Cantidad de contactos previos a esta campaña
- Si tiene deuda de crédito o no
- Si tiene préstamos adeudados o no
- Si tiene contratado un seguro de hogar o no
- Performance de la campaña de marketing anterior

Entre toda ésta información, se encuentra también el dato de si el cliente accede a la campaña de marketing o no. Éste último será de suma importancia para construir el modelo predictivo de clasificación ya que constituye la variable dependiente que se buscará anticipar.

Análisis exploratorio de datos (EDA)

En primer lugar, se elimina la variable denominada "Unnamed: 0" del dataset ya que no aporta para realizar machine learning. La misma actúa meramente como enumerador de cada uno de los registros.

Dado que las features denominadas “Contact” y “Poutcome” tienen mucha información faltante, un 29 y 82% respectivamente, se procede a eliminarlas.

Se eliminan luego todos los registros que poseen valores no asignados o desconocidos.

Después de lo mencionado, queda un dataset con 10.243 registros y 15 variables, entre ellas la dependiente que se quiere predecir.

Inicialmente, la variable dependiente que se quiere predecir toma valor 1 en caso que el cliente se suscriba a la campaña de MKT y 0 en caso contrario. Se la transforma de manera tal que tome valor 0 en el caso negativo y 1 en caso afirmativo.

En referencia a cómo se distribuye la variable dependiente “Subscription”, se puede afirmar que el 88% de los clientes de la base de datos obtenida luego de la limpieza no se suscriben a la campaña de marketing, mientras que el 12% restante si lo hace.

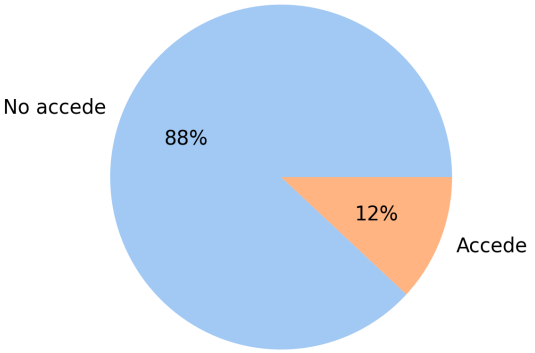

Fig. 1 Gráfico de torta representando la distribución de la variable “Subscription”.

Se analiza también cómo se distribuye la edad de los clientes en función de si se han suscritpo o no a la campaña. Se observa que la distribución es similar en ambos casos con una media cercana a los 40 años.

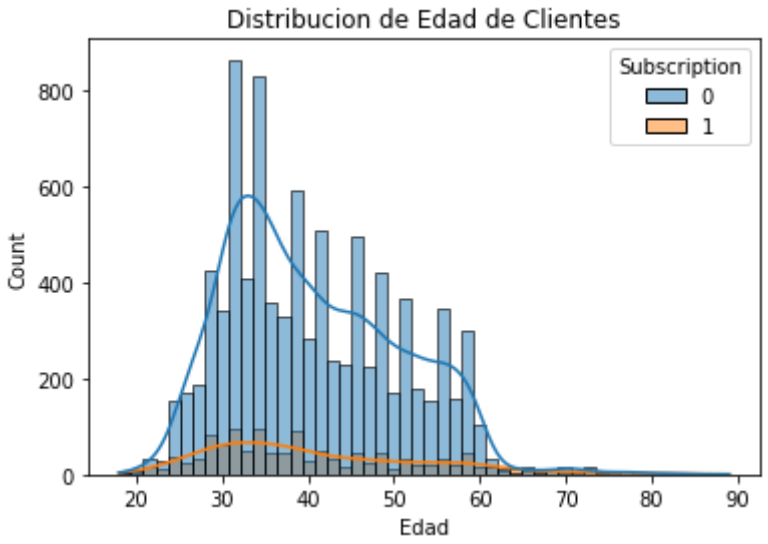

Fig. 2 Histograma de la edad de los clientes en función de la variable “Subscription”.

Posteriormente, se calcula el logaritmo del saldo promedio (“Balance (euros)”) en las cuentas de los clientes y se analiza su distribución en función de la variable dependiente.

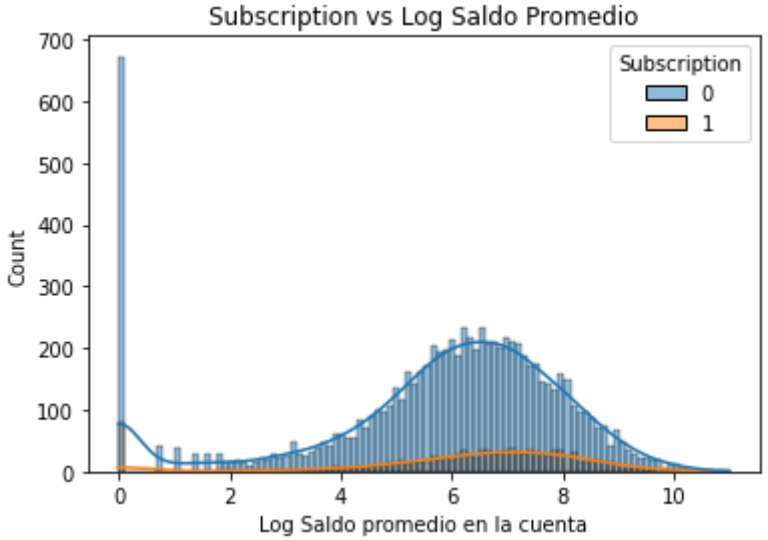

Fig. 3 Histograma del logaritmo del saldo promedio de los clientes en función de la variable “Subscription”.

Se percibe que quienes efectivamente se suscriben a la campaña de MKT suelen tener un saldo promedio mayor que quienes no lo hacen. La media del saldo promedio para aquellos registros en los que la variable “Subscription” toma valor 0 (no se suscribe) es 1.308 euros, mientras que para los otros registros (si se suscribe) la media es de 1.867 euros.

En referencia a la duración del último contacto con el cliente medido en segundos, se genera un histograma para evidenciar cómo se distribuyen los datos en función del outcome de la variable “Subscription”.

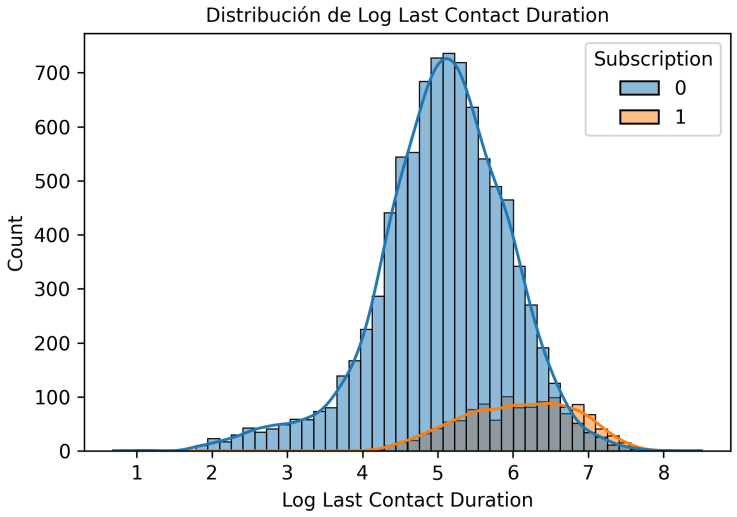

Fig. 4 Histograma del logaritmo la duración del último contacto con el cliente en función de la variable "Subscription".

En este caso, se percibe que el último contacto con los clientes que si se suscriben a la campaña de MKT suele tener una duración considerablemente mayor a la que tiene el contacto con aquellos clientes que no se suscriben.

Después de realizar los mencionados análisis, se separa del dataset a la variable dependiente "y", en este caso, "Subscription". Luego, se procede a dividir tanto la matriz de features "x" como la matriz "y" en Train y Test considerando que corresponden a testeo el 30% de las muestras. Finalmente, se obtiene un set de entrenamiento que cuenta con 7.170 registros y uno de testeo que cuenta con 3.073.

Por otra parte, se debe considerar que los algoritmos a utilizar requieren que las variables del dataset sean numéricas. Dado que en nuestro set existen algunas que son categóricas, se deben realizar transformaciones en pos de cumplir con dicha condición.

En el caso de la feature "Last Contact Month", se cuenta con el nombre del mes en el que ocurrió el último contacto. Se decide cambiar ese dato por el número del mes.

Para las variables categóricas restantes, "Job", "Marital Status", "Education", "Credit", "Housing Loan" y "Personal Loan", se crean variables dummies.

En el caso de las features numéricas, la diferencia de escalas puede generar problemas a la hora de aprender de los datos. Por ello, se recurre a la estandarización de las mismas para evitar que tengan rangos muy distintos. Esto es, se transforma a cada una de estas variables de manera tal que tengan media $\mu = 0$ y desvío estándar $\sigma = 1$.

$$x_i' = \frac{x_i - \mu}{\sigma}$$

Para aplicar las mencionadas transformaciones tanto a la matriz "x_{train}" como "x_{test}" utilizaremos *Column Transformer*. Esta herramienta permite transformar diferentes columnas o subconjuntos de columnas de un input por separado, concatenando las características generadas por cada transformador para formar un único espacio de características.

Una vez desarrollado todo el preprocesamiento, se cuenta con 31 variables independientes y 1 dependiente ("Subscription").

Materiales y métodos (algoritmos utilizados)

Como se mencionó con anterioridad, se aplicarán técnicas del aprendizaje estadístico supervisado con el objetivo de construir un modelo para predecir o estimar un output basado en uno o más inputs.

Dependiendo del tipo de variable respuesta, varía el enfoque del aprendizaje. En este caso, dado que la respuesta del modelo es de tipo categórica, se trata de un problema de clasificación.

En este enfoque, se parte de un conjunto de muestras x_i , las cuales tienen asociada una etiqueta y_i que hace referencia al grupo que pertenece.

$$\{(x_1, y_1), (x_2, y_2), (x_3, y_3), ..., (x_n, y_n)\}\ ,\ x \in R\ ,\ y\ \varepsilon\ \{0, 1\}$$

Con ese conjunto de datos, se busca dentro del espacio de hipótesis una función $\widehat{f}(x)$ caracterizada por parámetros *w* que pueda explicar lo mejor posible la relación entre input y output del problema. Dado que la realidad es compleja, resulta difícil llegar a la función ideal que explique perfectamente nuestros datos. Por ello mismo es que existe un grado de error en cada una de las aproximaciones del modelo.

Existe una combinación de parámetros que determinan una $\widehat{f}(x)$ que se aproxima a la verdadera $f(x)$ mejor que otras. Para encontrarla el algoritmo de aprendizaje debe aprender los parámetros w y b que minimicen la función de costo $L(Y, \widehat{Y})$. Esta última función (L) tomará valores altos cuando la estimación $\widehat{y}$ difiera mucho del valor real de la etiqueta y . Por el contrario, tomará valores pequeños cuando la estimación $\widehat{y}$ sea parecida al valor real de la etiqueta y

$$f(x) = w^T \cdot x + b \ \rightarrow\ min\, L(Y, \widehat{Y})$$

Las funciones de decisión toman como input un vector X (sample) con d features y le asigna una de las K clases.

$$D(x) = sing\left[w^T \cdot x + b\right] = sing\,[\,f(x)\,]$$

Para éste estudio, el valor 0 en la etiqueta estará asociado a que los clientes no accederán a la campaña de marketing, mientras que el valor 1 estará asociado al caso contrario.

Existen numerosos métodos de clasificación. En nuestro caso se opta por emplear los siguientes:

- **Regresión Logística:** clasificador de tipo lineal. Se basa en una regresión lineal precedida de una función de activación o de decisión “sigmoide”, la cual genera un output binario y no continuo como una regresión normal.
- **Support Vector Machines (SVM):** este método se basa en la idea de separar los datos mediante hiperplanos. El nombre del mismo proviene de la utilización de vectores que hacen de soporte para maximizar la separación entre los datos y el hiperplano.

Para cada uno de ellos, se definen diferentes hiper parámetros que serán utilizados más adelante en el proceso de Grid Search. Este consiste en generar una lista de combinaciones posibles para cada método, para que luego en el proceso de validación cruzada (Cross Validation), se compare cada una de ellas y se compruebe cual de todas las combinaciones es la que mayor Train Accuracy promedio genera. De esta manera, se elige el mejor método de clasificación con sus hiper parámetros identificadores para poder realizar las predicciones pertinentes a futuro.

Experimentos y resultados

En pos de desarrollar un modelo para predecir la variable “Subscription'' se crea un pipeline de Machine Learning.

Para ello, se utiliza la herramienta *Pipeline* de la librería *scikit-learn*. Esta permite aplicar secuencialmente una lista de transformaciones y un estimador final. En este caso, como estimador final se tienen los modelos de clasificación previamente mencionados: SVM y Logistic Regression. Las transformaciones a incluir en el pipeline serán las mencionadas con anterioridad que se juntaron con la herramienta *Column Transformer*.

Se definen los hiper parámetros a combinar para cada modelo de clasificación a comparar.

Luego, se construye el Grid Search y Cross Validation. El set de datos de train se dividirá en 5 porciones durante la validación cruzada.

Al realizar el entrenamiento se llega a la conclusión que, dados los hiper parámetros propuestos, el modelo con el mayor accuracy (89,1%) fue Support Vector Machines con la siguiente combinación de hiper parámetros:

C: 1 | kernel: ‘rbf' | gamma: 0.1

Análisis de Componentes Principales

El PCA es un método que permite reducir la cantidad de dimensiones (features) de un dataset, creando nuevas features llamadas “Componentes Principales” a partir de la

descomposición espectral. Esas nuevas dimensiones serán combinaciones lineales de las features originales creadas con el fin de representar la mayor cantidad de variación de los datos.

En este trabajo con el objetivo de filtrar el ruido y mejorar la eficiencia computacional se aplica dicho método. Se decide que los autovectores que se extraen deben explicar como mínimo un 80% de la variación de los datos.

Se construye un nuevo pipeline que incluye al final del preprocesamiento la implementación del PCA. Las transformaciones iniciales y los modelos de clasificación al final del pipeline, así como también los hiper parámetros a combinar, serán los mismos que en los utilizados en el pipeline inicial.

Luego de aplicar el preprocesamiento que incluye PCA, se pasa de tener una matriz con 31 dimensiones a una con 29.

Posteriormente, se construye el Grid Search y Cross Validation que se utilizarán para seleccionar el mejor modelo y la combinación de hiper parámetros. Durante la validación cruzada el set de datos de train, al igual que antes, se dividirá en 5 porciones.

Al realizar el entrenamiento, se llega a la conclusión que, dados los hiper parámetros propuestos, el modelo con el mayor accuracy (89,4%) fue un Logistic Regression con el siguiente hiper parámetro:

| C: 1 |

Discusión y conclusiones

La matriz de confusión permite evaluar el desempeño de los clasificadores construidos. La misma muestra qué tan bien cada clasificador categoriza las distintas clases.

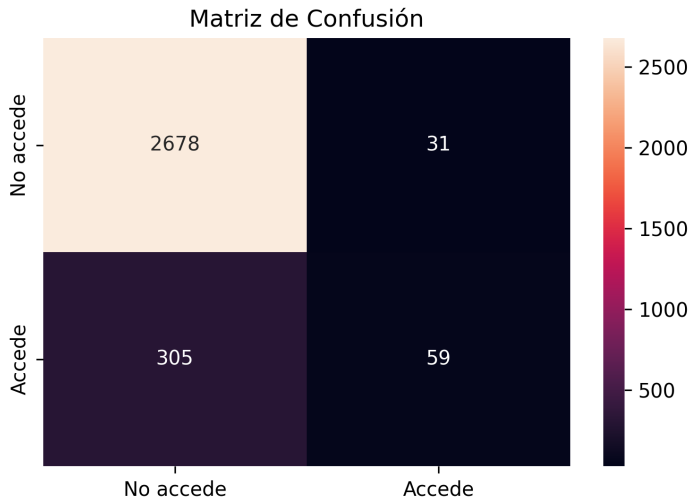

Fig. 5 Matriz de confusión para el primer modelo construido.

Fig. 6 Matriz de confusión para el segundo modelo construido.

Se observa que en ambos modelos obtienen valores similares. La exactitud obtenida por el primer clasificador fue 89,1%, mientras que la del segundo 89,4%.

Se procede también a comparar ambos clasificadores haciendo uso de la curva ROC. Los resultados indican que no hay diferencias significativas entre ambos modelos y que los resultados que se obtienen a partir de cada uno de ellos son buenos.

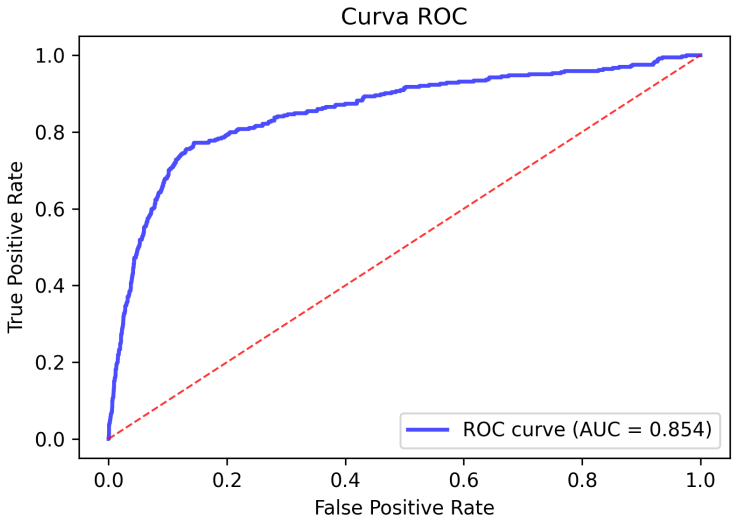

Fig. 7 Curva ROC del primer modelo construido.

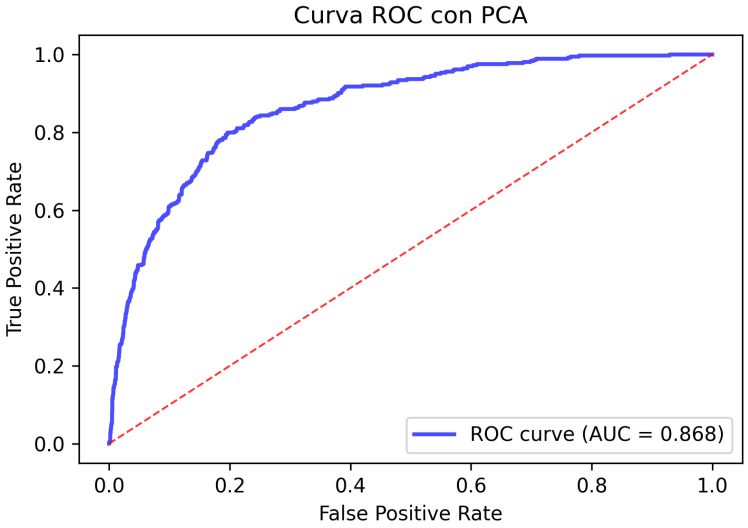

Fig. 8 Curva ROC del segundo modelo construido.

Los resultados de exactitud obtenidos para ambos clasificadores, 89,1% y 89,4% para el primero y el segundo respectivamente, son considerados buenos. La base de datos contiene información suficiente para predecir si un cliente del banco se suscribirá o no a una campaña de marketing.

La implementación del PCA permitió mejorar la eficiencia computacional y la performance del modelo. Es decir, la disminución de la dimensionalidad del dataset no implicó una merma en el desempeño del modelo que se construyó a posteriori respecto del construido inicialmente, si no que todo lo contrario, se logró una mayor precisión en la clasificación de las muestras.

Referencias

JAMES, G.; WITTEN, D.; HASTIE, T. & TIBSHIRANI, R., (2021). *An Introduction to Statistical Learning*.

VANDERPLAS, J., (2016). *Python Data Science Handbook*.

PALAZZO, M.; AGUIRRE, N. & CHAS, S. (2022). *clusterAI*. https://github.com/clusterai