UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL BUENOS AIRES

DEPARTAMENTO DE INGENIERÍA INDUSTRIAL

ASIGNATURA: Ciencia de Datos

CICLO LECTIVO **2024**

ESTUDIANTES				
	NOMBRE Y APELLIDO	**Leandro Del Sordo**	LEGAJO	**1755833**
	NOMBRE Y APELLIDO	**Nicolas Ferreyra**	LEGAJO	**1757556**
	NOMBRE Y APELLIDO		LEGAJO	
	NOMBRE Y APELLIDO		LEGAJO	
	NOMBRE Y APELLIDO		LEGAJO	

DIVISIÓN AÑO EQUIPO 11

2024

TRABAJO PRÁCTICO FINAL

OBSERVACIONES

FIRMA DEL ALUMNO

FIRMA DEL JTP / ATP

NOTA

ENTREGA	Fecha ENTREGA			Fecha VISADO			FECHAS DESTACADAS
1ra	29	11	24				FECHA DE INICIO
2da							
3ra							FECHA DE FINALIZACIÓN
4ta							

INTRODUCCIÓN

En un entorno altamente competitivo, las instituciones financieras enfrentan el desafío constante de diseñar estrategias de marketing más efectivas para captar y retener clientes. En este contexto, el presente trabajo aborda un problema planteado por un banco particular, que busca predecir cuáles de sus clientes es más probable que se suscriban a una campaña de marketing específica. La capacidad de identificar de manera precisa a estos clientes permitirá optimizar los recursos, reducir costos y aumentar la efectividad de las campañas, generando un impacto positivo tanto en los ingresos del banco como en la satisfacción de los clientes.

Para abordar este problema, se dispone de un conjunto de datos que comprende información de 45.211 clientes, descritos por 17 variables que reflejan características socioeconómicas, de comportamiento y de interacción con el banco. Este conjunto de datos representa la base para el desarrollo de un modelo predictivo que permita clasificar a los clientes según su probabilidad de suscripción.

OBJETIVOS

Objetivo general

Elaborar un **modelo** de data science que permita predecir la probabilidad de suscripción de los clientes del banco a una campaña de marketing.

Objetivos específicos

1. Realizar un análisis exploratorio de datos (EDA) que permita identificar patrones y características relevantes en la base de datos proporcionada.
2. Construir modelos predictivos de Machine Learning utilizando herramientas programadas en Python para estimar la variable objetivo.
3. Incorporar técnicas de reducción de dimensionalidad con el propósito de mejorar la eficiencia y el desempeño del pipeline de predicción.
4. Evaluar y comparar el desempeño de los modelos en términos de métricas relevantes, considerando los cambios introducidos por la reducción de dimensionalidad.

DATASET

El dataset contiene información sobre 45.211 clientes de un banco, descritos a través de 17 variables que reflejan sus características socioeconómicas, comportamientos financieros y su interacción con campañas de marketing del banco. El mismo está compuesto por las siguientes features:

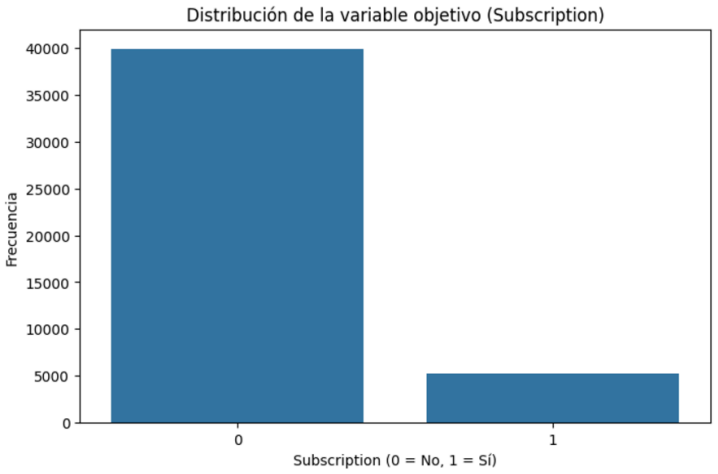
1. **Age**: Edad del cliente.
2. **Job**: Tipo de empleo del cliente.
3. **Marital Status**: Estado civil del cliente.
4. **Education**: Nivel educativo más alto alcanzado.
5. **Credit**: Indica si el cliente tiene deudas de crédito.
6. **Balance**: Promedio del saldo en la cuenta durante el año.
7. **Housing Loan**: Indica si el cliente tiene un préstamo hipotecario.
8. **Personal Loan**: Indica si el cliente posee un préstamo personal.
9. **Contact**: Tipo de contacto utilizado para comunicarse con el cliente.
10. **Last Contact Day**: Día del último contacto con el cliente durante la campaña.

11. **Last Contact Month**: Mes del último contacto con el cliente.
12. **Last Contact Duration**: Duración (en segundos) del último contacto.
13. **Campaign**: Número de contactos realizados durante la campaña actual.
14. **Pdays**: Días desde el último contacto con el cliente en una campaña previa (-1 si no hubo contacto previo).
15. **Previous**: Número de contactos realizados antes de la campaña actual.
16. **Poutcome**: Resultado de la campaña de marketing anterior.
17. **Subscription:** es la variables objetivo a predecir e indica si el cliente accedió a la campaña (1) o no (0).

ANÁLISIS EXPLORATORIO DE DATOS (EDA)

Primero analizar los tipos de datos que tiene cada feature y la cantidad de datos no nulos y como está distribuida la variable objetivo: **"Subcription"**

#	Column	Non-Null Count	Dtype
0	Age	40238 non-null	float64
1	Job	40238 non-null	object
2	Marital Status	40238 non-null	object
3	Education	40238 non-null	object
4	Credit	40238 non-null	object
5	Balance (euros)	40238 non-null	float64
6	Housing Loan	37525 non-null	object
7	Personal Loan	37525 non-null	object
8	Contact	45211 non-null	object
9	Last Contact Day	45211 non-null	int64
10	Last Contact Month	45211 non-null	object
11	Last Contact Duration	37525 non-null	float64
12	Campaign	45211 non-null	int64
13	Pdays	37525 non-null	float64
14	Previous	45211 non-null	int64
15	Poutcome	45211 non-null	object
16	Subscription	45211 non-null	int64



A primera vista, se observa que varias *features* contienen valores nulos y que algunas presentan el tipo de dato "Object". Por lo tanto, será necesario abordar estos dos aspectos durante el proceso de limpieza de datos.

Limpieza de datos

Análisis de Duplicados

Se verificó la existencia de duplicados, y se concluyó que no había ninguno.

Análisis de Valores Nulos

Se calculó el total de valores nulos por columna y el porcentaje que representan respecto al total de datos. Como se observa en la siguiente tabla, varias features contienen valores nulos, pero ninguna supera el 20% de valores nulos en la columna. Por lo tanto, no se descarta ninguna feature.

	Total	Percent
Pdays	7686	0.170003
Housing Loan	7686	0.170003
Personal Loan	7686	0.170003
Last Contact Duration	7686	0.170003
Age	4973	0.109995
Marital Status	4973	0.109995
Education	4973	0.109995
Credit	4973	0.109995
Balance (euros)	4973	0.109995
Job	4973	0.109995

Inicialmente, se eliminaron los registros con valores nulos, pero esto redujo el dataset de 45.211 registros a solo 10.630, lo que representa una disminución del 76%. Esta reducción resulta inviable para el análisis, ya que dejaría muy pocos datos.

Por este motivo, se optó por reemplazar los valores nulos (NaN) según el tipo de dato de cada feature:

- Para las columnas **"Age"** y **"Last Contact Duration"**, qué contienen números reales, los valores nulos se reemplazarán con el valor promedio.
- En la columna **"Pdays"**, los valores nulos se reemplazarán por "-1", lo que indica que no hubo contacto previo con el cliente.
- Para las columnas **"Marital Status"**, **"Personal Loan"**, **"Education"**, **"Housing Loan"** y **"Credit"**, que son de tipo object, los valores nulos se reemplazarán con la palabra "unknown", lo que significa que el valor de la categoría es desconocido.

Los registros restantes con valores nulos se eliminarán, dejando un dataset final de 35.814 filas.

Análisis de Capping y agrupamiento de variables

En las variables **"Campaign"** y **"Previous"** se identificaron valores atípicos (*outliers*), los cuales fueron reemplazados por el valor correspondiente al percentil 95. Este procedimiento se realizó para reducir la alta dispersión en los datos y evitar que estos valores extremos afecten el análisis.

Análisis de las variables categóricas (dummies)

Para la variable **“Jobs”**, los valores, que representen menos de un 10% de incidencia del total, se agruparán en una categoría denominada "Otros". De esta manera, la distribución quedará organizada de la siguiente forma:

```
Job
Other          10496
blue-collar     7746
management      7454
technician      6031
admin.          4087
```

Para la variable **“Last Contact Month”**, se creará un diccionario que mapeará los meses a números, reemplazando así los valores de la columna con sus equivalentes numéricos.

Por último, las demás variables de tipo *object* se transformarán en variables *dummy* (columnas de tipo booleana). Esto significa que cada categoría única de estas variables se convertirá en una columna separada, donde se asignará el valor 1 si el registro pertenece a esa categoría y 0 en caso contrario. Este proceso facilita su uso tanto en análisis gráficos como en la construcción del modelo predictivo.

MÉTODOS APLICADOS

Para predecir la variable objetivo se plantearon los siguientes métodos de aprendizaje supervisado

Random forest

"Random Forest es un algoritmo de aprendizaje automático de uso común, registrado por Leo Breiman y Adele Cutler, que combina el resultado de varios árboles de decisión para llegar a un único resultado. Su facilidad de uso y flexibilidad han impulsado su adopción, ya que maneja problemas tanto de clasificación como de regresión."

"Los algoritmos de bosque aleatorio tienen tres hiperparámetros principales que deben configurarse antes del entrenamiento. Estos incluyen el **tamaño del nodo**, la **cantidad de árboles** y la **cantidad de características muestreadas**. A partir de allí, el clasificador de bosque aleatorio se puede utilizar para resolver problemas de regresión o clasificación.

El algoritmo de bosque aleatorio se compone de una colección de árboles de decisión, y cada árbol del conjunto está compuesto por una muestra de datos extraída de un conjunto de entrenamiento con reemplazo, llamada muestra de arranque. De esa muestra de entrenamiento, un tercio se reserva como datos de prueba, conocida como muestra fuera de la bolsa. Luego se inyecta otra instancia de aleatoriedad a través del empaquetamiento de características, lo que agrega más diversidad al conjunto de datos y reduce la correlación entre los árboles de decisión. Dependiendo del tipo de problema, la determinación de la predicción variará. Para una tarea de regresión, se promedian los árboles de decisión individuales y, para una tarea de clasificación, un voto mayoritario (es decir, la variable categórica más frecuente) dará como resultado la clase predicha. Finalmente, la muestra fuera de la bolsa se utiliza para la validación cruzada, lo que finaliza esa predicción." (IBM SPSS, 2024)

Mediante un grid search se obtuvieron que los mejores hiperparametros son:

Hiper Parámetro	Valor
Max_depth	20
Min_Samples_Split	2
n_estimators	200

Logistic Regression

"La regresión logística es un algoritmo de aprendizaje supervisado que utiliza funciones logísticas para predecir la probabilidad de un resultado binario. Utiliza una función logística llamada función sigmoide para representar las predicciones y sus probabilidades. La función sigmoide se refiere a una curva en forma de S que convierte cualquier valor real en un rango entre 0 y 1, por lo tanto i el resultado de la función sigmoidea (probabilidad estimada) es mayor que un umbral predefinido en el gráfico, el modelo predice que la instancia pertenece a esa clase. Si la probabilidad estimada es menor que el umbral predefinido, el modelo predice que la instancia no pertenece a la

clase.” (Kanade, 2022). Por ejemplo, si el valor de salida de la función sigmoidea es superior a 0,5, se considera que el valor de salida es 1. Por otro lado, si el valor de salida es inferior a 0,5, se clasifica como 0.

Mediante un grid search se obtuvieron que los mejores hiperparametros son:

Hiper Parámetro	Valor
C	0.1
Solver	Liblinear

Neural Network

Una red neuronal es un modelo de *machine learning* inspirado en la estructura y el funcionamiento del cerebro humano. Se utiliza para resolver problemas complejos de clasificación, regresión y reconocimiento de patrones. Está compuesta por capas de **neuronas artificiales** interconectadas que procesan información de manera jerárquica. Estas capas incluyen:

1. **Capa de entrada**: Recibe los datos iniciales.
2. **Capas ocultas**: Procesan la información a través de cálculos matemáticos llamados activaciones. Estas capas permiten que la red aprenda patrones y relaciones complejas.
3. **Capa de salida**: Genera el resultado final, como una predicción o una clasificación.

“Entendamos con un ejemplo cómo funciona una red neuronal:

Considere una red neuronal para la clasificación de correo electrónico. La capa de entrada toma características como el contenido del correo electrónico, la información del remitente y el asunto. Estas entradas, multiplicadas por pesos ajustados, pasan a través de capas ocultas. La red, mediante entrenamiento, aprende a reconocer patrones que indican si un correo electrónico es spam o no. La capa de salida, con una función de activación binaria, predice si el correo electrónico es spam (1) o no (0). A medida que la red refina iterativamente sus pesos a través de la retropropagación, se vuelve experta en distinguir entre correos electrónicos spam y legítimos, lo que demuestra la practicidad de las redes neuronales en aplicaciones del mundo real como el filtrado de correo electrónico.” (guide & Jain, 2024)

Mediante un grid search se obtuvieron que los mejores hiperparametros son:

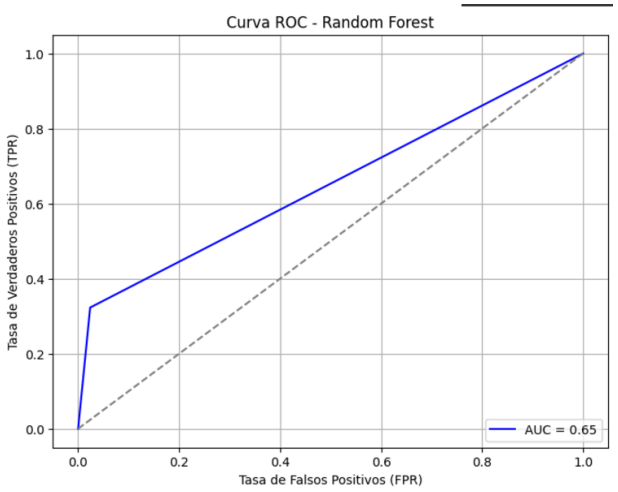
Hiper Parámetro	Valor
activation	tanh
alpha	0.0001
hidden_layer_sizes	50;50

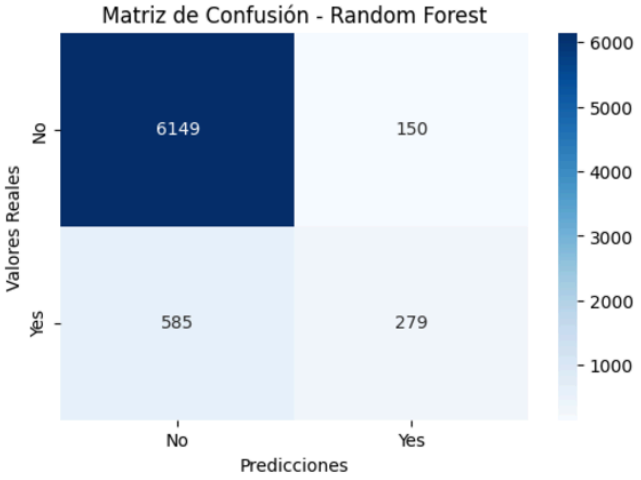
RESULTADOS

Modelo	Precisión	ReCall	F1- Score	Accuracy
Random forest	0,88	0,9	0,88	0,9

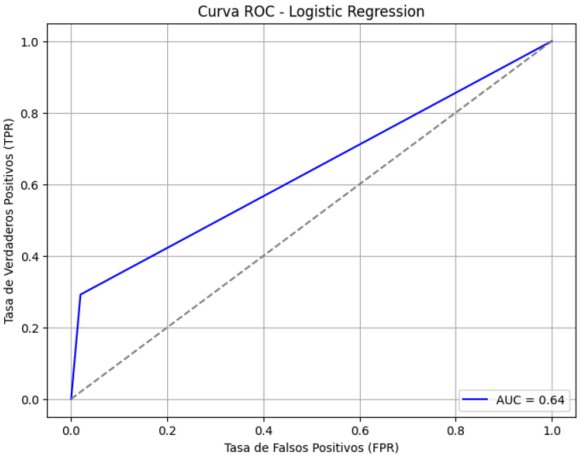
Logistic Regression	0,88	0,9	0,88	0,9
Neural Network	0,87	0,89	0,87	0,89

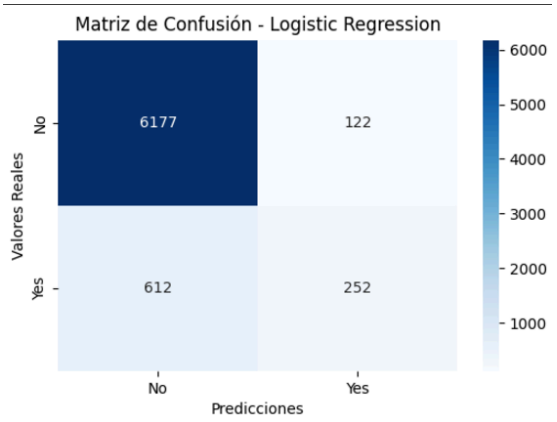
Random forest



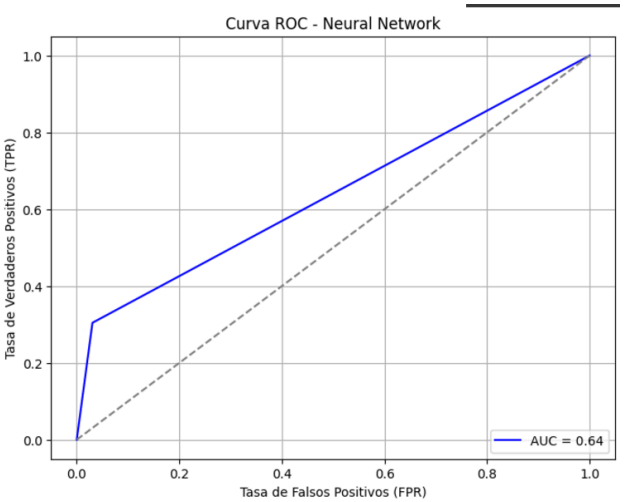


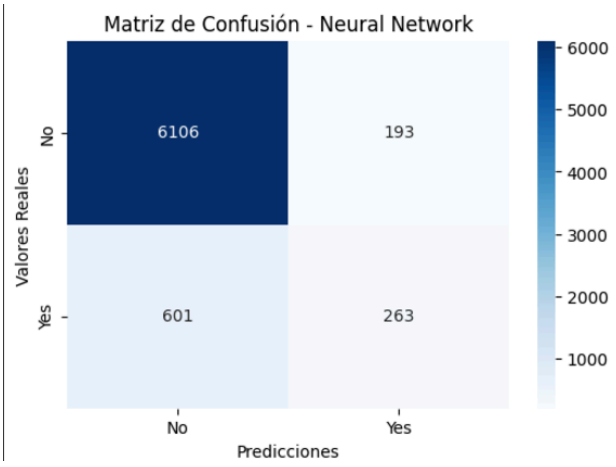
Logistic Regression





Neural Network





DISCUSIÓN Y CONCLUSIONES

En este proyecto, se evaluaron y compararon tres modelos de machine learning (Random Forest, Logistic Regression y Neural Network) con el objetivo de predecir la suscripción de clientes a un producto bancario luego de una campaña de marketing.

El análisis incluyó la optimización de hiperparámetros mediante Grid Search y la evaluación de métricas clave como accuracy, precision, recall, y F1-score. Además, se analizaron las matrices de confusión y las curvas ROC para entender mejor el desempeño de cada modelo, tanto en términos globales como específicos para cada clase, considerando un dataset desbalanceado.

El proceso permitió identificar las fortalezas y debilidades de cada modelo, con especial atención a la capacidad de detectar correctamente la clase minoritaria (clientes que sí se suscribieron), un aspecto crítico para este caso de negocio.

Los modelos tienen un desempeño similar en términos de accuracy:

- Random Forest: 90%
- Logistic Regression: 90%
- Neural Network: 89%

Sin embargo, al analizar más profundamente las métricas por clase (precisión, recall y F1-score), surgen diferencias significativas, especialmente en cómo manejan la clase minoritaria (1).

Dado que se considera más importante maximizar el rendimiento de las suscripciones (Subscription = 1), Random Forest resulta la mejor opción. Proporciona un buen balance entre las métricas, con el mejor F1-Score (43%) para la clase 1, superando a Logistic Regression (41%) y Neural Network (40%).

REFERENCIAS

guide, s., & Jain, S. (2024, January 3). *What is a neural network?* GeeksforGeeks. Retrieved December 1, 2024, from https://www.geeksforgeeks.org/neural-networks-a-beginners-guide/

IBM SPSS. (2024, Enero 22). *What Is Random Forest?* IBM. Retrieved December 1, 2024, from https://www.ibm.com/topics/random-forest

Kanade, V. (2022, April 8). *Everything You Need to Know About Logistic Regression*. Spiceworks. Retrieved December 1, 2024, from https://www.spiceworks.com/tech/artificial-intelligence/articles/what-is-logistic-regression/

MathWork. (2023, 9 26). *Support Vector Machine*. Support Vector Machine. Retrieved 12 1, 2024, from https://la.mathworks.com/discovery/support-vector-machine.html