Informe Proyecto Telco NN
Ciencia de Datos - UTN - FRBA - 2024

Introducción y Objetivo

En este trabajo se realiza un análisis de clasificación con machine learning, utilizando python, para llegar al objetivo planteado por la empresa Telco NN la cual nos pidió que los ayudemos a predecir qué clientes dejarán la compañía. Para ellos se nos presenta un dataset con una cartera de 7.043 clientes con 21 variables que muestran algunas características de estos en la empresa.

Dataset

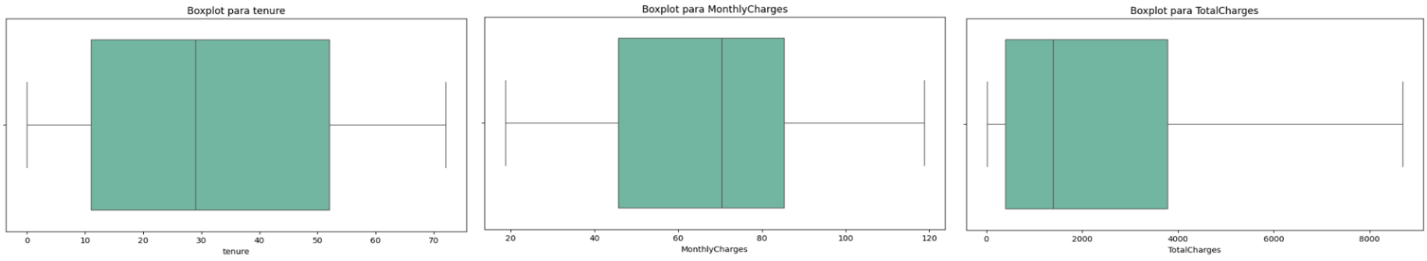
Cada columna representa una featured del cliente y cada fila representa uno diferente. Por lo tanto, como contamos con un dataset de 7043 filas y Columnas 21 podemos afirmar que contamos con 7043 clientes en la muestra y con 21 características de ellos. También vemos que tipo de features son, es decir, numéricas (float) o categóricas (object). Del total de 21 variables, 4 son numéricas y el resto categóricas. La variable al final del dataset llamada “Churn” es la que muestra si el cliente se fue o no, es decir, la clasificación buscada, por lo tanto es lo que buscamos predecir.

Exploratory Data Analysis (EDA)

La primera y segunda columna son de “numeración” como el ID del cliente; como en python ya se crea un índice no nos hace falta tener más columnas con ese propósito, entonces las sacamos. También vemos que tipo de features son, es decir, numéricas (float) o categóricas (object).

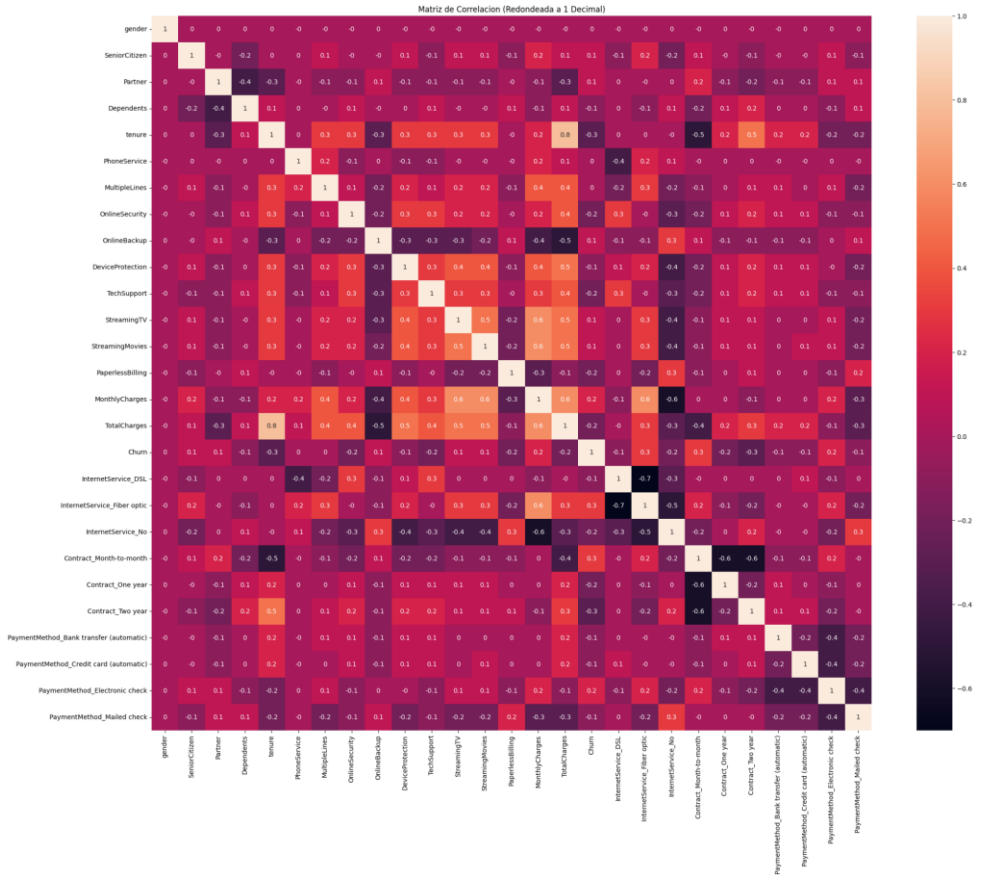
Luego pasamos a evaluar si hay valores nulos en el dataset y nos encontramos que en 15 de ellas si y que representan valores de un 13% a 17%.Teniendo en cuenta que son una importante parte del dataset, decidimos no borrarlas y computarlos la moda en los casos de las variables categóricas y la media para las numéricas. Ya sin nulos en el dataset, convertimos las variables categóricas a dummies, que son variables binarias para que puedan ser usados en modelos matemáticos. Cada categoría se convierte en una columna con valores 0 o 1, facilitando el análisis numérico.

En el EDA, analizamos la presencia de outliers en las features numéricas utilizando el Rango Intercuartílico (IQR), una medida que evalúa la dispersión de los datos centrales. El IQR se calcula como la diferencia entre el tercer cuartil (Q3), que marca el 75% más bajo de los datos, y el primer cuartil (Q1), que delimita el 25% más bajo. Los outliers se identifican si los valores están por debajo del límite inferior (Q1 - 1.5 × IQR) o por encima del límite superior (Q3 + 1.5 × IQR), indicando valores atípicos fuera del rango esperado.
Con código para encontrarlos y con boxplots para visualizarlos, no se encontró ningún outlier en las 3 variables.

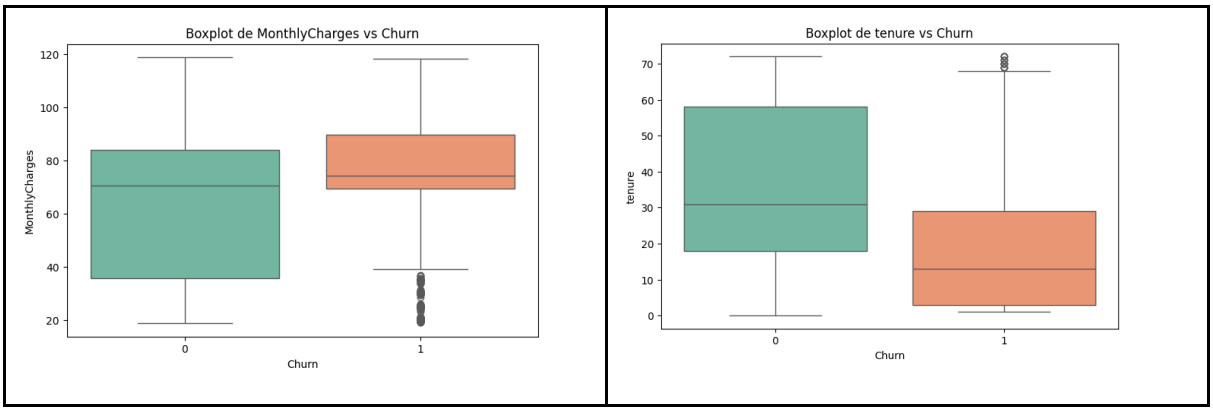


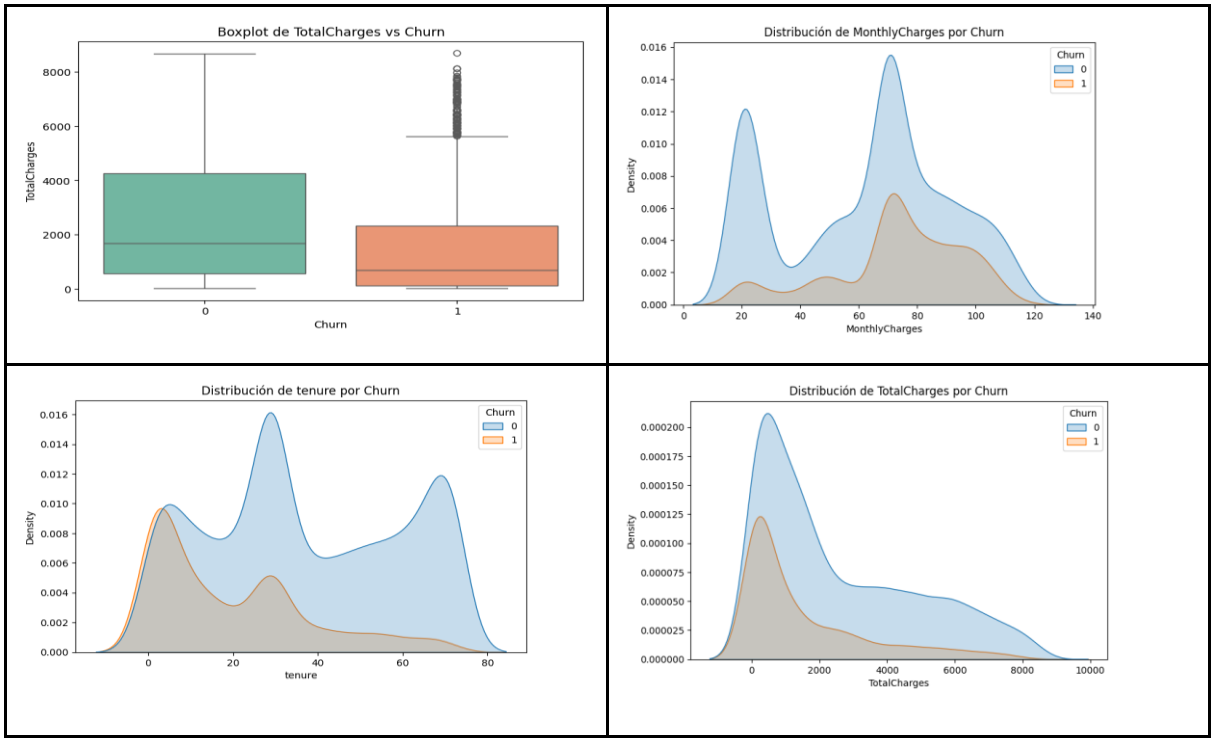
Además analizamos la correlación lineal entre las variables para entender sus relaciones y detectar dependencias lineales. Una correlación positiva indica que ambas variables aumentan o disminuyen

juntas, mientras que una negativa señala que una sube y la otra baja. Valores cercanos a cero indican poca relación lineal. Este análisis ayuda a identificar redundancias, permitiendo descartar variables altamente correlacionadas. En este caso, no se encontraron correlaciones elevadas, por lo que se mantuvieron todas las variables..



Buscamos una relación entre las features de “antigüedad del cliente”, “costo mensual” y “cargos totales” con la variable dependiente “Churn”. Utilizando gráficos como Boxplots y Kd Plots para comparar sus distribuciones y ver si hay diferencias.





Entonces los clientes con mayor permanencia(tenure) y mayor gasto total(TotalCharges) tienden a ser menos propensos a abandonar, mientras que los cargos mensuales más altos podrían estar asociados con una mayor probabilidad de abandono. También hay un churn temprano, con una mediana para clientes con tenure entre 10 y 20 (no se indica la unidad temporal en el diccionario, concluimos que son meses)

Por último calculamos la proporción de Churn para cada categoría de las variables dummy. Vemos que los más relevantes son los casos de:

- Servicio Internet: Optic-Fiber
- Método de Contrato: Month-to-month
- Método de Pago: Electronic Check

El 40% aproximadamente de estos entran en Churn, concluyendo luego que no hay simultaneidad entre ellas.

Modelos y Algoritmos

En la sección de machine learning se evaluó el modelo con siguientes tres algoritmos:

- **Logistic Regression:** funciona aplicando una función logística (sigmoide) para transformar los valores de entrada en probabilidades entre 0 y 1
- **Super vector Machine:** modelo de clasificación supervisada que busca encontrar un hiperplano que separe las clases de datos de manera óptima. Su objetivo es maximizar la distancia (o margen) entre el hiperplano y los puntos más cercanos de cada clase
- **Random forest:** es un modelo de aprendizaje supervisado que utiliza un conjunto de árboles de decisión para mejorar el rendimiento y la precisión del modelo. Cada árbol se construye a partir de una muestra aleatoria del conjunto de datos, y el resultado final se obtiene combinando las predicciones de todos los árboles.

Además se realizó un análisis adicional con una reducción de la dimensionalidad en cada uno de ellos. Primeramente se separa el dataset en train y test, luego se escalan las features para estandarizarlas.

Una vez realizado eso se siguen los mismos pasos para los tres algoritmos sin y con PCA, se lo entrena con los datos de entrenamiento, luego con el modelo entrenado se lo utiliza para predecir con los datos de testeo y finalmente se mide el rendimiento del mismo.

Uno de los algoritmos utilizados para el entrenamiento fue Grid Search, es una técnica para optimizar los hiper parámetros de un modelo. Consiste en probar de manera sistemática todas las combinaciones posibles de valores para los hiper parámetros especificados en una cuadrícula definida por el usuario y seleccionar la mejor.

- Define un conjunto de valores para los hiper parámetros.
- Entrena y evalúa el modelo con cada combinación.
- Selecciona la combinación que maximiza una métrica de rendimiento.

Además para el entrenamiento también le sumamos cross validation, la cual es una técnica utilizada para evaluar el rendimiento de un modelo de aprendizaje automático, dividiendo el conjunto de datos en varios subconjuntos o "folds". El modelo se entrena en algunos de estos subconjuntos y se evalúa en el restante, repitiendo este proceso varias veces para asegurar que cada fold se utilice tanto para entrenamiento como para evaluación

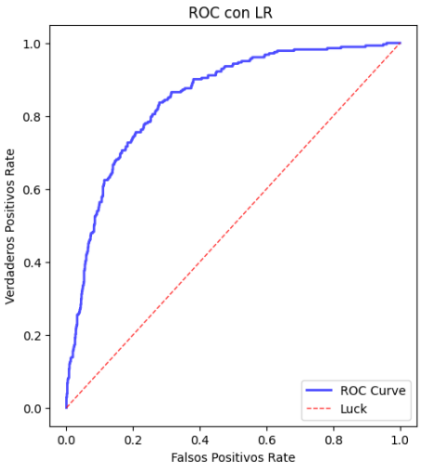
Para medir el rendimiento del modelo se utilizó la accuracy (o precisión) la cual es una métrica de evaluación que mide el porcentaje de predicciones correctas realizadas por un modelo en comparación con el total de predicciones. También se evaluó el rendimiento del modelo utilizando el AUC (Área Bajo la Curva ROC), una métrica que mide la capacidad del modelo para distinguir entre las clases. La curva ROC (Receiver Operating Characteristic) muestra la relación entre la Tasa de Verdaderos Positivos (Sensibilidad) y la Tasa de Falsos Positivos a diferentes umbrales de decisión. El AUC cuantifica el rendimiento del modelo con un valor entre 0 y 1, donde:

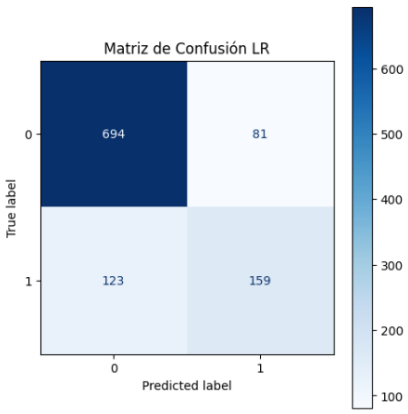
- AUC = 1.0 indica un modelo perfecto.
- AUC = 0.5 indica un rendimiento igual al azar.

Tener en cuenta que cuando hablamos de la accuary o precisión nos referimos al “weighted avg" de precisión, que es el promedio de las puntuaciones de precisión de cada clase, ponderado por el número de instancias reales en cada clase.

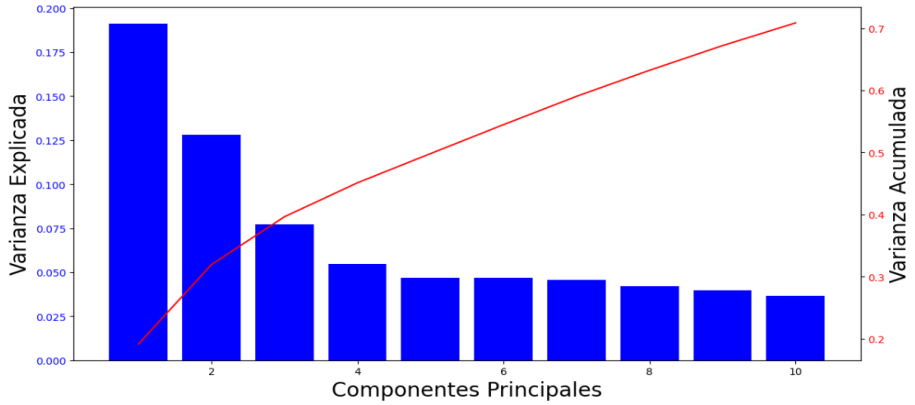
Resultados y conclusiones

Luego de entrenar y testear el modelo con los algoritmos ya mencionados, pudimos llegar a un **accuracy máximo de 0.807 y una área bajo la curva de 0.848**, correspondiente a Logistic Regression sin PCA. Para visualizarlo se muestra la curva ROC y la matriz de confusión.

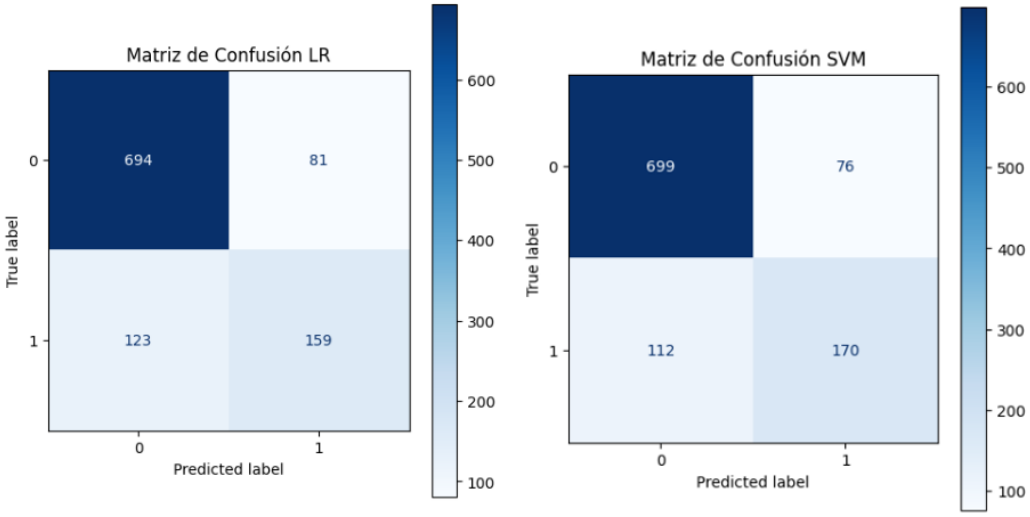




Una conclusión importantes es en este dataset la varianza está muy distribuida entre todas la variables, como vemos en el gráfico tomando las 10 (casi la mitad de las variables) con más varianza, llegamos una acumulada del 0.7 del total del modelo; por lo tanto aplicando la reducción con dimensionalidad los modelos tienden a empeorar su precisión por unos pocos puntos.



Los resultados muestran que la regresión logística fue el mejor modelo, aunque ninguno superó una precisión del 90%. Esto se debe a la desproporción en el dataset, donde solo el 26% de los casos corresponden a clientes que abandonaron la compañía (Churn = 1) y el 74% a los que permanecieron (Churn = 0). La falta de datos representativos, dificulta al modelo identificar correctamente esos casos donde el cliente se va. Esto se refleja en las matrices de confusión, donde los modelos son menos precisos al predecir el Churn, aunque logran un buen desempeño en los casos de clientes que permanecen, lo que eleva la precisión general.



Finalmente, puede suceder que el dataset no tenga la información más relevante para poder predecir el churn y podría faltar datos como:

- Información sobre la competencia(Precios y ofertas, Campañas de marketing)
- Interacciones con el servicio al cliente(Número de llamadas, Motivos, Satisfacción)
- Uso de la plataforma o servicio(Frecuencia, Duración, Dispositivos utilizados para acceder)
- Información demográfica(Nivel socioeconómico, Profesión, entre otros)

Los cuales podrían mejorar la precisión de los modelos.

Fuentes:

- Understanding logistic regression analysis
- An Introduction to Statistical Learning
- Python Data Science Handbook