Trabajo final ciencia de datos

M.Crivaro & M.Stamparin

Curso: I5521 Jueves noche

Fecha de envío
1 dic 2024

Profesores
M.Palazzo - N.Aguirre - S.Chas

Introducción

Contexto del estudio

El presente informe ha sido elaborado en el contexto del curso de Ciencia de Datos dictado en la Universidad Tecnológica Nacional. El caso que se aborda corresponde a la empresa *Telco Comunicaciones*, la cual solicita la **predicción de qué clientes dejarán de utilizar sus servicios.**

Objetivos

- Llevar a cabo un análisis exploratorio de los datos.
- Identificar las relaciones entre las variables.
- Visualización de métricas.
- Predicción de la tasa de abandono de clientes (“Customer Churn”) utilizando los datos proporcionados.

Datos de entrada

- Customer ID: Valor identificador de clientes
- gender: Género del cliente
- SeniorCitizien: Si el cliente es un SeniorCitizien o no
- Partner: Si el cliente tiene un socio o no
- Dependents: Si el cliente tiene dependientes o no
- tenure: Antigüedad del cliente
- PhoneService: Si el cliente tiene un servicio de telefono o no
- MultipleLines: Si el cliente tiene multiples lineas o no
- InternetService: Tipo de servicio de internet que recibe. Si es que recibe
- OnlineSecurity: Si el cliente tiene un servicio de seguridad online o no
- OnlineBackup: Si el cliente tiene un servicio de backup o no.
- DeviceProtection: Si el cliente tiene un seguro del dispositivo o no
- TechSupport: Si el cliente tiene soporte de tecnología o no.
- StreamingTV: Si el cliente tiene servicio de streaming o no
- StreamingMovies: Si el cliente tiene servicios de streaming de peliculas o no
- Contract: Tipo de contrato del cliente
- PaperlessBilling: Si el cliente recibe la factura en papel o no.
- PaymentMethod: Tipo de pago del cliente
- MonthlyCharges: Costo mensual
- TotalCharges: Cargos totales
- Churn: Si el cliente se fue de la compañía o no

Las dimensiones del conjunto de datos fueron visualizadas utilizando la función *np.shape*. El resultado mostró una matriz de **22 columnas** (nuestras variables, como se enumeraron anteriormente) y **7043 filas** (nuestros "samples", que en este caso corresponden a cada uno de los clientes).

Análisis Exploratorio (EDA)

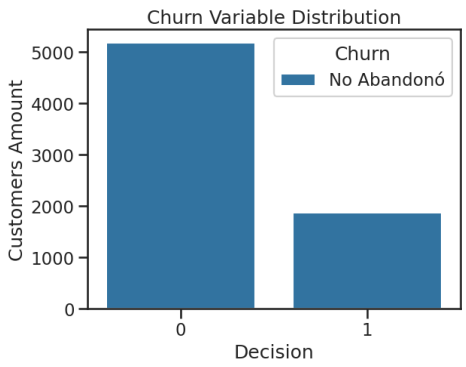
A continuación se enumeran las acciones realizadas en el contexto de "Data Cleaning"

1. **Verificamos que el numero de cliente** (Custumer ID) **no estuviera repetido** con la función *.duplicated* . La mismas nos permitió ver si esa columna presentaba algún valor duplicado. El resultado fue negativo. Con este resultado procedimos a eliminar la columna.
2. **Visualizamos el tipo de dato de cada feature y la cantidad de valores nulos** con las funciones *.dtypes y .isnull().sum()*
3. A partir de la información obtenida en el punto 2 procedimos a **trabajar sobre los valores NaN:**
   - En "MonthlyCharges", "tenure", "SeniorCitizien", "Multiplelines","Onlinesecurity" y "OnlineBackup" completamos los valores faltantes **con 0** *(.fillna(0))*
   - En "Contract" y "Paymethod" completamos los valores faltantes **con "Unknown"** *(.fillna("Unknown"))*
   - En "gender" , "partner", "dependents","paperlessbilling" completamos **con valores aleatorios** ya que entendimos son features que no son relevantes (*fill_random_yes_no(col))*
4. **Crearemos una nueva columna denominada 'ServicesQty'** para determinar la cantidad de servicios a los que está suscrito el cliente, lo que nos permitirá entender si esto influye en su decisión de continuar o cancelar los servicios. Para todas las columnas de servicios, contaremos los "YES", excepto en la columna de "Internet service", donde contabilizaremos las entradas distintas de "NO", ya que esta columna refleja los diferentes tipos de servicio de internet disponibles.
5. Finalmente **convertimos todos los "YES" y los "NO" presentes en el dataset por 1 y 0** respectivamente. De esta forma podemos generar el modelo de machine learning con el dataset completo. *(.replace)*

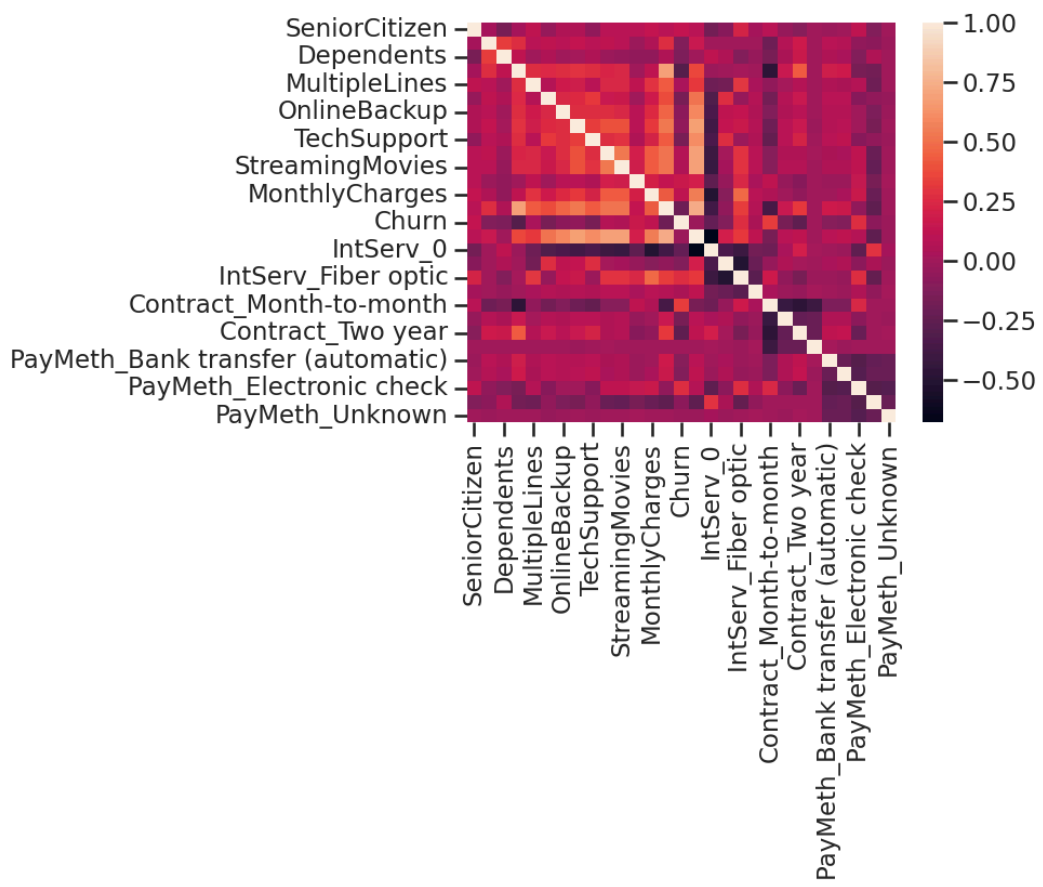
Visualización de métricas

Una vez que trabajamos nuestro dataset, procedimos a realizar diversas visualizaciones para extraer conclusiones sobre la información disponible.

- Primero, **examinamos la distribución de la característica "Churn"** con el fin de comprender cuántos clientes permanecen con el servicio y cuántos han decidido abandonarlo. Para esto utilizamos un *countplot.* El resultado fue 5174 personas permanecían con el servicio y otras 1869 lo habían abandonado.



- **Generamos dummies** de las features que eran categóricas como "Internet Service", "Gender", "Contract" y "Paymentmethods" *(.get_dummies (df.['colum1'], prefix='...')).* Posteriormente **filtramos las columnas numéricas** y guardamos en un variable denominada *"numeric_columns" (.select_dtypes(include=['number'])*
- Realizamos una **matriz de correlación** para identificar posibles relaciones entre nuestras variables. Tomamos la matriz "numeric_columns" *(Var_c_corr = numeric_columns.corr())*



Conclusiones:

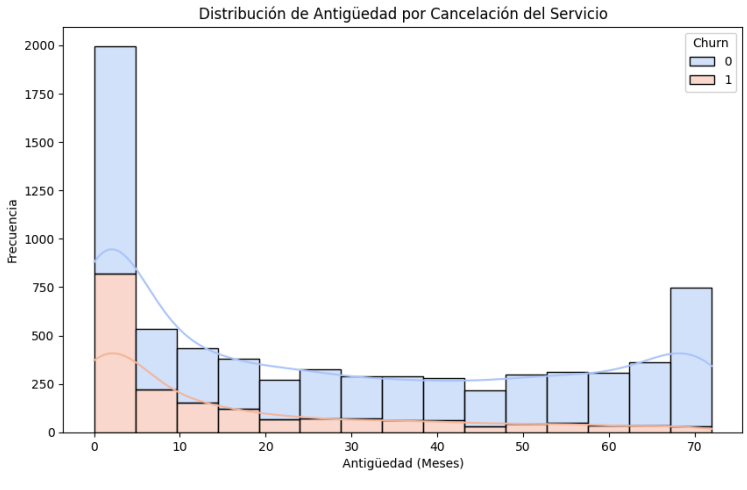
Tenure y Churn: Correlación negativa de -0.3; mayor antigüedad del cliente reduce la probabilidad de abandono, indicando que la fidelidad aumenta con el tiempo.

Churn y Contract: Contratos mes a mes tienen correlación positiva de 0.33 con la cancelación; contratos de 2 años muestran correlación negativa de -0.26, sugiriendo que la estabilidad contractual ayuda a retener clientes.

Churn y InternetService: Clientes de Fibra Óptica tienen correlación positiva de 0.27 con la cancelación, mientras que los de DSL tienen -0.11, indicando que los servicios más baratos podrían retener mejor a los clientes.

Churn y MonthlyCharges: Correlación positiva de 0.12, sugiriendo que aunque un servicio más caro puede aumentar la probabilidad de cancelación, el precio no es una variable significativa en el abandono.

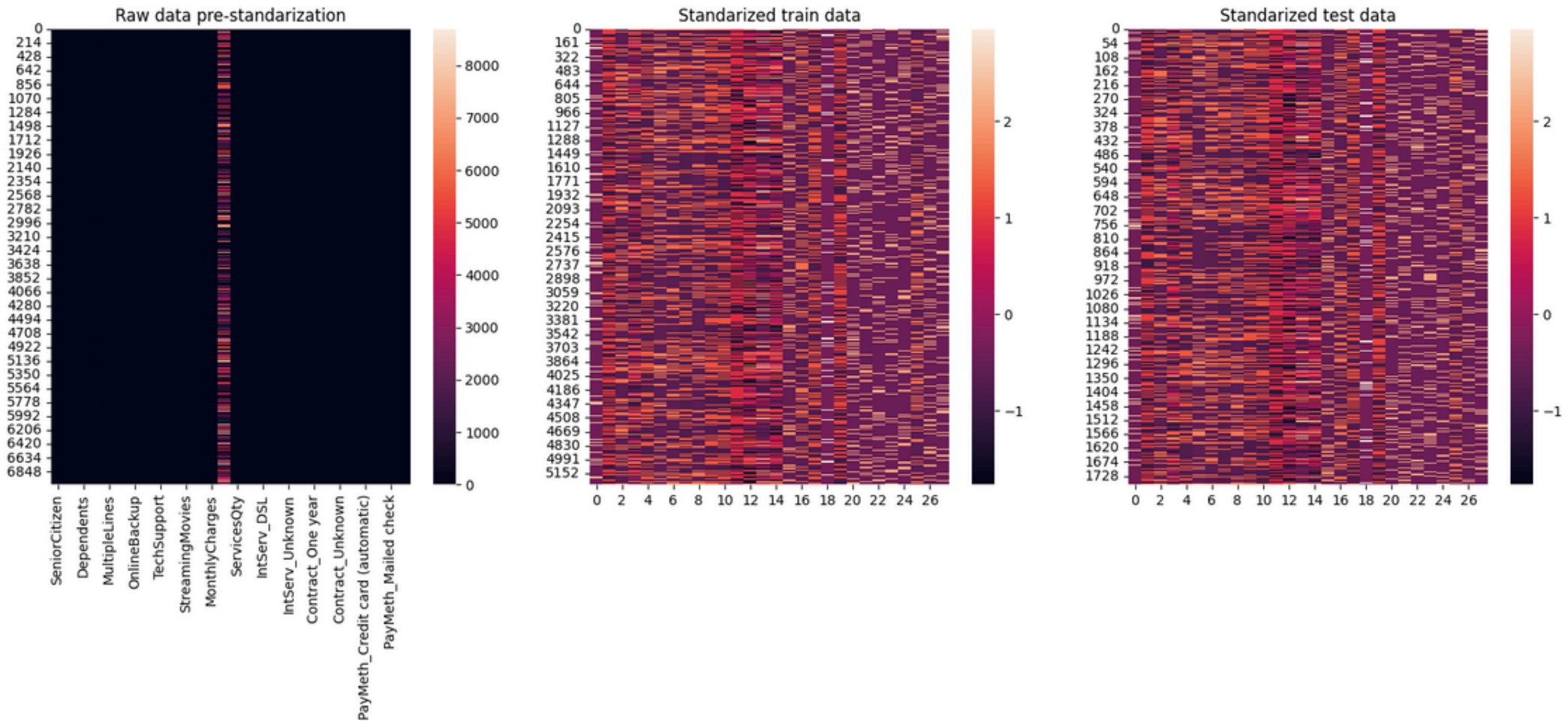
- Realizamos un **histograma combinado** para **visualizar la correlación** destacada entre la **antigüedad** de los clientes y **el nivel de abandono/permanencia** en el servicio



El histograma nos muestra que la mayoría de los clientes tienen una antigüedad menor a los 10 meses y que la proporción mas alta de clientes que abandona el servicio no llega a los 5 meses.

Machine Learning

1. **Definimos nuestra "y"** a predecir como la feature "Churn" y el resto de las features como**"x"**. Esto lo hicimos filtrando el dataset ya prefiltrado "numeric_columns"
2. **Separamos en train y test** (test set 25%) → *train_test_split(x, y, test_size=0.25, random_state=4)*
3. Para **dejar todas las features en los mismos rangos** utilizamos el **stdscaler** para que la media sea =0 y el desvio STD = 1 → *preprocessing.StandardScaler().fit(xtrain)*
4. **Aplicamos el scaler a los datos de test** y finalmente **visualizamos la matriz de datos** de entrenamiento previo y post a estandarizacion/preprocesamiento con un **mapa de calor.**

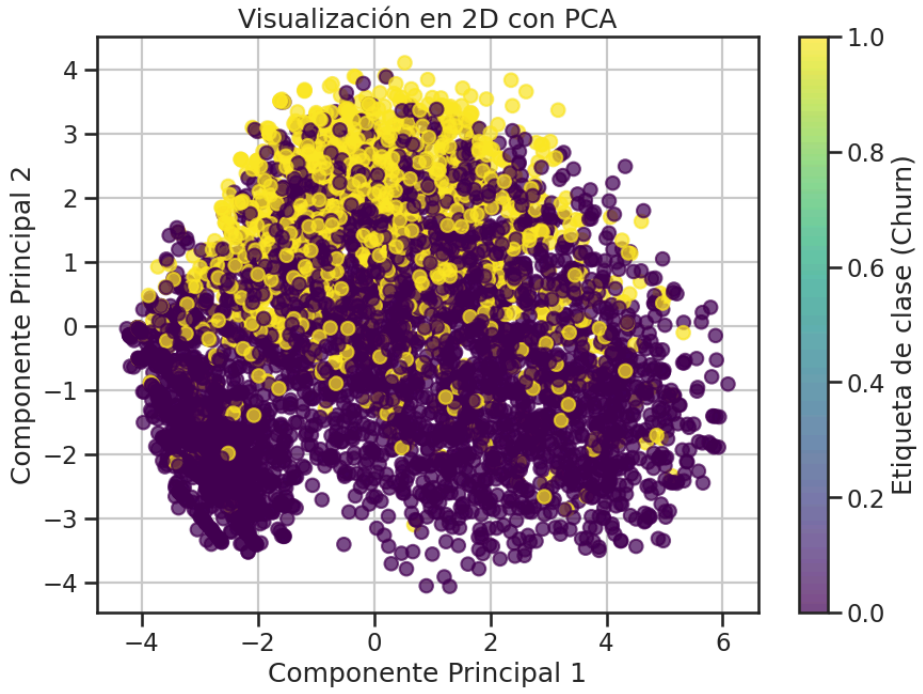


Aplicación del modelo

1. Definimos el **modelo de clasificacion → "Logistic Regression"** bajo el nombre de modelo_lr *(LogisticRegression().fit(xtrain_scal, ytrain)*
2. Realizamos un **Grid search y posteriormente un Cross Validation** para definir los mejores hiperparametos → *grid_search = GridSearchCV(estimator=modelo_lr, param_grid=param_grid, cv=5, verbose=1).*
3. Un vez detectados los mejores hiperparametros **volvemos a ajustar el modelo**

Aplicación PCA

1. **Realizamos un PCA con el 80% de los datos** en busca de que esta reducción de dimensionalidad nos de un mejor resultado → *pca = PCA(n_components=0.8)*
2. Aprovechamos esta herramienta para **visualizar en 2D nuestro Dataset** y entender donde se ubican los cliente. El PCA tomará los 2 componentes más importantes (que representan mayor % de varianza). En este caso fueron los features "Total Charges" y "tenure"

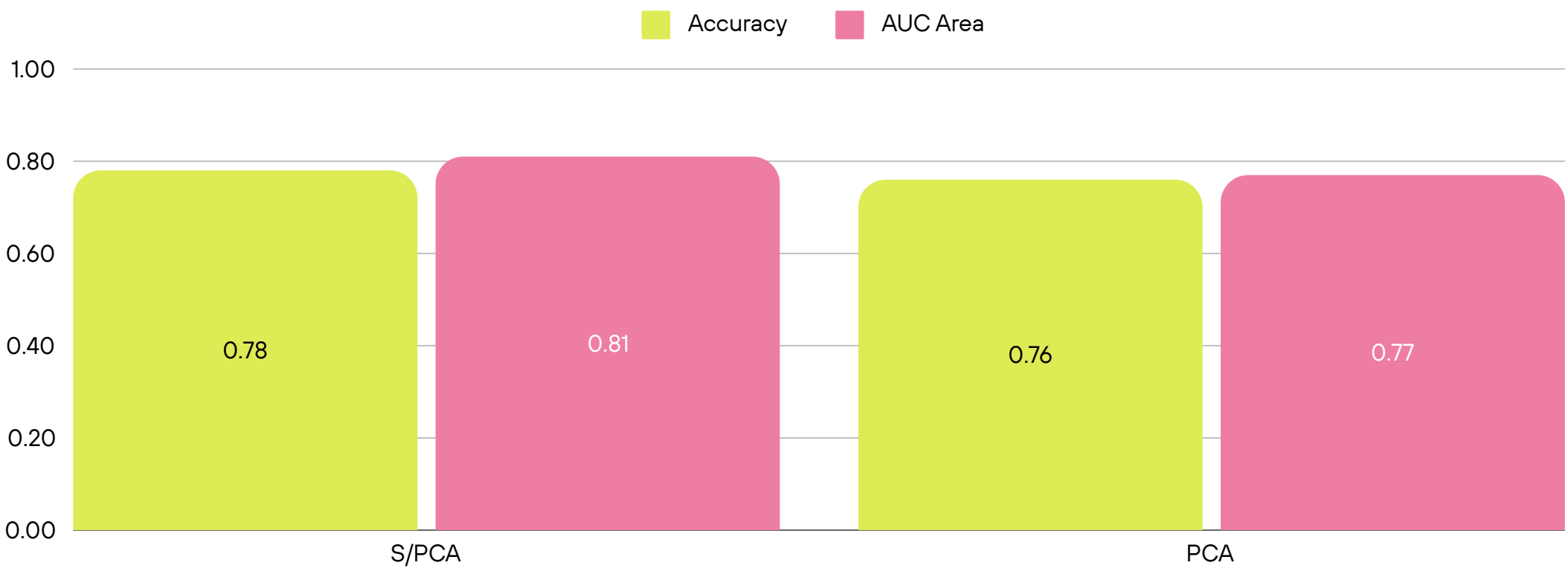


El gráfico muestra las dos direcciones principales de los datos: Componente Principal 1 (CP1) y Componente Principal 2 (CP2), que son combinaciones lineales de todas las features originales, con mayor peso en las más relevantes como TotalCharges y tenure.

La distribución bidimensional de los puntos organiza a los clientes según sus características originales, revelando patrones generales y variabilidad. Aunque este análisis en 2D ayuda a identificar patrones o clusters relacionados con el churn, la separación no es clara, sugiriendo que podría ser mejor considerar diferentes parámetros.

Resultados

A continuación se presenta en un grafico de barras la comparativa entre los resultados obtenidos con el modelo previo a ser aplicado el PCA y luego con PCA aplicado.



Conclusión

Esta investigación desarrolló un modelo de aprendizaje automático destinado a predecir la pérdida de clientes en el sector de telecomunicaciones. Se llevó a cabo un análisis exploratorio de datos (EDA) para identificar variables relevantes, seguido de un proceso de limpieza y normalización de los datos. Se entrenó un modelo de regresión logística que demostró un rendimiento notable en la diferenciación entre los clientes que permanecen y aquellos que se marchan. Se optimizaron los hiperparámetros utilizando GridSearchCV y se evaluó el modelo mediante métricas como la matriz de confusión y la curva ROC.

Aunque se aplicó análisis de componentes principales (PCA) para la reducción de dimensionalidad, no se observó una mejora significativa en el rendimiento. ya que con el modelo **sin PCA** obtuvimos un **mejor rendimiento** general, logrando una exactitud del **78%** y un AUC-ROC de **0.80 pero** con el modelo **con PCA**, que redujo la dimensionalidad de los datos, obtenemos un **menor desempeño**, alcanzando una exactitud del** 75%** y un AUC-ROC de **0.78**. Esto nos indica que la reducción de dimensiones mediante PCA no está siendo más eficiente, por el contrario puede haber perdido información relevante para la clasificación y por eso se redujo el rendimiento.

Este trabajo ilustra cómo el aprendizaje automático puede identificar clientes en riesgo de abandono, lo que permite a las empresas implementar acciones preventivas.

Recursos

- Repositorios de github de la catedra → GitHub - clusterai/Ciencia-de-Datos-UTN-FRBA
- https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression