# Introducción

La retención de clientes en el sector bancario es un desafío fundamental para la sostenibilidad financiera de cualquier institución. La competencia en la industria, junto con la facilidad con la que los clientes pueden migrar a otras entidades, hace que la fidelización se convierta en una prioridad estratégica.

Por otro lado, fidelización de clientes individualizada es difícil, la mayoría de las empresas tienen una gran cantidad de clientes y no pueden dedicar mucho tiempo a cada uno. Adicional los costes serían excesivos, superando los ingresos adicionales. Sin embargo, si una empresa pudiera prever qué clientes es probable que abandonen la empresa, podría centrar sus esfuerzos de fidelización únicamente en estos clientes de alto riesgo de deserción.

La pérdida de clientes es una métrica importante, porque a largo plazo para el banco es mucho más económico y seguro, retener a los clientes existentes que adquirir nuevos.

Este proyecto tiene como objetivo analizar un conjunto de datos bancarios para comprender los factores asociados al abandono de clientes y detectar señales tempranas de una posible pérdida, para desarrollar un modelo predictivo. Para ello, se realiza un análisis estadístico exploratorio detallado de las variables disponibles, enfocándose en aquellas que podrían ayudar a predecir el comportamiento de los clientes más propensos a deserción y de esta forma dedicar un plan de fidelización para estos clientes, disminuyendo la pérdida de clientes y minimizando los costos de campañas de fidelización.

# Justificación

El costo de adquirir un nuevo cliente bancario suele ser significativamente más alto que el de retener uno existente, además, un cliente nuevo no significa un cliente permanente y toma tiempo construir una relación de confianza y que la empresa recupere la inversión de captar ese cliente.Por tanto, identificar clientes en riesgo de abandono permite diseñar estrategias proactivas que reduzcan drásticamente la perdida de clientes en la empresa.

Mediante herramientas estadíticas y modelos predictivos, es posible construir modelos que anticipen esta conducta de abandono que no permita tomar acciones de perder un cliente.

Este proyecto busca sentar las bases para dicha predicción a partir de datos de los clientes, con una exploración de variables que serían utilizadas en el entrenamiento del modelo y levantar un modelo que permita obtimizar los recursos de fidelización y minimize las perdidas de clientes propensos a desertar.

# Antecedentes

Existen múltiples estudios que abordan la predicción del abandono de clientes (customer churn) mediante técnicas de machine learning.

Por ejemplo, un artículo publicado por IBM presenta el uso de árboles de decisión y modelos de boosting para detectar churn en telecomunicaciones.

<https://www.ibm.com/cloud/learn/customer-churn-prediction>

Otro estudio de Towards Data Science aplica redes neuronales y técnicas de reducción de dimensionalidad en bancos para este propósito.

<https://towardsdatascience.com/customer-churn-prediction-in-python-d6a5d5f520>d

Además, plataformas como Kaggle albergan competencias con datasets similares, mostrando que esta es una problemática ampliamente estudiada con alto impacto.

<https://www.kaggle.com/blastchar/telco-customer-churn>

# Definición del problema

¿Qué características definen a los clientes con mayor probabilidad de abandonar el banco? Esta investigación busca identificar y cuantificar esas variables clave a partir de un análisis exploratorio de datos históricos, en preparación para una posterior etapa de modelado predictivo.

# Avance del análisis predictivo y Visualizaciones

Hasta este punto se ha realizado una exploración detallada del conjunto de datos. Algunas variables muestran diferencias marcadas entre los clientes que abandonan y los que permanecen, como la edad y el saldo bancario. Los clientes que abandonan tienen en promedio 7 años más y saldos superiores por más de 18 mil euros. Esto sugiere que estas variables podrían ser útiles para predecir el abandono. También se observan diferencias por país y género, aunque de menor magnitud.

Estadíticas Generales





**Hipotesis de Variables**

Buscamos analizar las variables con el fin de identificar cual de estas varibles puede ayudar al modelo a distinguir una clase de la otra.

**Total\_Trans\_Ct** y **Total\_Trans\_Amt**: Los clientes que abandonaron (Attrited) suelen tener **menos transacciones** y **menor volumen total**.

A diagram of a graph

AI-generated content may be incorrect.

A graph of a graph showing different colored lines

AI-generated content may be incorrect.

A graph showing a number of different colored squares

AI-generated content may be incorrect.**Avg\_Utilization\_Ratio**: Tiende a ser **mayor** en clientes que se fueron → utilizan más porcentaje de su crédito disponible.

**Total\_Amt\_Chng\_Q4\_Q1** y **Total\_Ct\_Chng\_Q4\_Q1**: Los Attrited tienen cambios más bajos en montos y frecuencia ente trimestres → menor actividad reciente.

A graph of a graph showing a number of different colored lines

AI-generated content may be incorrect.

A graph of a bar chart

AI-generated content may be incorrect.

Total\_Revolving\_Bal por clase de cliente: tiende a ser mas bajo para los clientes Attrited.

A chart with a few squares

AI-generated content may be incorrect.

**Target**

|  |  |  |
| --- | --- | --- |
| **TARGET** | **Count** | **%** |
| Attrited Customer (1) | 1,627 | 16.07% |
| Existing Customer (0) | 8,500 | 83.93% |
| **Grand Total** | **10,127** | **100.00%** |

# Entrenamiento de Modelo

Limpieza de Datos se eliminaron, las columnas:

df = df.drop(columns=[

'CLIENTNUM',

'Naive\_Bayes\_Classifier\_Attrition\_Flag\_Card\_Category\_Contacts\_Count\_12\_mon\_Dependent\_count\_Education\_Level\_Months\_Inactive\_12\_mon\_1',

'Naive\_Bayes\_Classifier\_Attrition\_Flag\_Card\_Category\_Contacts\_Count\_12\_mon\_Dependent\_count\_Education\_Level\_Months\_Inactive\_12\_mon\_2'

])

Split data

Se dividió la data en 80/20, reservando 20% para el testeo.

#### TRAIN #####

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, stratify=y, test\_size=0.2, random\_state=42)

Modelos Entrenados

Se entrenaron 2 modelos de clasificación, RandomForestClassifier y LGBMClassifier y se compararon sus desempeños.

RANDOM FOREST

Métricas de Desempeño:

precision recall f1-score support

0 0.96 0.99 0.97 1701

1 0.94 0.77 0.85 325

accuracy 0.96 2026

macro avg 0.95 0.88 0.91 2026

weighted avg 0.95 0.96 0.95 2026

MAE: 0.0449

RMSE: 0.2119

MAPE: 4.49%

AUC ROC: 0.9841

Matriz de Confusión

A graph with numbers and a blue square

AI-generated content may be incorrect.

A graph of a curve

AI-generated content may be incorrect.Curva ROC

LIGHTGBM

Métricas de Desempeño:

precision recall f1-score support

0 0.98 0.99 0.98 1701

1 0.94 0.89 0.92 325

accuracy 0.97 2026

macro avg 0.96 0.94 0.95 2026

weighted avg 0.97 0.97 0.97 2026

MAE: 0.0262

RMSE: 0.1617

MAPE: 2.62%

AUC ROC: 0.9925

Matriz de Confusión

A blue squares with white text

AI-generated content may be incorrect.

Curva ROC

A graph of a curve

AI-generated content may be incorrect.

COMPARATIVA DE MODELOS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | precision | recall | f1-score | support |
| CLASE 1 | RANDOM FOREST | 0.94 | 0.77 | 0.85 | 325 |
| **LIGHTGBM** | **0.94** | **0.89** | **0.92** | **325** |

|  |  |  |
| --- | --- | --- |
| MEDIDAS | RANDOM FOREST | LIGHTGBM |
| MAE | 0.0449 | **0.0262** |
| RMSE | 0.2119 | **0.1617** |
| MAPE | 4.49% | **0.0262** |
| AUC ROC | 0.9841 | **0.9925** |

Si bien ambos modelos son muy buenos modelos y lograr separar las clases de buena manera, el modelo de Lightgbm es mejor que el Random Forest opteniendo mejores métricas y destaca en la captura de mayores casos de deserción. Tambien sus erros son mas bajos, por todo esto el escogeremos el modelo de Lightgbm.

Analisis de Modelo

A graph with blue and white bars

AI-generated content may be incorrect.FEATURE IMPORTANCE

Cabe mencionar que podemos ver cómo algunas de las features de nuestras hipotesis tienen alto nivel de relevancia para el modelo.

**Análisis de Shaps**

A logo with a circle and text

AI-generated content may be incorrect.

Un **análisis de SHAP (SHapley Additive exPlanations)** es una técnica para explicar cómo cada característica (variable) contribuye a la predicción de un modelo de machine learning. SHAP te dice **c**uánto y en qué dirección cada variable influye en la predicción de un modelo.

**¿Para qué sirve?**

* **Interpretar modelos complejos** (como Random Forest, XGBoost, etc.)
* Ver **qué variables son más importantes** globalmente.
* Entender **por qué un caso específico fue clasificado como positivo o negativo**.
* Detectar **sesgos o problemas de confianza** en modelos.

Realizaremos un análisis de shaps para entnder nuestro modelo y como las variables afectan su predicción.

Es una análsis sencillo puedes ver la documentacion aqui:

<https://shapash.readthedocs.io/en/latest/overview.html>

y\_pred\_series = pd.Series(model.predict(X\_test), index=X\_test.index)

xpl = SmartExplainer(model=model)

xpl.compile(

x=X\_test,

y\_pred=y\_pred\_series

)

**Dashboard Generado**

A screenshot of a computer

AI-generated content may be incorrect.

Principales Features

Total\_Trans\_Ct

A screen shot of a graph

AI-generated content may be incorrect.

En la linea vertical podemos observar el valor shap, cuando es mayor a 0 tiene impacto positivo en la predicción del modelo, cuando es negativo tiene un impacto negativo en la predicción y cero neutro.

En este caso, cuando los valores de la variable Total\_trans\_ct, estan entre 0 y 60 aproximadamente, aporta positivamente a la predicción del modelo, en cambio para valores mayores a 60 aporta negativamente a la predicción.

En otras palabras si analizamos esta feature en particular, los clientes con Total\_trans\_ct, tienen mas probabilidades de desertar.

A graph with orange and blue dots

AI-generated content may be incorrect.

A graph with numbers and a chart

AI-generated content may be incorrect.

**Conclusión**

El objetivo principal de este proyecto fue desarrollar un modelo de clasificación capaz de predecir qué clientes de un banco tienen alta probabilidad de desertar (churn), con base en variables disponibles.

Esta capacidad predictiva es crucial para permitirle al banco implementar estrategias preventivas de retención, con impacto directo en la reducción de pérdidas financieras.

Para la modelación, se emplearon dos algoritmos poderosos:

* **Random Forest Classifier**
* **LightGBM (Light Gradient Boosting Machine)**

Ambos modelos demostraron un alto poder predictivo y ambos clasificadores fueron capaces de separar correctamente las clases (clientes existentes vs. desertores).

Sin embargo LightGBM demostró tener mejores métricas generales.

Entre las ventajas observadas:

* Mayor AUC (más capacidad para separar clases en diferentes umbrales).
* Menor MAE y RMSE (más precisión en las predicciones).
* Mejores tasas de recall para detectar desertores.

Por estas razones, LightGBM fue elegido como el modelo final, al ser más adecuado para este tipo de clasificación desequilibrada y por su rendimiento computacional eficiente.

Contar con un modelo predictivo de churn permite al banco:

* Identificar clientes en riesgo de abandonar con anticipación.
* Lanzar campañas personalizadas de retención (como beneficios, llamadas, ofertas).
* Optimizar recursos de marketing, enfocando esfuerzos en los perfiles con más probabilidad de desertar.
* Reducir la pérdida de ingresos por cancelaciones de cuentas o tarjetas.

Este proyecto demuestra el gran valor que aportan las herramientas de machine learning al negocio:

Machine Learning es herramienta fundamental para la competitividad, especialmente en industrias como la banca, donde la fidelización del cliente es esencial.

**Próximos pasos recomendados**

* Integrar el modelo en procesos operativos del banco.
* Automatizar alertas para clientes en riesgo.
* Monitorear el rendimiento del modelo con nuevos datos en producción.
* Experimentar con estrategias de retención basadas en los factores más relevantes.