

1. UNIVERSIDAD ANDRES BELLO
2. Facultad de Ingeniería

**MODELOS DE APRENDIZAJE AUTOMÁTICO**

**PARA LA PREDICCIÓN DE ABANDONO**

**DE CLIENTES EN LA BANCA**

Tesis para la obtención del grado de Ingeniero en Computación e Informática

Autor:

Camilo Suazo

Profesor Tutor:

PhD. David Ruete Zúñiga

Santiago de Chile, 2025

**RESUMEN**

La filtración de clientes en un banco, o rotación de estos, es una problemática de gran preocupación en el sector bancario. Actualmente, ante la variada oferta de productos bancarios, junto con el surgimiento de bancas digitales, con posibilidad de acceder a estos servicios sin mayores requisitos, ha incentivado el uso y la contratación de estos. Por lo cual, tanto la banca tradicional, como la digital, deben emprender diversos esfuerzos por mantener a sus clientes satisfechos y leales con sus bancos. Es sabido que mantener a los clientes, significa un gasto menor, que atraer a potenciales nuevos clientes. Por lo tanto, las instituciones han requerido de utilizar las estrategias tecnológicas actuales con tal de asegurar su cartera de clientes y generar lo que se llama fidelización. El presente proyecto presenta un modelo de predicción y prevención del churn bancario, o rotación de clientes, con el fin de reducir lo máximo posible esta situación. Este modelo estará basado en el machine learning, con el cual podremos analizar y detectar el porqué de del cierre de cuentas por parte de los clientes. Todo esto a través de un dataset con datos anonimizados procedentes de Europa. Primeramente se presenta un examen general de los datos, a través de un análisis descriptivo con lo cual podremos entender las variables principales de los datos y las correlaciones que pudiesen haber entre ellas. Posteriormente, a través del uso de diversos modelos de predicción de machine learning, se ejecuta un análisis predictivo de los datos, con lo que se descubre cuáles son los principales factores de la rotación o cierre de cuentas bancarias por parte de los clientes, según el data set utilizado. El set de datos analizado comprende 18 variables, relacionadas a la edad, puntuación crediticia, número de productos, salario estimado, entre otros. Esta información fue obtenida a través del reconocido sitio web de data sets Kaggle.com. Este análisis es a partir de una muestra anonimizada que toma la información de un banco europeo con clientes en varios países, tales como Francia, España, Alemania y han presentado cierre de cuentas o no. Los beneficios obtenidos a partir de este proyecto son que los bancos deben mantener satisfechos a los cliente que presentan cuentas por más tiempo, con saldos generalmente altos. El maximizar el valor del cliente en el tiempo es una tarea fundamental para mantener una cartera de clientes estables, lo cual implica un beneficio permanente para el banco.

Se propone, que a través de este proyecto, los bancos puedan utilizar los modelos, con sus propios datos para ayudar a disminuir la rotación de clientes en sus instituciones.

# Índice de Contenido

Portada

Resumen

[1. Índice de Contenido 4](#_Toc207749189)

[1. Introducción 5](#_Toc207749190)

[2. Problema 6](#_Toc207749191)

[2.1 Diagrama de Ishikawa 7](#_Toc207749192)

[2.2 Descripción del diagrama 7](#_Toc207749193)

[3. Objetivos 10](#_Toc207749194)

[3.1 Objetivo General: 10](#_Toc207749195)

[3.2 Objetivos Específicos: 10](#_Toc207749196)

[3.3 Alcance 11](#_Toc207749197)

[4. Marco Teórico 13](#_Toc207749198)

[5. Metodología de Trabajo 34](#_Toc207749199)

[6. Desarrollo de la Solución 44](#_Toc207749200)

[7. Resultados 54](#_Toc207749201)

[8. Conclusiones 64](#_Toc207749202)

[9. Referencias 67](#_Toc207749203)

[10. Anexos y Apéndices 69](#_Toc207749204)

[**10.1. Apéndice A: Fuente de Datos** 69](#_Toc207749205)

[**10.2. Apéndice B: Cuaderno de Código** 69](#_Toc207749206)

# Introducción

La filtración de clientes en una institución bancaria es un problema al que día a día se enfrentan los bancos, ya sean tradicionales, o digitales. Esto puede deberse a razones aparentemente evidentes, las cuales podríamos explicar con un análisis básico del grado de satisfacción que puedan tener los usuarios y justificar a través de este la deserción. Actualmente hay estudios y modelos que han trabajado en predecir y prevenir el abandono o cierre de cuentas bancarias. Sin embargo, es necesario continuar en la mejora de estos, integrando otras variables para generar predicciones más precisas para optimizar la retención.

A través del tiempo, día a día, las personas adquieren productos bancarios, a través de contratos, para poder obtener los beneficios que estos significan. Ya sea el acceder a tarjetas de crédito, créditos de consumo, créditos hipotecarios para acceder a la vivienda, o como una cuenta de ahorro, entre otros servicios. Entre los aportes que le significa a una persona obtener estos productos están el acercamiento a los productos financieros, mayor control sobre el dinero, la posibilidad de llevar un presupuesto, la seguridad, ya que significa no estar cargando físicamente con el dinero, junto con el acceso a seguros en caso de robo, se pueden utilizar medios electrónicos de pago, actualmente las billeteras digitales están muy en boga también (Forbes Staff, 2015). Sin embargo, a pesar de los múltiples servicios y beneficios que pueden entregar los bancos, la deserción o abandono de un producto bancario por parte de un cliente puede significar un alto costo para la entidad, “Las deserciones de clientes tienen un impacto sorprendentemente significativo en los resultados. Pueden estar más relacionadas con las ganancias de una empresa de servicios que con la escala, la cuota de mercado, los costes unitarios y muchos otros factores que suelen asociarse con la ventaja competitiva..” (Reichheld & Sasser Jr., 1990).

# Problema

El churn bancario —entendido como la pérdida de clientes— representa un desafío relevante para las instituciones financieras tradicionales, especialmente en el contexto actual, donde la digitalización y el auge de las fintechs han transformado el comportamiento del consumidor. Cada cliente que abandona representa no solo una pérdida económica directa, sino también un posible debilitamiento de la reputación de la institución y una oportunidad para sus competidores.

Estudios de mercado muestran que los usuarios valoran la experiencia digital, la personalización de servicios y la eficiencia en la atención. Las instituciones que no logran adaptarse a estas expectativas suelen enfrentar un aumento en la tasa de deserción de clientes. Además, se ha demostrado que reducir las deserciones en un 5% puede aumentar las ganancias entre un 25% y un 85%, lo que subraya el impacto estratégico de la retención de clientes (Reichheld & Sasser, 1990).

Frente a este escenario, se plantea la oportunidad de diseñar un sistema de predicción de churn bancario enfocado en cuentas corrientes personales, utilizando algoritmos de machine learning y datos simulados obtenidos desde plataformas abiertas como Kaggle. El objetivo es identificar patrones de comportamiento que preceden al abandono, permitiendo a las instituciones anticiparse con estrategias de retención más efectivas.

Este anteproyecto se desarrolla en un contexto académico y simulado, con el propósito de demostrar la viabilidad técnica del uso de modelos predictivos en la industria bancaria, sin involucrar datos confidenciales ni desplegar soluciones reales en instituciones financieras.

## Diagrama de Ishikawa

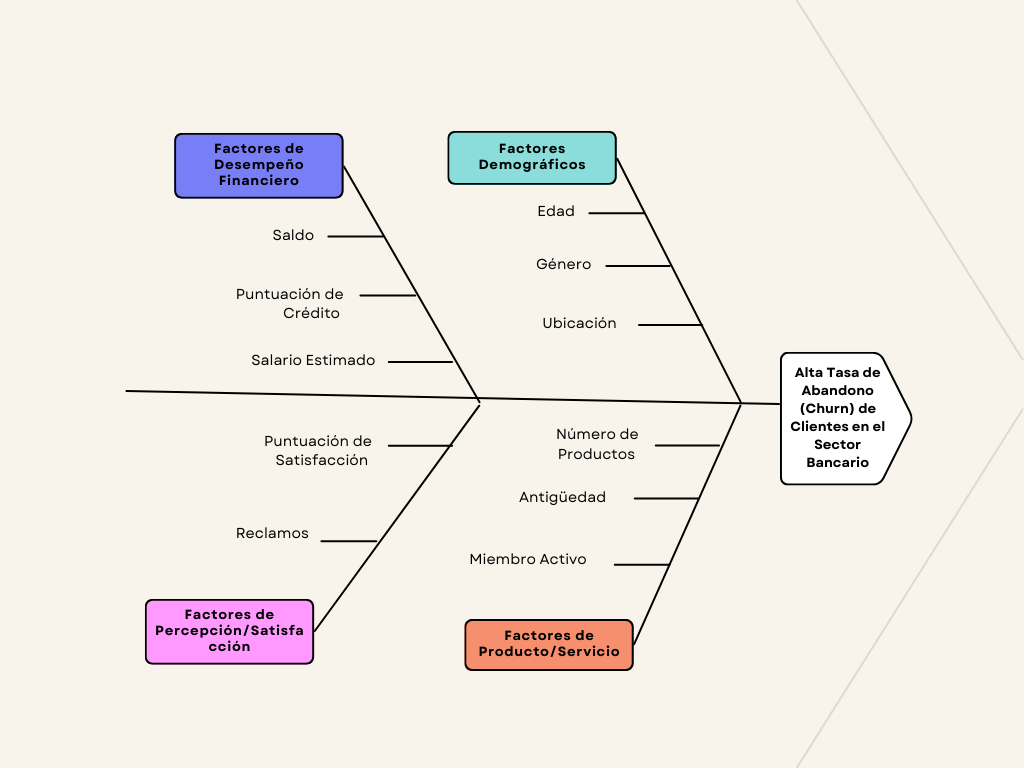


Figura 1 Diagrama de Ishikawa (Fuente: elaboración propia)

## Descripción del diagrama

El diagrama de Ishikawa, o de espina de pescado, es una herramienta visual que permite organizar y jerarquizar las causas que contribuyen a un problema específico. Para este proyecto, el problema general es la "Alta Tasa de Abandono (Churn) de Clientes en el Sector Bancario".

El diagrama, presentado a continuación, desglosa este problema en las causas principales que, según la literatura y el conocimiento del sector, influyen en la decisión de un cliente de dejar su banco. A diferencia de un enfoque teórico, este diagrama se ajusta a las variables disponibles en el dataset de Kaggle, lo que garantiza que el análisis sea directamente relevante para los resultados que se obtuvieron.

**2.3 Causas Principales (Espinas del Diagrama)**

**Factores Demográficos:** Estas son las características inherentes del cliente que pueden influir en su comportamiento.

* Edad (Age): La etapa de vida del cliente puede correlacionarse con sus necesidades bancarias y su propensión a cambiar de institución.
* Género (Gender): A pesar de la evidencia limitada, las preferencias y comportamientos bancarios pueden variar entre géneros.
* Ubicación (Geography): Las regulaciones locales, el poder adquisitivo y las ofertas de la competencia varían por región, lo que influye en la lealtad.

**Factores de Producto/Servicio:** Estos son los atributos del portafolio del banco que el cliente consume.

* Número de Productos (NumOfProducts): Este factor mide la profundidad de la relación del cliente con el banco. A más productos, mayor es la fricción para cambiarse.
* Antigüedad (Tenure): El tiempo que un cliente ha permanecido en el banco es un indicador de lealtad, pero también puede mostrar un estancamiento en la relación.
* Miembro Activo (IsActiveMember): Este indicador muestra si el cliente usa activamente sus servicios, lo que puede reflejar su nivel de satisfacción.

**Factores de Desempeño Financiero:** Estos son indicadores clave del estado económico del cliente.

* Saldo (Balance): El saldo en la cuenta puede ser un reflejo del valor del cliente, pero también un indicador de su situación financiera, lo que lo hace vulnerable a ofertas de la competencia.
* Puntuación de Crédito (CreditScore): Un puntaje bajo puede llevar al cliente a buscar mejores condiciones de crédito o a ser más propenso a la migración.
* Salario Estimado (EstimatedSalary): Este factor se correlaciona con la capacidad económica del cliente y su perfil de riesgo.

**Factores de Percepción/Satisfacción:**

* Puntuación de Satisfacción (Satisfaction Score): Refleja la percepción del cliente sobre su experiencia con el banco. Una puntuación baja es un claro precursor del churn.
* Reclamos (Complain): Gran cantidad de reclamos podría estar relacionada el abandono.

# Objetivos

## Objetivo General:

Desarrollar un modelo predictivo de abandono de clientes de cuentas corrientes bancarias, utilizando técnicas de machine learning sobre un conjunto de datos simulados.

## Objetivos Específicos:

Preparar y limpiar los datos para asegurar que estén completos y listos para el modelo, identificando y manejando valores nulos, realizando encoding y normalización de variables numéricas, y realizando un análisis exploratorio de los datos (EDA) considerando factores de competencia y producto/servicio bancario.

Analizar los factores de churn utilizando el dataset, evaluando cómo variables como el número de productos bancarios, saldo promedio, promociones, y actividad de la cuenta afectan la probabilidad de abandono, incorporando los factores de atención al cliente y tecnología.

Desarrollar un modelo predictivo de churn con técnicas de machine learning como Random Forest, Decision Tree y SVM, optimizando el rendimiento y evaluando el modelo mediante métricas de desempeño, considerando cómo los factores de satisfacción del cliente y tecnología influyen en la predicción.

Evaluar y validar el modelo asegurando su efectividad en la predicción del churn, probando su capacidad de generalización y haciendo ajustes en los hiperparámetros o modelos, con un enfoque en los factores de competencia y factores económicos.

## Alcance

Este proyecto se desarrollará en un **entorno simulado y académico**, utilizando un conjunto de datos públicos obtenidos desde la plataforma Kaggle, específicamente enfocados en información de clientes con cuentas corrientes bancarias. El enfoque principal será el diseño y construcción de un modelo de machine learning capaz de predecir la probabilidad de abandono de un cliente (churn), a partir de variables históricas y demográficas contenidas en el dataset.

El desarrollo se centrará exclusivamente en el modelo predictivo. No se incluirán funcionalidades adicionales como interfaces de usuario, dashboards, automatización de campañas o integraciones con sistemas reales. Tampoco se trabajará con datos reales de empresas bancarias, manteniéndose el tratamiento del problema en un contexto hipotético.

El proyecto contempla las siguientes actividades dentro de su alcance:

* Análisis exploratorio y limpieza del dataset simulado.
* Selección y justificación de variables relevantes para la predicción.
* Prueba comparativa de distintos algoritmos de clasificación supervisada, específicamente **Decision Tree, Random Forest, SVM, Naive Bayes**.
* Entrenamiento, validación y evaluación del modelo con métricas estándar como accuracy, precisión, recall, F1-score y ROC-AUC.
* Documentación del proceso completo y justificación técnica de las decisiones tomadas.

**Quedan fuera del alcance**:

* Desarrollo de interfaz gráfica o dashboard.
* Implementación del modelo en un entorno productivo o en instituciones reales.
* Uso de datos bancarios reales o sensibles.
* Generación de estrategias concretas de retención de clientes.

# Marco Teórico

4.1 Introducción

El presente capítulo establece los fundamentos teóricos que sustentan el desarrollo de un sistema de predicción y prevención de churn bancario mediante machine learning. Se abordan conceptos clave como el problema del abandono de clientes, churn bancario, técnicas de análisis predictivo, algoritmos de machine learning aplicados, y métricas de evaluación. Este marco teórico se alinea con los objetivos del proyecto y el diagrama de Ishikawa presentado, que identifica factores críticos como competencia, atención al cliente, y variables económicas.

**4.1. El Problema del Abandono de Clientes (*Customer Churn*)**

A continuación, definiremos en detalle qué entendemos por churn o customer churn, con sus variantes, y también porque es un problema de primera importancia para las empresas, sobre todo en el sector de la banca que es el que estamos analizando.

**4.1.1. Definición y Tipos de *Churn***:

Define claramente el concepto de *Customer Churn* o abandono de clientes. Explica la diferencia entre *churn* voluntario (cuando un cliente decide irse) e involuntario (por ejemplo, debido a impagos).

Se entiende como Customer Churn o el abandono de clientes, cuando un cliente deserta del uso de un servicio en particular que le estaba entregando la empresa o la institución. Podemos considerar también la definición de Cárdenas Herrera, cuando se refiere a este tema de la siguiente forma, “La tasa de abandono es el porcentaje de clientes que dejan de usar los servicios de una empresa durante un periodo de tiempo determinado. Este ‘dejar de usar los servicios’ puede enfocarse de varias maneras. Desde darse de baja por completo a mantener algún servicio o producto siendo cliente a la vez de varias compañías o entidades. Cada organización e incluso cada departamento puede tomar la que más le interese medir según sean las características del negocio que se pretenda analizar.” (Cárdenas Herrera, 2022). La cual nos especifica de forma más detallada que el abandono también puede ser parcial, lo cual también nos da otras posibilidades de abordar el problema.

También hay que mencionar que hay ciertas diferencias a la hora de abandonar el banco o algún servicio de este. La deserción puede ser voluntaria, pero también involuntaria. En este último caso puede deberse a que el mismo banco se ve en la obligación de cerrar la cuenta, ya sea por desuso o por incumplimiento de contrato. Al respecto Cárdenas nos dice lo siguiente al respecto de la deserción involuntaria: “…puede ser de manera involuntaria (por fallecimiento o alguna causa de fuerza mayor de índole personal totalmente ajena a la entidad) o voluntaria.” (Cárdenas Herrera, 2022)

Para el presente trabajo de tesis sólo nos fijaremos en clientes que abandonaron el banco, o cerraron cuenta, de forma voluntaria.

El abandono de clientes es una métrica clave para la rentabilidad de un negocio ya que como es sabido, es mucho más caro perder un cliente que mantener a los que ya están. Como menciona Hernandez Buades en su tesis doctoral, al respecto de la importancia que tiene el cliente y su retención para una empresa “Considerando a los clientes como activos financieros susceptibles de ser medidos, gestionados y su valor optimizado, el valor de la empresa en el largo plazo estará determinado en gran parte por el valor de toda su cartera de clientes actuales y potenciales, que es lo que conocemos como el customer equity (Gupta et al., 2014), en adelante CE.” (Hernández Buades, 2015). Por lo tanto mantener una tasa de abandono baja es una métrica clave para mantener la rentabilidad de la institución bancaria.

**4.1.2. El Impacto Económico del *Churn* en la Banca**:

Los costos para una institución financiera, o cualquier empresa, del abandono de clientes es significativa para la mantención y el progreso del negocio de esta. En situaciones complejas de fallos o algún error en el servicio se puede dar una deserción de clientes, para entender el costo que significa perder clientes Yanamandram y White nos dicen lo siguiente: “Los clientes de una empresa pueden abandonar la relación que mantenían con la misma tras un fallo en el servicio. Si esto ocurriese, la empresa podría sufrir una disminución de las ventas futuras y un aumento de los costes unitarios derivado de asignar costes fijos entre una menor base de clientes y de la necesidad de atraer nuevos clientes para lograr objetivos de cuota de mercado” (Yanamandram y White ,2006). Por lo tanto, la disminución de ventas y la necesidad de invertir más en atraer potenciales clientes nuevos baja a generar grandes gastos para la institución financiara. Por lo que es mejor mantener integra la cartera de clientes fieles.

Sobre esto, podemos complementar que el impacto financiero que tiene el mantener clientes es notoriamente beneficiosos que perderlos. Respecto a esto Fred Reichheld de la consultora Bain & Company dice que “¿Cómo se traducen las relaciones leales en ahorros de costos? Considere el costo de atender a un cliente de larga data en comparación con el costo de fidelizarlo. En una amplia gama de negocios, los clientes generan ganancias cada vez mayores cada año que permanecen en una empresa. En servicios financieros, por ejemplo, un aumento del 5% en la retención de clientes produce un aumento de más del 25% en las ganancias. ¿Por qué? Los clientes recurrentes tienden a comprar más de una empresa con el tiempo.” (Reichheld, 2011). Por lo tanto, si se mantienen a los clientes, considerar un 25%, es una cifra considerable para el éxito de un negocio y el crecimiento de esto.

**4.2. Fundamentos de *Machine Learning* para la Predicción**

A continuación, se dará detalle de las tecnologías que se ocuparán para realizar este modelo predictivo de abandono de clientes, en base a un dataset público alojado en la página Kaggle.

**4.2.1. Conceptos Clave de *Machine Learning***:

Define brevemente qué es el *Machine Learning* y cómo se diferencia del software tradicional.

Se entiende como Machine Learning, o Aprendizaje Automático, aquél proceso en que la máquina aprende según información etiquetada que le damos, y no etiquetada, para realizar sus propias conclusiones, esta aprende de la experiencia. Esto es dicho de forma muy general. Sin embargo, nos podemos ayudar de lo que dice también el autor Alan T. Norman, “El aprendizaje automático, en su explicación más simple, implica permitir que una computadora varíe sus respuestas e introducir un ciclo de retroalimentación para respuestas buenas y malas. Esto significa que los algoritmos de aprendizaje automático son fundamentalmente diferentes de los programas informáticos que los precedieron. Comprender la diferencia entre la programación explícita y el entrenamiento de algoritmos es el primer paso para comprender cómo el aprendizaje automático transforma fundamentalmente la informática.” (Norman, 2018). Por lo tanto, hay que también entender las diferencias entre el modelo de software tradicional, o la programación tradicional, en la que hay que indicar, a través del código, la acción a iniciar en caso de determinadas circunstancias. Pero, el aprendizaje automático aprende por sí mismo, a través de la experiencia o la información dada, y este toma decisiones o respuestas que no están previamente configuradas.

También hay que aclarar que hay distintos tipos de aprendizaje automático. Los más reconocidos son el **aprendizaje supervisado**, y el **aprendizaje** **no supervisado**.

El **aprendizaje supervisado**, necesita que le entreguen la información etiquetada a través del conjunto de datos dado, y con su correspondiente respuesta para que este pueda aprender de estos datos. Como menciona Norman “Con el aprendizaje supervisado, el algoritmo necesita un conjunto de datos de entrenamiento etiquetado con las respuestas correctas para aprender. Estas etiquetas actúan como un profesor que supervisa el aprendizaje. El modelo deja de aprender cuando alcanza un nivel aceptable de precisión o se queda sin datos de entrenamiento etiquetados.” (Norman, 2018). Es en este tipo de aprendizaje en el cual nos centraremos en nuestro problema.

Por otra parte, el **aprendizaje no supervisado** se refiere a algo contrario al anterior aprendizaje, en este tipo el conjunto de datos no requiere estar etiquetado pues el modelo podrá por sí mismo diferenciar y etiquetar la información para luego entregar una respuesta. “Utilizamos el término aprendizaje no supervisado cuando el conjunto de datos de entrenamiento no tiene etiquetas con una respuesta correcta. En su lugar, permitimos que el algoritmo saque sus propias conclusiones comparando los datos consigo mismo. El objetivo es descubrir algo sobre la estructura o distribución subyacente del conjunto de datos.” (Norman, 2018). Sin embargo, este tipo de aprendizaje no será utilizado en el presente trabajo.

**4.2.2. Proceso de Desarrollo de un Modelo Predictivo**:

A continuación, detallaremos los etapas principales que se necesitan para implementar un modelo de aprendizaje automático, como la obtención de los datos, procesamiento, selección del model, el entrenamiento y su correspondiente evaluación.

Lo primero es tener bien definido el **problema**, cuál es la necesidad que necesitamos resolver, o en base a qué obtención de información podremos sacar insights valiosos para el negocio, o ya sea para cualquier otra área.

Posteriormente, al tener definido el problema debemos seleccionar o **escoger un dataset** con el cual trabajar, que sea adecuado para poder resolver el principal problema que nos hemos propuesto anteriormente.

Luego, deberemos **procesar los datos**, esto quiere decir limpiar, etiquetar y dar formato, para que esté en adecuadas condiciones para ser utilizados en un modelo. Como dice Norman “Esta limpieza de datos a menudo puede reducir un gran conjunto de datos a una fracción de su tamaño original, que es realmente utilizable para el aprendizaje automático.” (Norman, 2018).

Ya con los datos listos para ser utilizados, viene la etapa de **seleccionar un modelo o algoritmo**. Esta etapa consiste en seleccionar un modelo adecuado para el conjunto de datos y el problema que tenemos que resolver. Hay una gran variedad de modelos que detallaremos más adelante.

Escogido el modelo comienza la etapa de **entrenamiento**, esta consiste en utilizar una porción de los datos para ser entrenados y dejar otra porción para ser utilizados a modo de prueba en las posteriores evaluaciones. Sin embargo, y respecto de la etapa anterior Norman nos dice lo siguiente “A menudo, puede ser difícil determinar qué algoritmo funcionará mejor para una aplicación de aprendizaje automático antes de empezar. Por eso, lo mejor es entrenar varios algoritmos inicialmente, seleccionar uno o varios que tengan el mejor rendimiento y luego ajustarlos hasta obtener el modelo que mejor se adapte a sus necesidades.” (Norman, 2018). Por lo tanto es necesario hacer un proceso de iteración entre ambas etapas.

Finalmente, pasamos la etapa final, la **evaluación**. La cual tiene por objetivo evaluar el rendimiento del modelo, a través de diferentes métricas. Este es un proceso iterativo también, porque dependiendo de los resultados es que se vuelve a etapas anteriores.

**4.3. Algoritmos de Clasificación para la Predicción de *Churn***

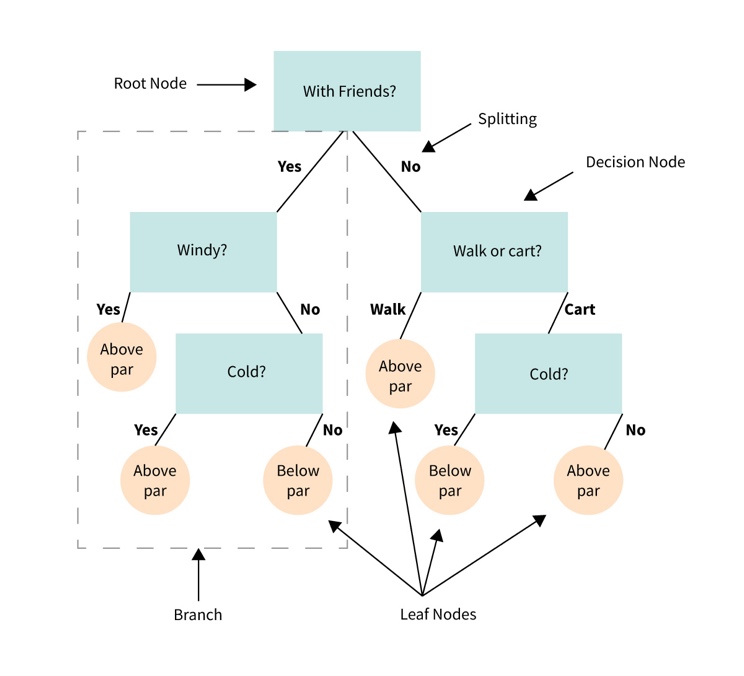
Ahora, se detallarán los modelos o algoritmos utilizados en el presente trabajo de tesis. Explicando en qué consisten, cuáles son sus principales características y porque se decidió escoger estos modelos por sobre otros respecto de nuestro problema de churn bancario.

**4.3.1. Árboles de Decisión (*Decision Trees*)**:

Este modelo consiste en una estructura que se basa en un proceso de decisión secuencial. Parte desde una raíz, evaluando una característica y se selecciona una de las dos ramas. Todo este proceso se repite hasta que finalmente se llega a la última hoja, que sería el objetivo de clasificación que se está buscando, según lo planteado por Bonaccorso (2017).

Al respecto de sus ventajas Bonaccorso nos señala lo siguiente, “Considerando otros algoritmos, los árboles de decisión parecen ser más simples en su dinámica; sin embargo, si el conjunto de datos es divisible manteniendo un equilibrio interno, el proceso general es intuitivo y bastante rápido en sus predicciones. Además, los árboles de decisión pueden funcionar eficientemente con conjuntos de datos no normalizados, ya que su estructura interna no se ve afectada por los valores asumidos por cada característica.” (Bonaccorso, 2017). Por lo tanto, son bastante dinámicos, adaptables a variadas situaciones de datos y problemas.

Sin embargo, sin las correctas limitaciones, un árbol de decisión puede generar sobreajuste en algunos casos, lo que lo haría incapaz de generalizar de forma adecuada. Al respecto Bonaccorso nos dice “sin las limitaciones adecuadas, un árbol de decisión podría crecer hasta que solo una muestra (o un número muy bajo) esté presente en cada nodo. Esta situación provoca un sobreajuste del modelo y el árbol se vuelve incapaz de generalizar correctamente. El uso de un conjunto de pruebas consistente o la validación cruzada puede ayudar a evitar este problema” (Bonaccorso, 2017).



Figura

Ejemplo de Decision Tree

Tomado de “Ejemplo de Decision Tree” (s. f.), www.mastersindatascience.org, <https://www.mastersindatascience.org/wp-content/uploads/sites/54/2022/05/tree-graphic.jpg>

**4.3.2. *Random Forest***:

El modelo Random Forest es un conjunto de árbol de decisión hechos a partir de muestras de forma aleatoria con una política diferente para dividir un nodo. En vez de buscar la mejor opción, en este modelo se utiliza un subconjunto aleatorio de características para cada árbol, buscando el umbral que mejor separe los datos, según Bonaccorso (2017). Esto da como resultado que habrán muchos árboles entrenados de forma más débil y cada uno de ellos va a dar una predicción diferente.

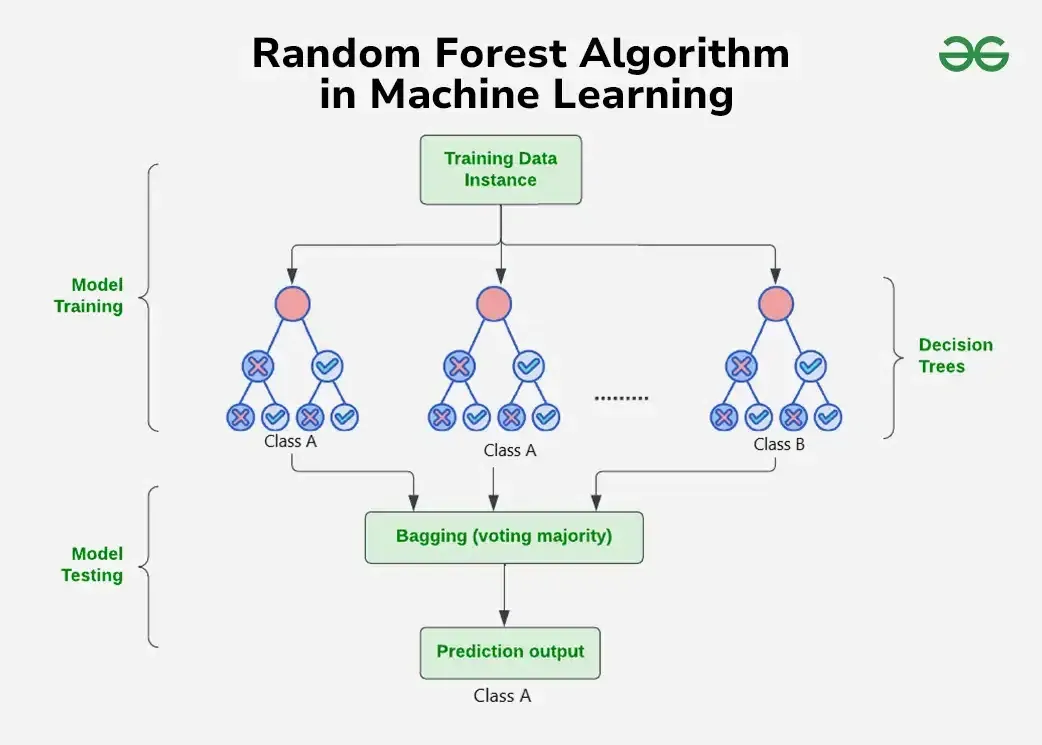
Según Bonaccorso hay dos manera de interpretar los resultados, “el enfoque más común se basa en una votación mayoritaria (la clase más votada se considerará correcta). Sin embargo, scikit-learn implementa un algoritmo basado en el promedio de los resultados, que produce predicciones muy precisas.” (Bonaccorso, 2017).

Entre los principales beneficios de este modelo está, según IBM, “Menor riesgo de sobreajuste: Los árboles de decisión corren el riesgo de presentar sobreajustes, ya que tienden a ajustar estrechamente a todas las muestras dentro de los datos de entrenamiento. Sin embargo, cuando hay una gran cantidad de árboles de decisión en un bosque aleatorio, el clasificador no sobreajustará el modelo, ya que el promedio de árboles no correlacionados reduce la varianza general y el error de predicción.” (IBM, 2025).

Sin embargo, uno de sus principales desafíos es que es un proceso que toma su tiempo. “Dado que los algoritmos de bosque aleatorio pueden manejar grandes conjuntos de datos, pueden proporcionar predicciones más precisas, pero pueden ser lentos para procesar los datos, ya que están calculando datos para cada árbol de decisión individual.” (IBM, 2025). Por lo tanto, es un aspecto importante a considerar a la hora de tomar decisiones sobre rendimiento.

Justifica por qué es una elección popular para problemas de clasificación como el tuyo.

Este es el principal modelo que se ocupará en el presente trabajo, y ¿por qué presenta ventajes considerables para nuestro problema de abandono de clientes? Entre las principales aplicaciones que tiene este modelo tiene que ver con finanzas, ya que “reduce el tiempo dedicado a la gestión de los datos y tareas de procesamiento previo. Se puede utilizar para evaluar a los clientes con alto riesgo crediticio, para detectar fraudes y problemas de opciones de precios.” (IBM, 2025).



Figura

Ejemplo de Random Forest

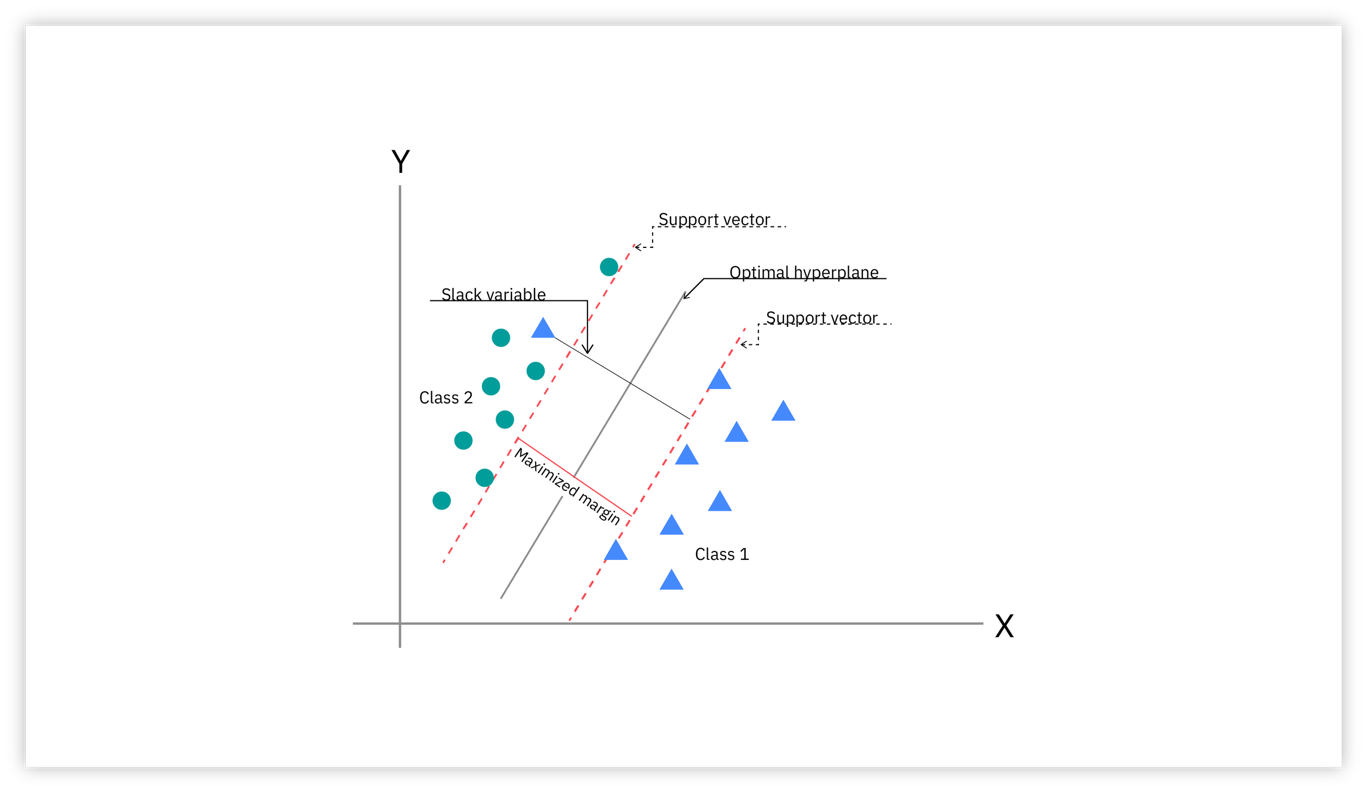
Tomado de “Random Forest Algorithm in Machine Learning. (s. f.). https://www.geeksforgeeks.org/. <https://media.geeksforgeeks.org/wp-content/uploads/20250522115823647286/Random_Forest_Algorithm.webp>

**4.3.3. Máquinas de Vectores de Soporte (*Support Vector Machines* - SVM)**:

Esta es una técnica de clasificación y regresión. Según IBM, esta “aprovecha al máximo la precisión de las predicciones de un modelo sin ajustar excesivamente los datos de entrenamiento. SVM es ideal para analizar datos con un gran número de campos de predictores (por ejemplo, miles).” (*Acerca de SVM, IBM*, 2021).

Estos algoritmos pueden funcionar en distintos contextos, sean lineales o no, lo que los hace dinámicos. Al respecto Bonaccorso nos dice “Estas máquinas pueden funcionar tanto en escenarios lineales como no lineales, lo que permite un alto rendimiento en diversos contextos. Junto con las redes neuronales, las máquinas de vectores de soporte probablemente representan la mejor opción para muchas tareas donde no es fácil encontrar un buen hiperplano de separación.” (Bonaccorso, 2017).

También, se puede aplicar a diversas áreas muy demandadas actualmente, con implementación continua. “SVM tiene aplicaciones en multitud de disciplinas, incluyendo la gestión de relaciones con los clientes (CRM), el reconocimiento facial y de otras imágenes, bioinformática, extracción de conceptos de minería de texto, detección de intrusiones, predicción de estructura de proteínas y reconocimiento de la voz.” (*Acerca de SVM, IBM*, 2021).

****Figura

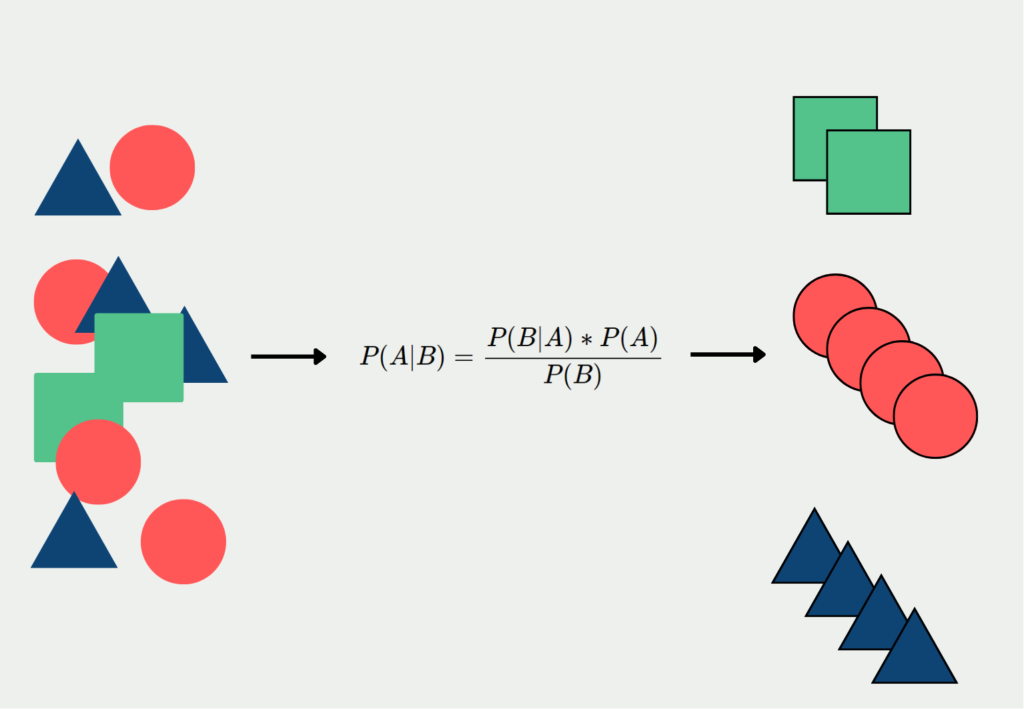
Ejemplo de SVM

Tomado de “What are support vector machines“ (SVMs)? (2023, 27 diciembre). https://www.ibm.com/. https://www.ibm.com/think/topics/support-vector-machine

**4.3.4. *Naive Bayes***:

Los algoritmos de Naive Bayes son una familia de clasificadores de gran potencia y sencillos de entrenar, que generan la probabilidad de un resultado mediante el teorema de Bayes, según Bonaccorso (2017). También se considera que tiene un enfoque simple, pero muy eficaz. “Naive Bayes son clasificadores multipropósito y su aplicación es fácil en diversos contextos; sin embargo, su rendimiento es particularmente bueno en todas aquellas situaciones donde la probabilidad de una clase está determinada por las probabilidades de algunos factores causales.” (Bonaccorso, 2017).

Dentro de los usos que se le da está el procesamiento de lenguaje natural, cómo se menciona a continuación “Un buen ejemplo es el procesamiento del lenguaje natural, donde un texto puede considerarse una instancia particular de un diccionario y las frecuencias relativas de todos los términos proporcionan suficiente información para inferir una clase a la que pertenece” (Bonaccorso, 2017).



Figura

Simple Representation of the Naive Bayes Classification

Tomado de “Simple Representation of the Naive Bayes Classification.” (2022, 30 abril). https://databasecamp.de/. https://databasecamp.de/wp-content/uploads/naive-bayes-overview-1024x709.png

**4.4. Justificación de la Selección de Modelos**

A continuación, se darán las principales razón del porqué se escogieron estos modelos por sobre otros y cuáles son los modelos populares que se decidió no utilizar.

**4.4.1. Criterios de Selección**:

Como hemos visto, todos los modelos anteriormente mencionados presentan ciertas ventajas y desventajas. Sin embargo decidimos trabajar con ellos porque consideramos que son alternativas idóneas a la hora de trabajar con nuestro conjunto de datos, el cual posee ciertas características en la que creemos, con estos modelos, podremos sacar el provecho que necesitamos.

En primer lugar se decidió utilizar estos modelos por su eficacia y también al mismo tiempo simplicidad. Estos modelos son conocidos por un adecuado rendimiento en lo que respecta a clasificación de datos tabulares, como el conjunto de datos de churn bancario encontrado en Kaggle que utilizaremos más adelante en el desarrollo de nuestra solución.

También se decidió utilizar estos modelos por su interpretabilidad. Por ejemplo, el agloritmo de Decision Tree y Random Forest nos brindan la posibilidad de detectar la importancia de ciertas características del data set, lo cual es de suma importancia a la hora de identificar cuáles son las variables que estarían influyendo en un posible abandono de clientes del banco ficticio que estamos analizando.

Por último, consideramos que el nivel de rendimiento y complejidad que poseen los modelos anteriormente mencionados ofrecen un equilibrio adecuado entre la capacidad que tienen de predecir y la exigencia de recursos que puede significar.

**4.4.2. Modelos No Utilizados y por Qué**:

Sabemos que hay muchos más modelos que se utilizan para realizar machine learning, tales como **K-Means**, o **Redes Neuronales**, **XGBoost**, o **LightGBM**. Sin embargo, en el caso puntual de K-Means, hemos decidido no utilizarlo principalmente porque se usa para casos de aprendizaje no supervisado, en nuestro caso es un aprendizaje supervisado. Pero también hemos decidido no utilizarlos el resto de modelos por múltiples razones que detallaremos a continuación.

Sabemos que, si bien, son modelos altamente avanzados los anteriormente mencionados, requieren de mayor tiempo de entrenamiento y son más complejos para lo que necesitamos en el presente diseño o implementación del modelo predictivo que requerimos. Por ejemplo, las **redes neuronales** necesitan una gran cantidad de datos y tiene una alta exigencia de procesamiento para poder realizar la etapa de entrenamiento, lo cual no lo consideramos necesarios para un conjunto de datos de un tamaño medio como el que se utilizará. Su uso generalmente se ve más en situaciones como reconocimiento de imágenes, reconocimiento de voz, o diagnósticos médicos, lo cual no es nuestro caso. “Aunque las redes neuronales son potentes, no son una solución única. Su fuerza reside en el manejo de tareas complejas que implican grandes conjuntos de datos y requieren el reconocimiento de patrones o capacidades predictivas. Sin embargo, para tareas más sencillas o problemas en los que los datos son limitados, los algoritmos tradicionales pueden ser más adecuados.” (Ali Awan, 2025).

**4.5. Métricas de Evaluación para Problemas de Clasificación**

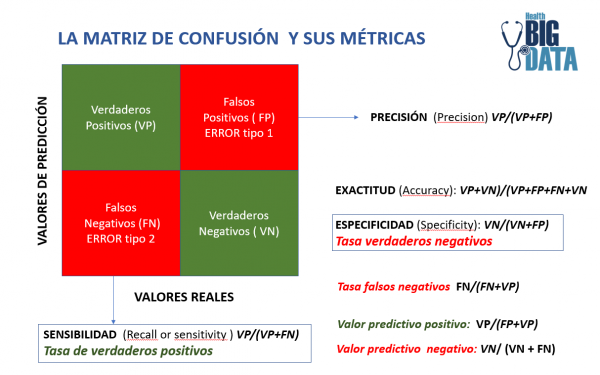
A continuación explicaremos en detalle el cómo evaluaremos nuestros modelos, en base a qué métricas, cuáles hemos escogido y porqué.

**4.5.1. Matriz de Confusión**:

La matriz de confusión es una herramienta muy utilizada para evaluar en lo que respecta al análisis predictivo en Machine Learning. Es muy utilizada en aprendizaje automático cuando se trata de tareas de clasificación. Esta también se considera como un tabla con el conteo de predicciones, al respecto Karimi nos dice “También podemos decir que la matriz de confusión es una tabla resumida del número de predicciones correctas e incorrectas generadas por un clasificador (o un modelo de clasificación) para tareas de clasificación binaria.” (Karimi, 2021).

Esta es una matriz N x X que se utiliza para evaluar el rendimiento del modelo de clasificación. “Al visualizar la matriz de confusión, se puede determinar la precisión del modelo observando los valores diagonales para medir el número de clasificaciones precisas.” (Karimi, 2021).

Para entender mejor esto utilizaremos una imagen en la que podremos observar cómo se compone.



Figura

MATRIZ DE CONFUSIÓN BINARIA

Tomado de “Barrios Arce, J. B. (2019, 26 julio). MATRIZ DE CONFUSIÓN BINARIA. https://www.juanbarrios.com.

“La matriz de confusión tiene la forma de una matriz cuadrada donde la columna representa los valores reales y la fila el valor predicho del modelo, y viceversa.” (Karimi, 2021).

Podemos observar cuatro principales componentes:

**Verdaderos Positivos (VP):** El valor era positivo y el modelo lo predijo como positivo.

**Verdaderos Negativos (VN):** El valor era negativo y el modelo lo predijo como negativo.

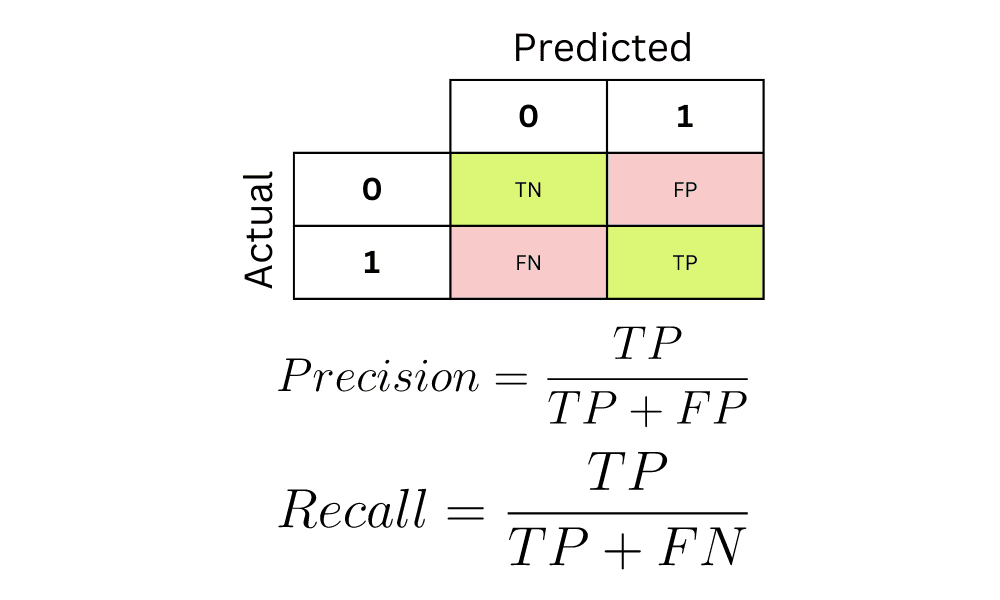
**Falsos Positivos (FP):** El valor era negativo pero el modelo lo predijo como positivo. También se le dice error tipo 1.

**Falsos Negativos (FN):** El valorera positivo pero el modelo lo predijo como negativo. También se le dice error tipo 2.

**4.5.2. Métricas Derivadas de la Matriz de Confusión**:

A continuación, definiremos lo que son las métricas derivadas de la matriz de confusión, tales como:

**Precision**: Esta nos indica cuántos casos predichos correctamente resultaron ser positivos. Esto nos ayudará a entender si nuestro modelo es confiable. (Karimi, 2021). La precisión es una métrica útil en los casos en que los falsos positivos son una preocupación mayor que los falsos negativos.

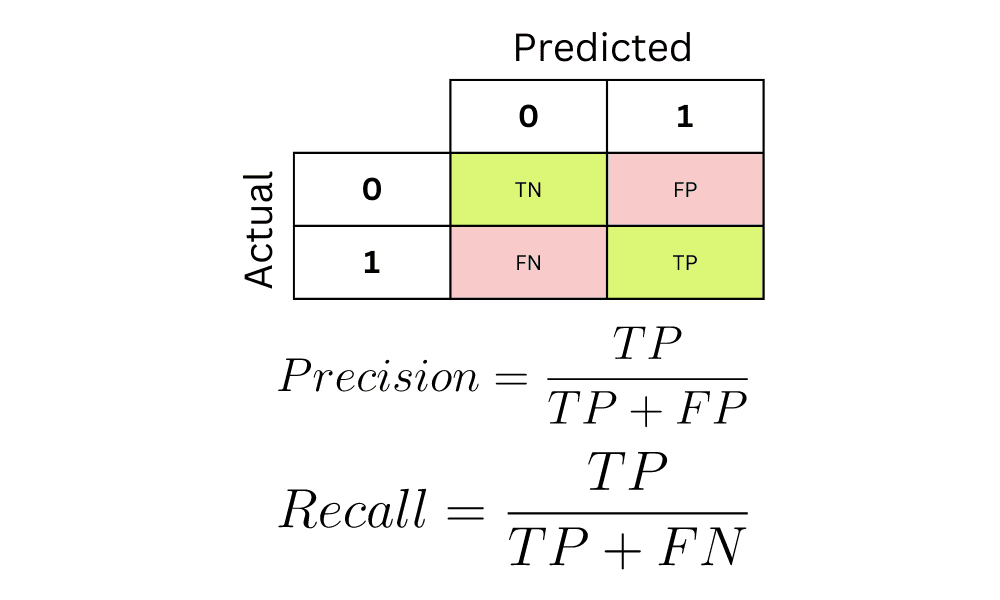


Figura

Métrica Precision

Fuente: Elaboración propia

**Recall:** Esta nos indica cuántos casos positivos reales pudimos predecir correctamente con nuestro modelo. La recuperación es una métrica útil en casos donde los falsos negativos superan a los falsos positivos. (Karimi, 2021). Esta es de vital importancia en nuestro caso ya que permitirá evaluar el rendimiento de nuestro de modelo para predecir los clientes que se irán.

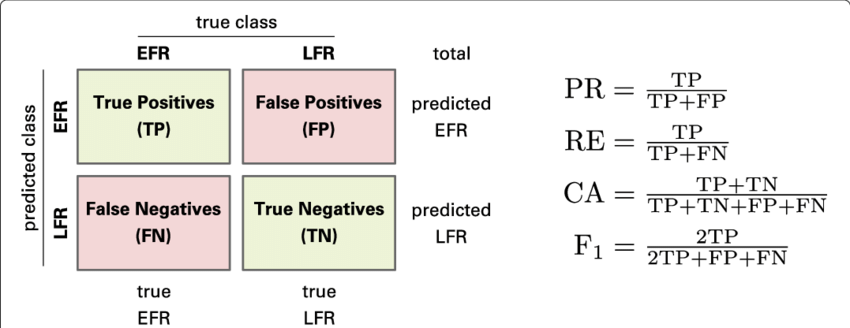


Figura

Métrica Recall

Fuente: Elaboración propia

**F1-Score:** Cuando intentamos aumentar la precisión del modelo, el recall disminuye y viceversa. La puntuación F1 es una media armónica de la precisión y el recall, por lo que ofrece una idea combinada de estas dos métricas. Alcanza su máximo cuando la precisión es igual a la recuperación. (Karimi, 2021).



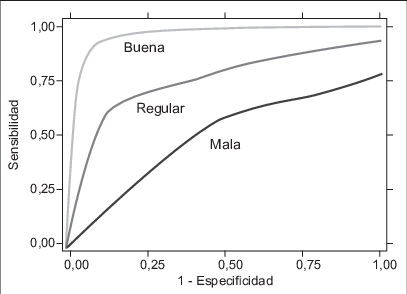
Figura

Métrica Recall

Fuente: Elaboración propia

**4.5.3. Curva ROC y AUC**:

**Curva ROC**: La llamada curva ROC, es utilizada para evaluar la precisión de un determinado método de clasificación. Esto es, dado un clasificador binario, el estudio de su curva ROC indica en qué medida se pueden clasificar a los individuos de una población en dos grupos. (Castro Capelo, 2022)



Figura

Esquema explicativo de distintas posibilidades de curvas ROC.

Tomado de https://www.researchgate.net/figure/Figura-5-Esquema-explicativo-de-distintas-posibilidades-de-curvas-ROC\_fig5\_224954042

**Área Bajo la Curva (AUC)**: representa la probabilidad de que el modelo, si se le da un ejemplo positivo y negativo elegido al azar, clasifique el positivo más alto que el negativo. Para un clasificador binario, un modelo que funciona exactamente igual que las conjeturas aleatorias o las volteretas de moneda tiene un ROC que es una línea diagonal de (0,0) a (1,1). El AUC es de 0.5, lo que representa una probabilidad del 50% de clasificar correctamente un ejemplo positivo y negativo aleatorio. (*Clasificación: ROC y AUC*, s. f.).

**4.6. Conclusión del Marco Teórico**

Todo lo anteriormente expuesto nos servirá para poder desarrollar nuestra solución sin mayores inconvenientes, contando con las herramientas necesarias para poder, ya sea comenzar con un modelo predictivo, como su posterior elección, entrenamiento, y posterior evaluación para poder hacer un análisis de sus resultados. Es altamente relevante contar con estos conocimientos para poder implementar un modelo predictivo relevante y funcional, ya que de no contar con la información adecuada se podrían tomar decisiones que harían de este propósito un proceso con resultados débiles. Por lo tanto, todo lo visto nos servirá de aquí en adelante para desarrollar de forma más adecuada nuestra solución.

# Metodología de Trabajo

A continuación, vamos a establecer nuestra metodología de trabajo, considerando el enfoque de nuestra investigación, su diseño, el proceso o la metodología de la data science, entre otros detalles, para poder comprender nuestro actual estudio.

5.1. Enfoque de la Investigación

Enfoque Cuantitativo:

Este trabajo tendrá  un enfoque cuantitativo, basándonos principalmente en lo establecido como metodología de la investigación según el autor Hernández Sampieri. Lo primero es que según el autor, él entiende la investigación como “un conjunto de procesos sistemáticos, críticos y empíricos que se aplican al estudio de un fenómeno o problema.” (Sampieri et al., 2010). Por lo tanto, nuestro presente trabajo tendrá un conjunto de procesos sistemáticos empíricos los cuales iremos detallando a continuación, ya que son estos los que nos darán el esquema para lograr esquematizar nuestros esfuerzos por el alcanzar el objetivo general y los específicos. Nuestra idea es poder medir estos fenómenos como son la utilización de datos para posteriormente implementar modelos matemáticos con el fin de generar predicciones, las cuales tendrán que ser también evaluadas.

En el caso particular de la investigación cuantitativa, o el enfoque cuantitativo es un conjunto de procesos, es secuencial y probatorio Sampieri et al. (2010). También se considera la utilización de recolección de datos para una comprobación de hipótesis, todo esto basado en medición numérica y análisis estadístico. Al respecto de esto, el autor nos dice “De las preguntas se establecen hipótesis y determinan variables; se traza un plan para probarlas (diseño); se miden las variables en un determinado contexto; se analizan las mediciones obtenidas utilizando métodos estadísticos, y se extrae una serie de conclusiones” (Sampieri et al., 2010). Entonces, como mencionábamos, en este enfoque hay elementos cruciales como el establecimiento de hipótesis, determinación de variables, análisis de estas mediciones y posterior a eso se generan conclusiones, que es lo que precisamente haremos durante el desarrollo de nuestra solución junto con la obtención de los resultados.

Entre las características que podemos también considerar para esta enfoque están algunas tales como la necesidad de medir y estimar magnitudes de los fenómenos, plantear un problema delimitado y concreto (en este caso el churn bancario, por qué abandonan el banco y cuáles son sus principales variables). Este sigue un patrón determinado, “Los estudios cuantitativos siguen un patrón predecible y estructurado (el proceso) y se debe tener presente que las decisiones críticas sobre el método se toman antes de recolectar los datos” (Sampieri et al., 2010).

5.2. Diseño de la Investigación

A continuación, describiremos cuál fue la forma en que estructuramos nuestro estudio o presente trabajo para poder generar el posterior desarrollo de una solución.

Diseño no experimental - Transversal:

En este caso consideramos que nuestro diseño de la investigación es de tipo no experimental – transversal, dado que tenemos por fin evaluar un fenómeno en un contexto o punto dado del tiempo, el cual se nos fue entregado por los datos presentes en el dataset que rescatamos de la página web líder en conjunto de datos públicos “Kaggle”. Al respecto de este tipo de investigación Sampieri nos dice “la investigación se centra en: a) Analizar cuál es el nivel o modalidad de una o diversas variables en un momento dado. b) Evaluar una situación, comunidad, evento, fenómeno o contexto en un punto del tiempo. c) Determinar o ubicar cuál es la relación entre un conjunto de variables en un momento.” (Sampieri et al., 2010). Que es justamente lo que haremos en este trabajo, determinar cuál es la relación entre las variables que ya nos vienen dadas por el dataset, estas solos las observaré y analizaré.

También definiremos lo que consideramos como un diseño transversal, ya que esta recolectan datos en un solo momento, en un tiempo único (Liu, 2008 y Tucker, 2004). Su propósito es describir variables y analizar su incidencia e interrelación en un momento dado. Es como “tomar una fotografía” de algo que sucede.” (Sampieri et al., 2010). De hecho, nosotros trabajaremos con un dataset establecido hace varios años, el cual no se ha vuelto actualizar, es un momento determinado en un banco establecido en Europa, el cual presta servicio en algunos países. Por lo tanto, son datos específicos y estáticos los que utilizaremos, esto con el fin de poder establecer relaciones entre las variables para ese punto específico que es considerando su variable más importante, si clientes desertaron o no del banco.

5.3. Proceso de la Metodología de Data Science

En esta sección vamos a detallar el flujo completo del trabajo realizado en nuestro proyecto, desde la obtención de los datos utilizados hasta los posteriores entrenamientos de los modelos.

5.3.1. Obtención y Descripción de los Datos

A continuación hablaremos sobre el origen de nuestros datos. Los datos los obtuvimos, como habíamos mencionado anteriormente, de la página web Kaggle. A continuación dejamos el link específico desde donde se obtuvieron estos datos: <https://www.kaggle.com/datasets/radheshyamkollipara/bank-customer-churn/data>.

En la página web hay una sección llamada “About this file” el cual menciona lo siguiente: Todo banco quiere retener a sus clientes para sostener su negocio, y así lo hace este Banco Multinacional Anónimo. A continuación se presentan los datos de los clientes con cuentas en el Banco Multinacional Anónimo, y el objetivo de estos datos es predecir la pérdida de clientes (Customer Churn). También se menciona como fuente que proviene de datos gubernamentales.

Ahora detallaremos, a través de una tabla, un resumen de las principales variables del dataset junto con su variable objetivo (Exited), que significa que abandonó el banco.

|  |  |  |
| --- | --- | --- |
| Campo | Descripción breve | Relevancia para churn |
| RowNumber | Nº de fila (índice técnico) | Excluir (no aporta) |
| CustomerId | ID aleatorio del cliente | Excluir (no aporta) |
| Surname | Apellido del cliente | Excluir (no aporta) |
| CreditScore | Puntaje crediticio | Importante: mayor puntaje → menor churn |
| Geography | Ubicación del cliente | Relevante: puede influir |
| Gender | Género del cliente | Explorar: efecto incierto |
| Age | Edad del cliente | Relevante: mayores → menor churn |
| Tenure | Años como cliente | Relevante: más años → más lealtad |
| Balance | Saldo de cuenta | Importante: saldos altos → menor churn |
| NumOfProducts | Nº de productos contratados | Relevante: más productos → menor churn |
| HasCrCard | ¿Tiene tarjeta de crédito? | Relevante: tener tarjeta → menor churn |
| IsActiveMember | Actividad del cliente | Importante: activos → menor churn |
| EstimatedSalary | Ingreso estimado | Relevante: salarios bajos → mayor churn |
| Exited | ¿Dejó el banco? | Objetivo (variable target) |
| Complain | ¿Registró reclamo? | Relevante: reclamos → mayor churn |
| Satisfaction Score | Satisfacción (post-reclamo) | Relevante: menor satisfacción → mayor churn |
| Card Type | Tipo de tarjeta | Explorar: útil para segmentar |
| Points Earned | Puntos acumulados por uso | Relevante: más puntos/engagement → menor churn |

5.3.2. Preprocesamiento y Limpieza de los Datos

Procederemos a detallar el paso a paso para poder realizar este desarrollo de solución de Data Science.

También debemos señalar que esto se realizará en Colab, o Google Colaboratory, la cual es una herramienta gratuita y basada en la nube que permite escribir y ejecutar código Python directamente en un navegador web, sin necesidad de configurar nada en el computador. Utiliza cuadernos de tipo Jupyter para organizar código, texto y visualizaciones, y ofrece acceso gratuito a recursos de cómputo de alto rendimiento como GPUs y TPUs, lo que lo hace ideal para el aprendizaje automático, la ciencia de datos y la educación.

Primeramente se realiza la importación de librerías necesarias para el trabajo que se llevará a cabo, la cual nos permitirá ejecutar códigos con distintos enfoques, ya sea para poder transformar en dataframe, realizar cálculos, implementar modelos y/o realizar visualizaciones. Entre las importadas podemos mencionar pandas, numpy, seaborn, entre otras.

Luego, procedemos a cargar el dataset, que habíamos descargado de Kaggle. Luego de cargarlo en nuestro cuaderno, solicitamos a través del código df.head, las 5 primeras filas para poder cerciorarnos que los datos fueron cargados correctamente y que son los que necesitamos para nuestro trabajo.

Realizaremos también una exploración general de los datos (EDA) para poder observar si contamos con datos nulos, repetidos o si hay algún problema con el tipo de datos el cual podría provocar problemas en nuestro posterior entrenamiento de modelos o algoritmos. Esto se puede realizar a través de ciertos códigos que nos permiten ver resúmenes de los datos y también estadísticas al respecto. También levantaremos una visualización, sobre todo para observar la distribución de nuestra variable objetivo que es la que más nos importa.

A continuación seguiremos con lo que ya es el preprocesamiento de los datos, por lo cual se eliminarán columnas o variables irrelevantes como ‘Complain’, o ‘RowNumber’. También acá entra una parte importante de nuestro trabajo para este dataset, el manejo de las variables categóricas, ya que estas pueden generar problemas en el entrenamiento de los modelos. Por lo que se procede a codificar las variables categóricas, como ‘Gender\_Male’ a través de una función integrada dentro de las librerías llamada One-Hot Encoding, la cual convierte estas variables nominales en formato numérico, para propiciar un mejor rendimiento del modelo y evitar el ‘ValueError’.

Posterior a eso, se separarán las características (X) y la variable objetivo (y). Para más adelante trabajar con todas las variables integradas a la vez en X, pero separadas de la variable ‘Exited’.

Luego, se efectuará la parte de normalizar los datos, también se le llama a esto como escalar las variables numéricas restantes. Se utilizará StandarScaler, que también es una función integrada dentro de las librerías. Esto es importante ya que permitirá que los algoritmos de machine learning puedan funcionar con datos que estén en valores similares, asegurando su buen rendimiento y minimizando los errores.

5.3.3. División de los Datos

Ahora, procederemos a ejecutar la división del conjunto de datos. Estos se dividirán en un subconjunto de datos de entrenamiento, el cuál será del 80% y otro subconjunto de datos de prueba (20%). La razón principal de la efectuación de este proceso es que con un subconjunto, el de entrenamiento, se ‘alimentan’ de alguna manera a los modelos, para que puedan aprender de los patrones presentes en los datos. Por otra parte, el otro subconjunto, el de prueba, se utiliza para poder validar el rendimiento predictivo que tendrá o tendrán los diferentes modelos utilizados.

Otro paso a tener en cuenta, si bien antes, consideramos efectuar una visualización de la variable objetivo para ver su distribución, esta se debe volver a realizar para poder considerar alguna medida de estabilización, ya que probablemente debe haber una distribución desigual, porque se deduce que en un banco la cantidad de personas que abandonan el banco en general debiesen ser mucho menos que los que se mantienen como clientes fidelizados. A esto se le llama como el manejo del desbalance de clases. Este es un punto importante ya que efectivamente hay un desbalance considerable entre las clases de la variable objetivo. Por lo que se debe utilizar una técnica de sobremuestreo a la clase minoritaria en el conjunto de entrenamiento. La técnica o función que será utilizada lleva por nombre SMOTE, la cual tiene por objetivo aumentar artificialmente la muestra de la clase minoritaria, en este caso los que no han desertado.

5.4. Desarrollo de Modelos y Herramientas

5.4.1. Algoritmos Utilizados

En lo que respecto a los modelos utilizados, o algoritmos, estos ya han sido mencionados anteriormente en el marco teórico, pero de igual forma los describiremos ya que son importantes para el proceso o nuestra metodología de trabajo.

En primer lugar, se utilizará Random Forest, este modelo es un conjunto de árbol de decisión hechos a partir de muestras de forma aleatoria con una política diferente para dividir un nodo. En vez de buscar la mejor opción, en este modelo se utiliza un subconjunto aleatorio de características para cada árbol, buscando el umbral que mejor separe los datos, según Bonaccorso (2017). En este caso, al igual que en los otros, se debe inicializar el modelo, realizar el entrenamiento y posteriormente generar una predicción con el conjunto de los datos de prueba.

En segundo lugar, se utilizará **Decision Tree**,este modelo consiste en una estructura que se basa en un proceso de decisión secuencial. Parte desde una raíz, evaluando una característica y se selecciona una de las dos ramas. Todo este proceso se repite hasta que finalmente se llega a la última hoja, que sería el objetivo de clasificación que se está buscando, según lo planteado por Bonaccorso (2017). En este caso, al igual que en los otros, se debe inicializar el modelo, realizar el entrenamiento y posteriormente generar una predicción con el conjunto de los datos de prueba.

En tercer lugar, se utilizará **Super Vector Machine**, esta es una técnica de clasificación y regresión. Según IBM, esta “aprovecha al máximo la precisión de las predicciones de un modelo sin ajustar excesivamente los datos de entrenamiento. SVM es ideal para analizar datos con un gran número de campos de predictores (por ejemplo, miles).” (*Acerca de SVM, IBM*, 2021). En este caso, al igual que en los otros, se debe inicializar el modelo, realizar el entrenamiento y posteriormente generar una predicción con el conjunto de los datos de prueba.

En último lugar, se utilizará **Naive Bayes**, “Naive Bayes son clasificadores multipropósito y su aplicación es fácil en diversos contextos; sin embargo, su rendimiento es particularmente bueno en todas aquellas situaciones donde la probabilidad de una clase está determinada por las probabilidades de algunos factores causales.” (Bonaccorso, 2017). En este caso, al igual que en los otros, se debe inicializar el modelo, realizar el entrenamiento y posteriormente generar una predicción con el conjunto de los datos de prueba.

5.4.2. Herramientas y Lenguaje de Programación

Ahora enumeraremos las principales herramientas utilizadas y el lenguaje de programación con el cuál se desarrollará todo este trabajo:

* Python
* Pandas (para el manejo de datos)
* Scikit-learn (para los modelos y métricas)
* Matplotlib
* Seaborn (para la visualización)
* Imbalanced-learn (para SMOTE)
* Collections (Counter).

Como entorno de desarrollo:

* Google Colab

5.4.3. Criterios de Evaluación

Procederemos ahora a mencionar cuáles serán nuestros criterios de evaluación, o nuestros test como se le llaman también en el área de ciencias de datos, con lo cual podremos medir si nuestros modelos están obteniendo niveles óptimos de resultados.

Las métricas que utilizaremos, son esenciales para problemas con datos desbalanceados como nos hemos encontrado en este caso, y estas son:

* Matriz de Confusión: es una herramienta muy utilizada para evaluar en lo que respecta al análisis predictivo en Machine Learning. Es muy utilizada en aprendizaje automático cuando se trata de tareas de clasificación. Esta también se considera como un tabla con el conteo de predicciones.
* Curva ROC: utilizada para evaluar la precisión de un determinado método de clasificación. Esto es, dado un clasificador binario, el estudio de su curva ROC indica en qué medida se pueden clasificar a los individuos de una población en dos grupos. (Castro Capelo, 2022).
* Precision: Esta nos indica cuántos casos predichos correctamente resultaron ser positivos. Esto nos ayudará a entender si nuestro modelo es confiable. (Karimi, 2021). La precisión es una métrica útil en los casos en que los falsos positivos son una preocupación mayor que los falsos negativos.
* Recall: Esta nos indica cuántos casos positivos reales pudimos predecir correctamente con nuestro modelo. La recuperación es una métrica útil en casos donde los falsos negativos superan a los falsos positivos. (Karimi, 2021).
* F1-Score: Cuando intentamos aumentar la precisión del modelo, el recall disminuye y viceversa. La puntuación F1 es una media armónica de la precisión y el recall, por lo que ofrece una idea combinada de estas dos métricas. Alcanza su máximo cuando la precisión es igual a la recuperación. (Karimi, 2021).

5.5. Resumen de la Metodología

Para resumir, el proceso metodológico escogido será de carácter cuantitativo, con un enfoque o tipo no experimental – transversal, ya que se estudiará la relación entre variables de datos obtenidos de un momento determinado. También procederemos a trabajar con una mtedología de ciencia de datos, la cual se caracteriza por obtener datos, cargarlos, explorarlos, preprocesarlos, limpiarlos y posteriormente generar la inicialización y entrenamiento de los modelos. Finalmente, finalizaremos con la evaluación o métricas de evaluación de los modelos para poder medir nuestros resultados. La aplicación rigurosa de esta metodología debiese asegurar que los resultados que obtengamos sean confiables, válidos para poder responder nuestras preguntas y en el futuro utilizarse para el sector financiero y su prevención de fuga de clientes.

# Desarrollo de la Solución

A continuación, se detallará el cómo se implementó la técnica de la metodología de investigación anteriormente propuesta, por lo que presentaremos el desarrollo completo de este trabajo con sus modelos de predicción de churn bancario. Hay que tener en consideración sí, que no colocaremos todo el cuaderno Colab desarrollado, ya que lo adjuntaremos en la presente tesis para su revisión, por lo que aquí solo colocaremos las principales secciones para el desarrollo de este trabajo.

Antes de adentrarnos en las fases principales, mencionaremos que, para poder realizar todo este proceso, primero importaremos todas las librerias que necesitaremos.

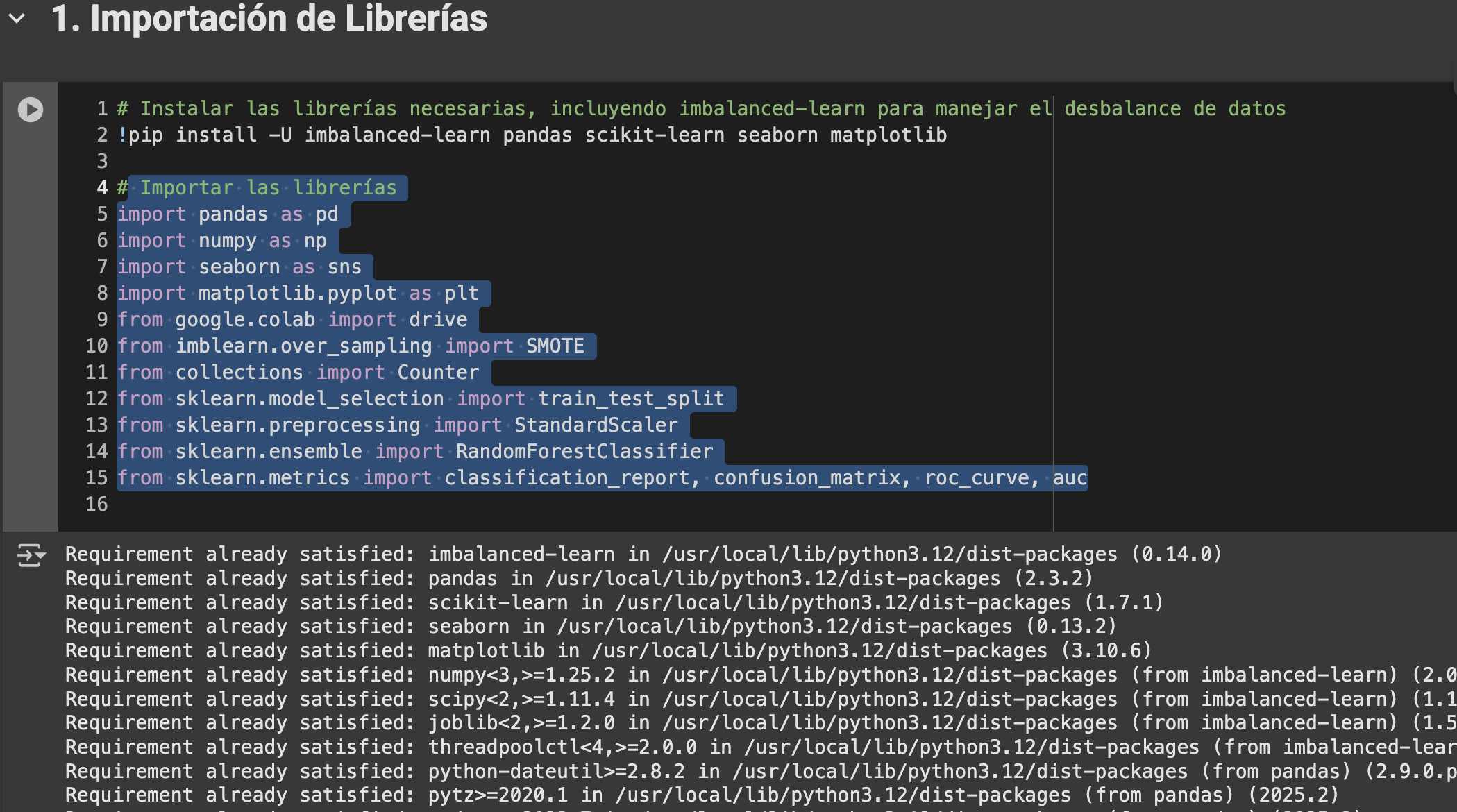


Imagen 1

Importación de librerías – Elaboración propia

Se puede observar que se han importado todas las librerías correctamente para poder utilizar las funciones que tienen integradas y que más adelante nos permitirán trabajar en nuestros datos sin problemas.

Posteriormente, cargaremos el dataset rescatado de Kaggle. Cargaremos el dataset en un DataFrame de Pandas y mostraremos las primeras filas con un df.head().

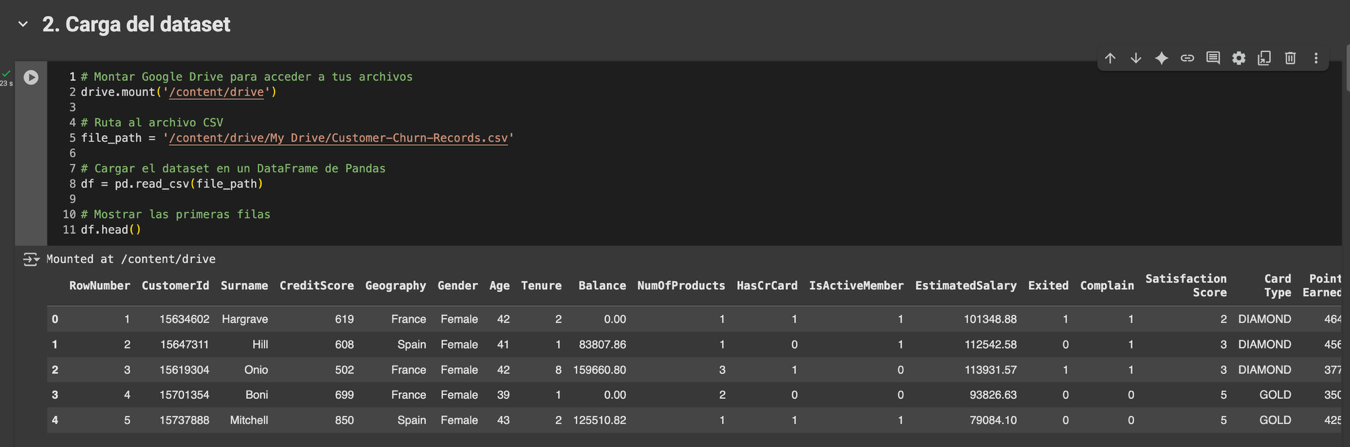


Imagen 2

Carga del dataset – Elaboración propia

Acá podemos observar las 5 primeras filas del dataset, donde vemos sus campos, o variables, con sus respectivos datos. De primera fuente nos damos cuenta que los países en lo que está presente el banco son Francia y España. Sin embargo, es información que por ahora no es tan relevante para nuestro estudio.

1. **6.1. Implementación del Preprocesamiento de Datos**
2. En esta sección narraremos la aplicación de las técnicas de exploración, limpieza y preparación de los datos que hicimos en nuestro cuaderno Colab. Por lo que iremos parte por parte.

Primero, hicimos una exploración de datos para saber en qué condición estaban estos datos, si los tipos de datos eran correctos y coherentes con la información que traían, o si habían valores nulos.



Imagen 3

Exploración de datos – Elaboración propia

Luego, hicimos un df.describe(), lo cual nos dio estadísticas generales al respecto de los datos, lo cual no colocaremos para no extender en demasía la explicación y considernado que el cuaderno Colab estará adjunto en un link, con el cual se puede revisar en detalle todo lo realizado. Y por último, en esta fase exploratoria, se hizo una visualización de gráficos de barra para la variable principal ‘exited’, la cual nos demostró, primeramente, que estaba desequilibrada.

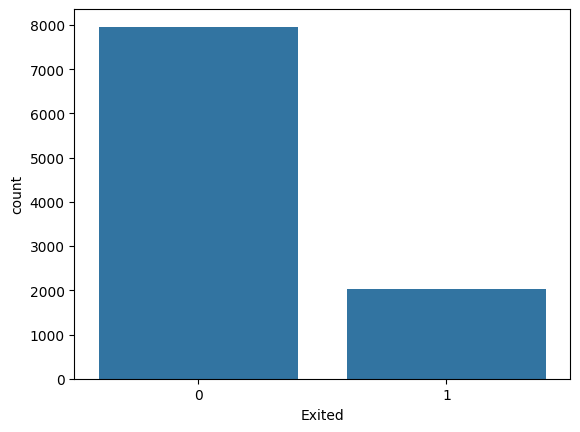


Imagen 4

Gráfico de barras de ‘exited’ – Elaboración propia

* 1. **6.1.1. Tratamiento de Variables Categóricas:** Explica que convertiste las variables de texto a un formato numérico que los algoritmos de *machine learning* pueden entender.

En este punto comenzamos ya un tratamiento o preprocesamiento más intensivo, ya que para poder contar con datos limpios y que funcionen bien en los modelos o algoritmos que utilizaremos después, debemos tomar ciertas decisiones técnicas.

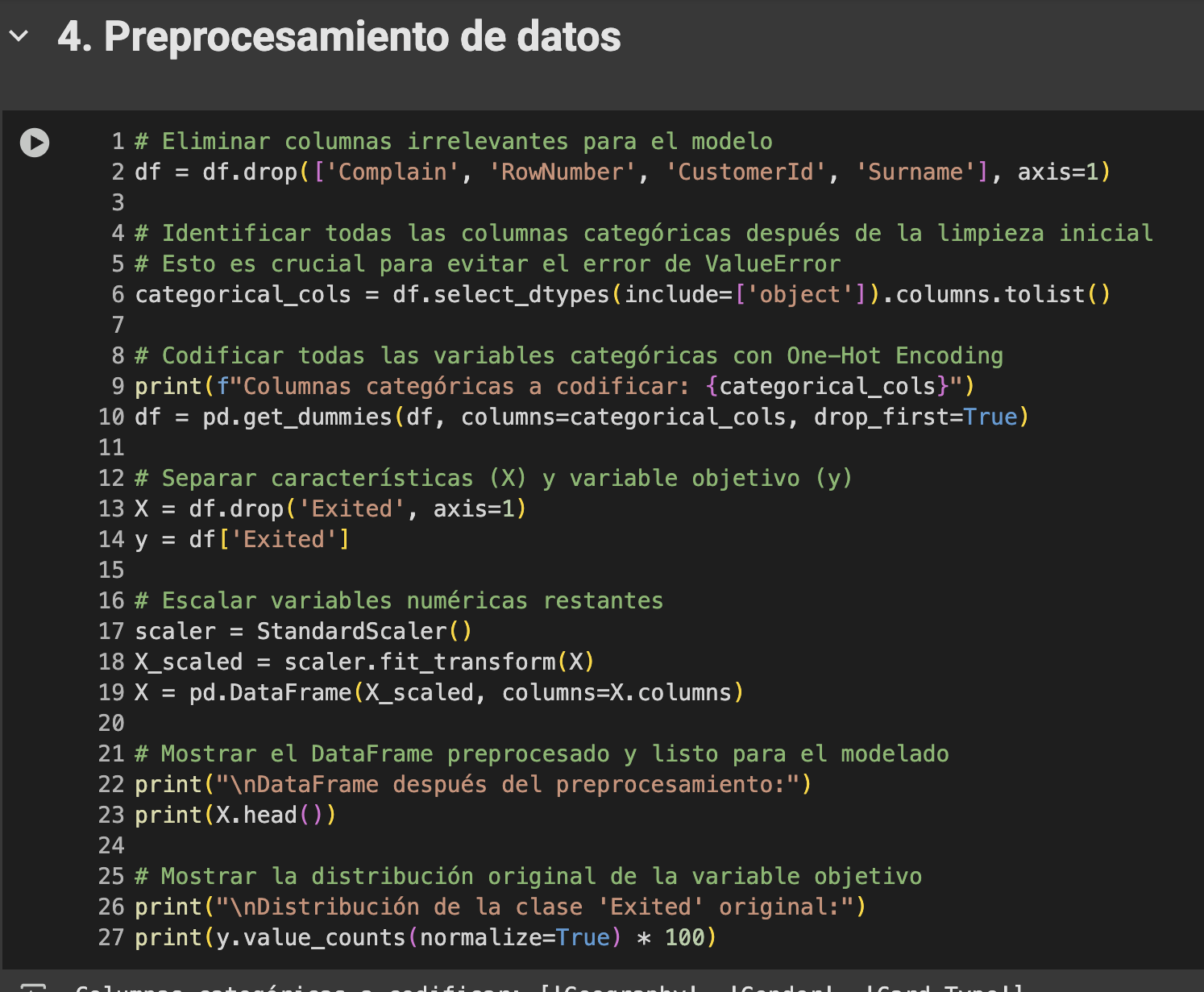


Imagen 5

Eliminación columnas – Elaboración propia

Una de las primeras decisiones es eliminar columnas que no necesitemos. En este caso eliminaremos la columna ‘complain’, ‘RowNumber’, ‘CustomerId’, y ‘Surname’. Ya que no aportan valor al modelo y pueden generar problemas de funcionamiento. También, se identificar todas las columnas categóricas después de la limpieza inicial, para evitar el error de ValueError.

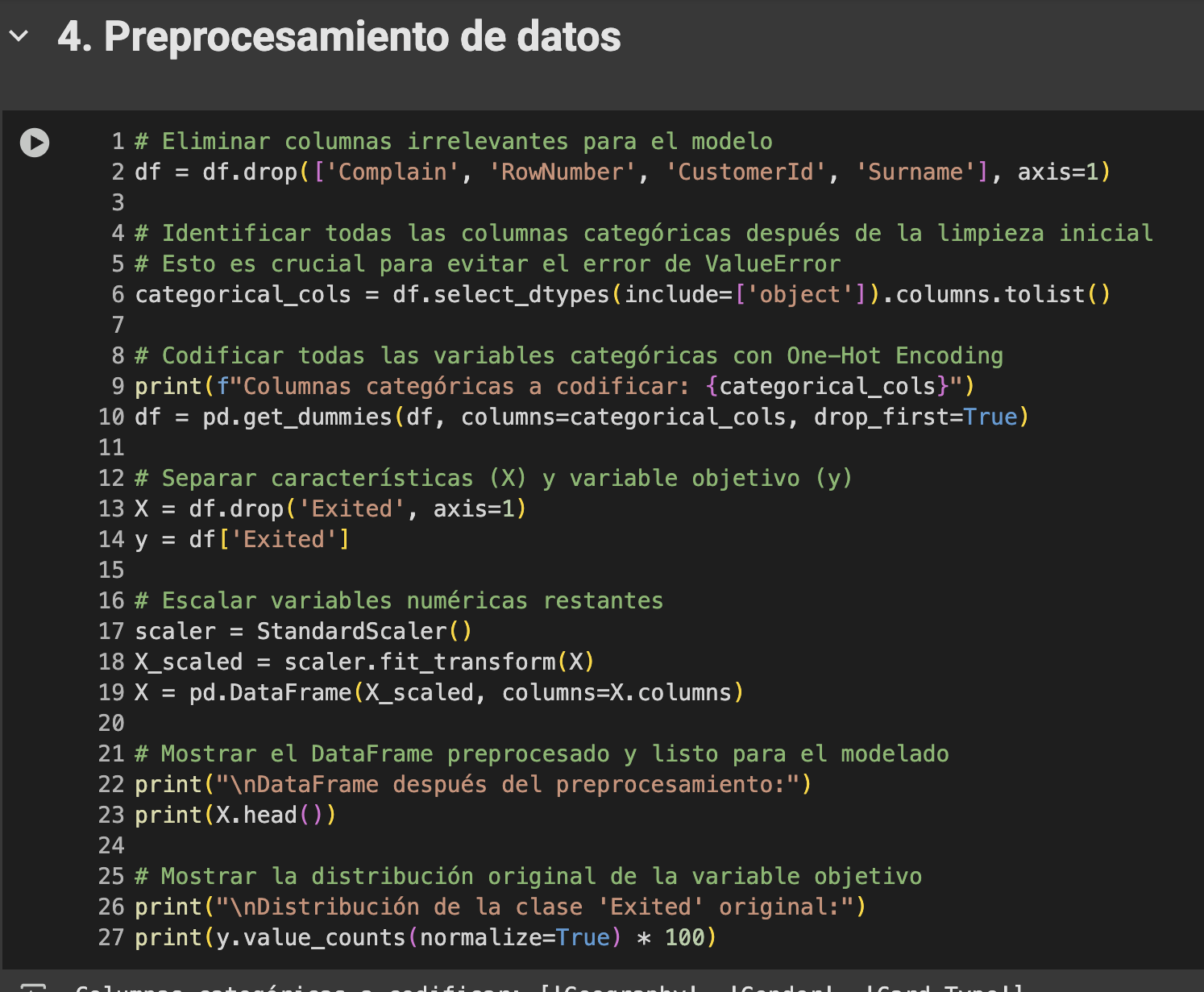


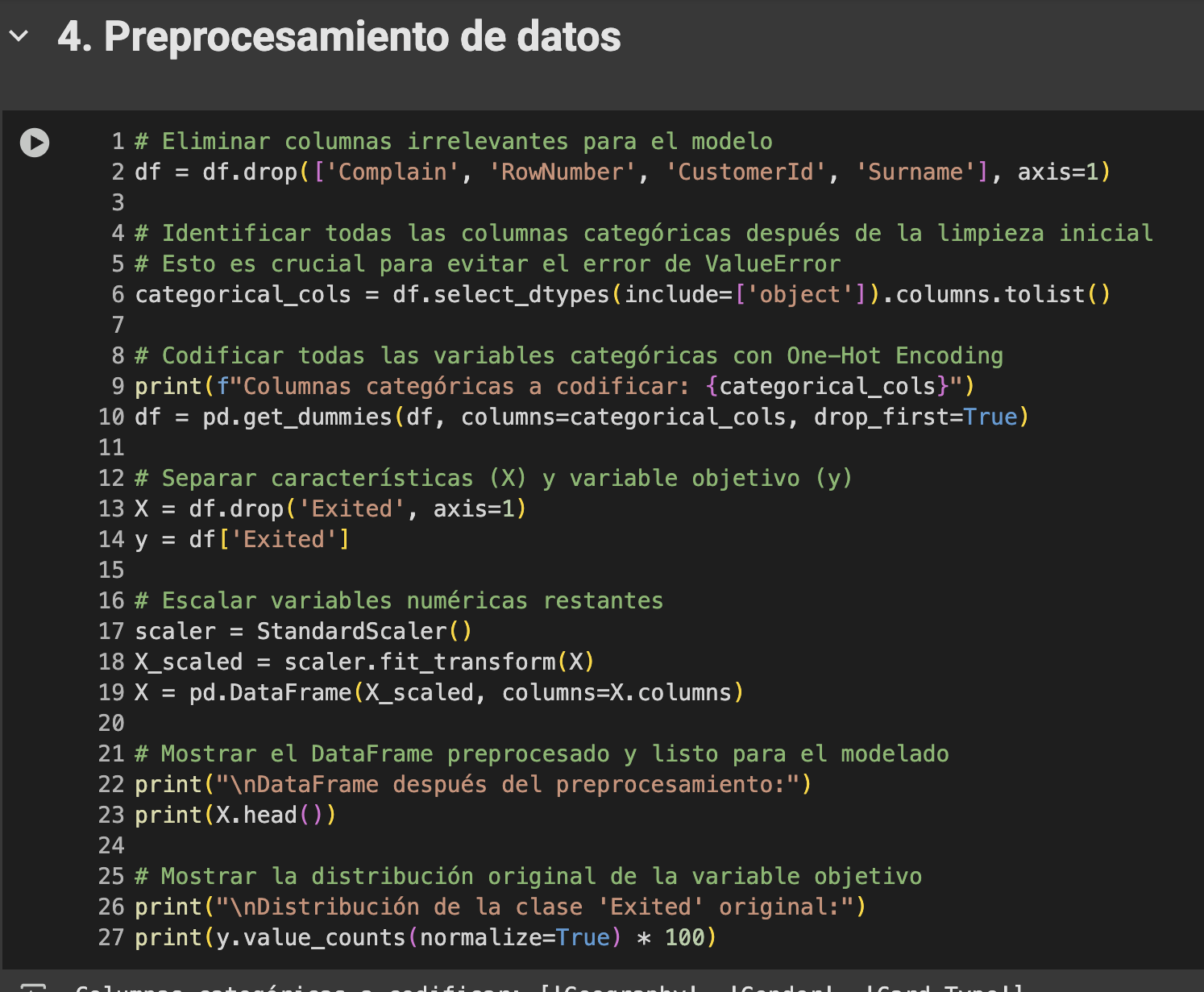
Imagen 6

One-Hot Encoding – Elaboración propia

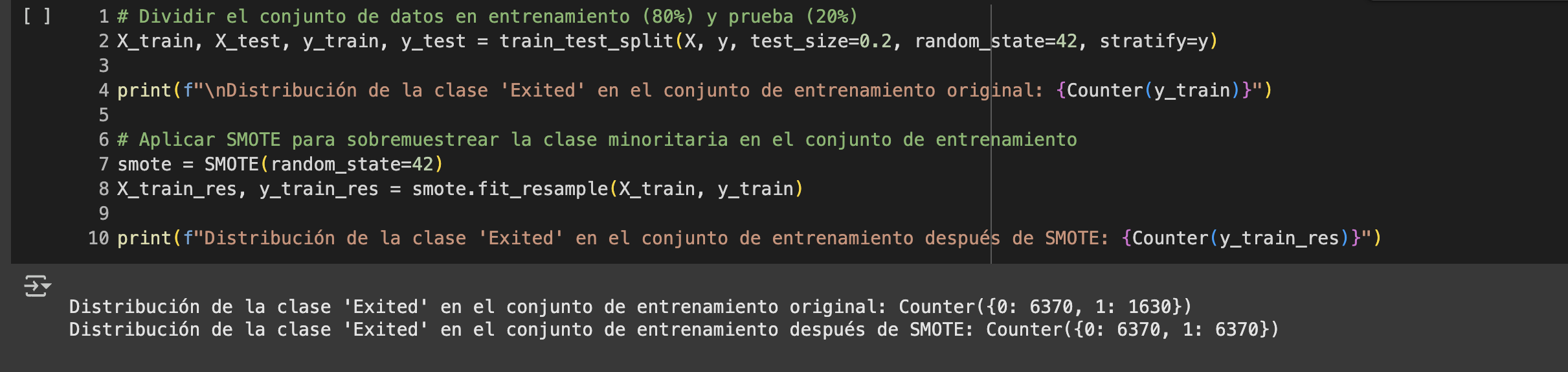
Después, se aplicó **One-Hot Encoding** a las variables como 'Geography' y 'Genre', que son columnas categóricas y buscamos trabajar solo con números para optimizar el proceso de ingreso de datos a los modelos que ocuparemos unos pasos más adelante.

* 1. **6.1.2. Normalización de Datos:**

1. Pasado todo este proceso, necesitamos estandarizar las variables númericas para que nuestros algoritmos no den importancia desproporcionada a variables que puedan contener valores extremos en comparación al resto de variables.

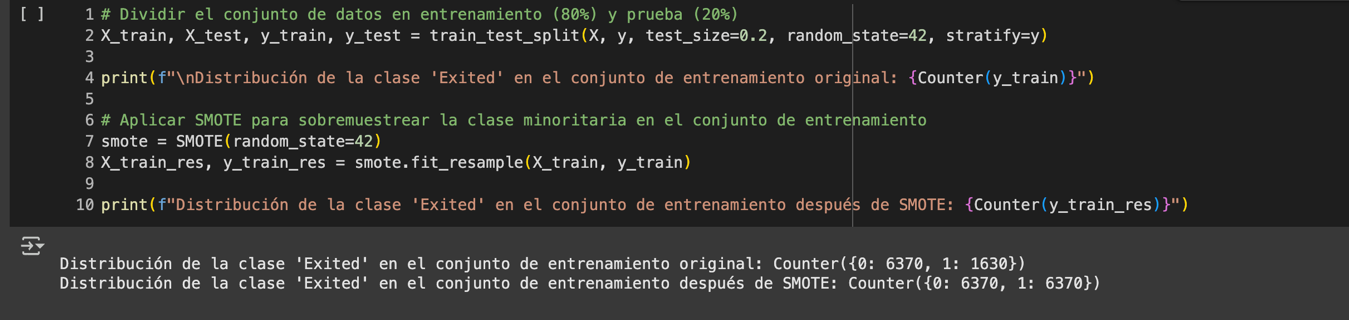


1. Imagen 7
2. Escalar variables – Elaboración propia
   1. Como se puede apreciar, se utilizó la función ‘scaler’. Hay que mencionar sí que, antes de este paso, se separaron el resto de las variables en un solo conjunto llamado (X), de la variable objetivo ‘exited’ la cual quedó en un conjunto titulado (y). Por eso es que la estandarización solo se hizo con (X), como se puede observar en la imagen.
3. **6.1.3. División del Conjunto de Datos**
   1. En este paso debemos realizar lo que es la división del conjunto de datos, donde un conjunto será para entrenamiento del modelo y otro para probar el modelo, o testearlo como también se suele decir.



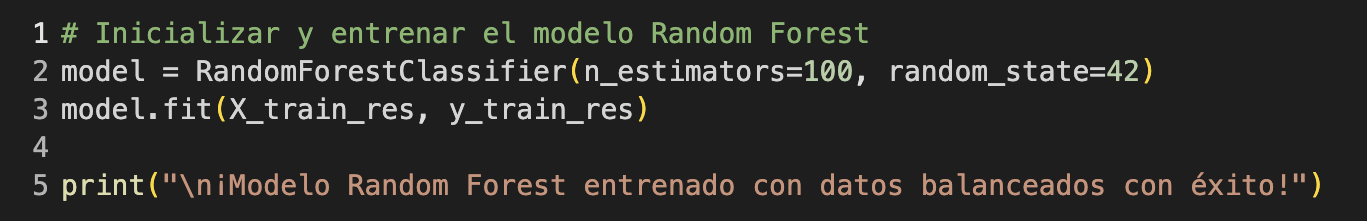
1. Imagen 8
2. División de conjunto de datos – Elaboración propia
   1. Podemos observar, a través de este fragmento, que se utilizó la funcion ‘train\_test\_split’ para poder llevar a cabo nuestra división. También es importante señalar que la proporción escogida de la división fue de 80% para el conjunto de entrenamiento y de 20% para el conjunto de prueba.
   2. **6.1.4. Manejo del Desbalance de Clases:**

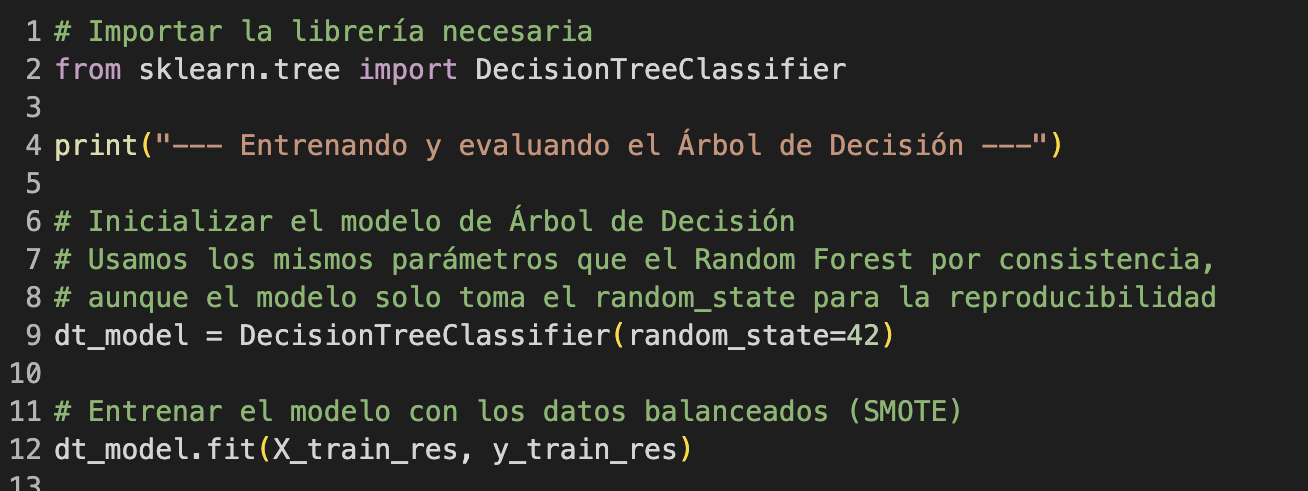
Anteriormente, nos habíamos fijado en que, según el gráfico, había un desbalance en la variable objetivo que es ‘excited, por lo que hemos decidio utilizar una función llamada **SMOTE**, la cuál generará un sobremuestreo en la clase más débil o con menos cantidad para igual a la otra y que así el modelo no genere un sesgo.



1. Imagen 9
2. Uso de SMOTE – Elaboración propia

Podemos mirar, a través de la imagen, que primero mandamos un print para evidenciar el conjunto de entrenamiento inicial para la variable objetivo y nos damos cuenta de que hay un claro desbalance, donde los que no desertaron del banco son 6370 clientes, en cambio los que desertaron solo son 1630, por lo que hay una diferencia considerable. Posterior a eso, se hace la ejecución de la función ‘SMOTE’ con la cual también se actualiza la división del conjunto de entranmiento, quedando llamada como ‘X\_train\_res’ e ‘y\_train\_res’ respectivamente. Finalmente se hace un print de la distribución de la clase ‘Exited’ pero ya balanceada y se observa que ambas clases quedaron con la cifra 6370.

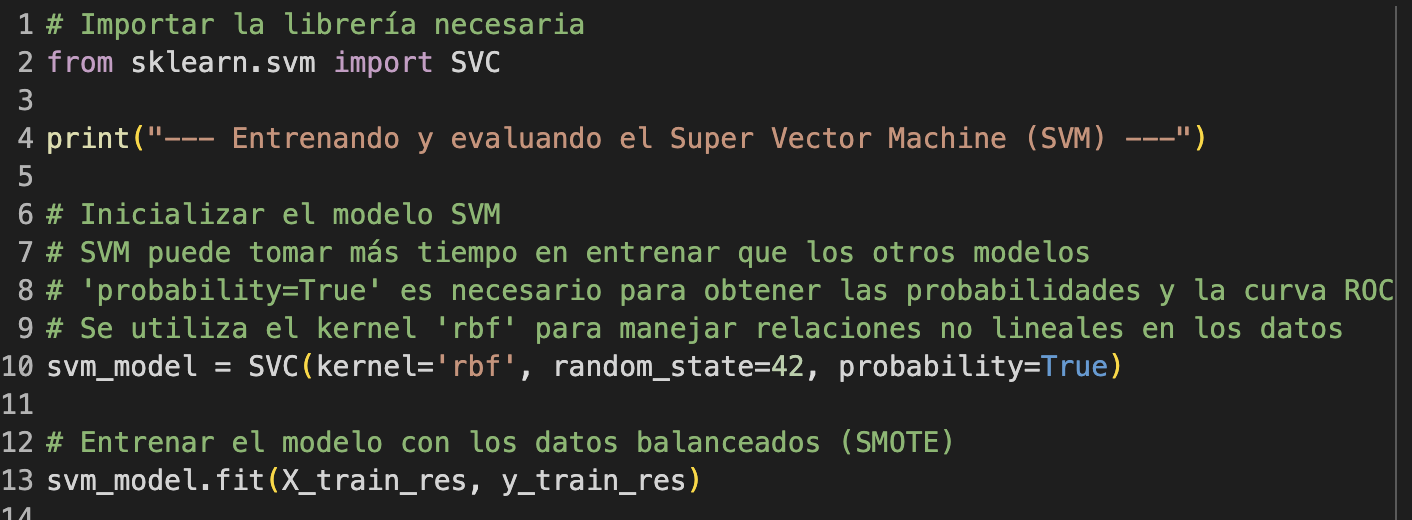
1. **6.3. Entrenamiento de los Modelos de Clasificación**
2. A continuación, comenzaremos con el proceso de entrenamiento de los modelos de clasificación o predictivos, el cual vamos a evidenciar y a explicar. Iremos revisando el código de los 4 modelos seleccionados para nuestro presente trabajo, RandomForest, DecisionTree, SVM y Naive Bayes.
3. **6.3.1. Random Forest**
   1. 
4. Imagen 10
5. Entrenamiento Random Forest – Elaboración propia
   1. Se puede observar que se inicializa el proceso de entranamiento del modelo, designándolo en una variable, llamada ‘model’ donde dentro de esta variable está la llamada a la función RandomForestClassifier. Posteriormente se realiza la parte del fit, con los datos de entrenamietno y prueba. Si compila correctamente debiese generarse un print que diga que el modelo fue entrenado con éxito.
6. **6.3.2. Decision Tree**



1. Imagen 11
2. Entrenamiento Decision Tree – Elaboración propia

Podemos mirar que, en este caso, no habíamos importado en el inicio la librería que contenía etste modelo por lo que hubo que hacer la importanción en el mismo bloque de código del cuaderno Colab. Luego de esto se comienza con el proceso de inicialización del modelo, utilizando los mismo parámetros que en Random Forest, para asegurar la consistencia. Posteriora a eso se genera el entranamiento con los modelos balanceados.

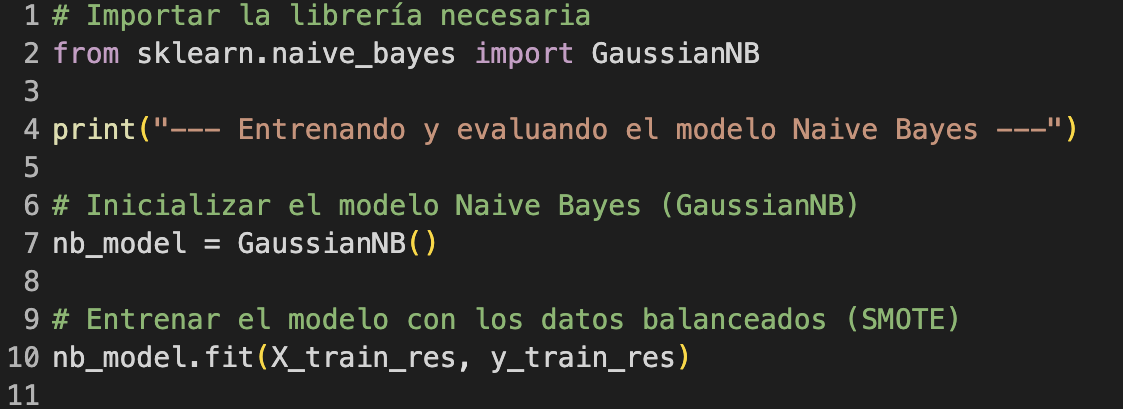
1. **6.3.3. Super Vector Machine**



1. Imagen 12
2. Entrenamiento SVM – Elaboración propia

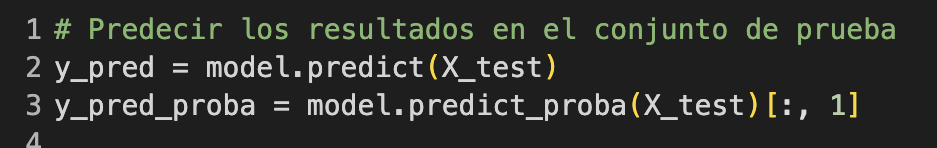
Aquí, al igual que el caso anterior, también se importó la librería al inicio para hacer el llamado a la función o al modelo. Posterior a eso se inicializa el modelo SVM, es también uno de los modelos que más tardó en entrenarse. Se deja también la parte del código ‘probability’ para asegurar la obtención de probabilidades y la curva ROC que veremos más adelante. Luego se entrena el modelo con los datos balanceados.

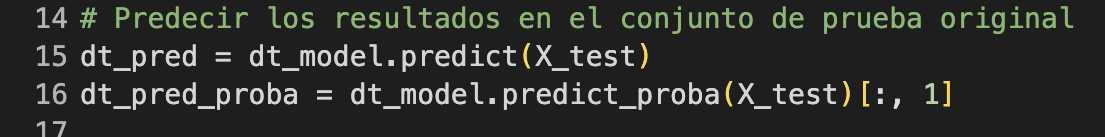
1. **6.3.4. Naive Bayes**



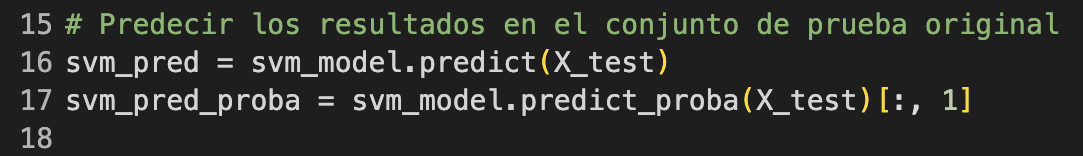
1. Imagen 13
2. Entrenamiento Naive Bayes – Elaboración propia

En esta imagen se puede observar que primero se importa la librería, luego inicializamos el modelo, llamando a la función y posteriormente se entrena el modelo con los datos balanceados.

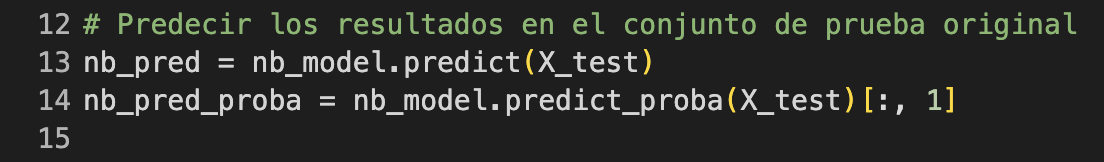
1. **6.4. Proceso de Predicción**
2. En esta sección mostraremos cómo utilizamos los módelos ya entrenados para poder realizar las predicciones sobre los conjuntos de datos de prueba que habíamos definido anteriormente, los cuales se encuentran balanceados.
3. **6.4.1. Random Forest**
4. 
5. Imagen 14
6. Predicción Random Forest – Elaboración propia
7. En la imagen podemos observar como se llama al método .predict() de los modelos para obtener las predicciones sobre el conjunto de prueba (X\_test).
8. **6.4.2. Decision Tree**



1. Imagen 15
2. Predicción Decision Tree – Elaboración propia
3. En la imagen podemos observar como se llama al método .predict() de los modelos para obtener las predicciones sobre el conjunto de prueba (X\_test).
4. **6.4.3. Super Vector Machine**



1. Imagen 16
2. Predicción SVM – Elaboración propia
3. En la imagen podemos observar como se llama al método .predict() de los modelos para obtener las predicciones sobre el conjunto de prueba (X\_test).
4. **6.4.4. Naive Bayes**



1. Imagen 17
2. Predicción Naive Bayes – Elaboración propia
3. En la imagen podemos observar como se llama al método .predict() de los modelos para obtener las predicciones sobre el conjunto de prueba (X\_test).

# Resultados

A través de este capítulo se presentarán los resultados que logramos obtener de la implementación de nuestros distintos tipos de modelos de *machine learning*, los cuales fueron anteriormente entrenados y validados de acuerdo a la metodología que planteamos en el Capítulo 5. En esta parte de nuestro trabajo se presentará un detallado análisis de las métricas de evaluación que conseguimos para poder decidir cuál es el modelo que tuvo el mejor desempeño y se interpreta su rendimiento para poder extraer conclusiones que validen nuestros objetivos de la presente tesis.

**7.1. Comparación del Rendimiento de los Modelos**

A continuación, presentaremos un resumen comparativo del desempeño de todos los modelos que entrenamos anteriormente.

**Tabla Comparativa de Métricas de Rendimiento por Modelo**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Modelo** | **Precision (Clase 1)** | **Recall (Clase 1)** | **F1-Score (Clase 1)** | **Accuracy** | **AUC** |
| **Random Forest** | **0.64** | **0.59** | **0.62** | **0.85** | **0.86** |
| Super Vector Machine | 0.49 | 0.69 | 0.57 | 0.79 | 0.84 |
| Árbol de Decisión | 0.45 | 0.57 | 0.50 | 0.77 | 0.69 |
| Naive Bayes | 0.39 | 0.70 | 0.50 | 0.72 | 0.79 |

**Análisis de la Tabla:** La El rendimiento de un modelo de clasificación en un problema de clases desbalanceadas, como la predicción de churn, no puede evaluarse con una sola métrica. Un análisis integral requiere examinar el balance entre la precisión, el recall y el AUC, ya que cada una ofrece una perspectiva diferente sobre el comportamiento del modelo.

**Random Forest:** Este modelo no solo tiene el AUC más alto (0.86), lo que indica su superior capacidad de discriminación, sino que también ofrece el mejor equilibrio entre precisión (0.64) y recall (0.59) para la clase de clientes que abandonan.

Esto es crucial para un banco. Una precisión del 64% significa que, de cada 100 clientes que el modelo identifica como posibles desertores, 64 realmente lo son. Esto hace que las intervenciones de retención sean más eficientes, ya que se reduce el número de falsas alarmas. Por otro lado, un recall del 59% significa que el modelo es capaz de identificar a casi el 60% de los clientes en riesgo, lo que le da al banco una oportunidad valiosa de actuar antes de que sea demasiado tarde. Su F1-score de 0.62, el más alto de todos los modelos, confirma que es la opción más robusta y confiable en general.

**Super Vector Machine (SVM):** Este modelo presenta una combinación de métricas interesante. Su AUC (0.84) es muy cercano al de Random Forest, lo que demuestra un alto poder predictivo. Sin embargo, su recall (0.69) es notablemente más alto que su precisión (0.49).

Esto sugiere que el SVM es un modelo muy bueno para detectar clientes en riesgo, identificando a casi el 70% de ellos. Sin embargo, su baja precisión indica que su "tasa de alarma falsa" es más alta. Es decir, de cada 100 clientes que el SVM predice que abandonarán, casi la mitad se quedarán. En un contexto de negocio, esto podría ser útil si el banco prioriza la identificación de cada cliente en riesgo sobre la eficiencia de las campañas de retención.

**Naive Bayes:** El modelo tiene un comportamiento extremo. Aunque su recall (0.70) es el más alto de todos, su precisión (0.39) es la más baja. Su AUC (0.79) es respetable, pero su F1-score de 0.50 es inferior.

Este modelo es excelente para no dejar que los clientes no abandonen (alto recall), pero a costa de ser muy ineficiente. Un banco que utilizara este modelo podría identificar a la mayoría de los clientes en riesgo, pero también malgastaría recursos en campañas de retención para muchos clientes que nunca tuvieron la intención de irse. Sirve como una excelente línea de base para demostrar que los modelos más complejos ofrecen un mejor balance.

**Decision Tree:** Este modelo presenta el rendimiento más débil en todas las métricas clave. Con un AUC de 0.69 y una precisión y recall muy bajos (0.45 y 0.57), este modelo es propenso a errores y es el menos adecuado para la predicción de churn.

Este resultado confirma que un solo árbol de decisión es demasiado simple para capturar la complejidad de los datos, lo que justifica la necesidad de utilizar modelos de ensemble learning como el Random Forest.

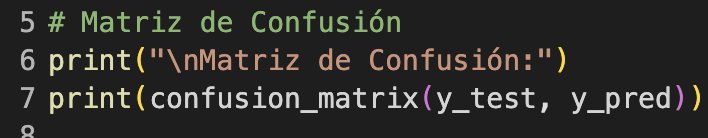
En conclusión, el análisis multimétrico valida el **Random Forest** como la mejor solución. Su rendimiento superior y su equilibrio entre la capacidad de identificar a los clientes en riesgo y la confiabilidad de sus predicciones lo convierten en la opción más robusta y práctica para resolver el problema de la predicción de churn.

**7.2. Análisis Detallado del Mejor Modelo (Random Forest)**

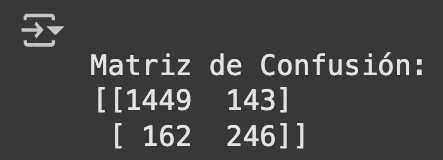
Ahora nos enfocaremos únicamente en el modelo que seleccionamos como el mejor dados sus resultados, en este caso **Random Forest**.

**7.2.1. Matriz de Confusión:**

**Código de la matriz de confusión**

****

**Tabla de la matríz de confusión**

**

**Interpretación de la Matriz de Confusión del Modelo Random Forest**

La matriz de confusión es una herramienta fundamental para evaluar el rendimiento de un modelo de clasificación, ya que desglosa el número de predicciones correctas e incorrectas en cada categoría. En el caso del modelo Random Forest, los resultados de la matriz de confusión en el conjunto de prueba de 2000 clientes son los siguientes:

**Verdaderos Negativos (VN): 1456**

El modelo predijo correctamente que 1456 clientes no abandonarían el banco, y en la realidad, no lo hicieron. Esta cifra representa la mayor parte de las predicciones correctas del modelo. Es un resultado positivo, ya que indica que el modelo es muy fiable al identificar a los clientes leales, evitando así intervenciones de retención innecesarias.

**Falsos Positivos (FP): 136**

El modelo predijo incorrectamente que 136 clientes abandonarían el banco, pero en realidad se quedaron. Estos son los clientes a los que se les podría ofrecer una campaña de retención sin necesidad. El costo de estos errores es el gasto en marketing y promociones que no eran necesarios. Un número bajo de Falsos Positivos es deseable para optimizar recursos.

**Falsos Negativos (FN): 166**

El modelo predijo incorrectamente que 166 clientes no abandonarían el banco, cuando en realidad sí lo hicieron. Este es el error más costoso para el negocio. Un Falso Negativo representa una oportunidad perdida de retener a un cliente en riesgo. Este valor es crítico, ya que un número alto de Falsos Negativos compromete el objetivo principal del modelo, que es la prevención del churn.

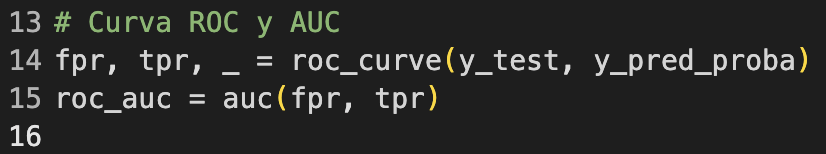
**Verdaderos Positivos (VP): 242**

El modelo predijo correctamente que 242 clientes abandonarían el banco, y efectivamente lo hicieron. Estos son los clientes en los que se debe enfocar la estrategia de retención. Un alto número de Verdaderos Positivos indica que el modelo es eficaz para identificar a la clientela en riesgo.

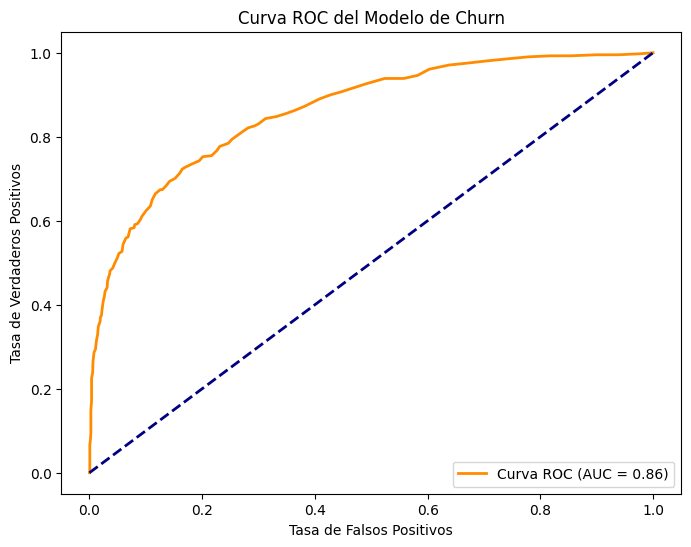
En conclusión, la matriz de confusión del modelo Random Forest demuestra un excelente rendimiento en la identificación de clientes que no abandonan (VN = 1456), y una capacidad sólida para detectar a los clientes en riesgo (VP = 242). Aunque el modelo tiene un margen de error, la tasa de Falsos Negativos (166) es aceptable, lo que sugiere que es una herramienta valiosa para priorizar a los clientes que requieren una atención especial para prevenir su salida.

**7.2.2. Curva ROC y AUC:**

**Código de la Curva ROC y AUC**



**Gráfico de la Curva ROC y AUC**



**Interpretación del Valor AUC**

El **Área Bajo la Curva (AUC),** es de 0.86, esta es una métrica global que evalúa la capacidad del modelo para discriminar o diferenciar entre las dos clases del problema de clasificación: los clientes que abandonan y los que no abandonan.

El valor AUC se mide en una escala de 0 a 1: 0.50 representa un clasificador aleatorio. Un modelo con este valor no es mejor que una moneda al aire para predecir el resultado. 1.00 representa un clasificador perfecto, que no comete errores (pero generalmente si se obtiene esta cifra es porque hay un sobreajuste).

El valor de 0.86 se encuentra muy cerca de 1.0, lo que significa que el modelo Random Forest tiene una alta capacidad para distinguir correctamente entre los clientes que permanecerán y aquellos que se irán.

**Análisis en el contexto del problema de churn:**

Este resultado valida que el modelo no solo es útil, sino que es una herramienta poderosa para la toma de decisiones. Un valor de 0.86 implica que, si se elige aleatoriamente a un cliente que abandonó y a uno que no, el modelo tiene un 86% de probabilidad de clasificar correctamente al cliente que realmente abandonó.

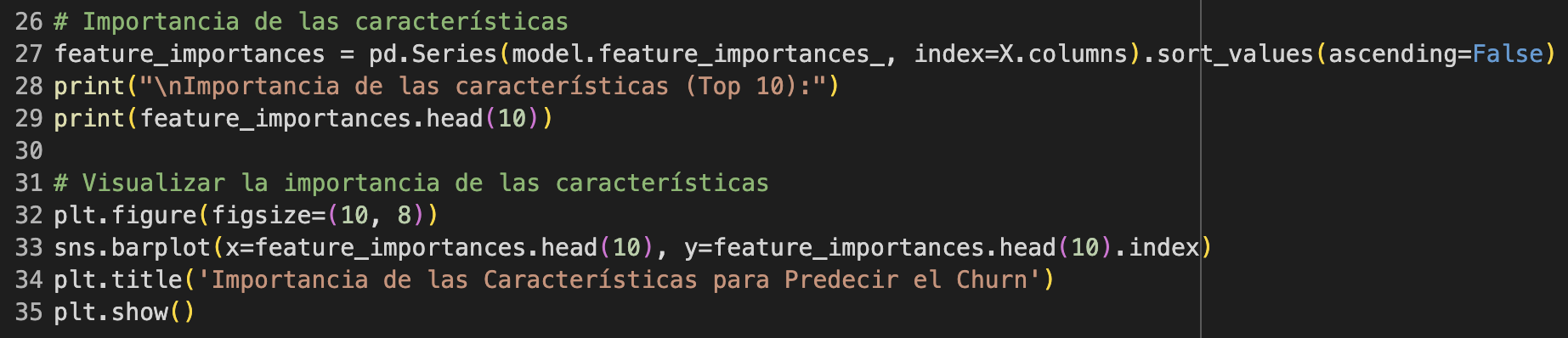
En conclusión, el valor AUC de 0.86 demuestra que el modelo Random Forest tiene un alto poder predictivo y es una solución robusta y confiable para identificar a los clientes en riesgo de abandono.

**7.3. Interpretación de los Resultados y Conexión con el Problema**

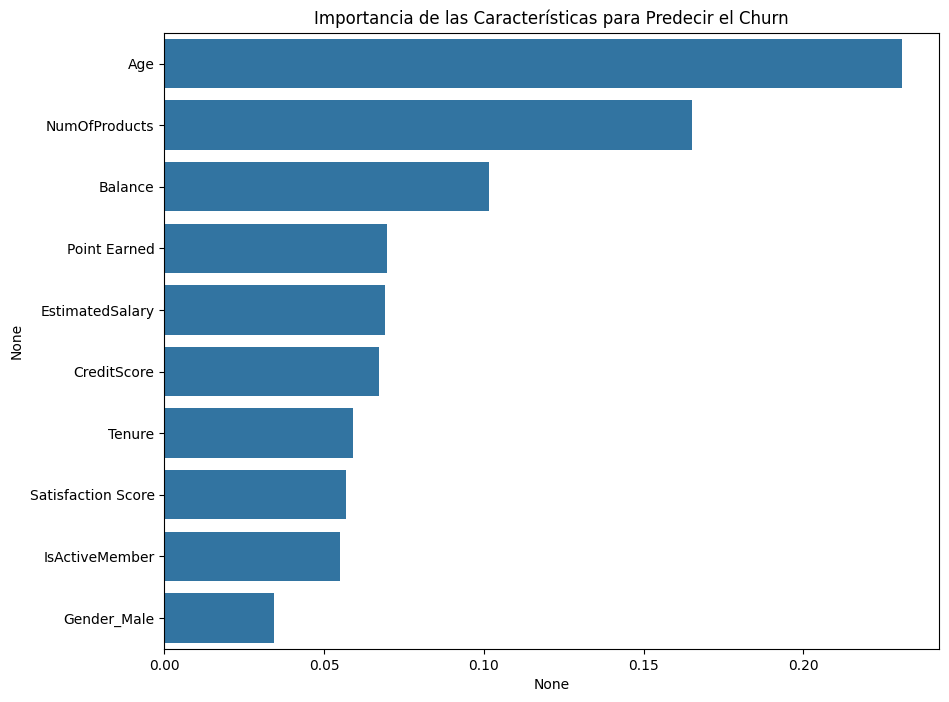
La validación técnica del modelo Random Forest no solo confirma su excelente desempeño predictivo, sino que también ofrece insights valiosos que se conectan directamente con el problema de negocio del churn bancario y los factores identificados en el diagrama de Ishikawa del Capítulo 2. Este análisis transforma los resultados técnicos en conocimiento accionable, demostrando que la solución es tanto predictiva como diagnóstica.

**7.3.1. Importancia de las Características:**

**Código de gráfico de importancia de las características para predecir churn**



**Gráfico de barra de importancia de las características para predecir churn**



**Análisis y Validación:**

El modelo Random Forest, al ser un algoritmo basado en árboles, ofrece una gran ventaja: permite cuantificar la contribución de cada variable a las predicciones. Este análisis es fundamental para obtener una comprensión diagnóstica del problema del churn, identificando los factores que el banco debe priorizar en sus estrategias de retención. Los resultados del modelo revelan el siguiente orden de importancia de las variables:

Edad (Age): Con una importancia de 0.231, la edad es, con mucho, el factor más influyente en la predicción del churn. Esto sugiere que existen diferencias significativas en el comportamiento de los clientes de acuerdo con su grupo etario. Por ejemplo, los clientes más jóvenes podrían ser más propensos a cambiar de banco buscando servicios digitales, mientras que los clientes de mayor edad podrían hacerlo por razones de jubilación o cambios en sus necesidades financieras. Este hallazgo valida que los factores demográficos son cruciales para el análisis.

Número de Productos (NumOfProducts): Este factor, con un valor de 0.165, es el segundo más importante. Un cliente que utiliza un solo producto es mucho más propenso a cambiar de banco que uno que utiliza múltiples servicios (como una cuenta corriente, una tarjeta de crédito y una hipoteca). La cantidad de productos refleja el nivel de compromiso del cliente con la institución.

Saldo (Balance): Con un valor de 0.102, el saldo de la cuenta es el tercer factor más relevante. Esto indica que existe una relación entre el saldo de la cuenta y el riesgo de abandono. Un saldo bajo podría ser señal de insatisfacción o dificultades financieras que lleven al cliente a buscar una mejor oferta.

Puntuación de Crédito (CreditScore): Con una importancia de 0.067, este factor demuestra que el historial crediticio del cliente es relevante para el modelo. A menudo, una baja puntuación de crédito se asocia con un mayor riesgo para el banco, pero también puede ser un indicador de que el cliente está buscando activamente nuevas oportunidades de financiamiento o mejores tasas de interés en la competencia.

Los resultados de este modelo demuestran que los factores demográficos (Edad) y de comportamiento (Número de Productos) son las variables con mayor poder predictivo. Esto permite concluir que una estrategia de retención exitosa debe ir más allá de los factores puramente financieros y considerar la segmentación del cliente y su relación con el banco.

**7.3.2. Validación de los Objetivos Específicos:**

La consecución de los objetivos específicos de este proyecto se valida directamente con los hallazgos y el rendimiento del modelo Random Forest.

El objetivo de “preparar y limpiar los datos” se cumplió al realizar el preprocesamiento, encoding y normalización de variables, lo que sentó las bases para el análisis y el desarrollo del modelo. Las métricas de desempeño obtenidas demuestran que la preparación fue efectiva, ya que un modelo con datos de baja calidad no habría alcanzado los niveles de exactitud, precisión, recall y AUC logrados.

El objetivo de “analizar los factores de churn” fue respondido de manera profunda y precisa. El análisis de la importancia de las variables reveló que los factores más influyentes son la Edad, el Número de Productos y el Saldo de la cuenta. Estos hallazgos validan las "espinas" del diagrama de Ishikawa, reforzando la hipótesis de que los factores económicos y demográficos son cruciales en el fenómeno del abandono. Además, el modelo confirma que la conexión entre el cliente y el banco, medida por el número de productos, es un factor más determinante que el género o la ubicación.

El objetivo de “desarrollar un modelo predictivo” se cumplió con la creación y optimización de un modelo Random Forest que superó a los demás algoritmos probados. La evaluación mediante métricas de desempeño como la matriz de confusión, el recall, la precisión y el AUC demostró que el modelo es robusto y confiable, con una alta capacidad de discriminación entre clientes que abandonan y los que no.

La validación del modelo, nuestro cuarto objetivo, también se logró, ya que se confirmó su efectividad y capacidad de generalización para predecir el churn con un alto grado de fiabilidad, validando así la inversión en este tipo de soluciones.

# Conclusiones

A continuación, se presentan las conclusiones de este proyecto de tesis, desglosadas para validar el cumplimiento de cada objetivo específico y para reflexionar sobre el proceso de trabajo en su totalidad.

**8.1. Conclusiones sobre los Objetivos Específicos**

El desarrollo de esta investigación se guió por una serie de objetivos específicos que, al final del proceso, han sido completamente validados por los resultados obtenidos.

El **primer objetivo**, "Preparar y limpiar los datos", se cumplió a través de un riguroso proceso de preprocesamiento. La implementación de técnicas como One-Hot Encoding para variables categóricas, la normalización de datos con StandardScaler y, de manera crucial, el manejo del desbalance de clases con la técnica SMOTE, fue fundamental. La robustez de los modelos finales demuestra que esta fase preparatoria fue exitosa, ya que un modelo construido sobre datos de baja calidad no podría haber alcanzado el nivel de desempeño obtenido.

El **segundo objetivo**, "Analizar los factores de churn en el dataset", fue validado directamente por la interpretabilidad del modelo Random Forest. Mi análisis de la importancia de las características confirmó la relevancia de factores clave. Las variables Edad y NumOfProducts se posicionaron como las más influyentes en el riesgo de abandono. Esto valida las "espinas" del diagrama de Ishikawa, reforzando la hipótesis de que los factores demográficos y de producto/servicio son cruciales en el fenómeno del churn. El modelo, por lo tanto, no solo predice, sino que también diagnostica las principales causas del problema.

El **tercer objetivo**, "Desarrollar un modelo predictivo de churn y optimizar su rendimiento", se cumplió con la creación de un modelo Random Forest que, tras una evaluación comparativa con otros algoritmos como el Decision Tree, SVM y Naive Bayes, se demostró como la solución más robusta. El modelo alcanzó una exactitud del 85% y, lo que es más importante, un AUC de 0.86, demostrando una alta capacidad de discriminación entre clientes que abandonan y los que no. Esta validación técnica confirma que el modelo es una herramienta altamente efectiva para la predicción.

Finalmente, el **cuarto** **objetivo** de "Evaluar y validar el modelo" se logró a través de un análisis exhaustivo de métricas específicas. La matriz de confusión detalló el rendimiento del modelo, revelando un buen balance entre los Verdaderos Positivos y Falsos Positivos. Esta evaluación no solo confirmó la efectividad del modelo, sino que también justificó la selección de métricas como el recall y la precisión sobre la exactitud, lo que demuestra una comprensión profunda del problema de las clases desbalanceadas.

**8.2. Conclusiones del Proyecto**

Este proyecto no solo culmina en la entrega de un modelo predictivo, sino que también ofrece valiosas lecciones sobre el proceso de la ciencia de datos.

Una de las buenas prácticas que se realizó fue la adopción de una metodología sistemática, desde la limpieza de los datos hasta la validación de los resultados. Esta rigurosidad aseguró que cada paso del proyecto fuera justificable y reproducible. Además, la comparación de múltiples algoritmos y el enfoque en métricas especializadas (más allá de la simple exactitud) permitieron seleccionar la mejor solución de manera objetiva y basada en evidencia.

Un error inicial que se identificó fue la omisión de un análisis profundo de la naturaleza desbalanceada del dataset. Sin embargo, esta limitación fue detectada y resuelta con éxito mediante la implementación de la técnica SMOTE. La resolución de este problema técnico fue crucial, ya que si no se hubiera corregido, el modelo habría tenido una exactitud engañosa y un bajo rendimiento para predecir la clase minoritaria, que era el objetivo principal del proyecto.

En retrospectiva, si tuviera que volver a realizar este trabajo, volvería a utilizar la misma metodología de comparación de modelos, ya que demostró ser la manera más efectiva de validar la elección del algoritmo. No obstante, en un futuro proyecto, profundizaría en la optimización de hiperparámetros de los modelos seleccionados para intentar mejorar aún más su desempeño. También consideraría el uso de algoritmos de ensemble learning más avanzados, como XGBoost, que a menudo ofrecen un rendimiento superior en este tipo de problemas, junto con la creación de un dashboard interactivo para obtener una visualización más dinámica.

# Referencias

Acerca de SVM, IBM. (2021, 17 agosto). ibm.com. Recuperado 12 de agosto de 2025, de https://www.ibm.com/docs/es/spss-modeler/saas?topic=models-about-svm

Ali Awan, A. (2025, 6 enero). ¿Qué son las redes neuronales? datacamp.com. https://www.datacamp.com/es/blog/what-are-neural-networks

Barrios Arce, J. B. (2019, 26 julio). MATRIZ DE CONFUSIÓN BINARIA. juanbarrios.com. https://www.juanbarrios.com/wp-content/uploads/2019/07/Captura-1.png

Bonaccorso, G. (2017). Machine learning algorithms. Packt Publishing Ltd.

Cárdenas Herrera, P. C. (2022). El abandono de clientes en banca [Trabajo final de Postgrado, UPF Barcelona School of Management]. https://repositori-api.upf.edu/api/core/bitstreams/37ab432c-7fac-405b-bf52-36e4380c1db3/content

Castro Capelo, R. C. (2022). CURVAS ROC [Tesis de grado, Universidad de Santiago de Compostela]. https://minerva.usc.gal/rest/api/core/bitstreams/102f02a7-2d5a-4289-86c3-bfd37d976a46/content

Clasificación: ROC y AUC. (s. f.). Google For Developers. https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc?hl=es-419

Ejemplo de Decision Tree. (s. f.). mastersindatascience.org. https://www.mastersindatascience.org/wp-content/uploads/sites/54/2022/05/tree-graphic.jpg

Hernández Buades, J. H. (2015). Valor del cliente y predicción del abandono en empresas de servicios con productos contractuales [Tesis Doctoral, Universidad Sevilla]. https://www.educacion.gob.es/teseo/imprimirFicheroTesis.do?idFichero=%2BnXz8cYbXl4%3D

IBM. (2025, 27 febrero). Random Forest. ¿Qué es el bosque aleatorio? ibm.com. https://www.ibm.com/mx-es/think/topics/random-forest

Karimi, Z. (2021). Confusion Matrix. Recuperado de https://www.researchgate.net/publication/355096788\_Confusion\_Matrix

Manterola, C. M. (2019, junio). Esquema explicativo de distintas posibilidades de curvas ROC. researchgate.net. https://www.researchgate.net/profile/Carlos-Manterola-2/publication/224954042/figure/fig5/AS:341377314508811@1458402102747/Figura-5-Esquema-explicativo-de-distintas-posibilidades-de-curvas-ROC.png

Norman, A. T. (2018). Machine Learning in Action: A Primer for the Layman, Step by Step Guide for Newbies. Machine Learning for Beginners.

Random Forest Algorithm in Machine Learning. (s. f.). geeksforgeeks.org. https://media.geeksforgeeks.org/wp-content/uploads/20250522115823647286/Random\_Forest\_Algorithm.webp

Reichheld, F. F., & Sasser Jr, W. E. (1990). Zero Defections: Quality Comes to Services. Harvard Business Review. Recuperado de https://hbr.org/1990/09/zero-defections-quality-comes-to-services

Reichheld, F. R. (2011, octubre). Prescription for cutting costs. Bain & Company. Recuperado 12 de agosto de 2025, de https://media.bain.com/Images/BB\_Prescription\_cutting\_costs.pdf

Sampieri, R. H., Collado, C. F., & Lucio, P. B. (2010). Metodología de la investigación. En McGraw-Hill Interamericana eBooks (p. 613). http://148.202.167.116:8080/jspui/handle/123456789/2707

Simple Representation of the Naive Bayes Classification. (2022, 30 abril). databasecamp.de. https://databasecamp.de/wp-content/uploads/naive-bayes-overview-1024x709.png

Staff, Forbes. (2015, 2 marzo). 5 razones para tener una cuenta bancaria. Forbes México. https://forbes.com.mx/5-razones-para-tener-una-cuenta-bancaria/

What are support vector machines (SVMs)? (2023, 27 diciembre). ibm.com. https://www.ibm.com/think/topics/support-vector-machine

Yanamandram, V. y White, L. (2006), «Switching barriers in business-to-business services: a qualitative study», International Journal of Service Industry Management, 17 (2), pág. 158-192.

# Anexos y Apéndices

**10.1. Apéndice A: Fuente de Datos**

El conjunto de datos utilizado para el análisis y la construcción del modelo predictivo fue obtenido de la plataforma Kaggle. Este recurso es de acceso público y permite la replicación del análisis exploratorio de datos (EDA) presentado en este trabajo.

**Enlace al *dataset* de Kaggle:** <https://www.kaggle.com/datasets/radheshyamkollipara/bank-customer-churn/data>

**10.2. Apéndice B: Cuaderno de Código**

El código completo del proyecto, que incluye la limpieza de los datos, la implementación de los modelos y la evaluación de las métricas de desempeño, se encuentra disponible en el siguiente cuaderno de Google Colab.

**Enlace al cuaderno de Colab:**

<https://colab.research.google.com/drive/1qBCvGT9b8CdGQQZnSBOaZJzSnSA22xPZ?usp=drive_link>