Pontificia Universidad Católica del Ecuador

Facultad De Ingeniería

TEMA:

DISEÑO DE UN MODELO PREDICTIVO DE FUGA DE CLIENTES UTILIZANDO ALGORITMOS DE MACHINE LEARNING

AUTOR:

JOSÉ RICARDO NAVAS AYALA

TRABAJO PREVIA A LA OBTENCIÓN DEL TÍTULO DE MAGISTER EN SISTEMAS DE INFORMACIÓN MENCIÓN DATA SCIENCE

Quito, Junio – 2023

DEDICATORIA

A mi familia, mi esposa, mis hijas por su apoyo incondicional en este proceso para poder cumplir mi meta propuesta, su confianza y afecto me han ayudado a superarme cada día más, a mi abuelita que desde el cielo sé que me sigue cuidando y guiando.

AGRADECIMIENTO

En primer lugar, a Dios por permitirme seguir aprendiendo día tras día y protegiendo a mi familia con amor, a la Corporación GPF por abrirme las puertas dándome la confianza para poder realizar este proyecto, a mi tutor que fue la persona que me guio en este proceso con su experticia y paciencia.

RESUMEN

El presente trabajo tiene como fin el desarrollo de un modelo predictivo utilizando algoritmos de Machine Learning para la predicción de los clientes que pueden llegar a convertirse en fuga, de tal manera que pueda ser una herramienta de alerta temprana para tomar acciones en cuanto a evitar que un cliente abandone la marca, incentivándolo nuevamente a realizar alguna compra ya sea esta por promociones o descuentos que se le pueda otorgar en determinados productos. Se empleó la metodología CRISP-DM para organizar de manera estructurada la información relevante y el flujo de actividades durante el desarrollo y evaluación del modelo predictivo. El modelo fue creado utilizando Python y se aprovecharon las bibliotecas de pandas y scikit-learn. Se eligió un algoritmo específico para desarrollar el modelo, el cual es el de árboles de decisión con 4 variables predictoras de tipo numérico, que fueron tomadas entre la recencia, frecuencia, valor monetario y RFM_Score. El modelo seleccionado es de suma utilidad para el negocio ya que ha resuelto un problema que venía manejando la corporación desde hace tiempo, al no poder identificar a sus clientes previamente antes de convertirse en fuga, siendo validado por los indicadores de precisión con 91.63%, una curva ROC_AUC de 95.40% y un recall de 96.3%. Con el modelo obtenido se obtuvo la base final que es exportada a Excel a través de una ruta compartida.

TABLA DE CONTENIDOS

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

CAPÍTULO I: INTRODUCCIÓN

1. Introducción

Durante los últimos años, Machine Learning o Aprendizaje Automático (ML) ha ganado popularidad en el ámbito tecnológico. Esto se debe en gran parte a la enorme cantidad de datos generados por las herramientas tecnológicas actuales, así como al crecimiento en el poder de cómputo de los datos y al desarrollo de algoritmos más avanzados. Como resultado, las aplicaciones del ML abarcan desde la automatización de tareas diarias hasta la optimización de procesos en diversos sectores industriales que buscan aprovechar esta tecnología.

En un contexto donde el mercado esta convirtiéndose más y más competitivo y a su vez agresivo en la búsqueda de clientes, la retención de clientes se ha convertido en uno de los desafíos más importantes para las empresas. En este escenario, donde el cliente ocupa un lugar central en cualquier negocio, las compañías necesitan llevar a cabo un análisis de gestión más avanzado con el objetivo de detectar y prevenir de manera temprana la pérdida de clientes.

En definitiva, el aprendizaje automático se convierte en una herramienta para medir de forma eficaz la probabilidad en que los clientes abandonen la marca para poder decidir dónde centrar sus esfuerzos y así evitar que en la medida de lo posible escapen a la competencia, dado que atraer nuevos clientes o tratar de recuperar a los perdidos es más difícil y costoso que retener a los clientes existentes.

1.1. Antecedentes

"La Corporación GPF fundada en 1930, con el propósito de generar bienestar con servicios y productos de calidad. Desde esa fecha, han transcurrido 90 años en los que Fybeca ha evolucionado e innovado, llevando a los hogares ecuatorianos los mejores productos y un

amplio portafolio de medicinas complementadas con otras soluciones integrales, para el cuidado de la salud y bienestar de las familias.

En el 2018, la multinacional mexicana Fomento Económico Mexicano (Femsa), adquirió el 100% de Corporación GPF, representando una de las mayores inversiones en el país durante los últimos años, y que permitió continuar con la expansión del negocio a nivel regional" (GPF, 2023).

Las principales áreas en la corporación son la comercial y atención al cliente estas cuentan con un papel fundamental en la capacidad de establecer relaciones comerciales para captar y retener clientes mediante campañas y beneficios que sean atractivos al consumidor final.

En la Corporación GPF no existe un modelo para predecir la fuga de clientes, por ende las actividades de retención se retrasan, son empíricas y no se basan en los patrones de comportamiento de los clientes. Un componente fundamental de un sistema de gestión de relaciones con los clientes, conocido en inglés como Customer Relationship Management (CRM), es un modelo que posibilita la predicción de la tasa de fuga de clientes.

Actualmente el porcentaje de fuga de los clientes equivale al 22% según el cálculo de churn (Peiró, 2022) que es (Clientes perdidos ÷ número total de clientes en un periodo de tiempo) x 100 = tasa de abandono de clientes, en un periodo de 12 meses, las causas se deben a:

- Stock de productos.
- Precios altos.
- Apertura de nuevos locales de la competencia
- Mejores ofertas por la competencia.
- Mayor gestión de marketing por la competencia.
- Atención al cliente.

Con lo expuesto la fuga de clientes es definida cuando un cliente tiene un tiempo de inactividad superior a 365 días, este tiempo de inactividad significa que el cliente no realizo compras

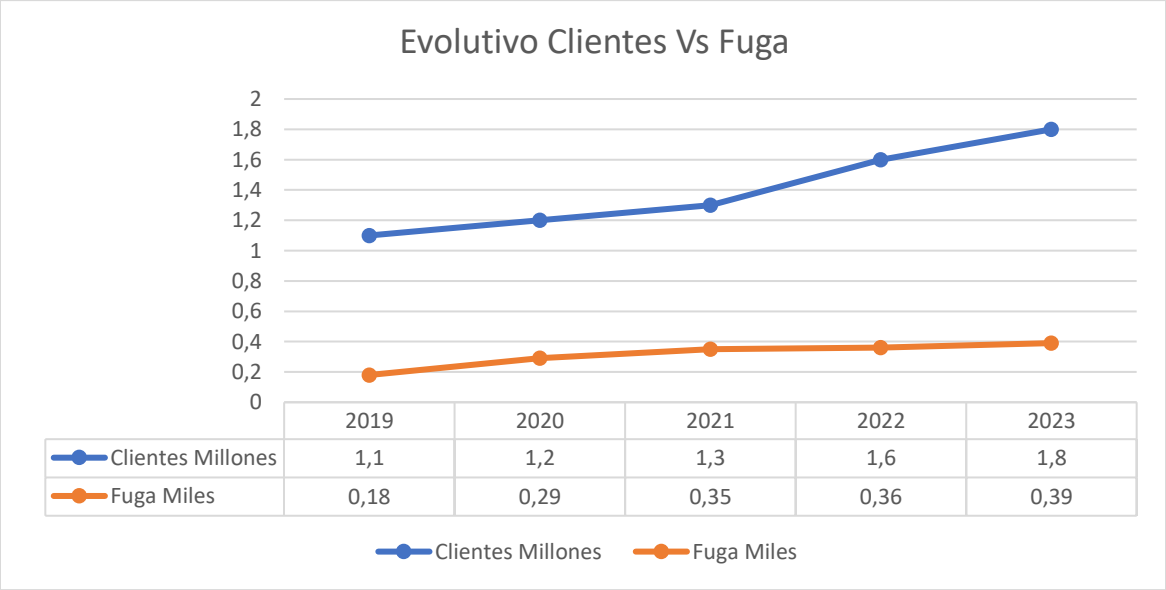
a través de ningún canal ya sea en un punto de venta (PDV) o a su vez utilizando algún medio tecnológico (app o web). Con lo mencionado desde ahora existen principalmente dos tipos de usuarios, los que dejan de comprar en la marca y aquellos que continúan comprando dentro de la misma de forma regular en los 365 días.

1.2. Planteamiento del Problema

Según los datos obtenidos ver figura 1, la empresa ha logrado un incremento de nuevos clientes a lo largo del tiempo y con un nivel de fuga en forma incremental.

Figura 1

Evolutivo histórico clientes vs fuga



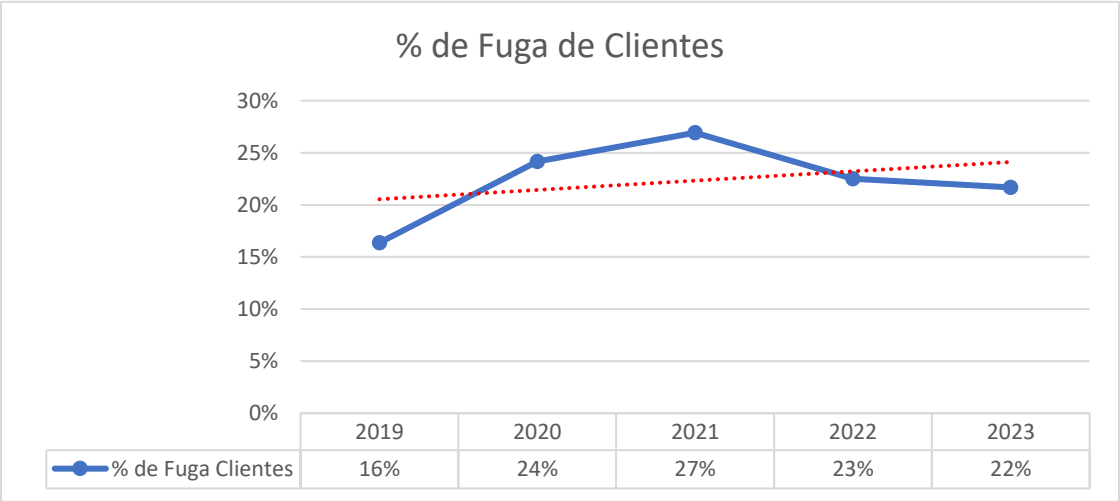
	2019	2020	2021	2022	2023
Clientes Millones	1,1	1,2	1,3	1,6	1,8
Fuga Miles	0,18	0,29	0,35	0,36	0,39

Nota. Gráfico que muestra el evolutivo anual de clientes vs el número de clientes fugados, autoría propia mediante datos obtenidos de las fuentes de la corporación, 2023.

Haciendo un análisis más de cerca a los clientes fugados ver figura 2 vemos que el porcentaje de fuga anual tiende a aumentar. Por lo tanto, si se complementa con lo expuesto en la figura 1, podemos concluir que cada vez se vuelve una prioridad el desarrollar herramientas que tengan como objetivo la prevención de la fuga de clientes.

Figura 2

Porcentaje anual de fuga de clientes histórico



Nota. Gráfico que muestra el evolutivo anual en porcentaje de clientes fugados, autoría propia mediante datos obtenidos de las fuentes de la corporación, 2023.

Con lo anteriormente dicho nos planteamos la siguiente pregunta ¿es posible que el modelo de predicción de fuga de los clientes usando algoritmos de machine learning disminuya la fuga?

1.3. Justificación

El actual trabajo contribuirá a generar un modelo para entender la fuga de clientes en el retail farmacéutico, dichos resultados obtenidos ayudaran a hallar los comportamientos originados por los consumidores y ejecutar acciones preventivas en conjunto con el área de marketing y CRM, creando una correlación directa con el cliente de ganar-ganar.

1.4. Objetivos

1.4.1. Objetivo General

Desarrollar un modelo predictivo usando algoritmos de machine learning para estimar la fuga de clientes que permita levantar alertas tempranas y atender necesidades insatisfechas de los usuarios.

1.4.2. Objetivos Específicos

- Analizar el segmento de clientes base a ser utilizado en el modelo predictivo.
- Determinar las variables más significativas a ser utilizadas en el modelo.
- Implementar el modelo de predicción de fuga.
- Recomendar a la empresa el segmento de clientes donde deberían enfocar sus acciones de fidelización.

CAPÍTULO II: FUNDAMENTACIÓN TEORICA

2. Marco Teórico

2.1. Fuga de Clientes

La fuga de clientes en su traducción al inglés (Churn) es un inconveniente común en las empresas, banca, industrias, telecomunicaciones y más. La rotación de clientes se produce cuando se utilizan estrategias de marketing agresivas para atraer clientes ofreciéndoles nuevos y mejores productos o servicios.

Como menciona (Rodríguez y Gallard, 2020) las principales causas de pérdida de clientes en la industria están relacionadas con factores creados por los servicios de la empresa, como la calidad de los productos, atención al cliente, el precio, etc.

La retención de clientes es un factor crucial, dado que conseguir nuevos clientes puede ser hasta siete veces más costoso que mantener a los clientes actuales. Una elevada tasa de pérdida de clientes, como se muestra en la Figura 3, representa una señal evidente de alarma para cualquier empresa. Por consiguiente, es imperativo implementar de inmediato estrategias de retención con el fin de evitar un incremento adicional en la rotación y las consiguientes pérdidas de ingresos.

Figura 3

Cálculo de Tasa de Abandono de Clientes

$$\text{Tasa de Abandono del Cliente} = \frac{\text{Núm. de clientes perdidos}}{\text{Núm. total de clientes (período)}} \times 100$$

Nota. Adaptado de ¿Qué es el abandono del cliente y cómo reducirlo? [Fotografía], Muguira A., 2018, QuestionPro. (https://www.questionpro.com/blog/es/abandono-del-cliente/). CC BY 2.0

En este caso para el presente trabajo la fuga de un cliente se entiende por aquel que ha dejado de comprar en la marca por cualquier canal por más de 12 meses y los clientes potenciales a fuga se los reconocerá como aquellos que han dejado de comprar de igual forma en la marca en los últimos 6 meses.

2.2. Análisis RFM

El análisis RFM es un método para segmentar el comportamiento del cliente en función de los datos según (Glotzer, 2022).

Cuando hablamos de RFM nos referimos a:

- R: Actualidad o Recencia
- F: Frecuencia
- M: Valor monetario.

Su objetivo es dividir a los clientes en diferentes grupos según su última compra, la cantidad de veces que realizaron compras anteriormente y el gasto total realizado.

Se ha demostrado que las tres medidas son efectivas para predecir la preparación del cliente para interactuar con mails y ofertas de marketing. El tema sobre el RFM apareció hace muy poco tiempo, es una herramienta de gran desempeño para los negocios de comercio electrónico de hoy.

El análisis RFM opera de manera similar a los porcentajes, ya que los valores relativos no proporcionan información ellos mismos, sino que deben ser relacionados con otro valor para poder interpretarlos. En tal caso, se utiliza un método de cuartiles que asigna valores del 1 al 4 según los umbrales de rendimiento definidos. Como resultado del segmento RFM se obtiene mediante la asignación de una calificación comúnmente utilizada.

Cundo se aplica el método del cuartil que es el más simple y útil, se procede a dividir a los clientes totales entre cuatro. Luego se asigna a cada uno de los cuartiles una puntuación que nos permita reflejar su posición.

- Cuartil primero: 1
- Cuartil segundo: 2
- Cuartil tercero: 3
- Cuartil cuarto: 4

Con los valores obtenidos por cuartiles para la recencia, frecuencia y valor monetario será utilizado para la implementación del RFM Score.

- **RFM Score**

  El RFM Score (Recency, Frequency, Monetary) es una métrica utilizada en marketing y análisis de clientes para evaluar y clasificar a los clientes en base al comportamiento en las compras que han realizado.

  El RFM score según nos indica (Team, 2021) asigna puntuaciones a cada uno de estos componentes (R, F y M) para cada cliente, y luego combina estas puntuaciones para generar un puntaje global que representa el valor o la importancia de ese cliente. Por lo general, se utiliza una escala del 1 al 5 (o del 1 al 10) para asignar puntuaciones a cada componente, donde una puntuación más alta indica un mayor valor o compromiso, a continuación, se presenta la formula del RFM score.

  ***RFM Score* =** (Recencia) + (Frecuencia)+ (Valor Monetario)

El valor obtenido anteriormente permitirá adquirir una segmentación rentable de los clientes como muestra (Glotzer, 2022) , a continuación, se presentan las mismas:

- **Core - Tus mejores clientes:** clientes altamente comprometidos, compradores finales más altos y mayores ingresos, puntuación RFM Score: 111.

- **Leales - Tus clientes más leales:** los clientes que compras con bastante frecuencia en los puntos de venta, puntuación RFM Score: X1X.
- **Ballenas - Clientes que más gastan:** Aquellos clientes quienes han generado más ingresos en los puntos de venta, puntuación RFM Score: XX1.
- **Prometedores - Clientes fieles:** clientes que regresan después de un periodo, pero no gastan mucho, puntuación RFM Score: X13, X14.
- **Novatos - Tus nuevos clientes:** clientes que realizaron su primera compra en los puntos de venta, puntuación RFM Score: 14X
- **Durmientes- Una vez leales, ahora ya no:** clientes que realizaban compras frecuentes en los puntos de venta, pero dejaron de hacerlo, puntuación RFM Score: 43X, 44X.

2.3. ¿Qué es Machine Learning?

Para explicar por qué el aprendizaje automático o su traducción en inglés (Machine Learning) es tan importante de nuestro desarrollo global, hay que volver a sus orígenes y básicamente profundizar en ellos ya que dicha herramienta deriva de la inteligencia artificial.

En el año de 1945, se creó una de las primeras computadoras de tipo electrónico de uso general. Se trataba de un dispositivo digital completamente basado en los principios de Turing, capaz de abordar diversos problemas de índole numéricos mediante su reprogramación. Esta computadora recibió el nombre de ENIAC (Electronic Numerical Integrator and Calculator). A pesar de que ENIAC era una máquina diseñada para llevar a cabo cálculos numéricos bastante intensivos, su desarrollo se fundamentó en la idea de construir alguna máquina que pudiera similar el razonamiento y pensamiento humano.

Según (SRINIVAS BANGALORE, 2017) en la década de 1950, cuando los científicos informáticos comenzaron a trabajar en algoritmos y métodos que permitían a las máquinas aprender por sí mismas y mejorar su desempeño en algunas de las tareas más determinadas, Alan Turing creó

el Test de Turing para con ello determinar si una computadora es capaz de ser verdaderamente inteligente. Para pasar la prueba, la computadora debe engañar a otra persona para que crea que también es humana.

En 1956, el matemático e informático estadounidense Samuel Arthur desarrolló el primer programa de aprendizaje automático que podía jugar al ajedrez y mejorar los resultados al jugar.

En la década de 1960, el señor John McCarthy informático estadounidense recalcó el término de inteligencia artificial para referirse al campo del aprendizaje automático.

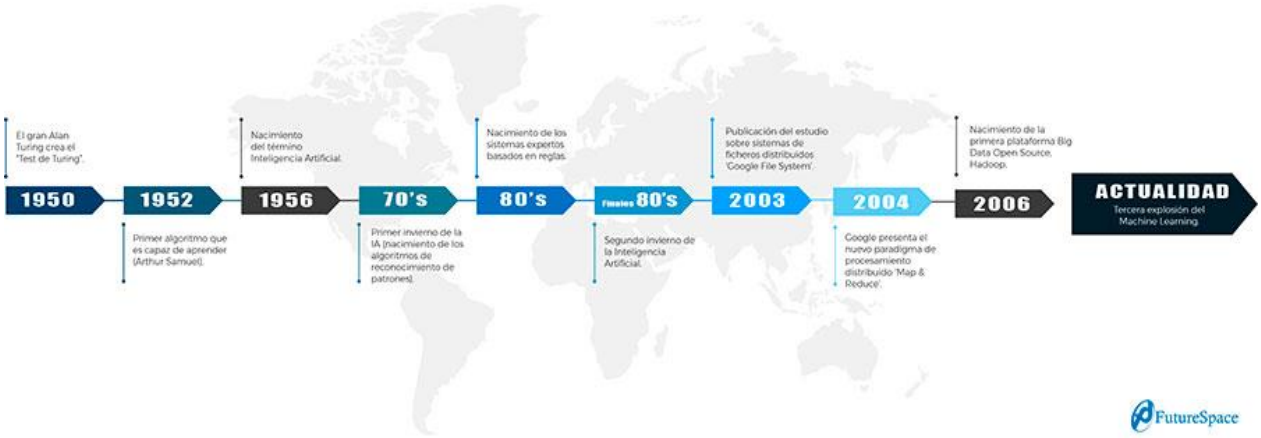
Durante las décadas de 1970 y 1980, el aprendizaje automático se centró en desarrollar algoritmos de aprendizaje supervisados y no supervisados. Los algoritmos supervisados se basan en entradas etiquetadas para aprender a predecir o clasificar, mientras que los algoritmos no supervisados se basan en entradas no etiquetadas para aprender patrones y estructuras.

En la década de 1990, el aprendizaje automático se centró en el aprendizaje por refuerzo, en el que una máquina aprende a partir de la retroalimentación de su entorno. También se han desarrollado métodos de aprendizaje profundo que permiten que las máquinas aprendan de formas más complejas.

Durante la última década, el aprendizaje automático ha crecido enormemente gracias a la disponibilidad masiva de los datos y el poder de procesamiento de cómputo en la nube. Se han desarrollado nuevos algoritmos y técnicas, como las redes neuronales profundas, que han logrado avances significativos en áreas como es la visión artificial y también procesamiento de lenguaje natural.

Figura 4

Gráfico Evolución Machine Learning



Nota. Adaptado de Machine Learning: Los orígenes y la evolución [Fotografía], por Nalda, V. ,2021, Future Space S.A. (https://www.futurespace.es/machine-learning-los-origenes-y-la-evolucion/). CC BY 2.0

Basándonos en lo mencionado previamente, el aprendizaje automático conocido en su siglas como ML, es una rama de la inteligencia artificial la cual se enfoca especialmente en el desarrollo de los algoritmos que habilitan a las computadoras para aprender a partir de datos y experiencias previas.

Otra manera de definirlo, tal como expresó Arthur Samuel en 1959, es que el aprendizaje automático es la capacidad de una máquina para aprender de forma automática a partir de datos, mejorar su rendimiento basado en las experiencias y realizar predicciones sin necesidad de ser programada explícitamente.

El aprendizaje automático reúne al campo informático y la estadística para la creación de los modelos de predicción donde generan o utilizan los algoritmos basados en datos históricos. Cuanto más ofrecemos información, mayor es el beneficio, porque se entiende que la máquina puede llegar a tener capacidad para mejorar la utilidad y conseguir un aumento de datos.

2.4. Clases de Algoritmos Machine Learning

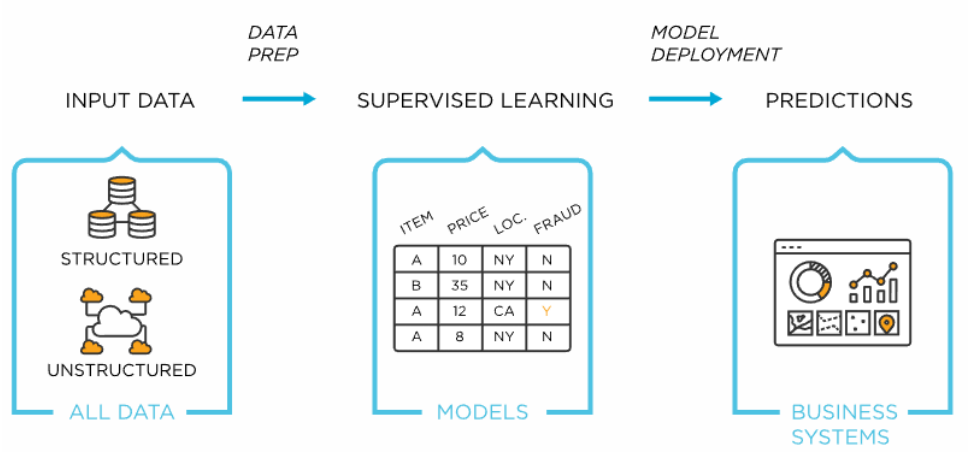
Dentro de este amplio campo, hay diversas modalidades de aprendizaje que se dedican a distintos procesos. El desarrollo sobre el aprendizaje automático se concentra sobre tres categorías fundamentales conocidas como son: aprendizaje supervisado, aprendizaje no supervisado y aprendizaje profundo. Estas categorías definen distintos enfoques dentro del ámbito del aprendizaje automático.

2.4.1. Aprendizaje Supervisado

En cuanto al aprendizaje supervisado podemos decir que se basa sobre el modelo particular de aprendizaje automático (ML) sobre la generación del conocimiento se elabora un grupo de ejemplos o de datos etiquetados, en donde la resolución de las operaciones se conoce previamente. Este tipo de modelo asimila de dichos resultados e implica el ajuste de los parámetros internos para acomodar los nuevos datos que se ingresan en el sistema tal cual se observa en la figura 5.

Figura 5

Aprendizaje supervisado



Nota. Adaptado de ¿Qué es el aprendizaje supervisado? [Fotografía], TIBCO,2023, TIBCO Software. (https://www.tibco.com/es/reference-center/what-is-supervised-learning). CC BY 2.0

Dicho algoritmo se procede a entrenar con los datos históricos y así aprender a asignar sobre la etiqueta de salida apropiada a un valor nuevo, otras palabras este puede predecir la salida con su respectivo valor (Simeone, 2018).

Según el tipo de etiqueta, existen dos tipos de modelos de aprendizaje supervisado:

- **Clasificación:** Los modelos generan una etiqueta de forma discreta, esto nos quiere decir que una etiqueta seleccionada de un conjunto específico de opciones. Además, los modelos de clasificación pueden llegar también a binarios cuando se requiere predecir entre dos etiquetas o clases (como si se estuviera enfermo o su vez no, clasificar correos electrónicos como si fueran correo basura o también como no correo basura), de igual forma pueden ser multiclase, estos necesitan clasificar dos clases o más (como la clasificación de imágenes sobre animales o de análisis de sentimientos, entre otros ejemplos).
- **Regresión:** El propósito de la regresión es predecir un valor numérico continuo para la muestra de entrada, el algoritmo de regresión utiliza datos con etiquetas y aprenden sobre patrones en esos datos que llegaran a ser utilizados para predecir en base a la variable de salida. Algunos ejemplos de algoritmos de regresión son la regresión lineal, los árboles de regresión y las redes neuronales.

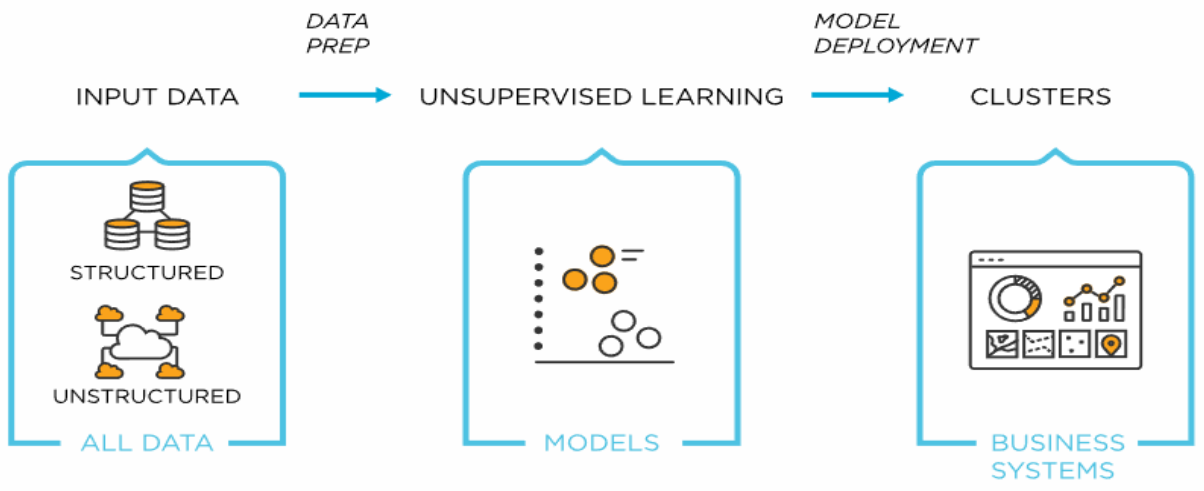
2.4.2. Aprendizaje No Supervisado

El modelo de aprendizaje no supervisado se da con datos etiquetados cuando no están disponibles en durante el entrenamiento. Ya que se conoce la entrada, pero ninguna salida correspondiente a la entrada dada. Por tal motivo solo se puede representar la estructura de

los datos, tratando de buscar alguna distribución que facilite el análisis. Así que son exploratorios en la naturaleza ver figura 6.

Figura 6

Aprendizaje no Supervisado



Nota. Adaptado de ¿Qué es el aprendizaje no supervisado? [Fotografía], TIBCO,2023, TIBCO Software. (https://www.tibco.com/es/reference-center/what-is-unsupervised-learning). CC BY 2.0

Este enfoque de entrenamiento puede aplicarse para reducir o simplificar el tamaño de un conjunto de datos. Al agrupar los datos en base a su similitud, el algoritmo determina la métrica de la similitud o la distancia que se utilizará para contrastar los datos. (datos.gob.es, 2020).

Hay dos tipos específicos de este tipo de aprendizaje llamados agrupamiento (clustering) y reducción dimensional:

- **El agrupamiento (clustering):** es un método exploratorio de análisis de los datos que la información se establece en grupos sin un conocimiento previo de su estructura. Este proceso se lleva a cabo con el fin de identificar grupos de datos que posean características de forma similar. Dicho tipo de análisis de los datos se utiliza frecuentemente en las estrategias de marketing, ya que

permite la creación de segmentos que utilizan variables determinadas en su análisis.

- **La reducción dimensional:** se usa para datos muy complejos que requieren más poder de cómputo. Funciona identificando correlaciones entre características que aparecen en un conjunto de datos, reduciendo la redundancia de la información y a su vez reduciendo el tiempo del análisis extrayendo de manera más eficaz la información que se considera más importante.

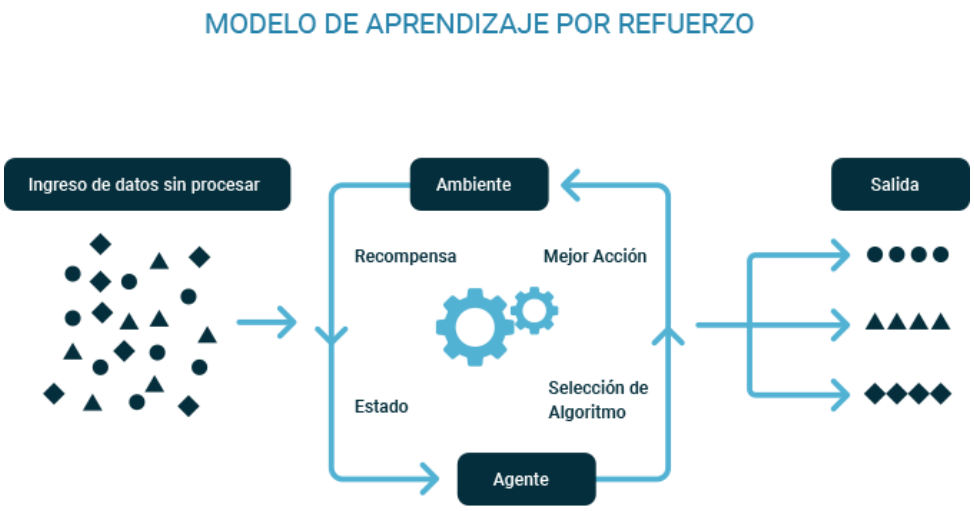
2.4.3. Aprendizaje por Refuerzo

El modelo de aprendizaje por refuerzo se basa en aprender a tomar decisiones en una situación determinada para alcanzar un objetivo específico. Está compuesto por dos componentes: el componente selectivo, que implica seleccionar la mejor operación entre algunas opciones disponibles, y el componente que se denomina combinado, en el cual las alternativas descubiertas son relevantes para contextos específicos en los que se están llevando a cabo en el presente.

El enfoque de aprendizaje por refuerzo resulta adecuado cuando no se dispone de un conocimiento previo sobre el entorno o cuando éste es demasiado complejo para emplear otros métodos, como se ilustra en la figura 7.

Figura 7

Aprendizaje por Refuerzo



Nota. Adaptado de Aprendizaje por Refuerzo [Fotografía], Velazquez N., 2021, Trustnet. (https://trustnet.com.mx/aprendizaje-por-refuerzo/). CC BY 2.0

El propósito del aprendizaje por refuerzo es enseñar al agente (en este caso, el software) a tomar decisiones en el entorno.

Como se puede ver en la figura anterior, sus elementos son:

- **Agente:** Software entrenado para adquirir conocimiento.
- **Ambiente:** El escenario en el que el agente se encuentra y realiza acciones.
- **Acciones:** Decisiones o acciones tomadas por un agente que pueden cambiar el estado del medio ambiente.
- **Recompensa:** Como se mencionó anteriormente, este es un estímulo reforzador positivo a través del cual se logrará su aprendizaje y reproducción.

En resumen, el objetivo del aprendizaje por refuerzo es desarrollar un comportamiento que conduzca a la resolución óptima de problemas. El comportamiento se refiere a un conjunto de labores tomadas para abordar y solucionar problemas, basándose en la experiencia adquirida al manejar diversas situaciones y tomar acciones en cada una de ellas.

2.5. Metodología CRISP-DM para la Creación de Modelos ML

Los métodos de ciencia de datos o data analytics, son altamente reconocidos y populares en la actualidad, en realidad surgieron en la década de 1990. En aquel entonces, se acuñó el término KDD con el significado de sus siglas en inglés (Knowledge Discovery in Databases), que se refiere al amplio concepto de descubrimiento de conocimiento sobre las bases de datos. Esta denominación se utilizó para describir el proceso de explorar, analizar y extraer conocimiento valioso de conjuntos de datos.

Al tratar de estandarizar el proceso de descubrimiento de conocimiento, al igual que la ingeniería de software se usa para estandarizar el proceso del desarrollo del software, a fines de la década de 1990 surgieron dos enfoques principales que son: CRISP-DM en inglés (Cross Industry Standard Process for Data Mining) y por otro lado SEMMA con su significado en inglés (Sample, Explore, Modify, Model, and Assess). Ambos definen que tareas se van a realizar en cada etapa que se encuentra descrita en el proceso, determinando las tareas específicas y definir lo que se desea después de cada etapa (Beatriz.Gil, 2019).

Se podría encontrar una similitud evidente entre ellas, CRISP-DM se destaca por su mayor nivel de exhaustividad, ya que considera la aplicación de los resultados en un entorno empresarial. Debido a esta ventaja, CRISP-DM es ampliamente utilizado en la práctica.

CRISP-DM se considera como un modelo para procesamiento de minería de datos el cual describe cómo los expertos en el campo indagan un problema. Para implementar la tecnología en los negocios, se necesita de una metodología.

Estos métodos generalmente se basan en la experiencia personal, así como en los procedimientos estándar más conocidos. Para proyectos de minería de datos, uno de los métodos que más apoyo ha recibido por parte de empresas privadas y organismos gubernamentales es CRISP-DM, como se puede ver en la figura 8, representa el nivel de utilización relacionados con los proyectos basados en el desarrollo de minería de datos que sean relevantes para la investigación realizada. Como se puede ver, CRISP-DM en los últimos años ha disminuido ligeramente, sin embargo, sigue siendo el más utilizado de los diversos métodos que existen.

Figura 8

Grado de utilización de las distintas metodologías de minería de datos

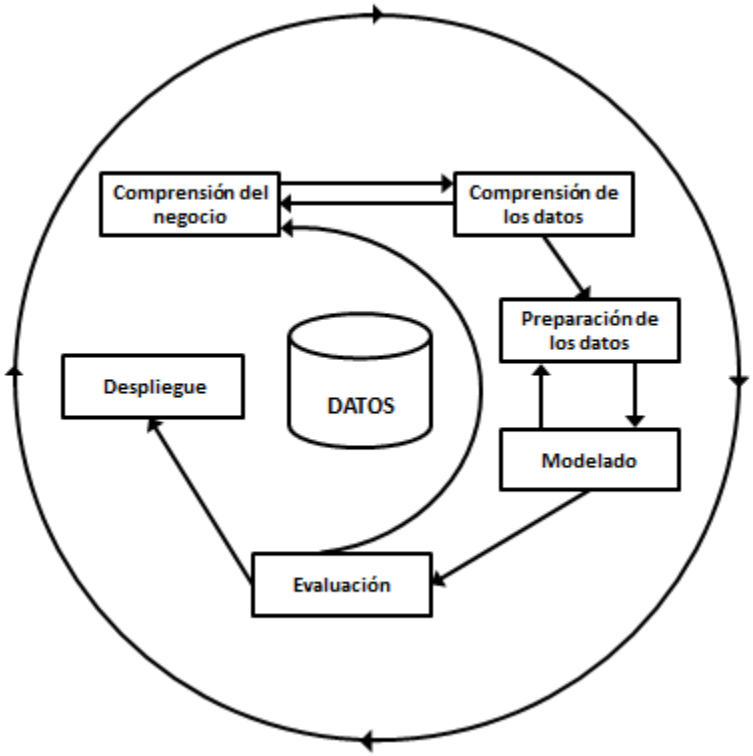
What main methodology are you using for your analytics, data mining, or data science projects ? [200 votes total] 2014 poll 2007 poll	
CRISP-DM (86)	43% 42%
My own (55)	27.5% 19%
SEMMA (17)	8.5% 13%
Other, not domain-specific (16)	8% 4%
KDD Process (15)	7.5% 7.3%
My organizations' (7)	3.5% 5.3%
A domain-specific methodology (4)	2% 4.7%
None (0)	0% 4.7%

Nota. Adaptado de CRISP-DM, still the top methodology for analytics, data mining, or data science projects [Fotografía], Gregory Piatetsky, 2014, KDNudgets. (https://www.kdnuggets.com/2014/10/crisp-dm-top-methodology-analytics-data-mining-data-science-projects.html). CC BY 2.0

CRISP-DM se basa sobre un manual dividido en seis pasos, algunos de los cuales van en ambas direcciones, lo cual significa que desde un determinado paso puede volver al paso anterior para revisarlo, por lo que el orden de los mismos no importa de principio a fin. En la figura 9 se muestra las fases por las que se divide CRISP-DM y las posibles secuencias entre ellos.

Figura 9

Fases de la Metodología CRISP-DM



Nota. Adaptado de Fases componentes de la metodología CRISP-DM [Fotografía], Federico Carlos Peralta, 2014, ResearchGate. (https://www.researchgate.net/figure/Fases-componentes-de-la-metodologia-CRISP-DM-7_fig12_284215308) . CC BY 2.0

El círculo exterior de la figura representa el carácter cíclico sobre proyectos del análisis de los datos. Pero, según este método, el plan no termina después de implementar la solución, sino que pasa por el proceso de verificación y seguimiento de los resultados.

CRISP-DM aborda el análisis de los datos a través de un plan profesional, lo cual implica crear un contexto más amplio que influye en el desarrollo del modelo. Este enfoque considera aspectos como la presencia de un cliente externo al equipo de desarrollo y la comprensión de que el proyecto no concluye una vez se encuentra el modelo perfecto, ya que posteriormente se requiere su implementación y mantenimiento. Además, se reconoce que los resultados y conocimientos obtenidos son relevantes para otros proyectos, por lo que deben documentarse de manera minuciosa para que otros equipos de desarrollo puedan utilizarlos y beneficiarse de ellos.

2.5.1. Fases de la Metodología CRISP-DM

I. Comprensión del negocio:

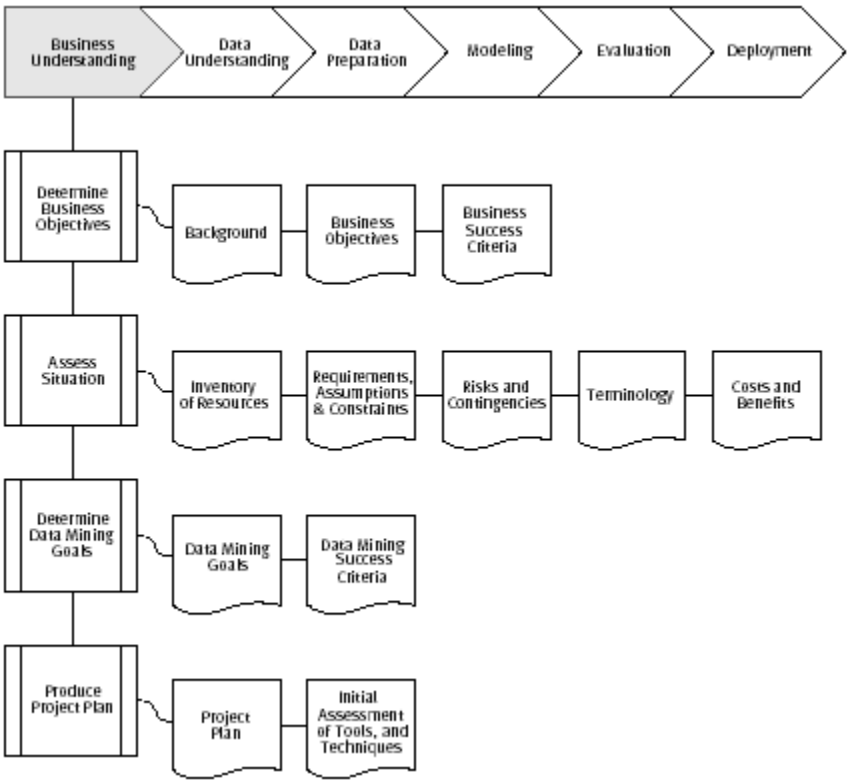
En esta fase el propósito principal será alinear los objetivos del proyecto de la minería de datos junto los objetivos comerciales. De esta forma, se trata de evitar iniciar algun proyecto que trate sobre minería de datos que no pueda llegar a tener un impacto verdadero dentro de la organización. Aquí se debe obtener lo siguiente:

- Perspectiva del negocio
- Establecer objetivos del negocio.
- Se evalúa la situación actual.
- Se establecen los objetivos de la minería de datos.
- Se crea un plan sobre el proyecto

Una vez que se establecen y acuerdan aquellos objetivos, ahora hay que crear un plan para el proyecto que incluye los hitos, tareas y las actividades que se necesitan para lograr dichos objetivos en la figura 10 se presenta el esquema descrito.

Figura 10

Comprensión del negocio



Nota. Adaptado de Metodología CRISP-DM para minería de datos [Fotografía], Pete Chapman, 2007, Dataprix. (https://www.dataprix.com/es/book/export/html/107). CC BY 2.0

En el caso de que no sea hecha esta fase de la manera correcta, el resto del proyecto puede ser invalidado a futuro. Por eso, es necesario que todos estén informados y totalmente alineados.

II. Comprensión de los datos:

Luego de iniciar con la primera fase, se puede empezar a pensar en los datos que se utilizarán en el proceso. Para lograr esto, podemos hacer algunas preguntas, por ejemplo: "¿La empresa tiene una base de datos? ¿Cómo estarán disponibles los datos? ¿Cuántas fuentes de datos se utilizarán? ¿Cuál será el formato de datos? ¿Están estructurados los datos? De

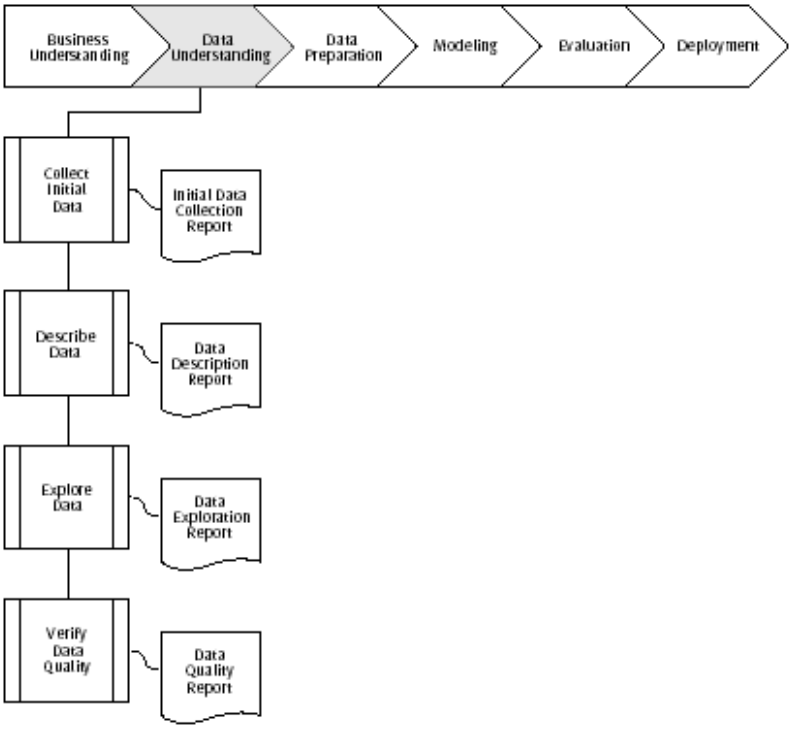
ellos se recopilan datos y es necesario asegurarse de que no se omita ninguna información importante. En esta fase se obtendrá:

- Ejecutar los procesos sobre recolección de datos.
- Proporcionar descripción del conjunto de datos.
- Realizar las actividades exploratorias de los datos.
- Gestionar procesos de calidad sobre los datos, detectando inconvenientes y ofreciendo soluciones.

Para extraer datos de manera más efectiva, se debe comprender completamente cual problema se intenta resolver, ya que permitirá recopilar datos exactos y con ello interpretar con precisión los resultados, a continuación, en la figura 11 verificamos el esquema.

Figura 11

Comprensión de los datos



Nota. Adaptado de Metodología CRISP-DM para minería de datos [Fotografía], Pete Chapman, 2007, Dataprix. (https://www.dataprix.com/es/book/export/html/107). CC BY 2.0

Durante esta fase, es fundamental comprender cómo aplicar los conceptos y requerimientos del negocio a la tarea de extracción de información a partir de los datos, y desarrollar un plan estratégico que guíe el proceso de minería de datos hacia el logro de dichos objetivos comerciales.

III. Preparación de los datos:

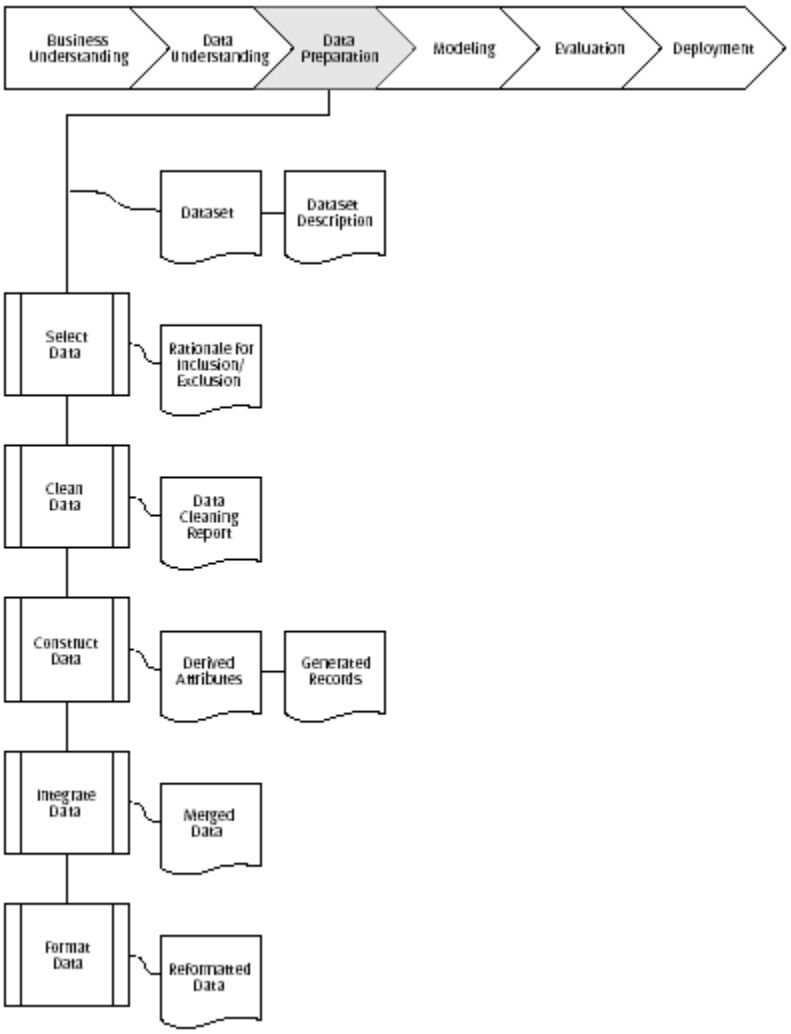
En esta fase se incluye la selección, limpieza y creación de conjuntos de datos apropiados, que se organizan y preparan para la fase de modelado. Este es un paso extremadamente importante para el proyecto de minería de datos. Aquí en esta fase deberemos tratar los siguientes pasos:

- Se establece el conjunto de los datos con los que se trabajara.
- Se realizan actividades sobre la limpieza de los datos.
- Se construye un conjunto de los datos idóneo que será usado con los modelos de la minería de datos.
- Integración de los datos de varias fuentes si el caso amerita.
- Cambio en el formato de los datos si lo amerita.

Los errores de datos faltantes que no se corrigen aquí serán trasladados a la fase de modelado, esto dará como resultado una menor precisión del modelo o incluso proporciona a los clientes resultados basados en datos que todavía contienen errores, la figura 12 muestra el gráfico de esta fase.

Figura 12

Preparación de los datos



Nota. Adaptado de Metodología CRISP-DM para minería de datos [Fotografía], Pete Chapman, 2007, Dataprix. (https://www.dataprix.com/es/book/export/html/107). CC BY 2.0

Esta etapa adquiere una gran importancia para el proyecto, ya que una mala interpretación de los datos puede resultar en un incremento del tiempo total al proyecto y disminuir la probabilidad de éxito. Por lo tanto, es crucial abordar esta fase con precisión y atención para garantizar un proceso fluido y exitoso.

IV. Modelado:

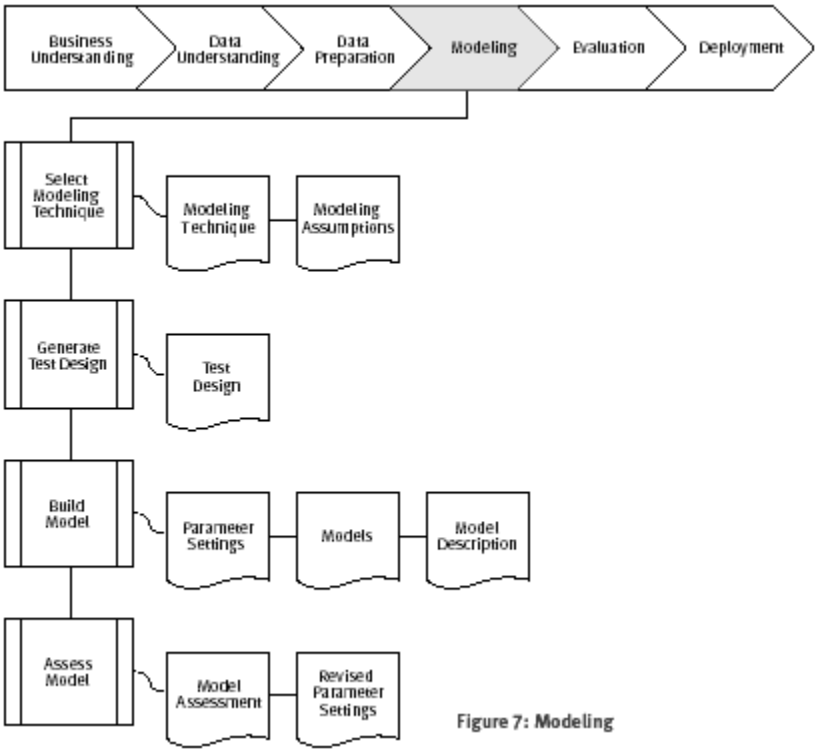
Durante esta fase, se generan modelos de conocimiento basados en los datos obtenidos en la etapa previa. Dichos modelos pueden adoptar diferentes formas, como modelos de clasificación o de regresión, que permiten estimar o a su vez derivar en el valor de una variable específica. En esta etapa, se abordan los siguientes aspectos:

- Elegir los métodos de modelado que mejor se adapten a nuestros objetivos y al conjunto de los datos.
- Establecer una metodología al modelo para el control de calidad.
- Crear un modelo basado en la aplicación de los métodos seleccionados al conjunto de datos.
- Ajustar mediante su evaluación el modelo en base a su solidez y a su vez el impacto sobre los objetivos predeterminados.

El tipo de modelo utilizado suele determinarse a partir de las necesidades que tiene el negocio y del tipo de variables que se analizan. Al determinar qué modelo usar, es importante especificar qué atributos serán las variables al construir ese modelo. En este caso, puede ser muy útil volver a la primera etapa para probar objetivos y encontrar nuevas oportunidades, en la figura 13 observamos la estructura de esta fase.

Figura 13

Modelado



Nota. Adaptado de Metodología CRISP-DM para minería de datos [Fotografía], Pete Chapman, 2007, Dataprix. (https://www.dataprix.com/es/book/export/html/107). CC BY 2.0

Algunos métodos requieren condiciones particulares en cuanto a la geometría de los datos. Por esta razón, es común volver a la fase de la preparación de los datos sobre el proyecto en el que se esté aplicando. Esta etapa se repite con frecuencia para garantizar que los datos estén adecuadamente estructurados y sean compatibles con los algoritmos y técnicas utilizados en el análisis.

V. Evaluación:

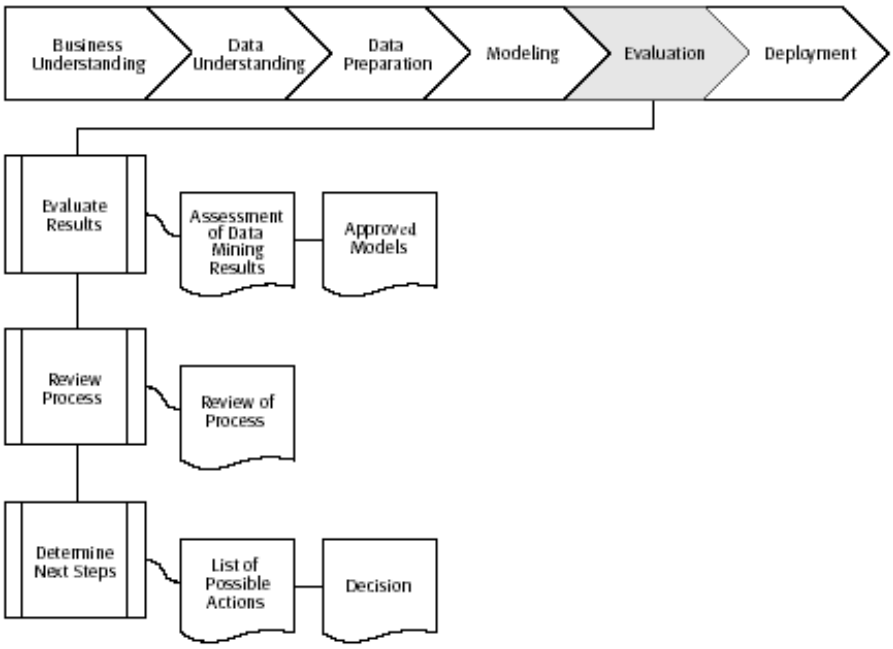
Culminadas las fases anteriores, el modelo se evalúa frente a los criterios de éxito de la tarea. También es importante tener en cuenta la confianza ya que esta se calcula para el modelo y solo es aplicada a los datos para los cuales el análisis se realiza. Este proceso debe validarse en base a los resultados para que se puedan repetir los pasos anteriores que podrían haber salido mal. En esta fase se deberá realizar los siguientes puntos:

- Se evalúa el modelo o los modelos que se han generado hasta esta instancia.
- Se revisa por completo el proceso de la minería de datos con la que nos encontramos a este punto.
- Se establece cuales serán pasos siguientes a ejecutar, en el caso de que se tengan que repetir fases anteriormente realizadas o a su vez incluir nuevos temas de investigación.

En base a los modelos resultantes si dichos están acorde a los objetivos comerciales, se acepta el modelo. Si no, este paso evalúa si repetimos los pasos anteriores nuevamente para encontrar nuevos resultados, la figura 14 observamos lo explicado en este punto.

Figura 14

Evaluación



Nota. Adaptado de Metodología CRISP-DM para minería de datos [Fotografía], Pete Chapman, 2007, Dataprix. (https://www.dataprix.com/es/book/export/html/107). CC BY 2.0

Si luego de evaluar el modelo y la precisión no es suficiente para lograr los objetivos sobre el proyecto, realizaremos una adecuación al mismo de forma iterativa con las fases anteriores para obtener un mejor modelo.

VI. Despliegue:

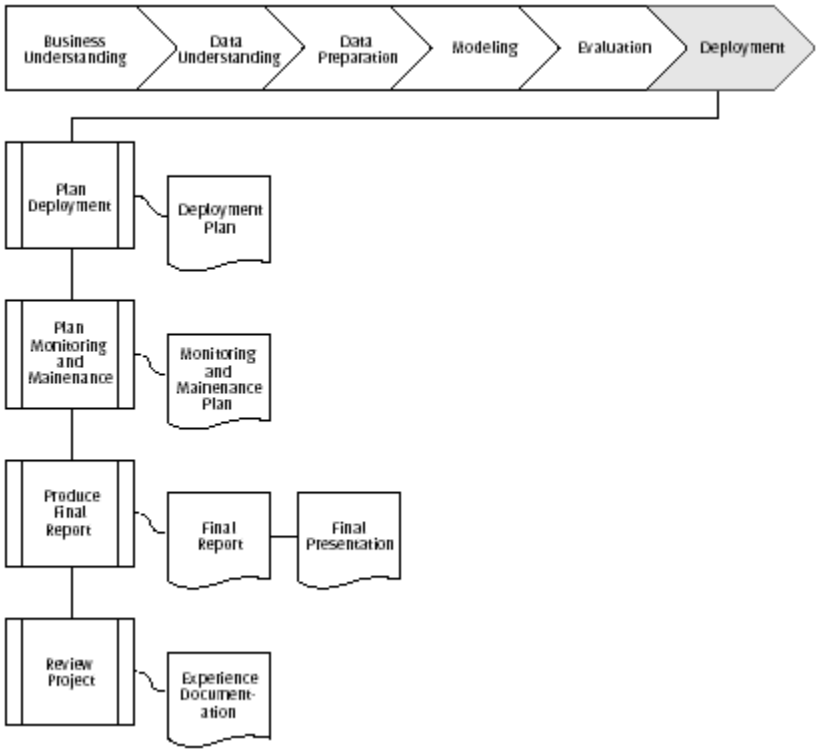
En este paso final, se determinan las estrategias de los modelos para implementar, monitorear y mantenerlos. Con lo anterior facilitara a monitorear el procedimiento anormal del modelo y poder corregirlo evitando que cause anomalías al servicio de algún cliente. En esta fase desarrollaremos los siguientes pasos:

- Se diseña el plan del despliegue del modelo seleccionado en la organización.
- Se realizará un mantenimiento y seguimiento operativo sobre el despliegue del modelo.
- Se revisará en su globalidad el proyecto esto con la finalidad de comprender y documentar las lecciones que hemos obtenido.

Si el procedimiento se realiza correctamente, este será el último paso. Aquí, el modelo debe ponerse en producción para agregar valor al negocio. La forma de hacerlo varía mucho según el tipo de modelo y el diseño, en la figura 15 podemos revisar los pasos de esta fase.

Figura 15

Despliegue



Nota. Adaptado de Metodología CRISP-DM para minería de datos [Fotografía], Pete Chapman, 2007, Dataprix. (https://www.dataprix.com/es/book/export/html/107). CC BY 2.0

Esta etapa de revisión final nos permite reflexionar sobre el proyecto en su conjunto, analizar los resultados obtenidos, identificar las fortalezas y debilidades del enfoque utilizado, y extraer conclusiones y recomendaciones para proyectos futuros.

2.6. Fundamentos de programación en Phyton

Python es un lenguaje muy popular para desarrollar aplicaciones de aprendizaje automático debido a sus múltiples librerías para este propósito y su sintaxis simple. Este lenguaje no compilado se considera uno de los mejores para tareas de investigación o prueba debido a su facilidad para realizar cambios en el programa sin gastar demasiado tiempo en nuevas compilaciones, ver figura 16 logo Python.

Figura 16

Logo Python

Nota. Adaptado de Python *Logo, symbol, meaning, history* [Fotografía], Python Org, 2023, Logos-World. (https://logos-world.net/python-logo/). CC BY 2.0

La generalización de los grandes datos de los últimos tiempos, a su vez por el auge de la inteligencia artificial, aprendizaje automático, aprendizaje profundo y la fusión en la ciencia de los datos como un nuevo campo interno, ha revolucionado el panorama en la actualidad.

Dado que Python se utiliza muy ampliamente, como un lenguaje de referencia en las instituciones educativas, su estudio ha llevado a ser necesario sin discusión en las mismas. Como consecuencia de esto ha derivado en la aparición de muchas herramientas en el campo como menciona (Robledano, 2022) basadas en el lenguaje y utilizadas tanto por personal de ingeniería que trabajan con los datos como por los especial alistas que estudian las ciencias de los datos. Entre los principales podemos encontrar a: PySpark para uso en big data o Pandas, NumPy, Matplotlib o a su vez Jupyter utilizado comúnmente en el desarrollo de ciencia de datos.

Un programador de aprendizaje automático tiene que realizar muchas tareas diferentes para las que utiliza diferentes librerías, después de analizar los datos dentro de Python, se procede a aplicar métodos de aprendizaje automático, en donde encontramos a Scipy y Sklearn, dos librerías que implementan una gran cantidad de algoritmos y permiten usarlos de manera simple. Finalmente, si la solución requiere el uso de métodos de aprendizaje profundo, se puede usar la librería Tensorflow, donde se desarrollará nuestros modelos basados en redes neuronales.

La librería Sklearn ha implementado algoritmos de clasificación, regresión y agrupamiento como son: la regresión lineal o logística, el Support Vector Machines, Vecinos Cercanos (K Nearest Neighbors), los Procesos Gaussianos, el clasificador Naive Bayes, los árboles de decisión, PCA y los modelos de ensamblaje.

Los siguientes códigos nos van a permiten ingresar los parámetros de la regresión, adicionalmente el paquete Scikit-learn permite dividir los datos de aprendizaje y prueba dentro de Python:

```
from sklearn.linear_model import LinearRegression

regressor=LinearRegression()

regressor=regressor.fit(X_train, y_train)
```

Con el código anterior programado en Python y previamente cargado los datos correspondientes el modelo puede será aplicado mediante la regresión lineal.

De igual forma cuando se desea aplicar la regresión logística, se usa la clase LogisticRegression del módulo sklearn.linear_model como un clasificador regular y lo entrenamos en datos limpios, se divide los datos para entrenamiento y prueba, en el código que utilizaremos en Python, por lo tanto sería así:

```
from sklearn.linear_model import LogisticRegression

clf = LogisticRegression()

clf.fit(X_train,y_train)

y_pred = clf.predict(X_test)
```

2.7. Modelos de Predicción

Podemos definir a los modelos de predicción aquellos que se basan en técnicas, en el campo del machine learning, la recopilación histórica de los datos junto con big data, apalancándose en el reconociendo de los patrones, con el fin de predecir resultados futuros, y obtener mejores resultados al momento de tomar una decisión utilizando herramientas para realizar análisis de los datos. La última década el campo de la predicción ha desempeñado un papel destacado en las industrias, la salud, en servicios financieros, así como políticas públicas, publicidad y marketing, redes sociales, y muchas otras aplicaciones.

La predicción que nos brinda el modelo se utiliza para poder identificar a tiempo oportunidades de mejora. Es muy utilizado por empresas a nivel de logística, previsión económica y reportes de ventas para saber dónde pueden estar los errores o cuáles pueden ser las ventas futuras. Por ejemplo, otro uso común es en marketing ya que nos ayuda a pronosticar el comportamiento que tendrán los consumidores en que realicen una compra debido a sus acciones pasadas.

Algunas de las técnicas más aplicadas en los Modelos Predictivos son las siguientes:

- Arboles de decisión
- Regresión Lineal y logística
- Redes neuronales
- Análisis bayesiano
- Series temporales y Datamining
- Máquinas de vectores de soporte
- K Nearest Neighbors
- Gradient boosting
- Repuesta incremental
- Razonamiento con base en la memoria
- Regresión de mínimos cuadrados parciales

Según nuestros objetivos de la investigación, la técnica a ser implementada podría variar.

Para el presente trabajo utilizaremos Regresión Lineal, Regresión Logística y Arboles de Decisión para el desarrollo de nuestros modelos a continuación una explicación de cada uno.

2.7.1. Regresión Lineal

Es un método que sirve para analizar datos que permite predecir un valor no conocido utilizando otro valor conocido asociado. Se establece una relación matemática entre la variable dependiente, que es la variable desconocida, y la variable independiente, que es la variable conocida, mediante una ecuación de carácter lineal. Como ejemplo, consideremos un conjunto de datos que incluye gastos y los ingresos del último año. Mediante la regresión lineal, se analizan dichos datos y se determina que los gastos representan la mitad de los ingresos. Utilizando esta relación, se pueden calcular los costos futuros desconocidos dividiendo los ingresos futuros conocidos por la mitad.

Este modelo lineal permite analizar y predecir cómo la variable dependiente *Y* varía en función de las variables independientes *Xi*, facilitando así la comprensión y el estudio de las relaciones entre las variables involucradas. El modelo de regresión lineal se representa mediante una ecuación que se muestra en la figura 17, donde se establece la forma matemática de la relación entre las variables.

Figura 17

Fórmula Regresión Lineal

$$Y = \beta_0 + \sum \beta_i X_i + \epsilon_i$$

Nota. Adaptado de ¿Qué es la regresión lineal? [Fotografía], MATLAB & Simulink, 2023, MathWorks. (https://la.mathworks.com/discovery/linear-regression.html). CC BY 2.0

En la ecuación mencionada, el símbolo β simboliza estimaciones de parámetros lineales que necesitan ser calculados, mientras que el término $\mathcal{E}$ representa los errores residuales.

Las clases que más se usan en la regresión lineal son las siguientes:

- **Modelo de regresión lineal simple:** Para (Ortega, 2023) la regresión lineal simple es una de las técnicas que más se utiliza para modelar una relación de dos o más variables. Como resultado se obtiene una ecuación y esta puede ser útil para predecir o evaluar los datos. En este modelo el predictor es considerado la variable x y Y como variable de respuesta, ahora observamos en la siguiente ecuación el cálculo del modelo:

$$\mathbf{Y=\beta_0+\beta_iX+\epsilon_i}$$

En la ecuación dada, los coeficientes desconocidos de la regresión son la ordenada al origen β0 y la pendiente βi.

- **Modelo de regresión lineal múltiple:** La regresión lineal múltiple es un enfoque estadístico utilizado para analizar escenarios que involucran múltiples variables. Esta técnica permite determinar qué variables independientes pueden influir en la variable dependiente, evaluar la relación causal entre ellas y realizar predicciones aproximadas de valores. (Ortega, 2023).

  La ecuación del modelo en cuestión se puede describir de la siguiente manera:

$$Y = \beta_0 + \beta 1X1 + \beta 2X2 + \ldots + \beta nXn + \varepsilon$$

  En la ecuación proporcionada, *Y* representa una variable dependiente, β representa los estimadores correspondientes y **ε** representa el residuo o error.

- **Modelo de regresión no lineal:** en este modelo el proceso es más complejo ya que la cantidad de los parámetros iniciales puede a su vez no coincidir en base a la cantidad de regresores, la siguiente función expresa el cálculo del modelo no lineal:

$$Y = \alpha X\beta$$

  La variables originales en algunos casos se pueden transformar en una función de carácter no lineal, y luego en una función lineal, de tal forma que se puedan emplear estos métodos. Entonces, si no hay linealidad se aplica solo a variables interpretativas mas no a sus factores, entonces se pueden determinar otras variables.

La regresión lineal también se puede utilizar para obtener información mediante la identificación de patrones y relaciones que compañeros de trabajo pueden haber visto y pensaron que entendían. Como nos muestra (IBM, 2023) Por ejemplo, analizar los datos de compras y ventas puede ayudar a descubrir patrones de compra específicos en ciertos días u horas. Dicha información que se obtiene del análisis de la regresión puede ayudar a líderes empresariales a predecir cuándo habrá una gran demanda de productos.

2.7.2. Regresión Logística

La regresión logística como menciona (Software, s.f.) se basa como método estadístico utilizado en la estimación de la probabilidad de ocurrencia en un evento específico. Este modelo analiza la relación entre las características o variables predictoras y calcular la probabilidad de obtener algún resultado particular.

Es una técnica utilizada en el aprendizaje automático para realizar predicciones efectivas. A diferencia de la regresión lineal, la regresión logística se utiliza cuando la variable de tipo objetivo es de carácter binaria, es decir, toma valores entre 1 o 0. En este contexto, existen dos tipos de variables: la variable explicativa o característica, que es la variable que se mide, y la variable objetivo binaria o de respuesta, que representa el resultado que se desea predecir.

A modo de ejemplo, cuando se trata de predecir si un estudiante aprobará o reprobará un examen, la característica es la cantidad de horas estudiadas y la variable objetivo a predecir obtendrá dos valores que son: aprobar o reprobar.

El logit como modelo predictivo también nos sirve para determinarla relación que existe entre la probabilidad de tener éxito o fracasar o la probabilidad de registro. Por ejemplo, si juega al póquer con amigos y gana 4 de 10 manos, sus probabilidades de ganar

son 4 de 6 o 4 de 6, que es su proporción de éxito y fracaso. De lo contrario, la posibilidad de ganar es de cuatro en diez.

Matemáticamente, sus probabilidades son probabilísticamente p/(1 - p) y sus probabilidades logarítmicas son (p/[1 - p]). La ecuación logística podemos definirla como una probabilidad logarítmica como se muestra a continuación:

$$\textit{Logit Function} = \log(p/1\text{-}p)$$

Existen tres clases principales en la regresión logística:

- **Regresión logística binaria:** en esta clase podemos encontrar dos resultados en la respuesta final. En el ejemplo de los estudiantes, los mismos pueden aprobar o reprobar el examen.

- **Regresión logística polinomial:** en este caso, las variables de respuesta incluyen más de tres categorías o tres sin un orden específico. Como ejemplo sería predecir en un restaurante a que clientes les gusta cierta clase de comida: carne, vegana o a su vez vegetariana.

- **Regresión logística convencional:** como en la regresión polinomial, la regresión convencional puede involucrar más de tres o tres variables. pero, en este caso, existe ordenanza para las medidas. Un ejemplo sería dar una calificación a un hotel que va desde 1 al 5, donde cada número representa un nivel de satisfacción creciente.

Regresión logística vs regresión lineal

Para (Amazon Web Services, s.f.) la regresión lineal se utiliza para predecir una variable objetivo continúa utilizando un conjunto de variables predictoras. la variable objetivo podría tener varios valores, pueden ser el precio, la edad, lo que permite a la regresión lineal que prediga de manera real los valores de la variable objetivo. Por lo tanto, se pueden dar solución a respuestas como ¿Cuál pude ser el precio del pan en 5 años?

Por otro lado, la regresión logística es un algoritmo para la clasificación en el que no se pueden predecir los valores reales ya que están basados en datos de tipo continuos. A diferencia de la regresión lineal, la regresión logística se utiliza para responder preguntas de clasificación, por ejemplo: ¿Aumentará el precio del petróleo en un 30% en 5 años?

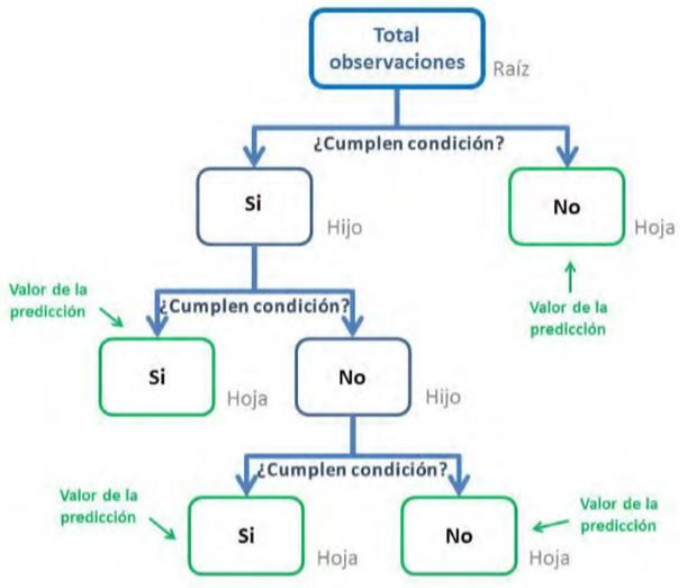
2.7.3. Árboles de Decisión

A los árboles de decisión se les considera un modelo que tiene dos propósitos: el primero como clasificación conocidos como árbol de clasificación y el segundo propósito el de regresión llamado árbol de regresión. Además, son muy utilizados por la sencillez y facilidad de comprensión que proporcionan a la hora de comparar resultados con una hipótesis o pregunta de negocio. Por lo tanto, es mucho más transparente para analizar y tomar decisiones sobre el fenómeno en estudio.

La composición que tiene el árbol de decisión podemos ver gráficamente en la Figura 18, donde el primer cuadrado corresponde al nodo donde se aplica la regla al elemento clasificado, dependiendo de si se cumple la regla, las flechas dirigen el elemento a la taxonomía adecuada o a la nueva regla de taxonomía.

Figura 18

Árbol de Decisión



Nota. Adaptado de Arboles Decisión [Fotografía], Veronica Chavez, 2018, RPubs. (https://rpubs.com/elfenixsoy/arbol-veronica). CC BY 2.0

Según menciona (Ferrero, 2020) el árbol de decisión consta de nodos y se lee de arriba abajo, hay diferentes clases de nodos que conforman un árbol de decisión:

- **Primer o nodo raíz:** crea la división inicial en base al campo principal.
- **Nodos internos o intermedios:** después de crear el primer nodo, ahora se volverán a dividir los nodos en base al grupo de datos por sus variables.
- **Nodos terminales (hojas):** estos nodos están ubicados al final del diagrama, la función es indicar el término de la clasificación.

Un término importante para tomarlo en cuenta es la profundidad que llegue a tener el árbol, ya que viene dado por el número máximo de los nodos en cada rama.

Los árboles de clasificación no representan reglas matemáticas muy complejas y son fáciles de interpretar. Además, es un modelo que permite mostrar y visualizar todas las

decisiones tomadas para lograr los resultados cuando el número de niveles del árbol lo permita.

Por otro lado, es fuerte en el manejo de valores atípicos y funciona muy bien con variables categóricas, en comparación a otro modelos. Sin embargo, una de las limitaciones es que requiere una cantidad significativa de observaciones de bases de datos bien entrenadas y bien ejecutadas como entrada.

2.8. Métricas de Desempeño

Anteriormente pudimos estableces que se evaluarán dos tipos de modelos, es importante definir en términos generales las métricas a utilizar para poder distinguir qué algoritmo tiene mejor desempeño de acuerdo al objetivo propuesto, determinar actividad de clientes activos potenciales a ser fuga y clientes activos. Primero, es necesario asignar una conjunto de datos que servirá al modelo de predicción de fuga para poder entrenarlo.

Para este propósito, la base de datos se dividirá con el enfoque Train-Test Split en un 70-30% para entrenamiento y prueba, respectivamente. El propósito de este paso intermedio antes de evaluar los modelos es poder usar alguna información para entrenar los modelos (70%). Luego probar con información que nunca se ha ingresado en los modelos (30%), probar su rendimiento con estos nuevos datos para que los modelos no recuerden la información que recibieron como entrada y tengan una previsibilidad real.

Después de completar el proceso anterior, los modelos se compararán con los valores del índice de rendimiento. Dado que el problema a ser resuelto en el presente trabajo tiene una clasificación binaria (activo o posible fuga) utilizaremos las siguientes métricas para su evaluación.

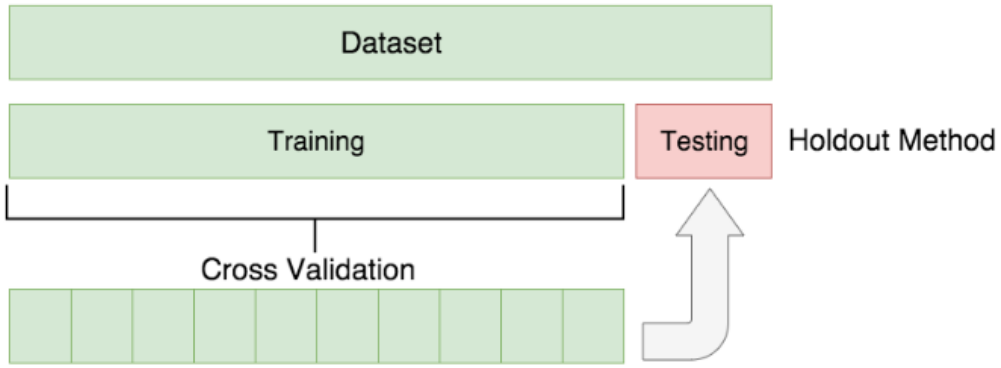
2.8.1. Validación Cruzada (Cross-Validation)

Los modelos predictivos se basan en parámetros estructurales asociados con los factores de ajuste y los parámetros de control de penalización en los datos para generar las mejores predicciones posibles.

El método de validación cruzada permite obtener los mejores parámetros, un modelo predictivo se considera bueno si puede predecir con precisión patrones invisibles. Los diferentes tipos de error de predicción son error de entrenamiento y error de prueba, el primero es la pérdida promedio en la muestra original y el segundo es el error de predicción en las muestras independientes, en la figura 19 podemos observar el método de validación cruzada.

Figura 19

Validación Cruzada



Nota. Adaptado de Árboles de decisión y Random Forest [Fotografía], Johanna Orellana Alvear, 2020, Bookdown. (https://bookdown.org/content/2031/arboles-de-decision-parte-i.html). CC BY 2.0

(Venturini, 2016) nos menciona que los modelos pueden variar según el sesgo y la varianza permitidos, los modelos con menos grados de libertad tendrán menos varianza, pero más sesgo y los modelos con más grados de libertad tendrán más varianza y menos

sesgo. En la actualidad existen 2 técnicas principales de la validación cruzada y son las siguientes:

- **Técnica de prueba dividida de entrenamiento (Train-Test Split):** este método divide aleatoriamente la serie de datos y la divide en dos partes. Los primeros datos, que representan del 70 % al 80 % de la serie, se utilizan para entrenar al modelo de aprendizaje automático, mientras que los segundos datos, que constituyen el 20 % al 30 % restante de los datos, permiten para el control Verificación de tasación.
- **Método K-Folds:** este método es el más fácil de entender y familiar porque es un modelo menos sesgado ya que garantiza todas las observaciones de la serie de datos original. Si su entrada es limitada, este método es ideal.

Para el aprendizaje automático, la validación cruzada es esencial, ya que se utiliza para comparar diferentes modelos y elegir el mejor para un problema en particular.

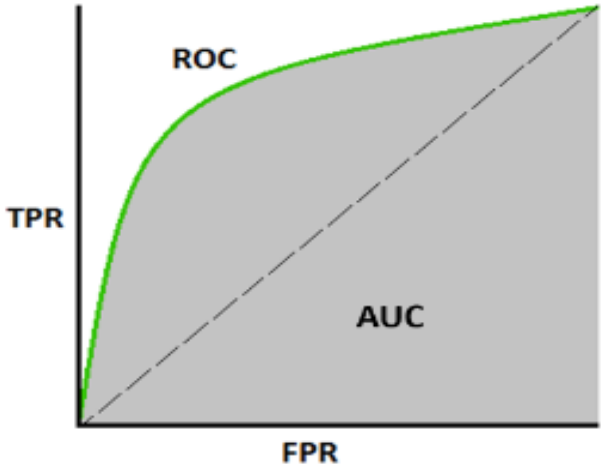
2.8.2. Curva ROC (AUC – ROC)

Otra forma de comparar el rendimiento de los modelos es utilizar la curva AUC - ROC nos ayuda a encontrar una medición en cuanto al rendimiento de clasificación basándose en diferentes niveles. ROC viene a ser la curva de probabilidad y mientras que AUC es el grado o medición de la separación, la cual nos muestra que, para diferentes niveles de umbral de probabilidad de clasificación, la interacción entre la tasa de los valores de verdaderos positivos TPR o también como recuperación versus la tasa de los valores de falsos positivos FPR genera una curva bidireccional que va de 0 a 1 en ambos ejes.

Cuanto mayor sea el AUC, mejor predice el modelo la clase 0 como 0 y la clase 1 como 1. De manera similar, cuanto mayor sea el valor de AUC, mejor será el rendimiento del modelo en cuanto a clasificación, lo dicho se observa en la figura 20.

Figura 20

Curva ROC-AUC



Nota. Adaptado de Understanding AUC - ROC Curve - Towards Data Science [Fotografía], Sarang Narkhede, 2018, Towards Data Science. (https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5). CC BY 2.0

La curva ROC se traza desde TPR versus FPR, donde TPR está en el eje Y y FPR está en el eje X.

Con base a los métodos de evaluación presentados, los modelos construidos se pueden comparar objetivamente para lograr cumplir con el propósito establecido en el presente trabajo.

2.8.3. Matriz de Confusión

Es un concepto fundamental para el ámbito de la inteligencia artificial y también el aprendizaje automático. Básicamente, es un instrumento que nos permite analizar y evaluar el rendimiento del algoritmo de aprendizaje supervisado. Esta matriz se presenta en forma de tabla, donde cada columna nos muestra el número de predicciones realizadas para cada una de las clases, y cada fila muestra el valor real de instancias que pertenecen a cada clase.

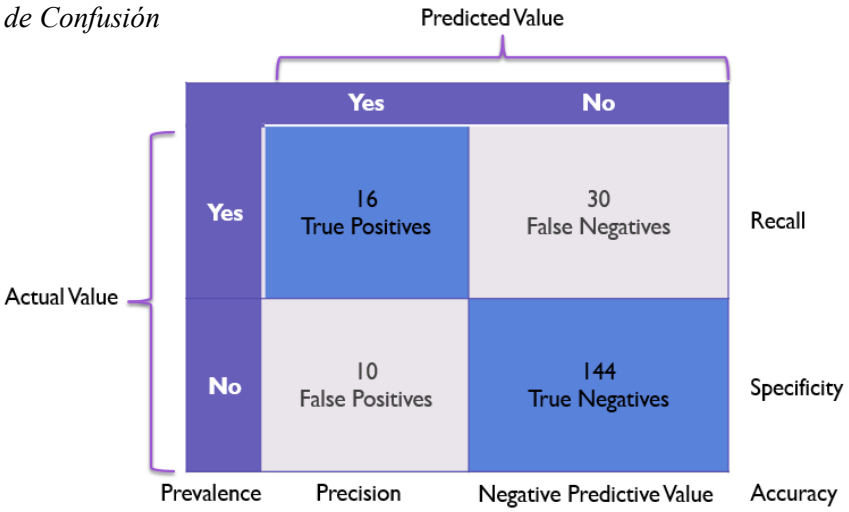
Según menciona (Shin, 2020) la matriz de confusión, es llamada como matriz de error, esta se crea como tabla concisa y se utiliza para evaluar y medir el rendimiento del

modelo de clasificación. Esta tabla genera el número de predicciones de forma correcta e incorrecta a manera de resumen y desglosa estos valores por cada clase.

En la figura 21 se muestra una imagen de la matriz de confusión de tamaño 2x2, a modo de ejemplo, supongamos que hay 16 casos cuando el modelo de clasificación dio como predicción correctamente el valor de SI cuando el valor era también el real SI. En este caso, el número dieciséis se ubicaría en la esquina superior del lado izquierdo del cuadrante de Verdaderos Positivos.

Figura 21

Matriz de Confusión



Nota. Adaptado de Interpreting Confusion Matrixes [Fotografía], Matt Eland, 2022, DataSource.ai. (https://accessibleai.dev/post/interpreting_confusion_matrixes/). CC BY 2.0

Composición de la Matriz de Confusión 2x2

- **Positivo (P):** la observación es catalogada de manera correcta como positiva (un ejemplo, son elefantes).
- **Negativo (N):** la observación es catalogada de manera negativa (ejemplo, no son elefantes).

- **Verdadero Positivo (TP):** la observación es predicha de manera correcta como una clase positiva.
- **Verdadero Negativo (TN):** la observación es predicha de manera correcta como una clase negativa.
- **Falso Positivo (FP):** se los conoce como error de tipo 1, la observación es predicha de manera incorrecta como una clase positiva y en realidad era negativa.
- **Falso Negativo (FN):** se los conoce como error de tipo 2, la observación es predicha de manera incorrecta como una clase negativa y en realidad era positiva.

CAPÍTULO III: DESARROLLO DEL MODELO

3. Desarrollo del modelo de predicción de fuga de clientes mediante CRISP-DM

Para lograr los objetivos establecidos en este trabajo, se utilizará una metodología CRISP-DM. Este tipo de método es una adaptación de un método comúnmente utilizado en ciencia de datos, llamado como Descubrimiento del Conocimiento en Base de Datos con sus siglas (KDD), en donde se ejecutan algunos pasos o etapas en un orden particular y extraen información valiosa de una base de datos. Además, el propósito del enfoque CRISP-DM es incorporar el conocimiento del negocio en la aplicación de modelos matemáticos. Por lo tanto, procedemos a describir a cada uno de los pasos (seis) de la aplicación para esta metodología.

3.1. Comprensión del Negocio

En la primera fase se ligan los factores de negocio con los factores técnicos para conseguir el resultado deseado y las decisiones tomadas en base de los criterios indicados anteriormente según menciona (Niño, 2016). Por lo tanto, la gestión y la creación del modelo se desarrollará para resolver el problema, lograr una explicación más nítida, asegurando el mayor uso práctico del proyecto por parte de las variables seleccionadas.

3.1.1. Perspectiva del Negocio

La corporación GPF tiene como objetivo incrementar sus ventas y convertirse en el principal retail farmacéutico elegido por los clientes al momento de realizar sus compras, para ello debe atraer a nuevos clientes y sobre todo retener a los que ya forman parte de la marca. Por tal motivo la corporación requiere de una herramienta de control de clientes para evitar su fuga utilizando un modelo de machine learning que será alimentado de parámetros seleccionados de los clientes convirtiéndose en un instrumento de predicción que ayudara a enmendar estas necesidades.

3.1.2. Objetivos del Negocio

Desarrollar un modelo predictivo usando algoritmos de machine learning para estimar la fuga de clientes que permita levantar alertas tempranas y atender necesidades insatisfechas de los usuarios en un periodo de 12 meses.

3.1.3. Evaluación de la situación

En la actualidad la corporación GPF registra sus ventas a través del sistema de facturación RESA en los diferentes puntos de venta a lo largo del país, esta información se almacena en la base de datos Teradata, con la información guardada se procede a realizar el análisis de ventas en cada cierre de mes, evaluando el comportamiento del cliente, sin embargo no existe un sistema o herramienta que permita detectar a tiempo si un cliente ha dejado de comprar antes de los 12 meses y se convertirá en posible fuga, por lo cual es necesario valorar al cliente y que permita aplicar acciones tempranas para retenerlo en la marca.

- **Recursos Necesarios**

Tabla 1

Personal del Proyecto

Cargo	Titulo	Cantidad
Jefe de BI	MSc. En Finanzas y Matemáticas	1
Coordinador de BI	Ing. De Sistemas	1
Analista de Procesos	Ing. en Estadística	1

Nota. Autor: José Navas A., 2023

- **Recursos Tecnológicos**

Tabla 2

Equipos Requeridos en el Proyecto

Descripción	Características	Cantidad
Laptop	Core I5 10 Gen Windows 10 Procesador 64 bits 8 GB de Memoria	3
Servidor	Core I7 9 Gen Windows 10 Procesador 64bits 64 Gb de Memoria	1

Nota. Autor: José Navas A., 2023

- **Riesgos, Supuestos y Contingencias**

Tabla 3

Riesgos, Supuestos y Contingencias del Proyecto

Riesgo	Contingencia
Uso de Datos de Clientes	Al no poder utilizar los datos personales de los clientes debido a cuestiones legales, se ha tenido que utilizar una base de datos con información que no muestre ningún tipo de dato que identifique al cliente.
Acuerdo de Confidencialidad	Se ha firmado un acuerdo de confidencialidad con la empresa para el uso de datos.
Eficacia del Modelo	No se incluirán datos o variables complejas que puedan afectar en el rendimiento del modelo

Nota. Autor: José Navas A., 2023

- **Costos**

Tabla 4

Costos del Proyecto

Rubro	**Valor**
Sueldo Coordinador BI por 2 meses	$ 2.900
Servicios Básicos por 2 meses	$ 100
Total	**$ 3.000**

Nota. Autor: José Navas A., 2023

Los datos de este proyecto no suponen ningún coste adicional para la empresa, ya que estos datos pertenecen a la misma desde el momento en que el cliente efectúa una compra en los puntos de venta.

3.1.4. Determinación de los objetivos de la modelación

- Analizar el segmento de clientes base a ser utilizado en el modelo predictivo.
- Determinar las variables más significativas a ser utilizadas en el modelo.
- Implementar el modelo de predicción de fuga.

3.1.5. Plan del proyecto

Tabla 5

Plan del Proyecto

	SEMANAS	3-7 Abril	10-14 Abril	17-21 Abril	24-28 Abril	2-5 Mayo	8-12 Mayo	15-19 Mayo	22-31 Mayo
Actividades del Proyecto									
Entendimiento del Negocio									
Comprensión de los Datos									
Preparación de los Datos									
Modelado									
Evaluación									
Despliegue									

Nota. Autor: José Navas A., 2023

El proyecto iniciará desde el mes de abril y su culminación en el mes de mayo con una duración de 2 meses excluyendo los días de feriados y fines de semana.

3.2. Comprensión de los datos

En este segundo paso de la metodología CRISP-DM, realizamos la recopilación inicial de los datos para establecer la primera asociación con el problema, y con ello poder familiarizarse con los datos, evaluar su disposición, e identificar relaciones obvias para formar las primeras hipótesis.

Al igual que en el punto anterior, esta etapa está encaminada a la comprensión, pero en esta ocasión se trata de la información acerca de la empresa que almacena en su base en este caso es Teradata. Para distinguir claramente los tipos de formatos de almacenamiento de los datos (fecha, texto, números, etc.), el tipo de información que contiene (características del cliente, datos de transacciones, etc.) esta información, combinándola con la comprensión del negocio.

3.2.1. Recolección inicial de los datos

Una vez que se completa toda la investigación sobre la empresa y la información que se tiene, se debe establecer la relación entre estos datos y lo que se construirá más adelante, el modelo de predicción de fuga de clientes. En otras palabras, la base de datos debe ser adecuada para futuros modelos, ya que dependiendo del tipo de teoría o modelo que se implemente, los requisitos para desarrollar adecuadamente esa teoría variarán.

Para este estudio, la data principal a utilizar es la facturación de los clientes almacenada como hemos mencionado anteriormente en la base de datos Teradata, en donde se tiene información de cada mes referente a las compras que está realizando en los puntos de venta y si el cliente sigue activo (o si se fugó), dentro del horizonte temporal en un periodo de 12 meses para el caso será desde mayo 2022 a la actualidad.

Esta data consta de alrededor de 11.5 Millones de filas para el periodo de 12 meses, donde cada una de estas representa las transacciones de compra de un cliente en un mes en particular. En cuanto a los clientes únicos, estos suman un total de 1.8 Millones activos a nivel nacional en el periodo antes mencionado.

En particular, esta sección trata sobre el manejo de los valores faltantes, la transformación de las variables, definición y demarcación de datos de línea base según la ventana de tiempo analizada o cualquier otra modificación. a la base de datos fuente necesaria para desarrollar el presente trabajo.

3.2.2. Exploración de los datos

La información obtenida corresponde a la base de datos de Teradata de nombre BIMKT, en la cual se presenta la tabla stage de ventas de nombre BI_VENTAS_DETALLE_FYBECA, donde podemos encontrar los datos de compra de los clientes en cada punto de venta donde efectuaron un registro de venta en la corporación.

Dicha tabla posee información de compras de clientes desde el año 2020 hasta la actualidad, el cual contiene un promedio de 10 Millones de registros por año, la tabla de ventas posee 37 variables, la cual se conoce como la tabla de hechos, las llaves principales de esta tabla se centran en:

- IDENTIFICACION_FIDELIZADA
- IDENTIFICACION_FACTURADA
- IDENTIFICACION_FINAL
- CODIGO_CLIENTE
- FECHA

A continuación, veremos una breve descripción de cada uno de los campos que se encuentran en la tabla fuente:

- **FECHA:** Fecha de transacción del documento de compra del cliente, tipo (date).

- **ANIO:** Anio de transacción del documento de compra del cliente, tipo (int).
- **MES:** Mes de transacción del documento de compra del cliente, tipo (int).
- **IDENTIFICACION_FIDELIZADA:** Nro. de Identificación del cliente aplicando un medio de descuento y afiliado al club para su compra, tipo (varchar).
- **IDENTIFICACION_FACTURADA:** Nro. de Identificación del cliente aplicado para la emisión de su factura, tipo (varchar).
- **IDENTIFICACION_FINAL:** Combinación de identificaciones (IDENTIFICACION_FIDELIZADA e IDENTIFICACION_FACTURADA) mediante reglas condicionales para generar una identificación final de las compras efectuadas por factura, tipo (varchar).
- **CODIGO:** Nro. de Identificación asociado al cliente otorgado de forma interna por el registro en las bases de la empresa de forma secuencial, tipo (varchar).
- **INDICADOR_CONSUMIDOR_FINAL:** Indicador que muestra si un cliente transacciono como consumidor final o no, tipo (binario).
- **CANAL:** Tipo de medio por el cual se efectuó la compra, PDV o WEB, tipo (varchar).
- **NUM_ORDEN_SF:** Numero de orden generada por la pagina web cuando se efectúa una compra, tipo (varchar).
- **CANAL_DIGITAL:** Tipo de medio por el cual se efectuó la compra, PDV, CALL o WEB, tipo (varchar).

- **COD_PRODUCTO_VITALCARD:** Tipo de medio de descuento usado por el cliente durante su compra, tipo (varchar).
- **COD_ITEM:** Código del producto adquirido en compra, tipo (varchar).
- **NOMBRE_PRODUCTO:** Nombre del producto adquirido en compra, tipo (varchar).
- **MARCA:** Marca del producto adquirido en compra, tipo (varchar).
- **NOMBRE_SUBCATEGORIA:** Subcategoría del producto adquirido en compra, tipo (varchar).
- **NOMBRE_CATEGORIA:** Categoría de producto adquirido en compra, tipo (varchar).
- **MACROCATEGORIA:** Macro categoría del producto adquirido, tipo (varchar).
- **PATOLOGÍA:** Patología asociada al producto adquirido, tipo (varchar).
- **NOMBRE_TIPO_NEGOCIO:** Unidad de producción a la que pertenece el producto, tipo (varchar).
- **COD_LOCAL:** Código del local donde se realizó la compra, tipo (varchar).
- **NOMBRE_LOCAL:** Nombre del local donde se realizó la compra tipo (varchar).
- **NOMBRE_PROVINCIA:** Nombre de la provincia asociada al punto de venta del lugar en donde se realizó la compra, tipo (varchar).
- **NOMBRE_CIUDAD:** Nombre de la ciudad asociada al punto de venta donde se realizó la compra, tipo (varchar).

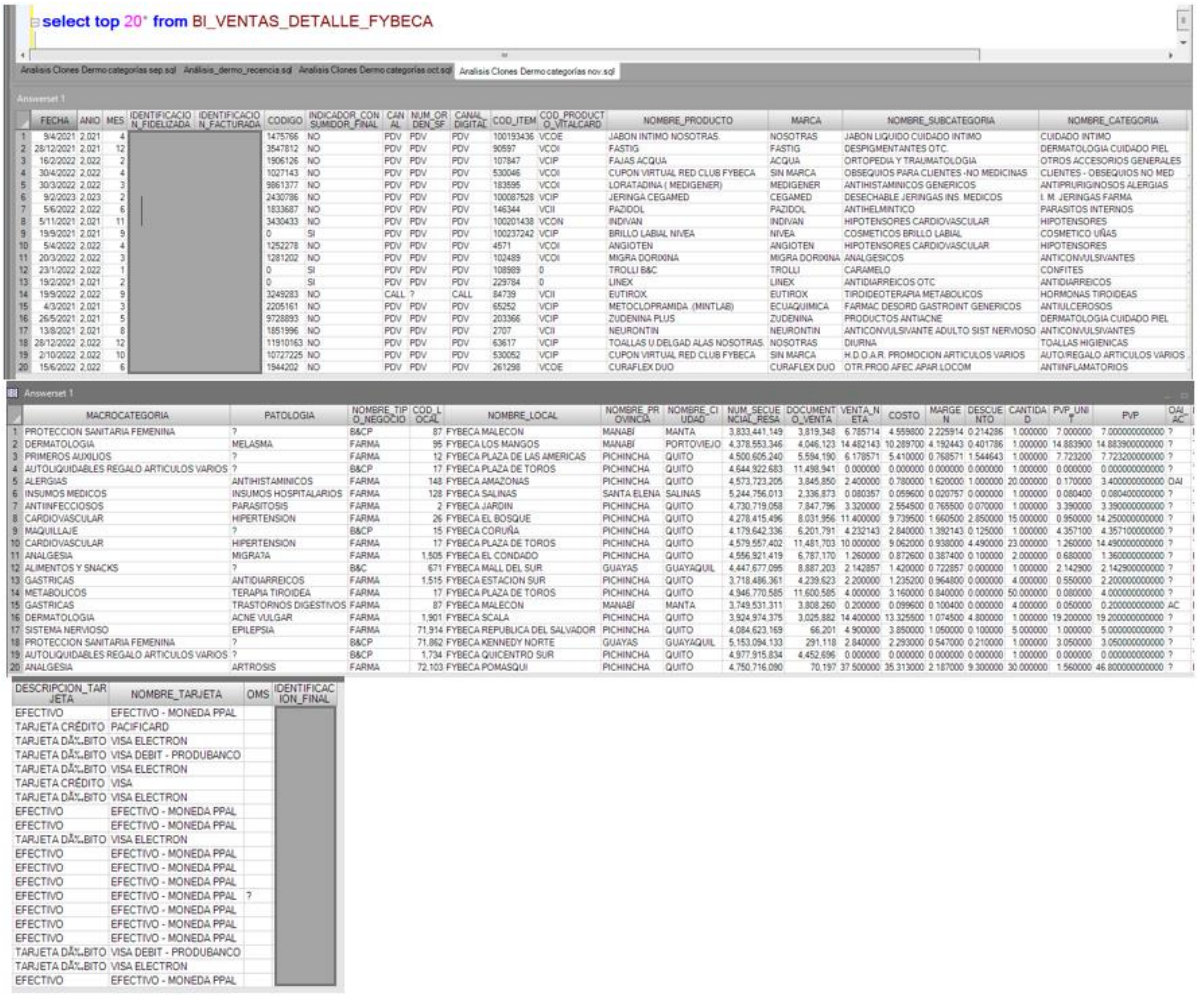
- **NUM_SECUENCIAL_RESA:** Nro. identificativo único asociado a la compra de forma interna en el registro de las bases de la empresa de forma secuencial, tipo (varchar).
- **DOCUMENTO_VENTA:** Nro. de documento único asociado a la factura entregada al cliente de forma impresa, tipo (varchar).
- **VENTA_NETA:** Valor neto de la compra realizada por producto sin IVA, tipo (decimal).
- **COSTO:** Valor real que invierta la empresa adquiriendo el producto imputado sobre la venta, tipo (decimal).
- **MARGEN:** Valor calculado luego de la resta entre la VENTA_NETA, el COSTO y DESCUENTO ingreso directo para la empresa cuando se realiza una compra por producto, tipo (decimal).
- **DESCUENTO:** Valor otorgado al cliente para realizar una rebaja en el valor de la VENTA_NETA por la adquisición de productos, en cada una de sus compras si amerita el caso, tipo (decimal).
- **CANTIDAD:** Nro. de productos adquiridos por el cliente en cada una de sus compras, tipo (int).
- **PVP_UNIT:** Precio de venta unitario por producto en las compras efectuadas, tipo (decimal).
- **PVP:** Precio de Venta total por producto en las compras efectuadas, tipo (decimal).
- **OAI_AC:** Descriptor de promoción confidencial aplicada por producto en las compras, tipo (varchar).
- **DESCRIPCION_TARJETA:** Medio de pago usado para realizar las compras de productos, tipo (varchar).

- **NOMBRE_TARJETA:** Nombre de la tarjeta usada al momento de realizar el pago de las compras, tipo (varchar).

Un vistazo de la base de datos inicial, donde se exponen registros de los años 2021-2022-2023 en un top 20 con la información de la fuente citada, en relación a los campos de identificaciones se encuentran restringidos por temas de privacidad de datos de los clientes, se observa a continuación en la figura 22:

Figura 22

Toma de captura de una muestra de la base de datos



```
select top 20* from BI_VENTAS_DETALLE_FYBECA
```

FECHA	ANIO	MES	IDENTIFICACION_FIDELIZADA	IDENTIFICACION_FACTURADA	CODIGO	INDICADOR_CONSUMIDOR_FINAL	CANAL	NUM_ORDEN_SF	CANAL_DIGITAL	COD_ITEM	COD_PRODUCTO_VITALCARD	NOMBRE_PRODUCTO	MARCA	NOMBRE_SUBCATEGORIA	NOMBRE_CATEGORIA
1	9/4/2021	2021	4		1475766	NO	PDV	PDV	PDV	100193436	VCDE	JABON INTIMO NOSOTRAS	NOSOTRAS	JABON LIQUIDO CUIDADO INTIMO	CUIDADO INTIMO
2	28/12/2021	2021	12		3547812	NO	PDV	PDV	PDV	90597	VCDI	FASTIG	FASTIG	DESPIGMENTANTES OTC	DERMATOLOGIA CUIDADO PIEL
3	16/2/2022	2022	2		1906126	NO	PDV	PDV	PDV	107847	VCIP	FAJAS ACQUA	ACQUA	ORTOPEDIA Y TRAUMATOLOGIA	OTROS ACCESORIOS GENERALES
4	30/4/2022	2022	4		1027143	NO	PDV	PDV	PDV	530046	VCDI	CUPON VIRTUAL RED CLUB FYBECA	SIN MARCA	OBSEQUIOS PARA CLIENTES -NO MEDICINAS	CLIENTES - OBSEQUIOS NO MED
5	30/3/2022	2022	3		9961377	NO	PDV	PDV	PDV	183595	VCDI	LORATADINA (MEDIGENER)	MEDIGENER	ANTIHISTAMINICOS GENERICOS	ANTIPRURIGINOSOS ALERGIAS
6	9/2/2023	2023	2		2430786	NO	PDV	PDV	PDV	100087528	VCIP	JERINGA CEGAMED	CEGAMED	DESECHABLE JERINGAS INS. MEDICOS	I. M. JERINGAS FARMA
7	5/6/2022	2022	6		1833687	NO	PDV	PDV	PDV	146344	VCII	PAZIDOL	PAZIDOL	ANTIHELMINTICO	PARASITOS INTERNOS
8	5/11/2021	2021	11		3430433	NO	PDV	PDV	PDV	100201438	VCDN	INDIVAN	INDIVAN	HIPOTENSORES CARDIOVASCULAR	HIPOTENSORES
9	19/9/2021	2021	9		0	SI	PDV	PDV	PDV	100237242	VCIP	BRILLO LABIAL NIVEA	NIVEA	COSMETICOS BRILLO LABIAL	COSMETICO UÑAS
10	5/4/2022	2022	4		1252278	NO	PDV	PDV	PDV	4571	VCDI	ANGIOTEN	ANGIOTEN	HIPOTENSORES CARDIOVASCULAR	HIPOTENSORES
11	20/3/2022	2022	3		1281202	NO	PDV	PDV	PDV	102489	VCDI	MIGRA DORIXINA	MIGRA DORIXINA	ANALGESICOS	ANTICONVULSIVANTES
12	23/1/2022	2022	1		0	SI	PDV	PDV	PDV	108969	0	TROLLI B&C	TROLLI	CARAMELO	CONFITES
13	19/2/2021	2021	2		0	SI	PDV	PDV	PDV	229784	0	LINEX	LINEX	ANTIDIARREICOS OTC	ANTIDIARREICOS
14	19/9/2022	2022	9		3249283	NO	CALL	?	CALL	84739	VCII	EUTIROX	EUTIROX	TIROIDEOTERAPIA METABOLICOS	HORMONAS TIROIDEAS
15	4/3/2021	2021	3		2205161	NO	PDV	PDV	PDV	65252	VCIP	METOCLOPRAMIDA (MINTLAB)	ECUAQUIMICA	FARMAC DESORD GASTROINT GENERICOS	ANTIULCEROSOS
16	26/5/2021	2021	5		9728893	NO	PDV	PDV	PDV	203366	VCIP	ZUDENINA PLUS	ZUDENINA	PRODUCTOS ANTIACNE	DERMATOLOGIA CUIDADO PIEL
17	13/8/2021	2021	8		1851996	NO	PDV	PDV	PDV	2707	VCII	NEURONTIN	NEURONTIN	ANTICONVULSIVANTE ADULTO SIST NERVIOSO	ANTICONVULSIVANTES
18	28/12/2022	2022	12		11910163	NO	PDV	PDV	PDV	63617	VCIP	TOALLAS U DELGAD ALAS NOSOTRAS	NOSOTRAS	DIURNA	TOALLAS HIGIENICAS
19	2/10/2022	2022	10		10727225	NO	PDV	PDV	PDV	530052	VCIP	CUPON VIRTUAL RED CLUB FYBECA	SIN MARCA	H.D.O.A.R. PROMOCION ARTICULOS VARIOS	AUTO/REGALO ARTICULOS VARIOS
20	15/6/2022	2022	6		1944202	NO	PDV	PDV	PDV	261298	VCDE	CURAFLEX DUO	CURAFLEX DUO	OTR.PROD.AFEC.APAR.LOCOM	ANTIINFLAMATORIOS

MACROCATEGORIA	PATOLOGIA	NOMBRE_TIPO_NEGOCIO	COD_LOCAL	NOMBRE_LOCAL	NOMBRE_PROVINCIA	NOMBRE_CIUDAD	NUM_SECUENCIAL_RESA	DOCUMENTO_VENTA	VENTA_NETA	COSTO	MARGEN	DESCUENTO	CANTIDAD	PVP_UNIT	PVP	OAL_AC
PROTECCION SANITARIA FEMENINA	?	B&CP	87	FYBECA MALECON	MANABÍ	MANTA	3,833,441,149	3,819,348	6.785714	4.559800	2.225914	0.214286	1.000000	7.000000	7.000000000000	?
DERMATOLOGIA	MELASMA	FARMA	95	FYBECA LOS MANGOS	MANABÍ	PORTOVIEJO	4,378,553,346	4,046,123	14.482143	10.289700	4.192443	0.401786	1.000000	14.883900	14.883900000000	?
PRIMEROS AUXILIOS	?	FARMA	12	FYBECA PLAZA DE LAS AMERICAS	PICHINCHA	QUITO	4,500,605,240	5,594,190	6.178571	5.410000	0.768571	1.544643	1.000000	7.723200	7.723200000000	?
AUTOLIQUIDABLES REGALO ARTICULOS VARIOS	?	B&CP	17	FYBECA PLAZA DE TOROS	PICHINCHA	QUITO	4,644,922,683	11,498,941	0.000000	0.000000	0.000000	0.000000	1.000000	0.000000	0.000000000000	?
ALERGIAS	ANTIHISTAMINICOS	FARMA	148	FYBECA AMAZONAS	PICHINCHA	QUITO	4,573,723,205	3,845,850	2.400000	0.780000	1.620000	1.000000	20.000000	0.170000	3.400000000000	OAI
INSUMOS MEDICOS	INSUMOS HOSPITALARIOS	FARMA	128	FYBECA SALINAS	SANTA ELENA	SALINAS	5,244,756,013	2,336,873	0.080357	0.059600	0.020757	0.000000	1.000000	0.080400	0.080400000000	?
ANTIINFECCIOSOS	PARASITOSIS	FARMA	2	FYBECA JARDIN	PICHINCHA	QUITO	4,730,719,058	7,847,796	3.320000	2.554500	0.765500	0.070000	1.000000	3.390000	3.390000000000	?
CARDIOVASCULAR	HIPERTENSION	FARMA	26	FYBECA EL BOSQUE	PICHINCHA	QUITO	4,278,415,496	8,031,956	11.400000	9.739500	1.660500	2.850000	15.000000	0.950000	14.250000000000	?
MAQUILLAJE	?	B&CP	15	FYBECA CORUÑA	PICHINCHA	QUITO	4,179,642,336	6,201,791	4.232143	2.840000	1.392143	0.125000	1.000000	4.357100	4.357100000000	?
CARDIOVASCULAR	HIPERTENSION	FARMA	17	FYBECA PLAZA DE TOROS	PICHINCHA	QUITO	4,579,557,402	11,481,703	10.000000	9.062000	0.938000	4.490000	23.000000	1.260000	14.490000000000	?
ANALGESIA	MIGRAÑA	FARMA	1,505	FYBECA EL CONDADO	PICHINCHA	QUITO	4,556,921,419	6,787,170	1.260000	0.872600	0.387400	0.100000	2.000000	0.680000	1.360000000000	?
ALIMENTOS Y SNACKS	?	B&C	671	FYBECA MALL DEL SUR	GUAYAS	GUAYAQUIL	4,447,677,095	8,887,203	2.142857	1.420000	0.722857	0.000000	1.000000	2.142900	2.142900000000	?
GASTRICAS	ANTIDIARREICOS	FARMA	1,515	FYBECA ESTACION SUR	PICHINCHA	QUITO	3,718,486,361	4,239,623	2.200000	1.235200	0.964800	0.000000	4.000000	0.550000	2.200000000000	?
METABOLICOS	TERAPIA TIROIDEA	FARMA	17	FYBECA PLAZA DE TOROS	PICHINCHA	QUITO	4,946,770,585	11,600,585	4.000000	3.160000	0.840000	0.000000	50.000000	0.080000	4.000000000000	?
GASTRICAS	TRASTORNOS DIGESTIVOS	FARMA	87	FYBECA MALECON	MANABÍ	MANTA	3,749,531,311	3,808,260	0.200000	0.099600	0.100400	0.000000	4.000000	0.050000	0.200000000000	AC
DERMATOLOGIA	ACNE VULGAR	FARMA	1,901	FYBECA SCALA	PICHINCHA	QUITO	3,924,974,375	3,025,882	14.400000	13.325500	1.074500	4.800000	1.000000	19.200000	19.200000000000	?
SISTEMA NERVIOSO	EPILEPSIA	FARMA	71,914	FYBECA REPUBLICA DEL SALVADOR	PICHINCHA	QUITO	4,084,623,169	66,201	4.900000	3.850000	1.050000	0.100000	5.000000	1.000000	5.000000000000	?
PROTECCION SANITARIA FEMENINA	?	B&CP	71,862	FYBECA KENNEDY NORTE	GUAYAS	GUAYAQUIL	5,153,094,133	291,118	2.840000	2.293000	0.547000	0.210000	1.000000	3.050000	3.050000000000	?
AUTOLIQUIDABLES REGALO ARTICULOS VARIOS	?	B&CP	1,734	FYBECA QUICENTRO SUR	PICHINCHA	QUITO	4,977,915,834	4,452,696	0.000000	0.000000	0.000000	0.000000	1.000000	0.000000	0.000000000000	?
ANALGESIA	ARTROSIS	FARMA	72,103	FYBECA POMASQUI	PICHINCHA	QUITO	4,750,716,090	70,197	37.500000	35.313000	2.187000	9.300000	30.000000	1.560000	46.800000000000	?

DESCRIPCION_TARJETA	NOMBRE_TARJETA	OMS	IDENTIFICACION_FINAL
EFECTIVO	EFECTIVO - MONEDA PPAL		
TARJETA CRÉDITO	PACIFICARD		
TARJETA DÃ‰BITO	VISA ELECTRON		
TARJETA DÃ‰BITO	VISA DEBIT - PRODUBANCO		
TARJETA DÃ‰BITO	VISA ELECTRON		
TARJETA CRÉDITO	VISA		
TARJETA DÃ‰BITO	VISA ELECTRON		
EFECTIVO	EFECTIVO - MONEDA PPAL		
EFECTIVO	EFECTIVO - MONEDA PPAL		
TARJETA DÃ‰BITO	VISA ELECTRON		
EFECTIVO	EFECTIVO - MONEDA PPAL		
EFECTIVO	EFECTIVO - MONEDA PPAL		
EFECTIVO	EFECTIVO - MONEDA PPAL		
EFECTIVO	EFECTIVO - MONEDA PPAL	?	
EFECTIVO	EFECTIVO - MONEDA PPAL		
EFECTIVO	EFECTIVO - MONEDA PPAL		
EFECTIVO	EFECTIVO - MONEDA PPAL		
TARJETA DÃ‰BITO	VISA DEBIT - PRODUBANCO		
TARJETA DÃ‰BITO	VISA ELECTRON		
EFECTIVO	EFECTIVO - MONEDA PPAL		

Nota. Gráfico que muestra la consulta realizada a la tabla base mediante un top 20, autoría propia mediante datos obtenidos de las fuentes de la corporación, 2023.

3.3. Preparación de los datos

Una vez que se completa toda la investigación sobre la empresa y la información que tienen, se debe establecer la relación entre estos datos y lo que se construirá más adelante, que es el modelo de predicción de fuga. En términos diferentes, la base de datos tendrá que adaptarse a los modelos futuros, ya que dependiendo del tipo de teoría o modelación que se realice, habrá diferentes requisitos para desarrollarla correctamente.

En concreto, este apartado trata sobre como tratar a los valores perdidos, formateo de variables, seleccionar variables apropiadas para el uso del modelo, definición y demarcación de datos de línea base según la ventana de tiempo analizada o cualquier otra modificación a la base de datos fuente que será útil para desarrollar el trabajo de título.

3.3.1. Selección de los datos

Se utilizarán los registros que conforma la base de datos BIMKT de la tabla BI_VENTAS_DETALLE_FYBECA en el periodo de 12 meses últimos según lo expuesto anteriormente. Sin embargo, hay campos en estos registros que no son necesarios para nuestros fines de extracción de datos, por lo que algunos campos pueden omitirse.

Los siguientes campos fueron seleccionados para el análisis:

- FECHA
- IDENTIFICACION_FINAL
- CODIGO
- COD_ITEM
- NOMBRE_PRODUCTO
- NUM_SECUENCIAL_RESA
- VENTA_NETA

- CANTIDAD

La razón para excluir ciertos campos, como se mencionó anteriormente, es analizar qué tan relevantes son estos en relación con los objetivos de la minería de datos identificados en la Fase 1 de la metodología.

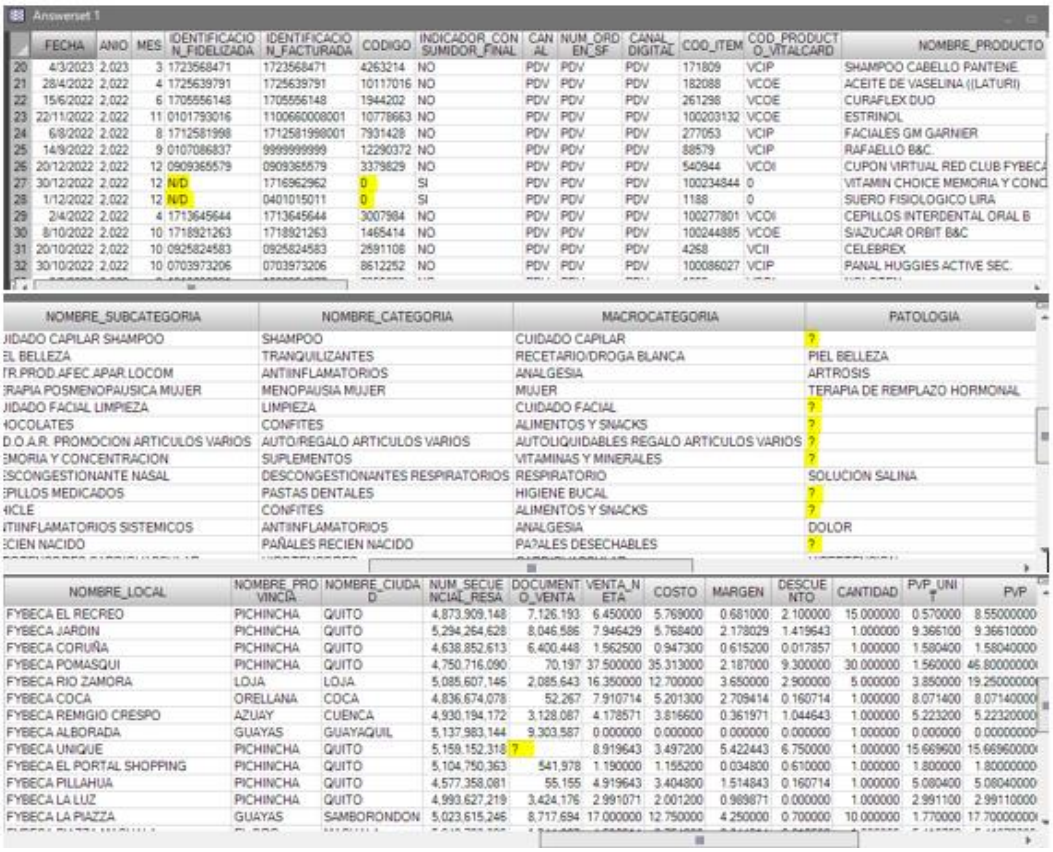
3.3.2. Limpieza de datos

La base de datos de la que dispone el trabajo presente tiene la información completa necesaria para lograr el propósito de la minería de datos, además, estos datos, específicamente referidos al caso presentado, son datos puros provenientes de las fuentes de los puntos de venta ya procesados y depurados por herramientas de ETL aplicadas por el departamento de tecnología de la coropración por lo tanto no se requiere una limpieza más profunda.

De igual manera tenemos campos con valores nulos o vacíos, que aparecen cuando en el registro de la venta no se ingresó dicho dato, por lo que no se consideran datos faltantes, entonces no hay que hacer ningún procedimiento en cuanto a valores perdidos. Estos valores nulos serán manejados por el proceso del modelo simplemente ignorándolos ya que no brindan información adicional para investigar, a continuación, observamos en la figura 23 resaltado en color amarillo un ejemplo de los datos faltantes.

Figura 23

Datos Faltantes en la base de datos



FECHA	ANIO	MES	IDENTIFICACION_FIDELIZADA	IDENTIFICACION_FACTURADA	CODIGO	INDICADOR_CONSUMIDOR_FINAL	CANAL	NUM_ORDEN_SF	CANAL_DIGITAL	COD_ITEM	COD_PRODUCTO_VITALCARD	NOMBRE_PRODUCTO
4/3/2023	2,023	3	1723568471	1723568471	4263214	NO	PDV	PDV	PDV	171809	VCIP	SHAMPOO CABELLO PANTENE
28/4/2022	2,022	4	1725639791	1725639791	10117016	NO	PDV	PDV	PDV	182088	VCOE	ACEITE DE VASELINA ((LATURI)
15/6/2022	2,022	6	1705556148	1705556148	1944202	NO	PDV	PDV	PDV	261298	VCOE	CURAFLEX DUO
22/11/2022	2,022	11	0101793016	1100660008001	10778663	NO	PDV	PDV	PDV	100203132	VCOE	ESTRINOL
6/8/2022	2,022	8	1712581998	1712581998001	7931428	NO	PDV	PDV	PDV	277053	VCIP	FACIALES GM GARNIER
14/9/2022	2,022	9	0107086837	9999999999	12290372	NO	PDV	PDV	PDV	88579	VCIP	RAFAELLO B&C.
20/12/2022	2,022	12	0909365579	0909365579	3379829	NO	PDV	PDV	PDV	540944	VCOI	CUPON VIRTUAL RED CLUB FYBECA
30/12/2022	2,022	12	N/D	1716962962	0	SI	PDV	PDV	PDV	100234844	0	VITAMIN CHOICE MEMORIA Y CONC
1/12/2022	2,022	12	N/D	0401015011	0	SI	PDV	PDV	PDV	1188	0	SUERO FISIOLOGICO LIRA
2/4/2022	2,022	4	1713645644	1713645644	3007984	NO	PDV	PDV	PDV	100277801	VCOI	CEPILLOS INTERDENTAL ORAL B
8/10/2022	2,022	10	1718921263	1718921263	1465414	NO	PDV	PDV	PDV	100244885	VCOE	SIAZUCAR ORBIT B&C
20/10/2022	2,022	10	0925824583	0925824583	2591108	NO	PDV	PDV	PDV	4268	VCII	CELEBREX
30/10/2022	2,022	10	0703973206	0703973206	8612252	NO	PDV	PDV	PDV	100086027	VCIP	PANAL HUGGIES ACTIVE SEC.

NOMBRE_SUBCATEGORIA	NOMBRE_CATEGORIA	MACROCATEGORIA	PATOLOGIA
IDADO CAPILAR SHAMPOO	SHAMPOO	CUIDADO CAPILAR	?
EL BELLEZA	TRANQUILIZANTES	RECETARIO/DROGA BLANCA	PIEL BELLEZA
TR.PROD.AFEC.APAR.LOCOM	ANTIINFLAMATORIOS	ANALGESIA	ARTROSIS
RAPIA POSMENOPAUSICA MUJER	MENOPAUSIA MUJER	MUJER	TERAPIA DE REMPLAZO HORMONAL
IDADO FACIAL LIMPIEZA	LIMPIEZA	CUIDADO FACIAL	?
OCOLATES	CONFITES	ALIMENTOS Y SNACKS	?
D.O.A.R. PROMOCION ARTICULOS VARIOS	AUTO/REGALO ARTICULOS VARIOS	AUTOLIQUIDABLES REGALO ARTICULOS VARIOS	?
EMORIA Y CONCENTRACION	SUPLEMENTOS	VITAMINAS Y MINERALES	?
ESCONGESTIONANTE NASAL	DESCONGESTIONANTES RESPIRATORIOS	RESPIRATORIO	SOLUCION SALINA
EPILLOS MEDICADOS	PASTAS DENTALES	HIGIENE BUCAL	?
NICLE	CONFITES	ALIMENTOS Y SNACKS	?
ITIINFLAMATORIOS SISTEMICOS	ANTIINFLAMATORIOS	ANALGESIA	DOLOR
ECIEN NACIDO	PAÑALES RECIEN NACIDO	PAÑALES DESECHABLES	?

NOMBRE_LOCAL	NOMBRE_PROVINCIA	NOMBRE_CIUDAD	NUM_SECUENCIAL_RESA	DOCUMENTO_VENTA	VENTA_NETA	COSTO	MARGEN	DESCUENTO	CANTIDAD	PVP_UNIT	PVP
FYBECA EL RECREO	PICHINCHA	QUITO	4,873,909,148	7,126.193	6.450000	5.769000	0.681000	2.100000	15.000000	0.570000	8.55000000
FYBECA JARDIN	PICHINCHA	QUITO	5,294,264,628	8,046.586	7.946429	5.768400	2.178029	1.419643	1.000000	9.366100	9.36610000
FYBECA CORUÑA	PICHINCHA	QUITO	4,638,852,613	6,400.448	1.562500	0.947300	0.615200	0.017857	1.000000	1.580400	1.58040000
FYBECA POMASQUI	PICHINCHA	QUITO	4,750,716,090	70,197	37.500000	35.313000	2.187000	9.300000	30.000000	1.560000	46.80000000
FYBECA RIO ZAMORA	LOJA	LOJA	5,085,607,146	2,085,643	16.350000	12.700000	3.650000	2.900000	5.000000	3.850000	19.25000000
FYBECA COCA	ORELLANA	COCA	4,836,674,078	52,267	7.910714	5.201300	2.709414	0.160714	1.000000	8.071400	8.07140000
FYBECA REMIGIO CRESPO	AZUAY	CUENCA	4,930,194,172	3,128,087	4.178571	3.816600	0.361971	1.044643	1.000000	5.223200	5.22320000
FYBECA ALBORADA	GUAYAS	GUAYAQUIL	5,137,983,144	9,303,587	0.000000	0.000000	0.000000	0.000000	1.000000	0.000000	0.00000000
FYBECA UNIQUE	PICHINCHA	QUITO	5,159,152,318	?	8.919643	3.497200	5.422443	6.750000	1.000000	15.669600	15.66960000
FYBECA EL PORTAL SHOPPING	PICHINCHA	QUITO	5,104,750,363	541,978	1.190000	1.155200	0.034800	0.610000	1.000000	1.800000	1.80000000
FYBECA PILLAHUA	PICHINCHA	QUITO	4,577,358,081	55,155	4.919643	3.404800	1.514843	0.160714	1.000000	5.080400	5.08040000
FYBECA LA LUZ	PICHINCHA	QUITO	4,993,627,219	3,424,176	2.991071	2.001200	0.989871	0.000000	1.000000	2.991100	2.99110000
FYBECA LA PIAZZA	GUAYAS	SAMBORONDON	5,023,615,246	8,717,694	17.000000	12.750000	4.250000	0.700000	10.000000	1.770000	17.70000000

Nota. Gráfico que muestra la consulta realizada a la tabla base donde se observan valores faltantes, autoría propia mediante datos obtenidos de las fuentes de la corporación, 2023.

Por ende, en esta parte lo que se procederá a realizar es aplicar un filtro a la consulta de la tabla BI_VENTAS_DETALLE_FYBECA, excluyendo aquellos clientes que son considerados como consumidor final, estos clientes no pueden ser incluidos en el modelo de fuga ya que su identificación corresponde a uno o varios números genéricos.

Las identificaciones a excluir son:

- '9999999999'
- 'N/D'

- '1791257049001'
- '1791927559001'
- '1791279352001'
- '0990017514001'
- '1790475247001'
- '1791988558001'
- '1792091705001'
- '1790093808001'
- '1791415132001'
- '1111111111'
- '0991189270001'
- '7777777770'
- '202020'
- 'CONSUMIDOR'
- '0992621915001'
- '05219B0010'
- '1792206979001'
- '0992794127001'
- '2002011787'
- '2002012041'
- '2002012042'
- '1777778888'
- '8989898989'
- '8120104060'
- '2002012045'

- '1792348684001'
- '2002012076'
- '01033358400'
- '1792493056001'
- ' '
- '0000'

También procederemos a filtrar a los clientes que realizan compras por RUC ya que al pertenecer a una sociedad u organización no son sujetos para ser aplicados en un posible análisis de fuga y por ende efectuar alguna acción personalizada, de tal manera que escogeremos aquellas identificaciones cuya longitud máxima sea de 10 caracteres.

3.3.3. Construcción de nuevos datos

Como se ha mencionado en el punto anterior no ha sido necesario generar nuevos atributos ni ingresar nuevos registros a la base de datos, ni la creación de nuevas estructuras (campos, registros, etc.), así también como la unión entre distintas tablas en la base de datos, la misma se encuentra depurada y preparada para su uso en el presente trabajo.

De igual manera no es necesario reordenar los campos de los registros ni reordenar los registros de la tabla. Tampoco es necesario cambiar el formato de los campos que se utilizarán para la extracción de datos, ya que el programa Python admite el formato actual y de ser el caso en que algún campo se cargue con una estructura diferente se realizar el cambio dentro del programa.

3.3.4. Selección de variables y preparación para modelar

Se requiere el poder seleccionar variables con el fin de generar los modelos de manera más simple y con ello optimizar el rendimiento de los métodos de aprendizaje automático.

El presente trabajo utilizará el análisis de recencia-frecuencia-monetaria (RFM) para modelar la fuga obteniendo en conjunto el RFM Score por cliente. Para hacer nuestros modelos, necesitaremos un marco de datos que consta de columnas de recencia, frecuencia y valor monetario.

Las definiciones de cada uno se encuentran a continuación.

- **Recencia:** tiempo entre la compra inicial y la compra más reciente (última)
- **Frecuencia:** número de compras repetidas realizadas por un cliente (compras totales-1)
- **Valor Monetario:** total gastado en compras en los puntos de venta.

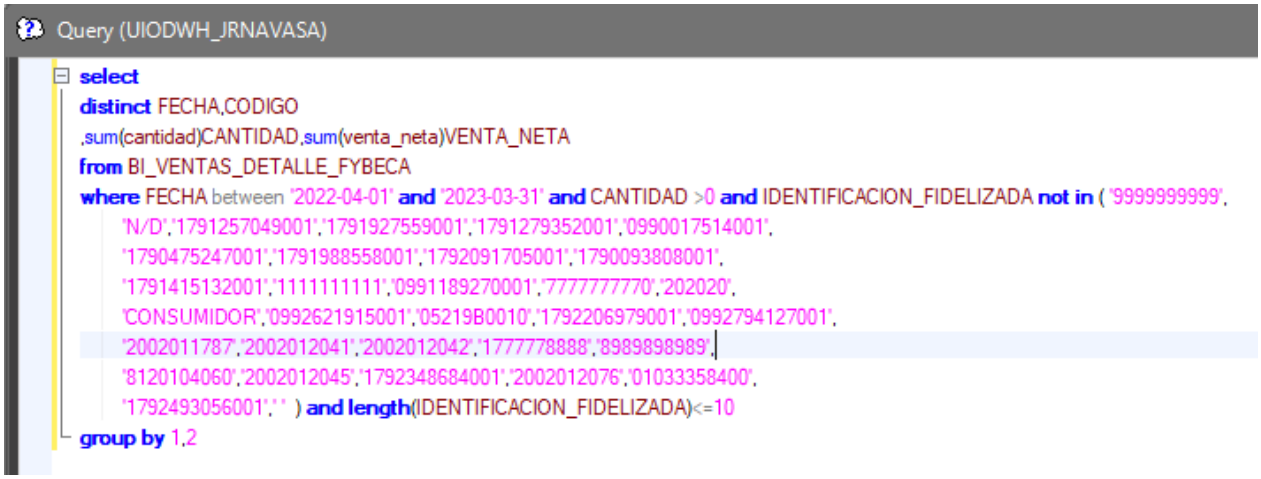
Con lo anteriormente dicho los campos solicitados y necesarios para comenzar con la construcción del modelo serían los siguientes:

- La información de identificación del cliente provendrá del campo **CODIGO**.
- La fecha de compra vendrá del campo **FECHA**.
- El valor gastado por cliente se obtendrá del campo **VENTA_NETA**.

La consulta final que será implementada en el lenguaje Python dará el comienzo en la construcción del modelo y esta quedaría de la siguiente manera, ver figura 24:

Figura 24

Consulta Teradata tabla ventas



Nota. Gráfico que muestra el query con el cual se procederá a incluir en el lenguaje Python dando inicio en la construcción del modelo, autoría propia mediante datos obtenidos de las fuentes de la corporación, 2023.

3.4. Modelado

En línea con lo expuesto anteriormente, la base de datos necesaria para aplicar los primeros modelos viables de predicción de fuga está lista para su uso. A continuación, después de planificar como se llevarán a cabo las pruebas para los modelos seleccionados, estos métodos se aplicarán a los datos para crear el modelo, y finalmente será necesario evaluar si el modelo en particular cumple con los criterios para tener éxito o no.

3.4.1. Selección de técnica de modelo

Antes de ejecutar los modelos de clasificación, al conjunto de datos se lo dividirá en dos grupos con características similares para probar el rendimiento de los modelos correctamente. Este proceso utiliza el 70 % del conjunto de los datos para poder entrenar el modelo, el 30 % restante para las pruebas. Para que el proceso cumpla su propósito, se debe

tener cuidado de que la división de datos corresponda a una distribución de instancias completa del set de datos a ser evaluado, de modo que no haya desequilibrio sobre la línea de base de los datos y los resultados se afecten.

Como se mencionó en la sección 2.6, se utilizarán tres modelos candidatos para predecir la fuga de clientes en este trabajo de titulación. Los modelos seleccionados serán el de regresión lineal, la regresión logística y el de árboles de decisión. Dichos modelos de clasificación utilizarán el mismo conjunto de variables basados en los campos anteriormente descritos en la sección 3.3.4.

3.4.2. Construcción del modelo

Para construir el modelo comenzamos a utilizar la herramienta de Python para la ejecución de los modelos, previamente a la ejecución de los modelos seleccionados debemos elaborar ciertos pasos previos, en primer lugar, procederemos a importar las librerías que son necesarias a lo largo de este trabajo, como observamos en la figura 25.

Figura 25

Librerías de Python

```
import time
import numpy as np
import pandas as pd
import teradatasql
import matplotlib.pyplot as plt
import datetime as dt
import warnings

from sklearn.model_selection import train_test_split, cross_val_predict
from sklearn.linear_model import LinearRegression, LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score, recall_score, roc_auc_score, roc_curve, auc,confusion_matrix
from sklearn import metrics
from collections import defaultdict


warnings.simplefilter(action="ignore", category=FutureWarning)
warnings.simplefilter(action="ignore", category=UserWarning)
```

Nota. Gráfico que muestra las librerías a ser usadas en el programa Python para iniciar con el desarrollo del modelo, autoría propia, 2023.

Procedemos a cargar la información proveniente de la base Teradata usando la librería teradatasql, con la estructura del query que definimos en la figura 22, a continuación, podemos observar en la figura 26 la forma en la que ese establecerá el código en Python.

Figura 26

Carga de Informacion desde Teradata

```
print("Hora Inicio:",time.strftime("%X"))
query="""select
distinct FECHA,CODIGO
,sum(cantidad)CANTIDAD,sum(venta_neta)VENTA_NETA
from BI_VENTAS_DETALLE_FYBECA
where FECHA between '2022-05-01' and '2023-04-30' and CANTIDAD >0 and IDENTIFICACION_FINAL not in ( '999999999
        'N/D','1791257049001','1791927559001','1791279352001','0990017514001',
        '1790475247001','1791988558001','1792091705001','1790093808001',
        '1791415132001','1111111111','0991189270001','7777777770','202020',
        'CONSUMIDOR','0992621915001','05219B0010','1792206979001','0992794127001',
        '2002011787','2002012041','2002012042','1777778888','8989898989',
        '8120104060','2002012045','1792348684001','2002012076','01033358400',
        '1792493056001',' '  ) and length(IDENTIFICACION_FINAL)<=10
group by 1,2"""
# BETWEEN  {d '2021-08-01'}  AND  {d '2022-08-03'}
# Se realiza la conexion y se trae la consulta que se realiza en el paso anterior
with teradatasql.connect(host='172.20.200.13', user='JRNAVASA', password='JRNAVASA') as connect:
    df = pd.read_sql(query, connect)
print("Total cargado:", df.shape[0])
print("Hora compilación:",time.strftime("%X"))
print("Fecha compilación:",time.strftime("%x"))
df.head(3)
```

```
Hora Inicio: 10:48:30
Total cargado: 10281403
Hora compilación: 10:56:16
Fecha compilación: 05/07/23
```

Nota. Gráfico que muestra la sentencia de Teradata colocada en el programa Python para la carga inicial de datos, autoría propia, 2023.

Una vez realizada la ejecución del código de la consulta, podemos visualizar los datos cargados en Python, con las variables seleccionadas previamente de la siguiente manera como se ve en la figura 27.

Figura 27

Datos Cargados a Python

	FECHA	CODIGO	CANTIDAD	VENTA_NETA
0	2022-09-10	3004576	2.0	1.062500
1	2023-01-04	3356694	3.0	3.910714
2	2023-01-18	7306385	2.0	47.680000

Nota. Gráfico que muestra la carga de datos en el lenguaje Python con el código head(), autoría propia, 2023.

Ahora procederemos a formatear las variables (CANTIDAD, FECHA) cargadas anteriormente aplicando el siguiente código en Python para que tengan una lectura optima dentro del programa, como se pueden ver en las figuras 28 y 29.

Figura 28

Formateo de datos

```
# df['doc_venta']=df['DOCUMENTO_VENTA'].apply(lambda x: str(x))
df['CANTIDAD']=df['CANTIDAD'].apply(lambda x: int(x))
# df['FECHA'] = pd.to_datetime(df['FECHA'])
df['FECHA']=pd.to_datetime(df['FECHA'],errors = 'coerce').dt.strftime('%d/%m/%Y')
df['FECHA']=df['FECHA'].astype('datetime64[ns]')
df.head()
```

	FECHA	CODIGO	CANTIDAD	VENTA_NETA
0	2022-09-10	3004576	2	1.062500
1	2023-01-04	3356694	3	3.910714
2	2023-01-18	7306385	2	47.680000
3	2023-03-12	11804517	5	11.446429
4	2023-03-07	12310441	5	27.633928

Nota. Gráfico que muestra el código ejecutado en el programa Python para el formateo de los campos, autoría propia, 2023.

Figura 29

Tipos de datos de las variables

```
df.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10325479 entries, 0 to 10325478
Data columns (total 4 columns):
 #   Column      Dtype
---  ------      -----
 0   FECHA       datetime64[ns]
 1   CODIGO      object
 2   CANTIDAD    int64
 3   VENTA_NETA  float64
dtypes: datetime64[ns](1), float64(1), int64(1), object(1)
memory usage: 315.1+ MB
```

Nota. Gráfico que muestra el código ejecutado en el programa Python con los tipos de datos modificados en los campos, autoría propia, 2023.

En este punto vamos a realizar una rápida validación en cuanto a fechas mínimas y máximas, así como el tamaño del data set cargado a Python tal como observamos en la Figura 30.

Figura 30

Validación y cantidad de datos

```
print('Rango de Fechas: %s - %s' % (df['FECHA'].min(), df['FECHA'].max()))

df1 = df.loc[df['FECHA'] < '2023-04-30']
df1.shape
```

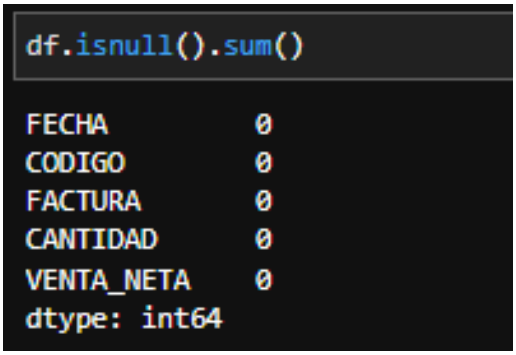
```
Rango de Fechas: 2022-04-01 00:00:00 - 2023-03-31 00:00:00
(10325479, 4)
```

Nota. Gráfico que muestra el código ejecutado en el programa Python con la validación de datos en cuanto a fecha y registros, autoría propia, 2023.

Procedemos a validar que no existen valores nulos en nuestra data que puedan causarnos inconvenientes más adelante, en la figura 31 podemos validar lo mencionado.

Figura 31

Comprobación de valores nulos

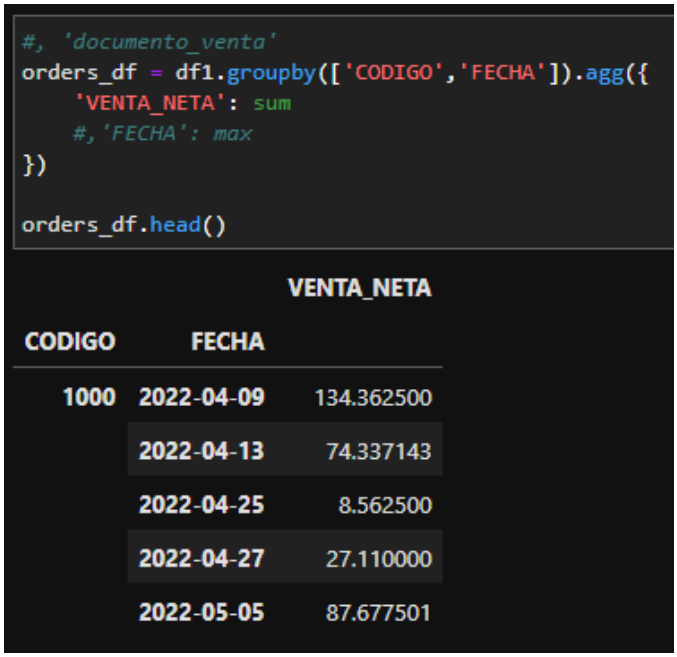


Nota. Gráfico que muestra el código ejecutado en el lenguaje Python con la validación de datos nulos aplicados al data frame, autoría propia, 2023.

Con la data ya cargada y sus campos formateados, lo que procedemos ahora es a realizar una agrupación por cliente con el campo CODIGO en base a la variable FECHA, y VENTA_NETA, dichos campos serán utilizados para la generación de los campos de frecuencia, recencia y valor monetario, en la figura 32 podemos observar el resultado obtenido.

Figura 32

Agrupación por cliente



```
#, 'documento_venta'
orders_df = df1.groupby(['CODIGO','FECHA']).agg({
    'VENTA_NETA': sum
    #,'FECHA': max
})

orders_df.head()
```

		VENTA_NETA
CODIGO	FECHA	
1000	2022-04-09	134.362500
	2022-04-13	74.337143
	2022-04-25	8.562500
	2022-04-27	27.110000
	2022-05-05	87.677501

Nota. Gráfico que muestra el código ejecutado en el lenguaje Python con la agrupación de datos por CODIGO en cuanto a FECHA y VENTA_NETA, autoría propia, 2023.

El siguiente paso será crear funciones que nos permita obtener los nuevos campos de frecuencia, recencia y valor monetario, para cada uno de los clientes que tenemos en nuestro data set, las funciones a crear serán: promedio, cantidad, recencia de compra, frecuencia de compra, tal como observamos en la figura 33.

Figura 33

Creación de funciones

```
fecha_max_analisis=df['FECHA'].max()
def groupby_mean(x):
    return x.mean()

def groupby_count(x):
    return x.count()

def purchase_duration(x):
    return (fecha_max_analisis - x.max()).days

def avg_frequency(x):
    return (x.max() - x.min()).days/x.count()


groupby_mean.__name__ = 'avg'
groupby_count.__name__ = 'cantidad'
purchase_duration.__name__ = 'recencia_de_compra'
avg_frequency.__name__ = 'frecuencia_de_compra'
```

Nota. Gráfico que muestra el código ejecutado en el lenguaje Python con la creación de funciones para la obtención de los campos frecuencia y recencia, autoría propia, 2023.

Procedemos a implementar las funciones anteriormente creadas, para obtener de forma lineal y única de cada uno de los clientes en el data set, con el campo de VENTA_NETA tendremos los siguientes campos nuevos que corresponden a la venta de los clientes (máximo, mínimo, promedio, suma, cantidad), y con el campo FECHA de igual forma obtenemos los siguientes campos (máxima, mínima, duración de compra, frecuencia de compra) en la figura 34 podemos observar el código en el lenguaje de Python para la creación de los nuevos campos al data set y su resultado obtenido.

Figura 34

Creación de nuevos campos

```
summary_df = orders_df.reset_index().groupby('CODIGO').agg({
    'VENTA_NETA': [min, max, sum, groupby_mean, groupby_count],
    'FECHA': [min, max, purchase_duration, avg_frequency]

})
```

	VENTA_NETA					FECHA			
	min	max	sum	avg	cantidad	min	max	recencia_de_compra	frecuencia_de_compra
CODIGO									
0000	0.071429	104.030000	2036.641745	8.854964	230	2022-05-01	2023-04-20	10	1.539130
00000	0.312500	64.236069	1470.711010	7.822931	188	2022-05-03	2023-03-23	38	1.723404
000000	0.428571	199.850712	2516.462123	12.645538	199	2022-05-01	2023-04-30	0	1.829146
0000001	2.589286	50.148214	78.867500	19.716875	4	2022-10-22	2023-04-23	7	45.750000
000001	0.258929	8.173214	80.835711	2.526116	32	2022-06-26	2023-03-23	38	8.437500
...	...	...	...	...	...	...	...	...	...
YE322208	10.358928	10.358928	10.358928	10.358928	1	2023-02-05	2023-02-05	84	0.000000
YS8615240	4.000000	4.000000	4.000000	4.000000	1	2023-03-06	2023-03-06	55	0.000000
Z3087638	24.300000	24.300000	24.300000	24.300000	1	2022-06-13	2022-06-13	321	0.000000
Z6042363	4.600000	22.115714	76.044998	12.674166	6	2022-12-27	2023-03-28	33	15.166667
Z7577627	45.000000	45.000000	45.000000	45.000000	1	2022-05-12	2022-05-12	353	0.000000

2094387 rows × 9 columns

Nota. Gráfico que muestra el código ejecutado en el lenguaje Python con la creación de los nuevos campos en base a las funcionas previas creadas, de frecuencia_de_compra y recencia_de_compra, autoría propia, 2023.

Ahora vamos a dar formato a las cabeceras de las columnas para quitar el efecto matriz y dejar en formato de tabla para ello aplicamos el siguiente código como se puede ver en la figura 35, junto con el resultado mostrado.

Figura 35

Datos en formato columnar

```
summary_df.columns = ['_'.join(col).lower() for col in summary_df.columns]
summary_df = summary_df.reset_index()
summary_df
```

	CODIGO	venta_neta_min	venta_neta_max	venta_neta_sum	venta_neta_avg	venta_neta_cantidad	fecha_min	fecha_max	fecha_recencia_de_compra	fecha_frecuencia_de_compra
0	0000	0.071429	104.030000	2036.641745	8.854964	230	2022-05-01	2023-04-20	10	1.539130
1	00000	0.312500	64.236069	1470.711010	7.822931	188	2022-05-03	2023-03-23	38	1.723404
2	000000	0.428571	199.850712	2516.462123	12.645538	199	2022-05-01	2023-04-30	0	1.829146
3	0000001	2.589286	50.148214	78.867500	19.716875	4	2022-10-22	2023-04-23	7	45.750000
4	000001	0.258929	8.173214	80.835711	2.526116	32	2022-06-26	2023-03-23	38	8.437500
...	...	...	...	...	...	...	...	...	...	...
2094382	YE322208	10.358928	10.358928	10.358928	10.358928	1	2023-02-05	2023-02-05	84	0.000000
2094383	YS8615240	4.000000	4.000000	4.000000	4.000000	1	2023-03-06	2023-03-06	55	0.000000
2094384	Z3087638	24.300000	24.300000	24.300000	24.300000	1	2022-06-13	2022-06-13	321	0.000000
2094385	Z6042363	4.600000	22.115714	76.044998	12.674166	6	2022-12-27	2023-03-28	33	15.166667
2094386	Z7577627	45.000000	45.000000	45.000000	45.000000	1	2022-05-12	2022-05-12	353	0.000000

2094387 rows × 10 columns

Nota. Gráfico que muestra el código ejecutado en el lenguaje Python dando el formato de tabla al data frame, autoría propia, 2023.

Ahora vamos a revisar un poco los datos de forma visual para mostrar la distribución de compra de los clientes en un numero promedio de días entre compras y con ello poder verificar el volumen de datos con los que vamos a trabajar, tal como observamos en la figura 36 junto con su código aplicado.

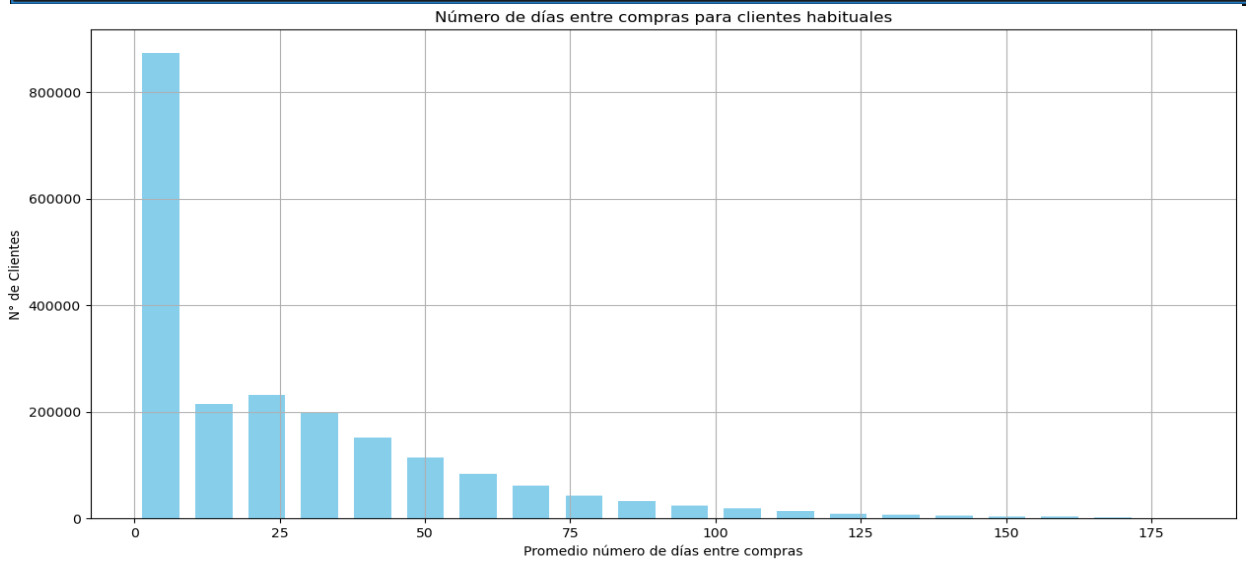
Figura 36

Gráfico distribución de cliente entre número promedio de días

```
'''
Esta gráfica muestra el número promedio de días entre compras para clientes habituales.
Es una vista general de la frecuencia con la que los clientes habituales realizaron compras históricamente.
'''
ax = summary_df['fecha_frecuencia_de_compra'].hist(
    bins=20,
    color='skyblue',
    rwidth=0.7,
    figsize=(15,7)
)

ax.set_xlabel('Promedio número de días entre compras')
ax.set_ylabel('N° de Clientes')
plt.title('Número de días entre compras para clientes habituales')

plt.show()
```



Nota. Gráfico que muestra el código ejecutado en el lenguaje Python obteniendo una imagen de barras por distribución entre clientes y días promedio entre compras, autoría propia, 2023.

En este punto vamos a proceder a filtrar la data a partir del data frame summary_df con el que hemos venido manejando hasta el momento, filtraremos a aquellos clientes que generan una recencia superior a 0 tanto en el campo fecha_recencia_de_compra así como en el fecha_frecuencia_de_compra, ya que dichos clientes que poseen 0 en dichas métricas son considerados clientes de paso es decir solo se acercaron puntualmente a realizar una compra en el local, pero no son considerados aun clientes fieles a la marca, y este ya será un tema de estudio distinto al que venimos presentando en este trabajo, como podemos

observar en la figura 37 los datos obtenidos luego del código aplicado, ya no constan clientes con valores 0 en los campos mencionados líneas más arriba.

Figura 37

Filtrado de datos en el data frame

```
summary_df1= (summary_df[summary_df.fecha_recencia_de_compra>0])
summary_df1=(summary_df1[summary_df1.fecha_frecuencia_de_compra>0])
summary_df1=summary_df1.reset_index(drop=True)
summary_df1
```

	CODIGO	venta_neta_min	venta_neta_max	venta_neta_sum	venta_neta_avg	venta_neta_cantidad	fecha_min	fecha_max	fecha_recencia_de_compra	fecha_frecuencia_de_compra
0	0000	0.071429	104.030000	2036.641745	8.854964	230	2022-05-01	2023-04-20	10	1.539130
1	00000	0.312500	64.236069	1470.711010	7.822931	188	2022-05-03	2023-03-23	38	1.723404
2	0000001	2.589286	50.148214	78.867500	19.716875	4	2022-10-22	2023-04-23	7	45.750000
3	000001	0.258929	8.173214	80.835711	2.526116	32	2022-06-26	2023-03-23	38	8.437500
4	000015738	12.550000	99.241072	294.811429	58.962286	5	2022-07-19	2023-03-27	34	50.200000
...	...	...	...	...	...	...	...	...	...	...
1315818	YC084976	20.112143	72.912500	163.124643	40.781161	4	2022-09-08	2022-12-08	143	22.750000
1315819	YC362939	9.508929	30.824643	68.092500	17.023125	4	2022-11-03	2023-02-18	71	26.750000
1315820	YC931590	29.348214	33.671429	63.019643	31.509822	2	2022-08-24	2022-09-05	237	6.000000
1315821	YE201422	8.000000	204.707857	526.430002	87.738334	6	2023-01-15	2023-04-12	18	14.500000
1315822	Z6042363	4.600000	22.115714	76.044998	12.674166	6	2022-12-27	2023-03-28	33	15.166667

1315823 rows × 10 columns

Nota. Gráfico que muestra el código ejecutado en el lenguaje Python obteniendo los datos una vez filtrados aquellos clientes que no poseen recencia ni frecuencia, valores en 0, autoría propia, 2023.

Ya con las columnas de recencia, frecuencia y valor monetario junto con los datos filtrados, es momento de comenzar a usar estas métricas que son vitales para nuestro modelo, en base a estas procedemos a crear cuartiles para cada una de las métricas mencionadas, en la figura 38 mostraremos el código utilizado y la salida del mismo con las nuevas columnas obtenidas en base a los cuartiles.

Figura 38

Creación de nuevas columnas basadas en cuartiles

```
quantiles = summary_df1.quantile(q=[0.75,0.5,0.25])
quantiles = quantiles.to_dict()

#Asignar un valor numérico a cada segmento
def r_score(x,p,d):
    if x<=d[p][0.75]:
        return 1
    elif x<=d[p][0.50]:
        return 2
    elif x<=d[p][0.25]:
        return 3
    else:
        return 4
summary_df1['r_quartile'] = summary_df1['fecha_recencia_de_compra'].apply(r_score, args=('fecha_recencia_de_compra',quantiles,))
summary_df1['f_quartile'] = pd.qcut(summary_df1['fecha_frecuencia_de_compra'], q=4, labels=range(1,5))
summary_df1['m_quartile'] = pd.qcut(summary_df1['venta_neta_sum'], q=4, labels=False) #summary_df1['venta_neta_sum'].apply(r_score, args=('venta_neta_sum',quantiles,))#
summary_df1['m_quartile'] = summary_df1['m_quartile'].max() -summary_df1['m_quartile']+1
```

```
summary_df1.head(100)
```

	CODIGO	venta_neta_min	venta_neta_max	venta_neta_sum	venta_neta_avg	venta_neta_cantidad	fecha_min	fecha_max	fecha_recencia_de_compra	fecha_frecuencia_de_compra	r_quartile	f_quartile	m_quartile
0	0000	0.071429	104.030000	2036.641745	8.854964	230	2022-05-01	2023-04-20	10	1.539130	1	1	1
1	00000	0.312500	64.236069	1470.711010	7.822931	188	2022-05-03	2023-03-23	38	1.723404	1	1	1
2	0000001	2.589286	50.148214	78.867500	19.716875	4	2022-10-22	2023-04-23	7	45.750000	1	3	2
3	000001	0.258929	8.173214	80.835711	2.526116	32	2022-06-26	2023-03-23	38	8.437500	1	1	2
4	000015738	12.550000	99.241072	294.811429	58.962286	5	2022-07-19	2023-03-27	34	50.200000	1	3	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...
95	0100003730	2.150000	120.883929	263.463215	32.932902	8	2022-05-20	2023-04-16	14	41.375000	1	3	1
96	0100003821	1.419643	49.996429	127.696787	18.242398	7	2022-05-27	2023-04-25	5	47.571429	1	3	2
97	0100003938	3.000000	9.786429	12.786429	6.393215	2	2022-10-30	2023-01-03	117	32.500000	4	3	4
98	0100003953	3.991071	20.553572	40.144643	13.381548	3	2023-03-30	2023-04-24	6	8.333333	1	1	3
99	0100004035	0.580357	50.055357	116.440000	10.585455	11	2022-06-03	2023-03-14	47	25.818182	1	2	2

100 rows × 13 columns

Nota. Gráfico que muestra el código ejecutado en el lenguaje Python obteniendo las nuevas columnas en cuartiles basadas en la recencia, frecuencia y valor monetario, autoría propia, 2023.

Las columnas que acabamos de crear son fundamentales para aplicar el concepto de RFM Score que vimos en el capítulo II, aplicamos la fórmula y creamos la columna de nombre rfm_score de tipo entero(int), tal como observamos en la figura 39.

Figura 39

Creación de columna rfm_score

```
#Calcular el RFM Score
summary_df1['rfm_score'] = summary_df1['r_quartile'].astype(str) + summary_df1['f_quartile'].astype(str) + summary_df1['m_quartile'].astype(str)
summary_df1['rfm_score'] = summary_df1['rfm_score'].astype(int)
summary_df1.head(100)
```

	CODIGO	venta_neta_min	venta_neta_max	venta_neta_sum	venta_neta_avg	venta_neta_cantidad	fecha_min	fecha_max	fecha_recencia_de_compra	fecha_frecuencia_de_compra	r_quartile	f_quartile	m_quartile	rfm_score
0	0000	0.071429	104.030000	2036.641745	8.854964	230	2022-05-01	2023-04-20	10	1.539130	1	1	1	111
1	00000	0.312500	64.236069	1470.711010	7.822931	188	2022-05-03	2023-03-23	38	1.723404	1	1	1	111
2	0000001	2.589286	50.148214	78.867500	19.716875	4	2022-10-22	2023-04-23	7	45.750000	1	3	2	132
3	000001	0.258929	8.173214	80.835711	2.526116	32	2022-06-26	2023-03-23	38	8.437500	1	1	2	112
4	000015738	12.550000	99.241072	294.811429	58.962286	5	2022-07-19	2023-03-27	34	50.200000	1	3	1	131
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
95	0100003730	2.150000	120.883929	263.463215	32.932902	8	2022-05-20	2023-04-16	14	41.375000	1	3	1	131
96	0100003821	1.419643	49.996429	127.696787	18.242398	7	2022-05-27	2023-04-25	5	47.571429	1	3	2	132
97	0100003938	3.000000	9.786429	12.786429	6.393215	2	2022-10-30	2023-01-03	117	32.500000	4	3	4	434
98	0100003953	3.991071	20.553572	40.144643	13.381548	3	2023-03-30	2023-04-24	6	8.333333	1	1	3	113
99	0100004035	0.580357	50.055357	116.440000	10.585455	11	2022-06-03	2023-03-14	47	25.818182	1	2	2	122

100 rows × 17 columns

Nota. Gráfico que muestra el código ejecutado en el lenguaje Python obteniendo la nueva columna rfm_score basada en los cuartiles por recencia, frecuencia y valor monetario, autoría propia, 2023.

Una vez que tenemos nuestra columna de RFM Score vamos a segmentar nuestro data frame en el segmento de clientes que necesitamos evaluar y pueden llegar a convertirse en posibles clientes fugados, de igual forma tomando la teoría del capítulo II de RFM Score nos centraremos en aquellos clientes cuyos segmentos se encuentran en 44X y 43X de la variable rfm_score, en la figura 40 podemos observar la segmentación realizada al data frame summary_rfmscore1 basado en el summary_df1 con el filtro aplicado.

Figura 40

Filtrado de datos basado en RFM Score

```
summary_rfmscore=summary_df1[summary_df1['rfm_score'].astype(str).str.startswith('43','44')]
summary_df1['filtro_score']=0 #inicializamos con 0 para todos los registros
summary_df1.loc[summary_df1['rfm_score'].isin(summary_rfmscore['rfm_score']),'filtro_score']=1
summary_df1.head(100)
```

	CODIGO	venta_neta_min	venta_neta_max	venta_neta_sum	venta_neta_avg	venta_neta_cantidad	fecha_min	fecha_max	fecha_recencia_de_compra	fecha_frecuencia_de_compra	r_quartile	f_quartile	m_quartile	rfm_score	filtro_score
0	0000	0.071429	104.030000	2036.641745	8.854964	230	2022-05-01	2023-04-20	10	1.539130	1	1	1	111	0
1	00000	0.312500	64.236069	1470.711010	7.822931	188	2022-05-03	2023-03-23	38	1.723404	1	1	1	111	0
2	0000001	2.589286	50.148214	78.867500	19.716875	4	2022-10-22	2023-04-23	7	45.750000	1	3	2	132	0
3	000001	0.258929	8.173214	80.835711	2.526116	32	2022-06-26	2023-03-23	38	8.437500	1	1	2	112	0
4	000015738	12.550000	99.241072	294.811429	58.962286	5	2022-07-19	2023-03-27	34	50.200000	1	3	1	131	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
95	0100003730	2.150000	120.883929	263.463215	32.932902	8	2022-05-20	2023-04-16	14	41.375000	1	3	1	131	0
96	0100003821	1.419643	49.996429	127.696787	18.242398	7	2022-05-27	2023-04-25	5	47.571429	1	3	2	132	0
97	0100003938	3.000000	9.786429	12.786429	6.393215	2	2022-10-30	2023-01-03	117	32.500000	4	3	4	434	1
98	0100003953	3.991071	20.553572	40.144643	13.381548	3	2023-03-30	2023-04-24	6	8.333333	1	1	3	113	0
99	0100004035	0.580357	50.055357	116.440000	10.585455	11	2022-06-03	2023-03-14	47	25.818182	1	2	2	122	0

100 rows × 17 columns

Nota. Gráfico que muestra el código ejecutado en el lenguaje Python filtrando los valores correspondientes a clientes posibles fuga de los segmentos 44X y 43X de RFM Score, autoría propia, 2023.

Con los datos segmentados por el RFM Score, lo siguiente es establecer un periodo de tiempo para estos clientes en el cual definiremos si su recencia es menor al valor elegido en este caso 180 días por definición del negocio, entonces crearemos una nueva columna de nombre churn para colocar 1 en caso de que cumpla dicha condición y 0 en el caso que no, tal como observamos en la figura 41, obtenemos un nuevo data frame de nombre summary_df1_filter1, el cual usaremos para la utilización en los modelos seleccionados.

Figura 41

Creación de variable churn

```
summary_df1['churn'] = ((summary_df1['fecha_recencia_de_compra'] >180) ).astype(int)
summary_df1
```

	CODIGO	venta_neta_min	venta_neta_max	venta_neta_sum	venta_neta_avg	venta_neta_cantidad	fecha_min	fecha_max	fecha_recencia_de_compra	fecha_frecuencia_de_compra	r_quartile	f_quartile	m_quartile	rfm_score	filtro_score	churn
0	0000	0.071429	104.030000	2036.641745	8.854964	230	2022-05-01	2023-04-20	10	1.539130	1	1	1	111	0	0
1	00000	0.312500	64.236069	1470.711010	7.822931	188	2022-05-03	2023-03-23	38	1.723404	1	1	1	111	0	0
2	0000001	2.589286	50.148214	78.867500	19.716875	4	2022-10-22	2023-04-23	7	45.750000	1	3	2	132	0	0
3	000001	0.258929	8.173214	80.835711	2.526116	32	2022-06-26	2023-03-23	38	8.437500	1	1	2	112	0	0
4	000015738	12.550000	99.241072	294.811429	58.962286	5	2022-07-19	2023-03-27	34	50.200000	1	3	1	131	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1315818	YC084976	20.112143	72.912500	163.124643	40.781161	4	2022-09-08	2022-12-08	143	22.750000	4	2	2	422	0	0
1315819	YC362939	9.508929	30.824643	68.092500	17.023125	4	2022-11-03	2023-02-18	71	26.750000	1	2	3	123	0	0
1315820	YC931590	29.348214	33.671429	63.019643	31.509822	2	2022-08-24	2022-09-05	237	6.000000	4	1	3	413	0	1
1315821	YE201422	8.000000	204.707857	526.430002	87.738334	6	2023-01-15	2023-04-12	18	14.500000	1	1	1	111	0	0
1315822	Z6042363	4.600000	22.115714	76.044998	12.674166	6	2022-12-27	2023-03-28	33	15.166667	1	1	2	112	0	0

1315823 rows × 17 columns

Nota. Gráfico que muestra el código ejecutado en el lenguaje Python con la creación de la variable churn que será utilizada para la evaluación de los modelos, autoría propia, 2023.

Con nuestro data frame final ahora comenzamos a segmentar la data en entrenamiento y prueba con la distribución 70/30 que habíamos definido con anterioridad 70% para datos de entrenamiento y 30% para los de prueba, con esto nuestra variables X y Y serán las siguientes, como podemos observar en la figura 42.

Figura 42

Segmentación de datos de entrenamiento y prueba

```
X = summary_df1[['r_quartile','f_quartile','m_quartile','filtro_score']]
y = summary_df1['churn']
trainX, testX, trainY, testY  = train_test_split(X, y, test_size=0.3)
```

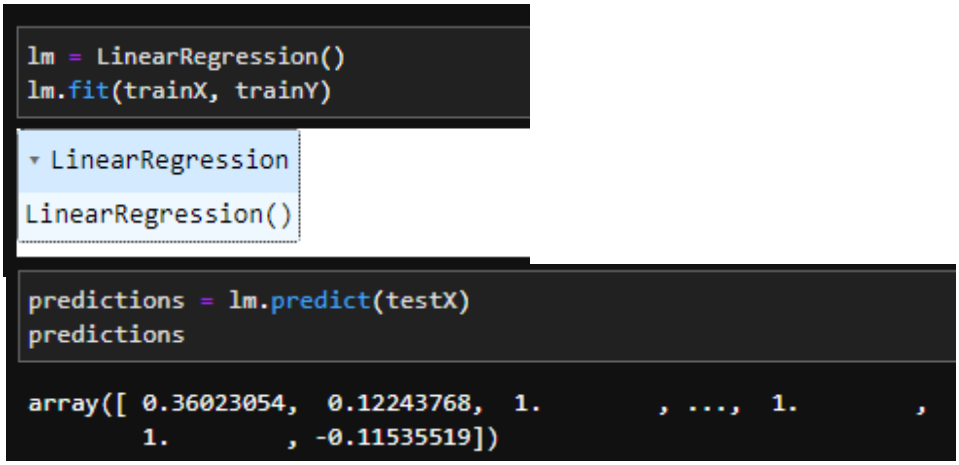
Nota. Gráfico que muestra el código ejecutado en el lenguaje Python con la segmentación de datos 70/30 para entrenamiento y prueba del modelo, autoría propia, 2023.

- **Regresión Lineal:**

  Nuestro primer modelo será la regresión lineal para ello utilizaremos, las variables anteriormente segmentadas para X y Y, el siguiente código que se presenta en la figura 43, permite crear el modelo de regresión lineal utilizando los datos de entrenamiento, junto con su predicción obtenida.

Figura 43

Modelo de Regresión Lineal



Nota. Gráfico que muestra el código ejecutado en el lenguaje Python con la creación del modelo de Regresión Lineal, en base al conjunto de prueba y entrenamiento, autoría propia, 2023.

- **Regresión Logística:**

  El segundo modelo a ser utilizado será la regresión logística, de la misma manera que el modelo anterior se utilizará las variables anteriormente segmentadas para X y Y, el código que se muestra en la figura 44, nos ayuda a generar el modelo de regresión logística utilizando los datos de entrenamiento, junto con su predicción obtenida.

Figura 44

Modelo de Regresión Logística

```
model = LogisticRegression()
model.fit(trainX ,trainY)
y_pred = model.predict(testX)
y_pred
```

```
array([0, 0, 0, ..., 1, 0, 0])
```

```
accuracy = accuracy_score(testY,y_pred)
print("Exactitud del modelo de Regresión Logística:",accuracy)
```

```
Exactitud del modelo de Regresión Logística: 0.9157080357798792
```

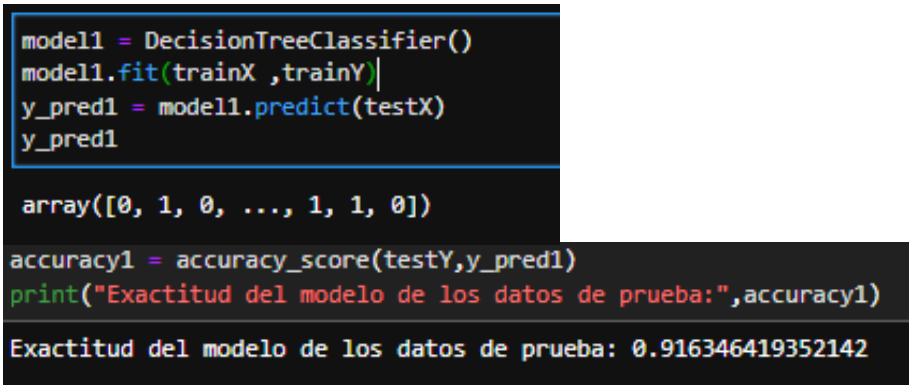
Nota. Gráfico que muestra el código ejecutado en el lenguaje Python con la creación del modelo de Regresión Logística, en base a los datos del conjunto de prueba y entrenamiento, autoría propia, 2023.

- **Árboles de Decisión:**

  Nuestro último modelo a ser utilizado será los árboles de decisión, de la misma manera que el modelo anterior se utilizará las variables anteriormente segmentadas para X y, en la figura 45 podemos observar el código que nos ayuda a generar el modelo de árboles de decisión para los datos de entrenamiento, junto con su predicción obtenida.

Figura 45

Modelo de Árboles de Decisión



Nota. Gráfico que muestra el código ejecutado en el lenguaje Python con la creación del modelo de Árboles de Decisión, en base al conjunto de prueba y entrenamiento, autoría propia, 2023.

3.5. Evaluación

Esta fase es muy importante para medir la calidad de las predicciones de nuestros modelos porque después de completar un entrenamiento, el modelo autocomprobará una pequeña porción de los registros que aprendió y que ya estaban previamente etiquetados como churn, con dicha prueba se puede determinar con precisión cómo se están calificando los clientes actuales.

3.5.1. Métricas de Desempeño Utilizadas

Como mencionamos anteriormente las métricas a ser utilizadas para la evaluación de los modelos son: matriz de confusión, cross-validation y curva ROC-AUC.

- **Regresión Lineal:**

  De manera general con las métricas principales arrojadas podemos validar en primera instancia el modelo realizado utilizando la regresión lineal, tal se observa en la figura 46.

Figura 46

Métricas de Valoración del Modelo

```
y_pred2 = lm.predict(testX)

print('Mean Absolute Error:', metrics.mean_absolute_error(testY, y_pred2))
print('Mean Squared Error:', metrics.mean_squared_error(testY, y_pred2))
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(testY, y_pred2)))
print("R-Square:",metrics.r2_score(testY, y_pred2))
```

```
Mean Absolute Error: 0.14476995858405015
Mean Squared Error: 0.058828369830121956
Root Mean Squared Error: 0.24254560360913976
R-Square: 0.3972207592661158
```

Nota. Gráfico que muestra el código ejecutado en el lenguaje Python con las métricas de valoración obtenidas por el modelo, autoría propia, 2023.

El valor más Relevante que podemos observar en la figura 45 es el valor de R-Square, el cual nos indica que nuestro modelo posee un 40% de probabilidad de acierto, por lo cual no se considera como un modelo aceptable.

Adicional a las métricas anteriores procedemos a aplicar una validación cruzada al modelo para obtener una segunda opinión en cuanto a la valoración del modelo, en la figura 47 observamos el código realizado en Python junto con la gráfica de la validación obtenida.

Figura 47

Gráfico de Validación Cruzada

```
from sklearn.model_selection import KFold
kfold_validacion = KFold(10)
resultados = cross_val_score(lm, X, y, cv = kfold_validacion)
print(f"Métricas validación cruzada: {resultados}")
print(f"Média de las métricas de validación cruzada: {resultados.mean()}")
```

```
Métricas validación cruzada: [0.39054409 0.40046556 0.41179282 0.40968471 0.39363907 0.39253701
 0.39475101 0.38471894 0.3763446  0.40613435]
Média de las métricas de validación cruzada: 0.39606121686105544
```

Nota. Gráfico que muestra el código ejecutado en el lenguaje Python con el código que genera la validación cruzada utilizando Kfold de 10 para el modelo, autoría propia, 2023.

- **Regresión Logística:**

  Para la regresión logística aplicaremos 2 conceptos vistos en el capítulo II que son las matrices de confusión y la curva ROC-AUC, en la figura 48 observamos el código que genera la matriz de confusión.

Figura 48

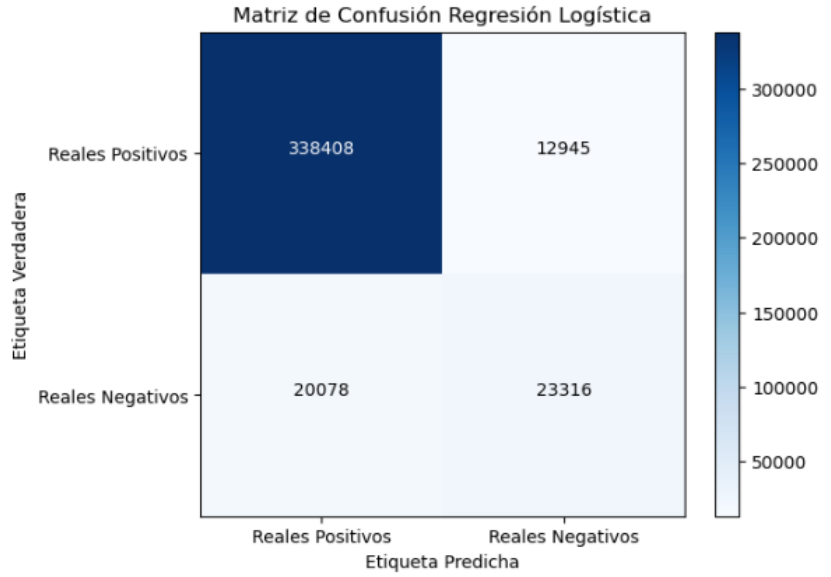
Matriz de Confusión Regresión Logística

```
cm1= confusion_matrix(testY,y_pred)
class_labels = ['Reales Positivos', 'Reales Negativos']

# Graficar la matriz de confusión
plt.imshow(cm1, interpolation='nearest', cmap=plt.cm.Blues)
plt.title('Matriz de Confusión Regresión Logística')
plt.colorbar()
tick_marks = np.arange(len(class_labels))
plt.xticks(tick_marks, class_labels)
plt.yticks(tick_marks, class_labels)

# Añadir los valores de la matriz en cada celda
thresh = cm1.max() / 2.0
for i, j in np.ndindex(cm1.shape):
    plt.text(j, i, format(cm1[i, j], 'd'),
             horizontalalignment="center",
             color="white" if cm1[i, j] > thresh else "black")

plt.ylabel('Etiqueta Verdadera')
plt.xlabel('Etiqueta Predicha')
plt.tight_layout()
plt.show()
```



Nota. Gráfico que muestra el código ejecutado en el lenguaje Python con el código que genera la matriz de confusión al modelo de regresión logística, autoría propia, 2023.

La matriz de confusión proporciona información detallada y cuantitativa sobre el desempeño de un modelo de clasificación, lo que ayuda a comprender su capacidad de clasificación y a realizar mejoras o ajustes en función de los resultados obtenidos.

- El valor 338.408 en la posición (0, 0) indica que hay 338.408 instancias que fueron clasificadas correctamente como positivas (verdaderos positivos).
- El valor 12.945 en la posición (0, 1) indica que hay 12.945 instancias que fueron clasificadas incorrectamente como negativas (falsos negativos).
- El valor 20.078 en la posición (1, 0) indica que hay 20.078 instancias que fueron clasificadas incorrectamente como positivas (falsos positivos).
- El valor 23.316 en la posición (1, 1) indica que hay 23.316 instancias que fueron clasificadas correctamente como negativas (verdaderos negativos).

Según los valores obtenidos y aplicando las formula de precisión el valor sería el siguiente: 338.408/(338.408+20.078) =94,4%, de igual manera calcularemos su Recall: 338.408/(338.408+12.945) =96,3%

Como segunda validación aplicaremos la curva ROC-AUC, como podemos observar en la figura 49, la curva alcanza un nivel favorable con un valor de 0.95.

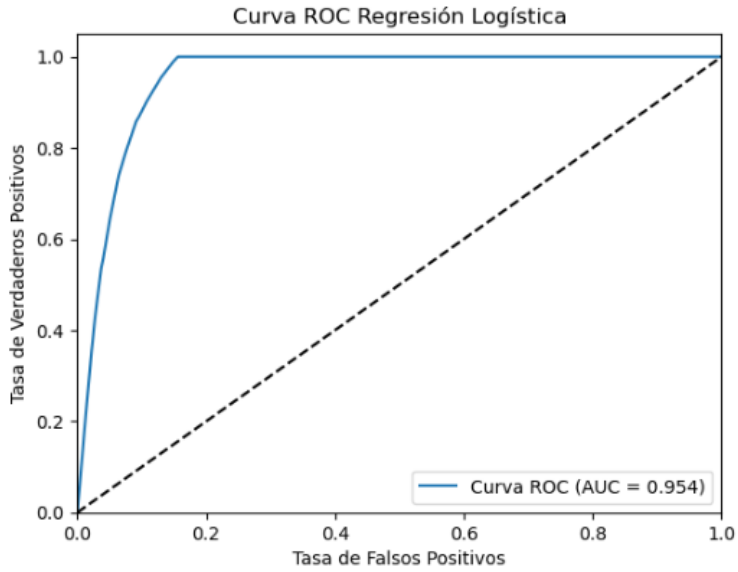
Figura 49

Curva Roc Regresión Logística

```
y_prob1 = model.predict_proba(testX)[:, 1]

fpr, tpr, thresholds = roc_curve(testY, y_prob1)
roc_auc1 = auc(fpr, tpr)

# Graficar la curva ROC
plt.figure()
plt.plot(fpr, tpr, label='Curva ROC (AUC = %0.3f)' % roc_auc1)
plt.plot([0, 1], [0, 1], 'k--')
plt.xlim([0.0, 1.0])
plt.ylim([0.0, 1.05])
plt.xlabel('Tasa de Falsos Positivos')
plt.ylabel('Tasa de Verdaderos Positivos')
plt.title('Curva ROC Regresión Logística')
plt.legend(loc="lower right")
plt.show()
```



Nota. Gráfico que muestra el código ejecutado en el lenguaje Python con el código que genera la curva ROC-AUC para el modelo de regresión logística, autoría propia, 2023.

La medición anterior proporcionada por la Curva ROC-AUC es de 95.4% lo que nos da a entender que el modelo tiene una predicción muy buena en base a los datos proporcionados.

- **Árboles de Decisión:**

  De igual forma que la regresión logística aplicaremos 2 conceptos vistos en el capítulo II que son las matrices de confusión y la curva ROC-AUC para los árboles de decisión, en la figura 50 observamos el código que genera la matriz de confusión.

Figura 50

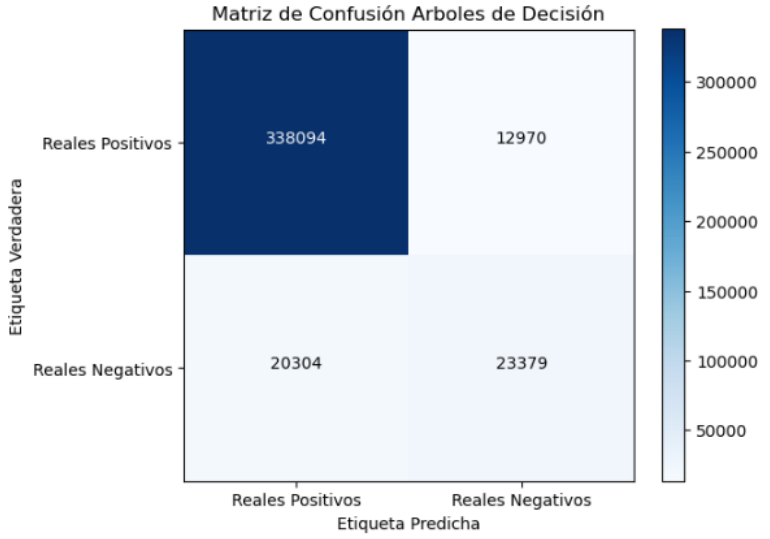
Matriz de Confusión Árboles de Decisión

```
cm= confusion_matrix(testY,y_pred1)
class_labels = ['Reales Positivos', 'Reales Negativos']

# Graficar la matriz de confusión
plt.imshow(cm, interpolation='nearest', cmap=plt.cm.Blues)
plt.title('Matriz de Confusión Arboles de Decisión')
plt.colorbar()
tick_marks = np.arange(len(class_labels))
plt.xticks(tick_marks, class_labels)
plt.yticks(tick_marks, class_labels)

# Añadir los valores de la matriz en cada celda
thresh = cm.max() / 2.0
for i, j in np.ndindex(cm.shape):
    plt.text(j, i, format(cm[i, j], 'd'),
             horizontalalignment="center",
             color="white" if cm[i, j] > thresh else "black")

plt.ylabel('Etiqueta Verdadera')
plt.xlabel('Etiqueta Predicha')
plt.tight_layout()
plt.show()
```



Nota. Gráfico que muestra el código ejecutado en el lenguaje Python con el código que genera la matriz de confusión al modelo de regresión logística, autoría propia, 2023.

La matriz de confusión proporciona información detallada y cuantitativa sobre el desempeño de un modelo de clasificación, lo que ayuda a comprender su capacidad de clasificación y a realizar mejoras o ajustes en función de los resultados obtenidos.

- El valor 338.094 en la posición (0, 0) indica que hay 338.094 instancias que fueron clasificadas correctamente como positivas (verdaderos positivos).
- El valor 12.970 en la posición (0, 1) indica que hay 12.970 instancias que fueron clasificadas incorrectamente como negativas (falsos negativos).
- El valor 20.304 en la posición (1, 0) indica que hay 20.304 instancias que fueron clasificadas incorrectamente como positivas (falsos positivos).
- El valor 23.379 en la posición (1, 1) indica que hay 23.379 instancias que fueron clasificadas correctamente como negativas (verdaderos negativos).

Según los valores obtenidos y aplicando las formula de precisión el valor sería el siguiente: 338.094 /(338.094+20.304) =94,3%, de igual manera calcularemos su Recall: 338.094 /(338.094+12.970) =96,3%

Como segunda validación al igual que el modelo anterior, aplicaremos la curva ROC-AUC, como podemos observar en la figura 51, la curva alcanza un nivel favorable con un valor de 0.954.

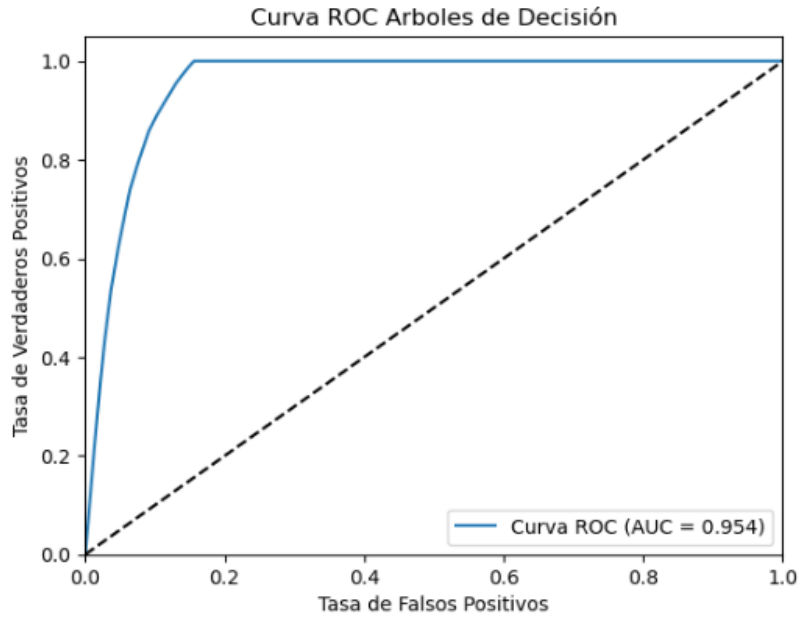
Figura 51

Curva Roc Árboles de Decisión

```
y_prob = model1.predict_proba(testX)[:, 1]

fpr, tpr, thresholds = roc_curve(testY, y_prob)
roc_auc = auc(fpr, tpr)

#Graficar la curva ROC
plt.figure()
plt.plot(fpr, tpr, label='Curva ROC (AUC = %0.3f)' % roc_auc)
plt.plot([0, 1], [0, 1], 'k--')
plt.xlim([0.0, 1.0])
plt.ylim([0.0, 1.05])
plt.xlabel('Tasa de Falsos Positivos')
plt.ylabel('Tasa de Verdaderos Positivos')
plt.title('Curva ROC Arboles de Decisión')
plt.legend(loc="lower right")
plt.show()
```



Nota. Gráfico que muestra el código ejecutado en el lenguaje Python con el código que genera la curva ROC-AUC para el modelo para árboles de decisión, autoría propia, 2023.

La medición anterior proporcionada por la Curva ROC-AUC de 95,4%, nos da a entender que el modelo tiene una predicción muy buena en base a los datos proporcionados.

Posterior a la ejecución de los modelos con las variables seleccionadas que fueron descritas anteriormente, se procede a elaborar un síntesis de los datos previamente obtenidos mediante las métricas de evaluación en la tabla 6.

Tabla 6

Cuadro Comparativo Modelos

Modelo	Precisión	Roc-Auc	Recall
Regresión Lineal	39.72%	N/A	N/A
Regresión Logística	91.57%	95.35%	96.3%
Árboles de Decisión	91.63%	95.40%	96.3%

Nota. Autor: José Navas A., 2023

Según los datos obtenidos en el cuadro anterior, se establece que el modelo seleccionado considerado el de mejor desempeño en conjunto con los datos ingresados es el de árboles de decisión con una precisión del 91,63%, a lo largo de este trabajo y con todo lo desarrollado, el modelo tiene sentido para el negocio, cumpliendo así con el objetivo propuesto al identificar a aquellos clientes a convertirse en fuga.

3.6. Implementación

Se acordó que el modelo seleccionado realizaría una evaluación mensual de sus clientes, la cual sería analizada durante un lapso de 6 meses (180 días) para considerar aquellos clientes que pueden llegar a ser posible fuga. Las tareas incorporarán fechas transcurridas en un periodo de tiempo de 12 meses para construir un conjunto de datos de clientes activos que serán evaluados, los resultados de la predicción serán enviados como una base de datos al equipo de CRM quien administrará las campañas adecuadas para dichos clientes mediante personalización, donde se asociará el código del usuario con su indicador de churn otorgado por el modelo.

CAPÍTULO IV: RESULTADOS

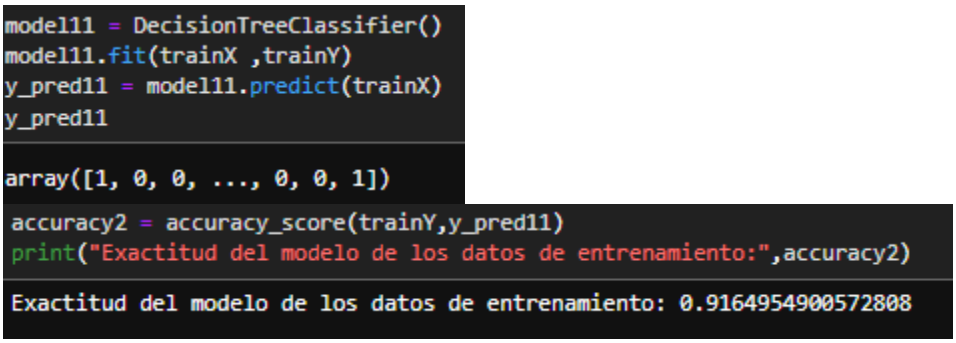
4.1. Resultados Obtenidos

Con el modelo previamente seleccionado según sus métricas de evaluación fue el de Arboles de Decisión con una precisión del 91,63% para los datos de prueba, lo que indica que el modelo tiene una precisión muy buena al momento de realizar la predicción de los clientes que pueden llegar a fugarse.

De igual forma se aplicó el modelo al conjunto de datos de entrenamiento obteniendo una precisión de 91.64% como se puede observar en la figura 52, lo que nos da a entender que un resultado de precisión alto en ambos conjuntos de datos sugiere que el modelo ha aprendido patrones y características relevantes para la clasificación de clientes en fuga.

Figura 52

Validación del Modelo datos de Entrenamiento



Nota. Gráfico que muestra el código ejecutado en el lenguaje Python con el que genera la evaluación para el modelo para árboles de decisión con los datos de entrenamiento, autoría propia, 2023.

Sin embargo, es importante aclarar que el rendimiento en los datos de prueba es decisivo para evaluar la capacidad de generalización del modelo. Si el modelo tiene una precisión similar en los datos de entrenamiento y prueba, es una buena señal de que no está sobreajustando (overfitting) los datos de entrenamiento y el mismo es apto para generalizar bien a los datos no vistos.

Con lo mencionado, podemos asegurar que nuestro modelo es confiable en un 91,6% junto con su matriz de confusión y su curva ROC_AUC previamente evaluadas en el capítulo anterior, lo cual es aceptable para el negocio y en conjunto con el objetivo el cual planteamos al comienzo de este trabajo que es predecir una posible fuga de clientes en un determinado periodo de tiempo.

Pues bien, ahora aplicaremos el modelo a todos los datos para aplicar la predicción creando una nueva columna de nombre Prediccion_de_Churn al data set principal summary_df1, como podemos observar en la figura 53.

Figura 53

Creación de Columna de Predicción

```
model1.fit(X ,y)
y_pred_final = model1.predict(X)
summary_df1['Prediccion_de_Churn']=pd.Series(y_pred_final)
summary_df1
```

	CODIGO	venta_neta_min	venta_neta_max	venta_neta_sum	venta_neta_avg	venta_neta_cantidad	fecha_min	fecha_max	fecha_recencia_de_compra	fecha_frecuencia_de_compra	r_quartile	f_quartile	m_quartile	rfm_score	filtro_score	churn	Prediccion_de_Churn
0	0000	0.071429	104.030000	2036.641745	8.854964	230	2022-05-01	2023-04-20	10	1.539130	1	1	1	111	0	0	0
1	00000	0.312500	64.236069	1470.711010	7.822931	188	2022-05-03	2023-03-23	38	1.723404	1	1	1	111	0	0	0
2	0000001	2.589286	50.148214	78.867500	19.716875	4	2022-10-22	2023-04-23	7	45.750000	1	3	2	132	0	0	0
3	000001	0.258929	8.173214	80.835711	2.526116	32	2022-06-26	2023-03-23	38	8.437500	1	1	2	112	0	0	0
4	000015738	12.550000	99.241072	294.811429	58.962286	5	2022-07-19	2023-03-27	34	50.200000	1	3	1	131	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1315818	YC084976	20.112143	72.912500	163.124643	40.781161	4	2022-09-08	2022-12-08	143	22.750000	4	2	2	422	0	0	0
1315819	YC362939	9.508929	30.824643	68.092500	17.023125	4	2022-11-03	2023-02-18	71	26.750000	1	2	3	123	0	0	0
1315820	YC931590	29.348214	33.671429	63.019643	31.509822	2	2022-08-24	2022-09-05	237	6.000000	4	1	3	413	0	1	1
1315821	YE201422	8.000000	204.707857	526.430002	87.738334	6	2023-01-15	2023-04-12	18	14.500000	1	1	1	111	0	0	0
1315822	Z6042363	4.600000	22.115714	76.044998	12.674166	6	2022-12-27	2023-03-28	33	15.166667	1	1	2	112	0	0	0

1315823 rows × 17 columns

Nota. Gráfico que muestra el código ejecutado en el lenguaje Python con el que se genera la nueva columna de predicción al data set principal summary_df1, autoría propia, 2023.

Posteriormente vamos a filtrar aquellos clientes cuya predicción es igual a 1 para poder iniciar el proceso de exportación de la base final creando un nuevo data set de nombre summary_churn_final, en la figura 54 observamos el resultado obtenido.

Figura 54

Creación de Data set Final

```
summary_df1_churn =summary_df1[(summary_df1['Prediccion_de_Churn']==1)]
summary_churn_final =summary_df1_churn[['CODIGO','Prediccion_de_Churn']]
summary_churn_final=summary_churn_final.reset_index(drop=True)
summary_churn_final
```

	CODIGO	Prediccion_de_Churn
0	00056150	1
1	000878649	1
2	0011178952	1
3	001698870	1
4	0023304516	1
...	...	...
145360	YB6506649	1
145361	YB704898	1
145362	YB8152338	1
145363	YC069658	1
145364	YC931590	1

145365 rows × 2 columns

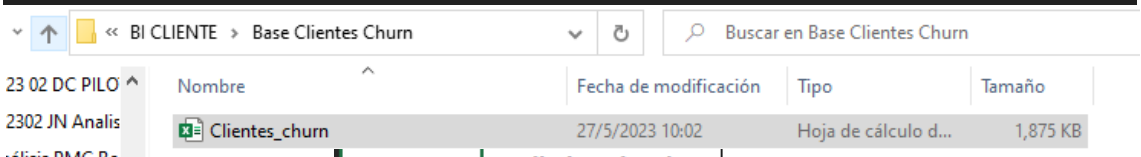
Nota. Gráfico que muestra el código ejecutado en el lenguaje Python con el que se genera el nuevo data set de nombre summary_churn_final, autoría propia, 2023.

Por último, exportamos el resultado final obtenido del data set anterior a un archivo de Excel para poder almacenarlo en una ruta compartida al cual tiene acceso el área de CRM para que pueda realizar las gestiones respectivas de personalización a cada uno de los clientes obtenidos por el modelo de predicción de fuga, lo expuesto se puede ver en la figura 55.

Figura 55

Archivo Final Obtenido

```
summary_churn_final.to_excel('P:\BI CLIENTE\Base Clientes Churn\Clientes_churn.xlsx',index=False)
```



CODIGO	Prediccion_de_Churn
00056150	1
000878649	1
0011178952	1
001698870	1
0023304516	1
0025564320	1
002871178	1
0043089689	1
005313	1
0057484438	1
006162	1
006442707	1
00661765	1
0089001	1
0100	1
0100000686	1
0100001155	1
0100001858	1
0100003466	1

Nota. Gráfico que muestra el código ejecutado en el lenguaje Python con el que se genera archivo final en Excel que es almacenado en una ruta compartida, autoría propia, 2023.

4.2. Discusión

En este capítulo, discutiremos los resultados y las conclusiones derivados del estudio de predicción de fuga de clientes utilizando el modelo de árboles de decisión el cual arrojo las mejores métricas. Como objetivo principal en este presente trabajo fue el desarrollar un modelo preciso y confiable que pudiera identificar a los clientes propensos a abandonar la marca.

Los resultados obtenidos fueron altamente prometedores, con una precisión del 91,63% para la predicción en la fuga de clientes. Este resultado demuestra la capacidad del modelo de árboles de decisión para distinguir entre los clientes que probablemente abandonarían la marca. Una precisión del 91,63% indica que el modelo es apto para poder predecir de manera correcta la fuga en una gran proporción de casos.

Además de la precisión, se evaluaron otras métricas de rendimiento, como la curva ROC_AUC y la matriz de confusión, para obtener una visión más completa del desempeño del modelo. Estas métricas también mostraron resultados alentadores, confirmando la eficacia del modelo en la detección de clientes propensos a la fuga.

Es importante destacar que la precisión obtenida debe considerarse en el contexto del problema y los datos utilizados. La calidad y la representatividad de los datos son factores clave que pueden afectar la precisión del modelo. Se llevaron a cabo esfuerzos significativos para recopilar y preparar los datos de manera adecuada, lo que contribuyó a la robustez del modelo.

Se identificaron características y patrones importantes que influyen en la fuga de clientes. Estos hallazgos pueden brindar información muy valiosa para tomar decisiones estratégicas y acciones preventivas. Por ejemplo, se observó que el tiempo de permanencia del cliente, el valor monetario gastado y la frecuencia con la que los clientes realizan una compra son factores clave que influyen en su propensión a abandonar.

A pesar de los resultados prometedores, también se reconocen las limitaciones del estudio. Por ejemplo, la precisión del modelo puede variar en diferentes contextos o industrias. Además, el cambio fluctuante del mercado junto con las preferencias del cliente pueden afectar la capacidad del modelo para mantener su precisión en el tiempo. Se recomienda realizar estudios de seguimiento y actualización periódica del modelo para garantizar su validez y relevancia continua.

Como resumen de lo expuesto hasta el momento podemos decir que el modelo de árboles de decisión confirma ser el más adecuado para predecir la fuga de los clientes, con una precisión del 91,63%. Este estudio ha proporcionado información valiosa en comprender que tipo de factores pueden influenciar en la fuga de clientes y puede servir de fuente para poder implementar estrategias en la retención de clientes más efectivas. Sin embargo, es fundamental tener en cuenta las limitaciones y considerar el contexto específico al aplicar estos resultados en la práctica empresarial.

CAPÍTULO V: CONCLUSIONES Y RECOMENDACIONES

5.1. Conclusiones

En este estudio, se utilizó el modelo de árboles de decisión para poder realizar la predicción de fuga de clientes con una precisión del 91,63%. Basado en los resultados y análisis realizados, podemos llegar a concluir lo siguiente:

- El periodo de tiempo elegido de 12 meses para poder recolectar los datos iniciales y poder evaluar al cliente en un rango de 180 días que haya superado desde su última fecha de compra fue adecuado para la generación del modelo.

- El modelo basado en árboles de decisión demostró ser eficaz en la predicción de la fuga de clientes. La precisión del 91,63% indica que este modelo es apto para identificar de una manera correcta a la mayor parte de los clientes propensos a abandonar la marca, antes de obtener dicho modelo se tuvo que aplicar otros tipos de modelado que incluyen la Regresión Logística y la Regresión Lineal.

- Las características utilizadas en el modelo, como la recencia, la frecuencia de compra y el valor monetario de las transacciones, resultaron ser importantes para predecir la fuga de clientes. Estas variables reflejan aspectos clave del comportamiento y como el cliente interactúa con la marca, en conjunto con el RFM_Score fueron determinantes para la culminación del presente trabajo.

- La interpretación de los resultados reveló que el tiempo que un cliente permanece en la marca y el nivel de satisfacción del cliente son factores críticos que influyen en su propensión a abandonar. Estos hallazgos nos sirven de fuente para poder elaborar estrategias de retención de clientes más efectivas.

- Podemos ver que existe una relación de sentido contrario entre la frecuencia de compra y la probabilidad de fuga de clientes. Los clientes que realizan compras con mayor frecuencia tienden a ser más leales y menos propensos a abandonar.

- El uso del modelo metodológico CRISP-DM en el desarrollo del modelo predictivo no solo fue eficiente, sino que también permitió organizar de manera estructurada y coherente los diferentes pasos del proceso, incluyendo la exploración de los datos, la planificación, la evaluación y ejecución. Además de ser ágil, el enfoque CRISP-DM facilitó la síntesis y el manejo ordenado de todos estos elementos, garantizando un desarrollo eficaz del modelo.

5.2. Recomendaciones

Con lo obtenido en base a los resultados en el presente trabajo, podemos proponer las siguientes recomendaciones:

- Utilizar el modelo de árboles de decisión como herramienta en la detección temprana de clientes propensos a la fuga. Esto permitirá a la empresa tomar medidas preventivas y diseñar estrategias personalizadas para retener a estos clientes.

- Realizar un seguimiento continuo y actualización periódica del modelo de predicción. El mercado cambiante en conjunto con las preferencias de los clientes pueden variar con el tiempo, por lo tanto, es importante mantener el modelo actualizado y ajustado a los nuevos datos y tendencias.

- Ampliar el conjunto de características utilizadas en el modelo. Además de las variables de recencia, frecuencia, valor monetario y RFM_Score, considerar otras variables relevantes, como la interacción del cliente en redes sociales, el comportamiento de navegación en el sitio web, o las respuestas a encuestas de satisfacción del cliente. Estas características adicionales pueden mejorar aún más la precisión del modelo.

- Complementar el modelo de predicción con estrategias de retención de clientes. Una vez identificados los clientes propensos a la fuga, implementar acciones personalizadas para mejorar su satisfacción, ofrecer incentivos o promociones especiales, o brindar un servicio al cliente excepcional. Esto ayudará a aumentar las posibilidades de retener a estos clientes y fortalecer su lealtad.

- Realizar estudios de seguimiento y evaluando el impacto sobre las estrategias de retención implementadas. Medir la efectividad de las acciones tomadas y realizar ajustes en función de los resultados obtenidos. Esto permitirá mejorar continuamente las estrategias y optimizar los esfuerzos de retención de clientes.

BIBLIOGRAFÍA

Amazon Web Services, I. (s.f.). *¿Qué es la regresión logística? - Explicación del modelo de regresión logística - AWS*. Amazon Web Services, Inc.: https://aws.amazon.com/es/what-is/logistic-regression/

Beatriz.Gil. (28 de 8 de 2019). *CRISP-DM: La metodología para poner orden en los proyectos*. Sngular: https://www.sngular.com/es/data-science-crisp-dm-metodologia/

BERSON, A. S. (2000). *Building data mining applications for CRM.* New York: McGraw-Hill.

Chapman, P. (9 de 2007). *Metodología CRISP-DM para minería de datos* . Dataprix: https://www.dataprix.com/es/book/export/html/107

datos.gob.es. (9 de 12 de 2020). *datos.gob.es*. ¿Cómo aprenden las máquinas? Machine Learning y sus diferentes tipos: https://datos.gob.es/es/blog/como-aprenden-las-maquinas-machine-learning-y-sus-diferentes-tipos

Ferrero, R. (5 de 2020). *Qué son los árboles de decisión y para qué sirven*. Máxima Formación: https://www.maximaformacion.es/blog-dat/que-son-los-arboles-de-decision-y-para-que-sirven/

Glotzer, S. (30 de 08 de 2022). *Guía de análisis RFM 2021 - Ejemplos de segmentación predictiva*. Barilliance: https://www.barilliance.com/es/guia-de-analisis-por-rfm-6-segmentos-clave-para-el-rfm-basado-en-marketing/

GPF, C. (2023). *Corporación GPF*. https://www.corporaciongpf.com/la-corporacion/

IBM. (2023). *Acerca de la regresión lineal*. México | IBM: https://www.ibm.com/mx-es/analytics/learn/linear-regression

J. Miranda, P. R. (2015). Predicción de Fugas de Clientes para una Institución Financiera Mediante Support Vector Machines. *Revista Ingeniería de Sistemas Volumen XIX*, 49-68.

Niño, M. (18 de 11 de 2016). *CRISP-DM: Fase de "Comprensión del negocio" (Business Understanding)*. El blog de Mikel Niño: https://www.mikelnino.com/2016/11/crisp-dm-metodologia-data-mining-comprension-negocio-business-understanding.html#:~:text=de%20cada%20fase.-,Fase%20de%20%E2%80%9CComprensi%C3%B3n%20del%20negocio%E2%80%9D%20(Business%20Understanding),preliminar%20para%20a

Ortega, C. (25 de 2 de 2023). *Análisis de regresión: Qué es, tipos y cómo realizarlo*. QuestionPro: https://www.questionpro.com/blog/es/analisis-de-regresion/

Peiró, R. (24 de 11 de 2022). *Churn*. Economipedia: https://economipedia.com/definiciones/churn.html

Robledano, A. (3 de 11 de 2022). *Qué es Python: Características, evolución y futuro*. OpenWebinars.net: https://openwebinars.net/blog/que-es-python/

Rodríguez, A. R., y Gallard, J. C. (2020). Uso del algoritmo Adaboost y la regresión logística para la predicción de fuga de clientes en una empresa de telefonía móvil. *Natura@economía, ISSN 2226-9479*, 102-117. https://doi.org/ http://dx.doi.org/10.21704/ne.v5i2.1610

Shin, T. (15 de 05 de 2020). *Comprensión de la Matriz de Confusión y Cómo Implementarla en Python*. DataSource.ai: https://www.datasource.ai/es/data-science-articles/comprension-de-la-matriz-de-confusion-y-como-implementarla-en-python

Simeone, O. (2018). A Very Brief Introduction to Machine Learning. *IEEE, 1*, 2. https://doi.org/1808.02342v4

Software, T. (s.f.). *¿Qué es la regresión logística?* TIBCO Software: https://www.tibco.com/es/reference-center/what-is-logistic-regress

SRINIVAS BANGALORE, P. H. (2017). THE FUNDAMENTALS OF MACHINE LEARNING. *Interactions*, 5-7.

Team, V. (18 de 06 de 2021). *Demo - Scoring De Clientes | Blog Visionarios*. Visionarios: https://blogvisionarios.com/impulsa-tu-negocio/casos-de-uso/scoring-clientes/

Venturini, S. (2016). *Cross-Validacion for Predictive Analytics*. Milano R: http://www.milanor.net/blog/cross-validation-for-predictive-analytics-using-r/

ANEXOS

Anexo A. Carta de autorización de uso de datos en la empresa

CONVENIO DE CONFIDENCIALIDAD

Comparecen a la celebración del presente "Convenio de Confidencialidad":

Por una parte, el señor **NAVAS AYALA JOSE RICARDO**, con C.C. No. **0603885104**, por sus propios y personales derechos, en calidad de EMPLEADO de la compañía **PROVEFARMA** S.A. a quien en adelante y para efectos del presente contrato, se la denominará simplemente como "EL EMPLEADO"; y,

Por otra parte, la señora **HELLEN ANDERSEN JIRON**, en su calidad de APODERADA ESPECIAL de la Compañía **PROVEFARMA S.A.**, a quien en adelante y para efectos del presente convenio, se la denominará simplemente como "LA COMPAÑÍA".

EL EMPLEADO Y LA COMPAÑÍA (en adelante las PARTES), en forma libre y voluntaria, por sus propios derechos y en las calidades invocadas de representación, suscriben el presente convenio de confidencialidad al tenor de las siguientes cláusulas:

PRIMERA: ANTECEDENTES.-

1.- Con fecha 14 de diciembre del 2020 LA COMPAÑÍA contrató los servicios lícitos y personales del Sr. **NAVAS AYALA JOSE RICARDO** a fin de que desempeñe las funciones de **COORDINADOR DE BI.**

2.- EL EMPLEADO ha solicitado a la COMPAÑÍA que se le permita acceder a la siguiente información **"INFORMACIÓN DE ID DE CLIENTES"** con la finalidad de llevar a cabo el **"PROYECTO DE INVESTIGACIÓN MODELO DE PREDICCIÓN DE FUGA DE CLIENTES"** toda vez que la información a la que se le dará acceso al EMPLEADO es catalogada como información secreta y confidencial que es propiedad exclusiva de LA COMPAÑÍA, empresas relacionadas, asociados, proveedores, y/o de terceros que hubieren confiado información confidencial a LA COMPAÑÍA, misma que no se encuentra disponible para el público en general, EL EMPLEADO mediante el presente se compromete al cumplimiento de lo estipulado en el presente instrumento.

SEGUNDA: CONFIDENCIALIDAD.-

En orden de mantener la confidencialidad de la información, el EMPLEADO reconoce que:

1.- La información confidencial a la cual tendrá acceso forma parte del patrimonio de la COMPAÑÍA.

2.- Acepta que por la especialidad de la información a la que tendrá acceso en el desempeño de sus tareas, en principio debe considerar confidencial toda la información, por vía escrita, oral o electrónica, a la que accederá. Particularmente, se obliga a guardar secreto de toda aquella información que se refiera directa o indirectamente al desarrollo informático, a su método de trabajo, a los negocios, a la organización interna de la COMPAÑÍA o de los clientes, proveedores de la misma o de los terceros que le comunicaron secretos en general.

3.- EL EMPLEADO, conoce que la propiedad de la información entregada a éste, pertenece a la COMPAÑÍA, en todos sus derechos, por lo que se obliga a no divulgarla ni revelarla en forma alguna, información, datos, especificaciones, técnicas, secretos, métodos, sistemas y en general cualquier mecanismo relacionado con la información y tecnología a la cual tuvo acceso, sujetándose a las responsabilidades que marca la ley en caso de contravenir lo dispuesto en esta cláusula.

En este sentido, se entenderá como confidencial toda la información que EL EMPLEADO reciba u obtenga acceso por cualquier forma, como consecuencia de las labores específicas que realizó en ejercicio del cargo que ostentó, ya sea en forma directa o indirecta, escrita, oral o de cualquier otra forma o por cualquier otro medio posible.

4.- EL EMPLEADO se compromete a no divulgar cualquier tipo de información a la que haya tenido acceso, misma que será mantenida con carácter reservado y estrictamente confidencial y que bajo ningún concepto podrá ser vendida, comercializada, publicada ni revelada de ningún modo a terceros, incluyendo fotocopias o reproducciones.

En concordancia con esto, EL EMPLEADO se compromete a no realizar a terceras personas ningún comentario, acotación, advertencia u observación sea escrita o verbal sobre la información a la que la COMPAÑÍA le dio acceso.

EL EMPLEADO declara también conocer que cualquier revelación de la información confidencial a la que tuvo acceso, podrá ser objeto de responsabilidad civil, administrativa e incluso se podrá constituir en un ilicito de naturaleza penal, de acuerdo a lo que establece la Ley, sujetándose en consecuencia a las sanciones que para dichos ilícitos prevé el Código Orgánico Integral Penal COIP. De la misma forma se acuerda que de verificarse la existencia de una revelación no autorizada, EL EMPLEADO pagará por concepto de penalidad la cantidad de Dos Cientos Mil Dólares de los Estados Unidos de América (USD.200.000,oo), sin perjuicio de las acciones por daños y perjuicios que la COMPAÑÍA pueda entablar de conformidad con la Ley aplicable.

TERCERA: PROPIEDAD DEL PRODUCTO DEL TRABAJO.-

Todos los derechos de propiedad intelectual e industrial sobre los programas de computación, descubrimientos, invenciones, ideas, conceptos, diseños, mejoras de cualquier tipo, patentes, modelos industriales o de utilidad, presentaciones comerciales, etiquetas, planes de marketing, estrategias, proyectos comerciales, datos técnicos, know-how, bocetos, dibujos de ingeniería, antecedentes y, en general, cualquier información que tenga valor e incidencia comercial, legal o administrativa, así como aquellas actividades que se encuentren contenidas o relacionadas o sean consecuencia o el resultado del trabajo realizado corresponden exclusivamente a la COMPAÑÍA.

EL EMPLEADO realizará la devolución de la información que le hubiere sido entregada, hecho que este se compromete a realizar de forma inmediata a la suscripción de este convenio, esta devolución deberá realizarse por escrito, mediante un Acta de Entrega Recepción que contendrá el detalle de la documentación e información devuelta.

CUARTA. - PROTECCIÓN DE DATOS PERSONALES:

Las Partes expresamente reconocen que, para el desarrollo y cumplimiento de los objetivos del presente Convenio, deberán entregar a la otra, información confidencial de distinta naturaleza, que comprende, además, datos de carácter personal, que han sido recolectados, almacenados, comunicados, transferidos, y en general, tratados en estricta observancia de las disposiciones contenidas en la Ley Orgánica de Protección de Datos Personales.

En caso de presentarse una vulneración de seguridad de los datos personales que se encuentren bajo el control o responsabilidad de una de las partes, aquella deberá informar de manera inmediata a la parte que hubiere transferido los datos inicialmente, sin perjuicio de las obligaciones establecidas en la Ley de notificación al titular y a la Autoridad de Control.

QUINTA.- VIGENCIA DE OBLIGACIONES:

LAS PARTES de común acuerdo expresan que el presente convenio tendrá vigencia de 10 años, contados a partir de la suscripción del presente instrumento. Cada parte asume y reconoce la responsabilidad que pudiera hacerse acreedor por el mal uso o divulgación de la información.

SEXTA.- DUDA O DISCREPANCIAS:

En caso de existir dudas o discrepancias en cuanto a si alguna información es un secreto comercial o laboral y, por lo tanto, si ésta se encuentra sujeta a los términos del presente convenio de voluntades, ésta deberá de ser tratada como confidencial y por ende, estará sujeta a los términos de este convenio y a lo establecido en la Ley Orgánica de Protección de Datos Personales, Código Orgánico de la Economia Social de los Conocimientos Creatividad e Innovación y demás normativa conexa.

SEPTIMA.- NORMATIVA APLICABLE:

El presente Acuerdo se rige y deberá ser interpretado conforme a las disposiciones contenidas en la Decisión 486 de la Comunidad Andina de Naciones, Ley Orgánica de Regulación y Control del Poder de Mercado y demás normativa legal, civil y/o penal contemplada en la legislación ecuatoriana.

OCTAVA.- JURISDICCIÓN Y COMPETENCIA:

Cualquier controversia que surja en relación con este acuerdo, se someterá en forma obligatoria a mecanismos alternativos de solución de conflictos, tales como: a) Por el acuerdo transaccional de las partes. B) En caso de persisitr la controversia, este se someterá a la decisión de un Tribunal de Arbitraje del Centro de Arbitraje y Mediación AMCHAM de la Cámara de Comercio Ecuatoriano Americana con sede en la ciudad de Quito, para su organización y funcionamiento se sujetará a lo dispuesto en la Ley de Arbitraje y Mediación, al Reglamento del Centro de Arbitraje y Mediación y a las siguientes normas:

1. El Tribunal estará conformado por un solo árbitro. El árbitro será seleccionado de la lista de Árbitros del Centro de Arbitraje y Mediación, en la ciudad de Quito por sorteo dirigido por el Director del Centro;
2. Las partes se comprometen a aceptar el laudo arbitral que se expida;
3. El procedimiento arbitral será confidencial;
4. El Árbitro fallará en Derecho;
5. Para la ejecución de medidas cautelares, el Árbitro está facultado para solicitar de los funcionarios públicos, judiciales, policiales y administrativos su cumplimiento, sin que sea necesario recurrir a juez ordinario alguno;
6. El lugar de arbitraje será las instalaciones del Centro de Arbitraje y Mediación AMCHAM de la Cámara de Comercio Ecuatoriano Americana con sede en la ciudad de Quito.

Las PARTES acuerdan que todos y cada uno de los costos que se generen por el procedimiento arbitral, tales como derechos del Centro de Arbitraje y Conciliación, honorarios de peritos, árbitros, abogados, etc., serán asumidos íntegra y totalmente por la parte que incumplió o vulneró este Convenio de Confidencialidad

NOVENA.- ACEPTACIÓN:

Las partes conociendo el contenido y alcance del presente convenio, lo aceptan de forma expresa completa e íntegra, mismo que sustituye e invalida todas las comunicaciones, entendimientos y acuerdos anteriores entre las Partes, ya sean escritos, orales, expresos o implícitos con respecto a la revelación de información, prueba de lo cual suscriben en unidad de acto en Quito, a los 24 días de enero del 2023.

LA COMPAÑIA

SRA. HELLEN ANDERSEN JIRON
APODERADA ESPECIAL
Provefarma S.A.
Ruc: 1791050665001

EL EMPLEADO

NAVAS AYALA JOSE RICARDO
C.C. 0603885104

Anexo B. Sentencia SQL utilizada en Python para la carga inicial de datos.

```
select
distinct FECHA,IDENTIFICACION_FINAL as CODIGO
,sum(cantidad)CANTIDAD,sum(venta_neta)VENTA_NETA
from BI_VENTAS_DETALLE_FYBECA
where FECHA between '2022-05-01' and '2023-04-30' and CANTIDAD >0 and
IDENTIFICACION_FINAL not in ( '9999999999',
'N/D','1791257049001','1791927559001','1791279352001','0990017514001',
'1790475247001','1791988558001','1792091705001','1790093808001',
'1791415132001','1111111111','0991189270001','7777777770','202020',
'CONSUMIDOR','0992621915001','05219B0010','1792206979001','0992794127001',
'2002011787','2002012041','2002012042','1777778888','8989898989',
'8120104060','2002012045','1792348684001','2002012076','01033358400',
'1792493056001',' '  ) and length(IDENTIFICACION_FINAL)<=10
group by 1,2
```

Anexo C. Creación de funciones para el cálculo de las columnas de recencia y frecuencia en el lenguaje de Python.

```
fecha_max_analisis=df['FECHA'].max()
def groupby_mean(x):
    return x.mean()

def groupby_count(x):
    return x.count()

def purchase_duration(x):
    return (fecha_max_analisis - x.max()).days

def avg_frequency(x):
```

```
    return (x.max() - x.min()).days/x.count()


groupby_mean.__name__ = 'avg'

groupby_count.__name__ = 'cantidad'

purchase_duration.__name__ = 'recencia_de_compra'

avg_frequency.__name__ = 'frecuencia_de_compra'
```

Anexo D. Código en lenguaje de Python para el filtrado de valores mayores a 0 para el data frame summary_df1 basado en las columnas recencia y frecuencia de compra.

```
  summary_df1= (summary_df[summary_df.fecha_recencia_de_compra>0])

  summary_df1=(summary_df1[summary_df1.fecha_frecuencia_de_compra>0])

  summary_df1=summary_df1.reset_index(drop=True)

  summary_df1
```

Anexo E. Código en lenguaje de Python para la creación de los cuartiles por recencia, frecuencia y valor monetario.

```
quantiles = summary_df1.quantile(q=[0.75,0.5,0.25])

quantiles = quantiles.to_dict()


def r_score(x,p,d):
    if x<=d[p][0.75]:
        return 1
    elif x<=d[p][0.50]:
        return 2
    elif x<=d[p][0.25]:
        return 3
    else:
        return 4
```

```
summary_df1['r_quartile'] = summary_df1['fecha_recencia_de_compra'].apply(r_score, args=('fecha_recencia_de_compra',quantiles,))

summary_df1['f_quartile'] = pd.qcut(summary_df1['fecha_frecuencia_de_compra'], q=4, labels=range(1,5))

summary_df1['m_quartile'] = pd.qcut(summary_df1['venta_neta_sum'], q=4, labels=False)
#summary_df1['venta_neta_sum'].apply(r_score, args=('venta_neta_sum',quantiles,))#

summary_df1['m_quartile'] = summary_df1['m_quartile'].max() -summary_df1['m_quartile']+1
```

Anexo F. Código en lenguaje de Python para la creación de la columna rfm_score basada en los cuartiles anteriormente creados.

```
summary_df1['rfm_score'] = summary_df1['r_quartile'].astype(str) +
summary_df1['f_quartile'].astype(str) + summary_df1['m_quartile'].astype(str)

summary_df1['rfm_score'] = summary_df1['rfm_score'].astype(int)

summary_df1.head(100)
```

Anexo G. Código en lenguaje de Python para la creación de la columna filtro_score, para aquellos clientes que cumplan el perfil de clientes durmientes según los rangos del RFM_Score.

```
summary_rfmscore=summary_df1[summary_df1['rfm_score'].astype(str).str.startswith('43','44')]

summary_df1['filtro_score']=0 #inicializamos con 0 para todos los registros

summary_df1.loc[summary_df1['rfm_score'].isin(summary_rfmscore['rfm_score']),'filtro_score']
=1

summary_df1.head(100)
```

Anexo H. Código en lenguaje de Python para la creación de la columna churn necesario para comenzar a modelar fijando el segmento de recencia de compra mayor a 180 días.

```
summary_df1['churn'] = ((summary_df1['fecha_recencia_de_compra'] >180) ).astype(int)

summary_df1
```

Anexo I. Código en lenguaje de Python para la división de la data en 70/30 para entrenamiento y prueba de los modelos a evaluar.

```
X = summary_df1[['r_quartile','f_quartile','m_quartile','filtro_score']]
#'r_quartile','f_quartile','m_quartile'

y = summary_df1['churn']

trainX, testX, trainY, testY  = train_test_split(X, y, test_size=0.3)
```

Anexo J. Código en lenguaje de Python para la creación del modelo de regresión logística.

```
model = LogisticRegression()

model.fit(trainX ,trainY)

y_pred = model.predict(testX)

y_pred
```

Anexo K. Código en lenguaje de Python para la creación del modelo de Árboles de Decisión.

```
model1 = DecisionTreeClassifier()

model1.fit(trainX ,trainY)

y_pred1 = model1.predict(testX)

y_pred1
```

Anexo L. Código en lenguaje de Python para la creación del modelo de Regresión Lineal.

```
lm = LinearRegression()

lm.fit(trainX, trainY)

predictions = lm.predict(testX)

predictions
```

Anexo M. Código en lenguaje de Python para la creación del campo final de nombre Prediccion_de_Churn al data set principal summary_df1.

```
model1.fit(X ,y)

y_pred_final = model1.predict(X)

summary_df1['Prediccion_de_Churn']=pd.Series(y_pred_final)

summary_df1
```

Anexo N. Código en lenguaje de Python para la separación de los clientes que en la columna Prediccion_de_Churn tuvieron valor de 1 y colocarlos en un nuevo data set, junto con la columna CODIGO de nombre summary_shurn_final.

```
summary_churn_final =summary_df1_churn[['CODIGO','Prediccion_de_Churn']]

summary_churn_final=summary_churn_final.reset_index(drop=True)

summary_churn_final
```

Anexo O. Código en lenguaje de Python para la exportación del data set anterior de nombre summary_shurn_final a una ruta compartida.

```
summary_churn_final.to_excel('P:\BI CLIENTE\Base Clientes
Churn\Clientes_churn.xlsx',index=False)
```