UNIVERSIDAD INTERNACIONAL DE LA RIOJA

**Universidad Internacional de La Rioja**

**Escuela Superior de Ingeniería y Tecnología**

**Máster Universitario en Análisis y Visualización de Datos Masivos**

# Análisis de Productos Financieros Basado en CRM para Banco Atlántida

**Trabajo Fin de Máster**

**Tipo de trabajo:** Piloto Experimental

**Presentado por:** Martínez Cerrato, Shirley Claudette

**Director:** Cervantes Rovira, Alejandro

# Resumen

En la actualidad cada vez es más importante ofrecer experiencias personalizadas a los clientes en cualquier sector económico, estas experiencias se pueden construir en base a diferentes factores de los clientes como ser nivel socioeconómico, hábitos de compras o preferencias personales.

En la realización del presente trabajo se realizará un análisis de productos financieros que ayude a determinar, a través de distintas técnicas de inteligencia artificial, que productos son más propensos para ser adquiridos por los clientes de una institución bancaria en Honduras, también se tratará de identificar que clientes son más propensos a abandonar determinados productos que ya han adquirido anteriormente para impulsar campañas de retención de productos. Los datos utilizados en los diferentes algoritmos serán extraídos de un sistema CRM e información histórica de la tenencia de los diferentes productos por parte de los clientes, esta extracción de datos se realizará a través de consultas SQL de esta forma se buscará obtener beneficios de los datos demográficos que el sector bancario suele almacenar de las personas y por ende mejorar la promoción de su cartera de productos.

**Palabras Clave:** CRM, Almacén de Datos, SQL, KDD, Agrupamiento, Abandono de productos, RFM

# Abstract

Nowadays every time is more important to offer personalized experiences to customers in any economic sector, these experiences can be built based on different customer factors such as socioeconomic status, shopping habits or personal preferences.

In the elaboration of this actual work will be performed an analysis of financial products that helps to determine, through different artificial intelligence techniques, which products are more likely to be acquired by the customers of a banking institution in Honduras, it will also try to identify which customers are more likely to abandon certain previously acquired products to boost product retention campaigns. The data used in the different algorithms will be extracted from a CRM system and historical information about the ownership of the different products by customers, this data extraction will be done by SQL queries, in this way it will aim to obtain benefits from the demographic data that the banking sector usually stores of people and therefore improve the promotion of their product portfolio.

**Keywords:** CRM, Data Warehouse, SQL, KDD, Clustering, Product Churn, RFM

# Índice de contenidos

# Índice de tablas

# Índice de figuras

# 1. Introducción

Banco Atlántida es una institución financiera fundada en Honduras el 10 de febrero de 1913, actualmente cuenta con un gran abanico de productos y servicios que están disponibles para sus clientes y el público en general, sumado a la solidez con la cual se caracteriza, estos factores le han permitido consolidarse como uno de los bancos más importantes a nivel nacional.

En Honduras las distintas instituciones del sector bancario históricamente han elaborado campañas de promoción de sus productos financieros a través de métodos tradicionales como ser listados de personas que se encuentran con un historial crediticio aceptable, independientemente las características y gustos personales de cada cliente existente o potencial. Debido a estos métodos simplistas utilizados para impulsar sus productos, cada vez es más difícil para las instituciones bancarias que los clientes estén dispuestos a adquirir los productos que estas les ofrecen, otro problema que se genera de dicha situación es que muchos de los productos que logran ser colocados por los diversos gestores de ventas son abandonados o cancelados al poco tiempo por los clientes, pues muchas veces estos productos no satisfacen sus necesidades financieras según sus preferencias personales.

En base a lo anterior surge una necesidad de eficientizar la forma en que las empresas hondureñas del sector bancario ofrecen sus productos a la población, para ello se pretende hacer uso de la inteligencia artificial y el machine learning, ya que hoy en día han cobrado gran auge a nivel global como un pilar fundamental en varios procesos del sector bancario, de hecho un estudio efectuado por Purdy, M., & Daugherty, P. en 2016, revela que la inteligencia artificial y las técnicas de machine learning dictarán como los bancos interactuaran con sus cliente en el futuro.

## 1.1 Justificación

Los dos problemas principales que se pretenden solventar son la falta de precisión al momento de enfocar las campañas de colocación de productos y la pronta cancelación de los productos recientemente adquiridos por los clientes.

En el caso de la aquerencia de productos, Banco Atlántida de Honduras suele dividir sus clientes en tres bancas principales, cada banca de cliente tiene disponibles productos que son comunes a todas las bancas, así como productos que son pensados para un tipo de banca en específico. Para el caso de los productos que son exclusivos para un solo tipo de banca, el listado de clientes potenciales se reduce drásticamente, si la banca a la cual pertenece el

producto posee una cartera de clientes reducida, el problema radica cuando el producto a colocar puede ser adquirido por más de una banca y por ende posee un amplio repertorio de posibles clientes, en estos casos es necesario enfocar la promoción del producto que se desea ofrecer a un conjunto de clientes que tengan mayor posibilidad de aceptar la oferta de adquisición inicial, de esta forma se evitara desperdiciar tiempo y esfuerzo invertido en la etapa de contacto y negociación de un producto bancario determinado.

Para el caso del abandono o cancelación de un producto en específico, en la banca hondureña en general, no se suele llevar un control minucioso de las causas por las cuales los clientes deciden cancelarlo, tampoco se ha trabajo en identificar que segmento especifico de clientes en concreto decide abandonar un producto, este inconveniente complica enfocar a los clientes correctos las campañas de fidelización y retención de productos, las cuales a largo plazo representarían un aumento en las utilidades de la empresa.

Por medio de la información histórica que se guarda de los clientes en el Data Warehouse, se puede tratar de identificar a través de la agrupación de clientes similares, que productos en común poseen dichos clientes y cuales productos tienen un menor nivel de tenencia, también se pueden aplicar técnicas de clasificación para conocer que clientes abandonarán un producto de forma relativamente inmediata.

## 1.2 Planteamiento del trabajo

En el presente trabajo se propone elaborar un análisis de los diversos productos del banco para la banca de personas específicamente, el cual estaría encaminado a determinar la colocación y retención de productos financieros. Esta meta se puede lograr haciendo uso de técnicas de inteligencia artificial que utilicen la información bancaria sobre los clientes que se encuentra almacenada en el datawarehouse empresarial y que es extraída del sistema CRM de la empresa.

Los sistemas CRM suelen almacenar una gran cantidad de información concerniente a los clientes, mucha de esta información la constituyen datos de tipo demográfico, los cuales son datos generales sobre las personas como ser edad, genero, estado civil, nivel educativo. El cometido principal es dar un uso útil a la información almacenada en este tipo de sistema, de forma que al final represente un beneficio para la empresa.

Para poder lograr el cometido propuesto en este trabajo se pretende identificar inicialmente las variables más representativas de los clientes a nivel demográfico dentro de la base de datos, luego se desea desarrollar algoritmos de inteligencia artificial que permitan identificar de manera más precisa los clientes que serán más propensos a adquirir o abandonar un

producto, los algoritmos a aplicar lo conforman las técnicas de agrupamiento y clasificación. Por último, se desea medir el nivel de eficacia de los algoritmos desarrollados a través de diferentes métricas de evaluación.

La obtención de los datasets para poder llevar a cabo estos cometidos se realizará haciendo uso del lenguaje SQL, con lo cual se crearán consultas que obtendrán los datos deseados del Data Warehouse empresarial, esto resultará en la novedad de tener datasets personalizados para cada caso, según se configuren los parámetros deseados en las consultas SQL.

## 1.3 Estructura de la memoria

El presente trabajo está constituido por 6 apartados principales, en cada uno de estos apartados se desglosa un aspecto importante para la correcta compresión de la propuesta que se está presentando.

En el capítulo 1 se ofrece una introducción sobre el tema a tratar, aquí se expone el contexto sobre el cual se estará desarrollando la parte práctica de la investigación, con lo cual se busca dar una idea rápida de la intención final del documento al lector. Este capítulo se divide en tres subapartados en los cuales se especifica con mayor medida los objetivos al alcanzar, los procesos y herramientas que será utilizados para alcanzar esos objetivos y la importancia de la propuesta en general.

En el capítulo 2 se expone el contexto y estado del arte sobre el tema a tratar, este apartado consiste exclusivamente en explicar por medio de estudios previamente efectuados y literatura ya existente la propuesta de trabajo a realizar. Con esta sección se fundamenta de forma teórica la importancia del trabajo actual, esta sección es de vital importancia ya que, a través de los trabajos y desarrollos previamente realizados sobre el tema a desglosar, se establece un pivote o punto de partida con el cual será más sencillo lograr los objetivos finales del trabajo.

El capítulo 3 abarca los objetivos y la metodología de trabajo a emplear. En este apartado se expondrán la finalidad principal que se busca alcanzar con el desarrollo del trabajo actual, se propondrá un objetivo principal, el cual deberá ser logrado al final del desarrollo del análisis de los productos financieros con técnicas de Machine Learning, también se plantearán una serie de objetivos secundarios, los cuales ayudarán a fortalecer el alcance del objetivo principal. Este apartado también contiene la explicación de la metodología a utilizar, con ello se buscará definir el proceso de desarrollo del análisis a efectuar, así como las herramientas,

lenguaje de programación y librerías a utilizar para completar dicho desarrollo, también se definen las métricas y métodos a utilizar para medir el nivel de precisión y efectividad de los modelos predictivos creados.

En el capítulo 4 se explica el desarrollo práctico del trabajo final, esto incluye descripción de la forma de conexión a la fuente de datos de donde se extraerá la información y la forma de procesar la misma, obtención de los distintos conjuntos de datos a través de consultas SQL, explicación de los algoritmos utilizaos y cálculo de las métricas para medir la precisión de los algoritmos empleados.

En el capítulo 5 se aborda el tema de conclusiones y trabajo futuro. Aquí se deberá especificar cuáles fueron los resultados obtenidos del análisis previamente efectuado, se deberá exponer en cual medida los objetivos expuestos previamente fueron alcanzados con el resultado final, también se elaborará un resumen final de todo el proceso del análisis de la información donde se describirá los inconvenientes surgidos durante las fases de obtención de los datasets finales hasta la fase se obtención de las métricas de precisión y los hallazgos encontrados. Este apartado también abarca las líneas a futuro del trabajo efectuado, en donde se planteará las mejoras que se podrían hacer a los algoritmos y consultas SQL utilizadas para obtener los datos finales, así como su aplicación en diversas áreas por medio de la adecuación de mejoras que permitan utilizar el código escrito para desarrollar aplicaciones de IA Bancaria.

En el capítulo 6 se especifica la bibliografía consultada para poder elaborar el trabajo final actual, está bibliografía consiste mayoritariamente en papers de carácter científico y académico, así como otras fuentes como ser libros y en última instancia sitios web que abordan la materia estudiada.

# 2. Contexto y estado del arte

## 2.1. CRM (Customer Relationship Management)

Las CRM traducidas como gestión de relaciones con el cliente, vistas desde el lado de la informática se tratan de Software para la administración o gestión de la relación con los clientes que se integran en los llamados Sistemas de Gestión Empresarial (SGE). Actualmente, las empresas almacenan y reciben grandes cantidades de datos a través de correos electrónicos, sesiones de chat en línea, llamadas telefónicas entre otros. Sin embargo, muchas empresas no hacen un uso adecuado de esta gran cantidad de datos pese a haber mucha teoría que trata sobre el manejo de estos datos. Uno de los modelos para el manejo adecuado de CRM es el modelo RATER (Reliability, Assurance, Tangibles, Empathy, Responsiveness) que surge de mejor el modelo SERVQUAL que trabaja con 10 dimensiones. Las 10 dimensiones fueron restablecidas a la mitad dado la existencia de una superposición entre sus objetivos (Zeithaml et al., 1985). Las dimensiones como Assurance, Tangibles, Empathy tienen un carácter más interpersonal, las CRM tienen objetivos que solo son alcanzables con el uso de tecnologías, si bien para las CRM no son un fin sino un medio, sin el medio no se puede llegar al objetivo (Czaplewski et al., 2002). Relativo a el caso de estudio presente, sistemas de recomendaciones basados en las necesidades del cliente y la satisfacción con el servicio de recomendaciones, buscando evitar el hostigamiento y siendo capaz de sugerir solo los servicios que el cliente es propenso a adquirir.

Otro modelo es el modelo RFM (Recencia, Frecuencia y Monetario), en inglés Recency, Frequency, Monetary. El modelo RFM, es un modelo basado en el comportamiento utilizado para analizar el comportamiento de un cliente y luego hacer predicciones basadas en el comportamiento en la base de datos. Además, la recencia representa la duración de un período de tiempo desde la última compra, mientras que la frecuencia denota el número de compras dentro de un período de tiempo específico y monetario significa la cantidad de dinero gastado en este período de tiempo especificado. De hecho, estos tres factores pertenecen a las variables de comportamiento y pueden ser utilizados como variables de segmentación mediante la observación de los clientes actitudes hacia el producto, la marca, el beneficio o incluso la lealtad de la base de datos (Wei, J. T., Lin, S. Y., & Wu, H. H., 2010). El modelo RFM ayuda a establecer quienes son los mejores clientes, los clientes que tienen más posibilidades de abandonar los servicios buscando retener a ambos sin sacrificar a los más provechosos.

Al final los que todos estos modelos buscan es mejorar la tasa de respuesta en atención al cliente y la creación de campañas especiales para los grupos encontrados valiéndose de medios tradicionales como llamadas, anuncios e inclusive servicios remotos evitando la necesidad de encuestas sino más bien analizando los datos reflejados en el historial de los clientes de forma no invasiva sino estadística.

## 2.2. Historia del Aprendizaje Automático

El Machine Learning es un subcampo de la IA. Se puede definir a una IA como el producto final, mientras que el Machine Learning son los métodos y algoritmos internos que dotan a una IA con la capacidad de aprender y tomar decisiones. En 1959 Arthur Samuel define al Machine Learning como; *"la rama del campo de la IA, que busca como dotar a las máquinas de capacidad de aprendizaje"* (*Samuel, A., 1959*).

Samuel es el precursor de las IA en juegos de estrategia creando el primer programa en jugar a las damas con el primer ordenador comercial disponible, el IBM 701. Él detalla sus investigaciones en su artículo *"Some Studies in Machine Learning Using the Game of Checkers"* y hay que destacar el alcance de sus métodos de aprendizaje. Se escogió el juego de damas por sus reglas relativamente simples, pero a su vez con estrategias profundas. En lugar de buscar cada camino hasta llegar al final del juego y teniendo en cuenta la muy limitada memoria de computadora disponible, Samuel implementó un árbol de búsqueda con una función de puntuación basada en la posición del tablero en un momento dado, tomando en cuenta el número de piezas en cada lado y la interacción entre ellas. Esta función mide la posibilidad de ganar para cada lado en la posición dada. El programa elige su movimiento basado en una estrategia minimax, lo que significa hacer un movimiento que optimiza el valor de esta función, asumiendo que el oponente estaba tratando de optimizar el valor de la misma función desde su punto de vista. Esto se conoce actualmente como poda alfa-beta por reducir los nodos evaluados en un árbol de juego por el algoritmo Minimax y es muy utilizada en IA de ajedrez, tres en raya o el Go (Richard S., 1990).

En su momento el programa de Samuel llego a superar a los jugadores de nivel medio en damas, pero casi 4 décadas más tarde la poda alfa-beta como método de aprendizaje le daría la victoria a Deep Blue (Hsu et al. *1995*) contra el campeón mundial de ajedrez de Gari Kaspárov en 1997 (Campbell, M., 1998). Este acontecimiento marca una antes y un después donde las máquinas superan a los humanos jugando ajedrez. En un contexto más actual el nivel ajedrecístico se mide con un cálculo estadístico conocido como Elo, el mejor Elo de Kaspárov que batió récord fue de 2851 (Bill W., 2008) superado en 2014 por Magnus Carlsen con 2882. Mientras tanto la IA Stockfish considerada la mejor en el ajedrez contaba con un

Elo de 3400 cuando perdió en 2017 contra otra IA AlphaZero que solo conocía las reglas del juego y había entrenado 4 horas contra sí misma jugando cerca de 5 millones de partida. AlphaZero es una IA basada en aprendizaje profundo que usa un árbol de búsqueda Monte Carlo (Silver et al. 2017).

## 2.3. Paradigmas del Machine Learning

Los paradigmas son un conjunto de teorías, estándares y métodos que permiten interpretar una materia o el mundo mismo a través de dichas ideas. Para el machine learning esto se traduce como los múltiples enfoques de aprendizaje y herramientas que se pueden implementar según lo que se desea que una maquina realice. Previamente se dio un ejemplo de un método de aprendizaje que enseño a la diferentes IA jugar damas, ajedrez entre otros y superaron a los grandes maestros humanos, sin embargo, en los juegos de mesa las instrucciones están bien definidas, así como las victorias, derrotas y empates, hay otros procesos de aprendizaje donde esas ideas no se ajustan al objetivo de estudio. La robótica por ejemplo es un subcampo del estudio de IA que llega a mezclarse con otros subcampos, pero por si sola es un área de estudio lo suficientemente extensa como para abordar temas fuera de aprender las reglas de un juego y no requerirlas jamás en un proyecto dado. En la Figura 1 podemos apreciar los distintos subcampos de la IA y el Machine Learning.

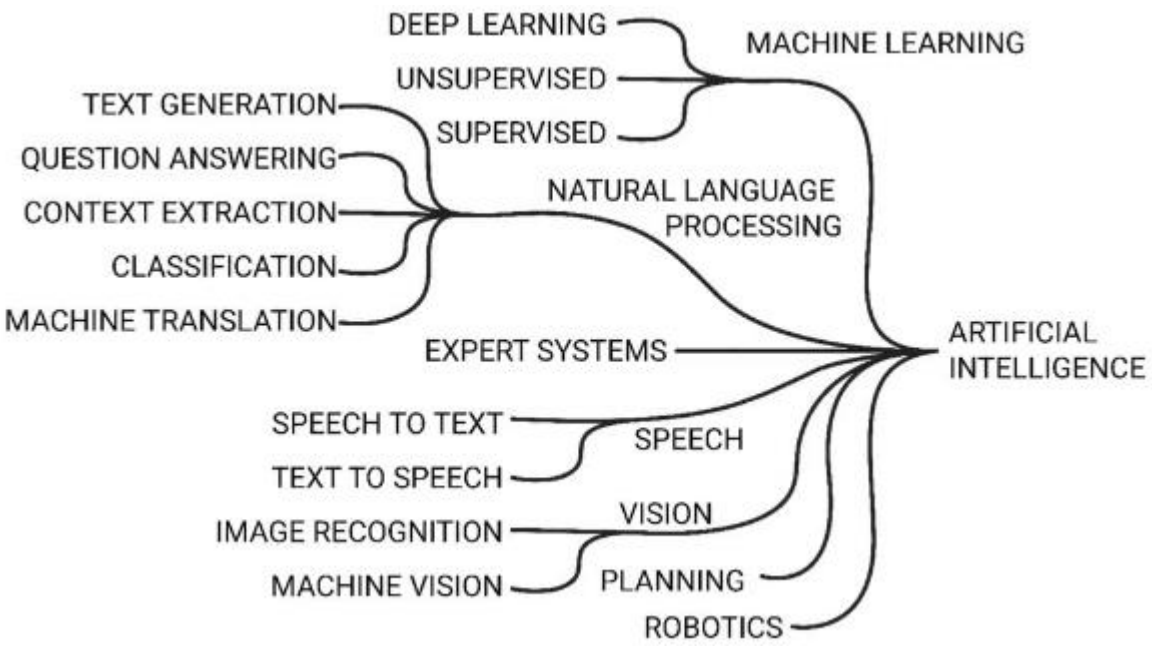



*Figura 1. Paradigmas de la inteligencia artificial y machine learning. Fuente: Panesar, 2019.*

En paralelo la programación tiene sus propios paradigmas sobre la estructura de las instrucciones y las variables que operan, uno de los más utilizados hoy día en la programación orientada a objetos, OOP por sus siglas en inglés (Object-Oriented Programming). Los

paradigmas de programación no son lo mismo que a los paradigmas del machine learning, aunque conceptos como clases abstractas u objetos se pueden abordar en las redes neuronales de una forma muy parecida a las relaciones entre clases en un OOP.

## 2.3.1. Aprendizaje Supervisado

Este enfoque se conoce de antemano que es lo que se quiere predecir, como los datos deben de estar previamente clasificados a su posterior utilización, este enfoque requiere de la intervención humana para su correcto funcionamiento.

### *2.3.1.1. CART Classification and Regression Trees*

Este método busca crear un modelo que de predice el valor de una variable de destino en función de diversas variables de entrada. Ilustradas como un árbol las variables de entradas hacen el papel de ramas que parten de la raíz del proceso y las hojas seria las variables de destino o predicciones (Figura 2). Árboles de Clasificación y Regresión (en español ACR) es un término genérico que aborda los dos tipos de árboles de decisión; Árboles de Clasificación y Árboles de Regresión. Su aparición se deriva de la publicación Classification and regression trees (Beiman et al. 1984) dada algunas similitudes en los métodos, sin embargo, cuentan con sus diferencias en la clasificación de los datos. Los Árboles de Clasificación crean clases de datos que contendrán los resultados de predicciones, ejemplo de ello la clase silla para el reconocimiento de imágenes. Los Árboles de Regresión buscan establecer los resultados como números reales, tales como pueden ser precios, días etc.

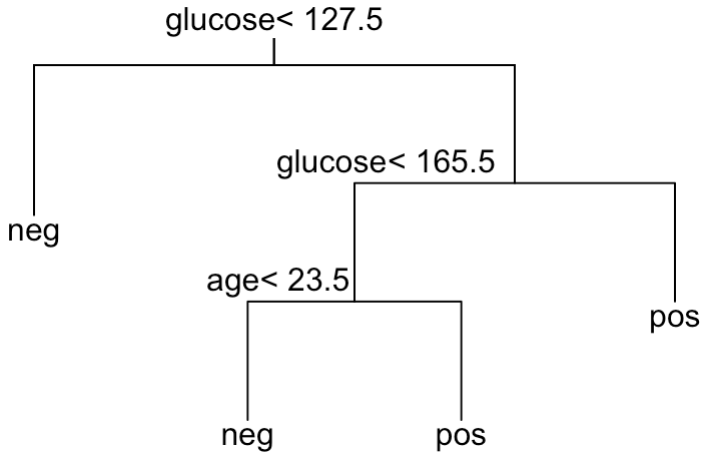



*Figura 2. Árbol de predicción según niveles de glucosa, elaborado partir de Kassambara A. 2017.*

### *2.3.1.2. Random Forest*

Es un método conjunto basado en árboles de decisión que busca mejorar una tasa de clasificación, en español, bosques aleatorios, se pueden entender como un conjunto enorme de árboles de predicción tal que cada uno depende de valores de un vector aleatorio probado

independientemente y con la misma distribución para todos ellos. La idea central es construir una larga colección de árboles no correlacionados y luego promediar sus resultados. Por sus similitudes se puede comparar con un meta-algoritmo (conjuntos de algoritmos) conocido como boosting que parte de una pregunta sencilla; ¿Puede un conjunto de clasificadores débiles crear un clasificador robusto? (Kearns et al. 1989). En cuanto a problemas de rendimiento random forest y boosting son también similares, pero random forest es más simple de entrenar y ajustar lo que en consecuencia lo hace más popular y ampliamente utilizado.

Uno de los problemas en los datos es conocidos como ruido, información que se sale de los parámetros y es difícil interpretar. Los árboles de predicción pueden llegar a ser realmente grandes teniendo relativamente baja parcialidad o sesgo. Producto de su tamaño los árboles son ruidosos, por lo cual un método de promediar el ruido les beneficia, para ello se utiliza una idea llamada bagging, bolsa, (Breiman L. 2001) que promedia modelos ruidosos, pero aproximadamente libres de sesgos.

### *2.3.1.3. Regresión Logística*

Su nombre sugiere que es un método de regresión, pero para el machine learning no es un algoritmo aplicado a problemas de regresión sino un algoritmo de clasificación. Al igual que otros métodos estadísticos depende de variables independientes y en particular su variable dependiente es del tipo binaria con los estados 0 y 1. La función que relaciona la variable dependiente con las independientes es una función sigmoidea, es decir una curva en forma de S (Figura 3), que puede tomar cualquier valor entre 0 y 1, pero nunca valores fuera de estos límites. La ecuación que define la función es f(x) = \frac(1){1 + e^(-x)} donde x es un número real. En la ecuación se puede ver que cuando x tiene a menos infinito el cociente tiende a cero. Por otro lado, cuando x tiende a infinito el cociente tiende a la unidad.

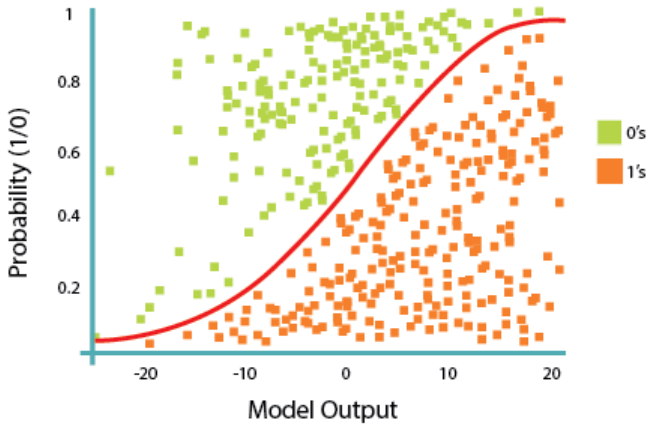



*Figura 3. Función sigmoidea, extraído de statdeveloper.com*

#### *2.3.1.4. K-Nearest-Neighbors*

Conocido en español como K-vecinos más cercanos. Es un algoritmo que se basa en la distancia, con lo cual asume que los puntos que se encuentran a una distancia menor son similares.

El aprendizaje con los métodos del vecino más cercano es diferente de otros métodos de aprendizaje automático en el sentido de que solo está almacenando todos o un subconjunto de ejemplos de aprendizaje. Cuando se presenta un nuevo ejemplo al predictor vecino más cercano, un subconjunto de ejemplos de aprendizaje similares al nuevo ejemplo es utilizado para hacer una predicción. Desde que la fase de aprendizaje casi no tiene lugar, los métodos del vecino más cercano a menudo se caracterizan como métodos de aprendizaje perezosos. La carga principal del cálculo se transfiere a la predicción del valor de la variable objetivo de un nuevo ejemplo; por lo tanto, para el método del vecino más cercano, la complejidad computacional de una predicción es considerablemente mayor que para otros algoritmos de Machine Learning (Kononenko, I., & Kukar, M., 2007).

### 2.3.2. Aprendizaje no Supervisado

A diferencia del enfoque supervisado los humanos no validan los datos, es decir únicamente se ingresa la información y el programa usa procesos que van clasificando la información por sí solo. Este paradigma ha resultado muy interesante dado que descubre patrones muchas veces ocultos en los datos, tanto así que generaciones de investigadores pueden pasarlos por alto. Uno de estos casos fue publicado en la revista Nature Communications (Mondal et al. 2019) donde los autores implementaron una IA de aprendizaje profundo que analizó el genoma humano de la población en Eurasia mediante el método de aproximación bayesiana encontrando un rastro de genomas ajenos a otros ancestros que se encuentra aún presentes en poblaciones de humanos modernos.

#### *2.3.2.1. Agrupamiento DBSCAN*

El agrupamiento espacial basado en densidad de aplicaciones con ruido o Density-based spatial clustering of applications with noise (DBSCAN) es un algoritmo que encuentra los grupos, clústeres, comenzando por una estimación de la distribución de la densidad de los nodos correspondientes entre un punto y otro.

El algoritmo comienza seleccionando un punto p arbitrario, si p es un punto central, se comienza a construir un grupo y se ubican en su grupo todos los objetos denso-alcanzables desde p. Si p no es un punto central se visita otro objeto del conjunto de datos. El proceso continúa hasta que todos los objetos han sido procesados. Los puntos que quedan fuera de

los grupos formados se llaman puntos ruido, los puntos que no son ni ruido ni centrales se llaman puntos bordes. De esta forma DBSCAN construye grupos en los que sus puntos son o puntos centrales o puntos borde, un grupo puede tener más de un punto central (Pascual, D., Pla, F., & Sánchez, S., 2007).

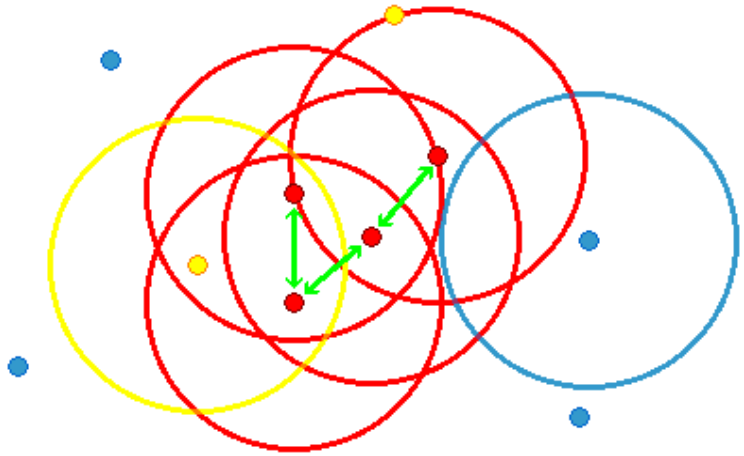


*Figura 4. Tipo de puntos en DBSCAN, puntos centro (rojos), puntos ruidosos (azules). Elaborado a partir de Huang et al. 2021.*

### *2.3.2.2. Agrupamiento K Medias*

Conocido como KMeans en inglés, es uno de los algoritmos de clustering más populares y utilizados por su rapidez y sencillez de implementación.

K-means utiliza la medida de distancia euclidiana y asigna iterativamente cada registro en los grupos derivados. Es un algoritmo muy rápido ya que no necesita calcular las distancias entre todos los pares de registros. Los clústeres se refinan a través de un procedimiento iterativo durante el cual los registros se mueven entre los clústeres hasta que el procedimiento se vuelve estable. El procedimiento comienza seleccionando k registros iniciales bien espaciados como centros de los agrupamientos y asigna cada registro a su agrupamiento "más cercano". A medida que se agregan nuevos registros a los agrupamientos, los centros de los agrupamientos se vuelven a calcular para reflejar sus nuevos miembros. Luego, los casos se resignan a los grupos ajustados. Este procedimiento iterativo se repite hasta que converge y la migración de registros entre clústeres ya no refina la solución (Tsiptsis et al. 2009).

### *2.3.2.3. Agrupamiento Basado en Mezcla Gaussiana*

Es un algoritmo de agrupamiento que trabaja en base a probabilidades y asumiendo que los datos presentan una distribución normal o gaussiana.

Para explicar la forma en que GMM realiza las asignaciones de las observaciones a los clústeres primero se debe mencionar que k-means lleva a cabo una asignación dura. Es decir que, k-means simplemente asigna observaciones a los clústeres, es un modelo determinista.

Se conoce como asignación dura cuando el algoritmo no duda sobre la pertenencia de un evento a un clúster, simplemente se destina la observación al clúster. GMM realiza una asignación suave, esto quiere decir que además de los clústeres que se generan en k-means, también entrega probabilidades de pertenencia de un evento a cada uno de los clústeres (Ríos Carrillo, D. A., 2022).

El uso de los modelos de mezcla gaussiana presenta dos ventajas clave. En primer lugar, son mucho más flexibles en términos de covarianza de clústeres debido al parámetro de desviación estándar, los clústeres pueden adoptar cualquier forma de elipse (Figura 5), en lugar de limitarse a círculos. En segundo lugar, dado que los modelos de mezcla gaussiana utilizan probabilidades, pueden tener conglomerados múltiples por punto de datos. Por lo tanto, si un punto de datos está en medio de dos grupos superpuestos, podemos simplemente definir su clase diciendo que pertenece al X por ciento de la 1 y al Y por ciento de la 2. Es decir, los modelos de mezcla gaussiana apoyan una composición mixta.

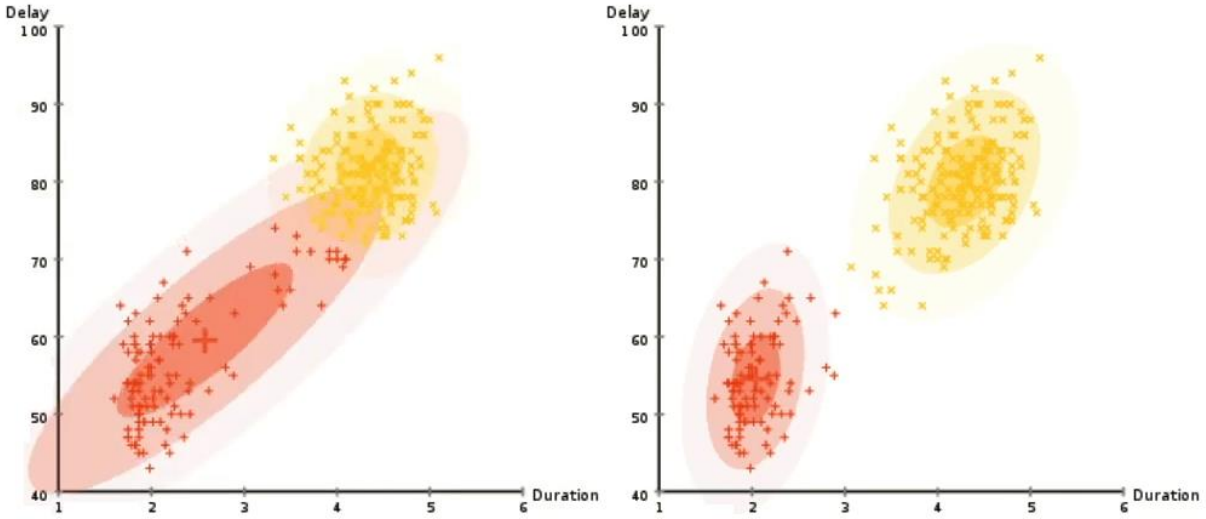



*Figura 5. Ejemplo de optimización gaussiana en su forma inicial hasta su forma final. Elaborado a partir de González, Ligdi. 2020*

## 2.4. Implementación de la inteligencia Artificial en el Sector Bancario

Hoy en día la inteligencia artificial cada vez ha tomado un mayor auge en la industria bancaría. Son muchas las implementaciones que se han llevado a cabo con el uso de técnicas de inteligencia artificial y han venido a eficientizar los procesos y toma de decisiones dentro de las instituciones. Según Donepudi, P. algunas de estas implementaciones son las siguientes (Donepudi, P., 2017):

**AML y Detección de Patrones de Fraude:** AML por sus siglas en inglés (Anti-Money Laundering) es el conjunto de métodos, leyes y regulaciones que existen para evitar que se cometa lavado de activos, lo cual implica el enriquecimiento ilegal por parte de las personas. Actualmente las técnicas de inteligencia artificial están siendo utilizadas por una gran cantidad de entidades bancarias para mejorar la detección de lavado de activos, en comparación a la forma tradicional que se utilizaba para detectar estos casos, gracias a la inteligencia artificial se puede concluir de forma más rápida y efectiva la detección de un mayor número de casos de lavado de activos (Donepudi, P., 2017).

Uno de los principales problemas que enfrenta la banca con sus transacciones bancarias son los casos de fraude, especialmente los casos de fraude cometidos con tarjetas de crédito son muchas las técnicas de fraude que se utilizan para llevar a cabo estos fines, por lo cual se necesitan mecanismos avanzados y robustos para su detección inmediata. Debido a que algunas técnicas de minería de datos son muy lentas en su implementación, lo más recomendable en este tipo de casos es utilizar redes neuronales previamente entrenadas que puedan ofrecer una respuesta en tiempo real con respecto a las transacciones efectuadas (Patidar, R., & Sharma, L., 2011).

**Automatización y Personificación de Banca:** Este es uno de los campos donde la inteligencia artificial se ha destacado mayormente en la industria bancaria, aplicaciones como los chatbots, los cuales son asistentes artificiales bancarios que ofrecen soluciones a los clientes por medio de mensajes telefónicos, permiten a las personas solucionar sus problemas de forma personal sin tener que llamar a los centros de ayuda, lo cual reduce la carga de trabajo, una modalidad aún más novedosa en la industria son los asistentes artificiales por medio de voz, los cuales simulan con mayor precisión la experiencia de ser atendido por un asistente humano para recibir asesoría sobre las transacciones, productos y servicios que los clientes desean gestionar. En esta categoría también se ha implementado sistemas que ofrecen consejos personalizados de finanzas a los clientes basado en su actividad financiera como ser ingresos personales, gastos mensuales hábitos de gastos. Las empresas también están apuntando a utilizar RPA (Robotic Process Automation) con lo cual se planea utilizar software basado en inteligencia artificial que será capaz de aprender como ejecutar diferentes procesos empresariales de manera artificial, de esta forma se podrá disminuir el costo operacional, aumentar la productividad y eliminar el error humano que están relacionados con la ejecución de procesos dentro de la organización (Donepudi, P., 2017).

**Recomendaciones de Cliente:** Gracias a la información que se posee de los clientes y a la interacción que estos realizan con los productos y servicios bancarios que se ofrecen, es posible hacer uso de los sistemas de recomendación, los cuales predicen que productos o

servicios tendrán mayor oportunidad de ser adquiridos por una persona en particular. Los sistemas de recomendación juegan un papel fundamental en las campañas de fidelización de los clientes, ya que ayudan a identificar productos más precisos para los clientes de acuerdo con sus gustos y necesidades.

**Gestión del Riesgo:** La inteligencia artificial y las técnicas de Machine learning han venido a revolucionar como las empresas gestionan el riesgo en el sector financiero, desde tomar decisiones sobre otorgamiento de préstamos a clientes, proveer señales de alerta a comerciantes del mercado financiero sobre posiciones de riesgo, detectar fraude interno y proveniente de clientes hasta mejorar el cumplimiento de metas y reducir el riesgo (Aziz S., Dowling M., 2019). Algunos ejemplos que se puede mencionar sobre modelos predictivos aplicados a la gestión de riego abarcan su aplicación al riesgo de crédito donde existe la posibilidad que un cliente falle a sus obligaciones contractuales, también hay ejemplos donde las técnicas de machine learning se han aplicado al riesgo de mercado, con lo cual se busca minimizar perdidas en la inversión de negocios que sean poco factibles para la empresa, también se puede destacar su implementación en el riesgo operativo, el cual puede surgir a lo interno de la empresa y en muchos casos puede ser más complejo de detectar si no se cuenta con las métricas correctas para medirlo.

## 2.5. Retos de la Inteligencia Artificial en el Sector Bancario

A pesar de que se ha tenido una gran cantidad de avances con respecto a la inteligencia artificial en el sector financiero, debido a la naturaleza de las empresas de este rubro con respecto a la forma en que se deber almacenar y manipular la información de sus clientes, existen ciertos retos que se deben afrontar para que los algoritmos de machine learning y demás técnicas de inteligencia artificial tengan éxito en su implementación.

### 2.5.1. Reglamento General de Protección de Datos

Uno de los retos más significativos hace referencia a las políticas de seguridad de los datos, lo cual limita la información disponible para poder ser utilizada con la inteligencia artificial. La ley RGPD (Reglamento General de Protección de Datos), es una ley de protección de datos que surgió en la Unión Europea en el 2016 y entró en vigor en el año de 2018, con este reglamento se busca salvaguardar la identidad y demás información sensible de los individuos. Esta ley contiene clausulas preventivas sobre automatización de toma de decisiones, esto afecta, no sólo a la industria financiera, sino a todos los sectores en general.

El artículo 22 de la ley RGPD sostiene lo siguiente:" El interesado deberá tener el derecho a no estar sujeto a una decisión basada únicamente en el procesamiento automatizado, incluida la elaboración de perfiles", esto representa un problema particularmente para aplicaciones de inteligencia artificial, cuya toma de decisiones por definición es enteramente automática, para sobrellevar estas restricciones la intervención de un humano en algún punto en específico podría significar una solución (Kaya O., 2019).

Existen otros artículos de la ley RGPD que también podrían significar un inconveniente para una fácil aplicación de la inteligencia artificial en múltiples campos, un ejemplo de ellos es el artículo 25, en el cual se establece que el responsable del tratamiento aplicará, tanto en el momento de determinar los medios de tratamiento como en el momento del propio tratamiento de los datos personales, medidas técnicas y organizativas apropiadas, como la seudonimización, concebidas para aplicar de forma efectiva los principios de protección de datos, como la minimización de datos, e integrar las garantías necesarias en el tratamiento de los datos (art. 25, GDPR), esto supone mayores costos de tiempo, dinero y recursos en la elaboración de aplicaciones que brinden una protección optima de los datos desde su diseño.

También se debe prever, tal como lo dispone el artículo 32 de la ley RGPD la aplicación de medidas técnicas y organizativas apropiadas para garantizar un nivel de seguridad adecuado al riesgo, que en su caso incluya, entre otros: la seudonimización y el cifrado de datos personales, la capacidad de restaurar la disponibilidad y el acceso a los datos personales de forma rápida en caso de incidente físico o técnico y cualquier otra medida para asegurar la seguridad del tratamiento de los datos. (art. 32, GDPR).

## 2.6. Trabajos Relacionados

Actualmente existe un amplio repertorio de trabajos centrados a aplicar los beneficios de la inteligencia artificial, específicamente del Machine Learning en las instituciones bancarias a nivel internacional, algunos de los estudios realizados sobre segmentación de clientes y detección de abandono basado en Machine Learning para este sector son los siguientes:

### 2.6.1. Segmentación de Clientes Bancarios a Través de Mapas Autoorganizados:

Esta segmentación de cliente se basa en la utilización de redes neuronales, llamadas mapas autoorganizativos (Self Organizing Maps) capaces de mapear los datos demográficos, de comportamiento, personales y operacionales de los clientes, de tal forma que son altamente capaces de segmentar de forma eficiente en base a un mayor número de atributos, se dice

que este método mapea a los clientes en mapas autoorganizativos porque a diferencia de los métodos de agrupamiento tradicionales no requiere de un análisis previo del número de clústeres a generar (Bach, M. P., 2013).

### 2.6.2. Segmentación de Clientes a través de Minería de Datos y Segmentación de Mercado

Este enfoque se basa en utilizar los datos disponibles de los clientes para segmentarlos a través de técnicas de agrupamiento clásico como ser k-Medias, también se combina el análisis RFM de los clientes para conocer que conjunto de clientes posee mayor relevancia para los bancos (Zakrzewska, D., & Murlewski, J., 2005).

En este enfoque también es posible utilizar mapas autoorganizativos para segmentar a los clientes en base a conjuntos de datos de alta dimensionalidad.

### 2.6.3. Predicción de Abandono de Productos Bancarios Usando Agrupamiento C-Medias Difuso

Debido a la naturaleza de C-Medias Difuso (Fuzzy CMeans) donde un ítem puede pertenecer a más de un clúster, este enfoque de detección de abandono, a diferencia de un modelo de clasificación, utiliza probabilidades para determinar a que grupo de clientes tiene mayor pertenencia un cliente en específico, estos grupos de clientes se pueden clasificar en clientes desertores o clientes y clientes no desertores, por lo tanto un cliente puede pertenecer a ambos grupos, pero con diferentes niveles de pertenencia, se toma el nivel más alto para determinar a qué grupo pertenecerá (Popović, D., & Bašić, B. D., 2009).

### 2.6.4. Predicción de Abando de Clientes en Bancos Utilizando Minería de Datos y Redes Neuronales

Este enfoque utiliza las técnicas de minería de datos para poder seleccionar las mejores variables de las cuales de disponen en la fuente de datos, luego se utiliza redes neuronales por su alta capacidad para procesar información que no lineal (Bilal Zorić, A., 2016).

# 3. Objetivos concretos y metodología de trabajo

A continuación, se expones el objetivo principal de la elaboración del trabajo actual, junto con los objetivos específicos que ayudaran a alcanzar el cumplimiento del objetivo principal.

## 3.1. Objetivo general

Realizar un análisis de los productos financieros mediante el uso de técnicas de inteligencia artificial que permita reconocer o identificar que clientes que son más propensos a adquirir o abandonar los productos y servicios que son ofrecidos por una institución bancaria, de manera que los resultados obtenidos por los algoritmos sean de utilidad al momento de elaborar las campañas para vender o fidelizar un producto en específico, con lo cual se espera mejorar productividad de los ejecutivos que crean dichas campañas y la efectividad de las mismas.

## 3.2. Objetivos específicos

- Identificar las características demográficas más representativas de los clientes dentro del sistema CRM para que sean utilizadas en los algoritmos de machine learning a emplear
- Elaborar las consultas SQL con las cuales se obtendrán los datasets finales a utilizar en los diferentes algoritmos de agrupamiento y clasificación
- Establecer e implementar los algoritmos de agrupamiento que se utilizarán para segmentar los clientes y poder observar la tenencia de productos a través de cálculos estadísticos.
- Implementar y comparar las diferentes técnicas de clasificación para determinar el mejor algoritmo a utilizar en el modelo predictivo de abandono de productos bancarios.
- Establecer las métricas a utilizar para evaluar la efectividad de los algoritmos utilizados, aquí de deberá evaluar cada algoritmo de machine learning utilizado para cada producto estudiado.
- Analizar y plantear las mejoras que se podrían realizar a la investigación final y los campos en los cuales se podría ampliar su uso a futuro.

## 3.3. Metodología del trabajo

A continuación, se explica todo el proceso seguido para desarrollar la investigación actual, así como las herramientas, algoritmos y métricas utilizadas para obtener los resultados finales.

### 3.3.1. Proceso de KDD

Para desarrollar la presente investigación se proponen una serie de pasos los cuales están basados en el Proceso de Descubrimiento de Conocimiento en Bases de Datos, conocido por sus siglas en inglés como proceso de KDD (Knowledge Discovery from Databases).

El término Descubrimiento de Conocimiento en Bases de Datos empezó a utilizarse en 1989 para referirse al amplio proceso de búsqueda de conocimiento en bases de datos, y para enfatizar la aplicación a "alto nivel" de métodos específicos de minería de datos. En general, el descubrimiento es un tipo de inducción de conocimiento, no supervisado, que implica dos procesos: Búsqueda de regularidades interesantes entre los datos de partida y formulación de leyes que las describan (Martínez, G., 2001).

El proceso global consiste en transformar información de bajo nivel en conocimiento de alto nivel (Figura 6). El proceso KDD es interactivo e iterativo conteniendo los siguientes pasos (Riquelme Santos, J. C., Ruiz, R., & Gilbert, K., 2006):

1. Comprender el dominio de aplicación: este paso incluye el conocimiento relevante previo y las metas de la aplicación.

2. Extraer la base de datos objetivo: recogida de los datos, evaluar la calidad de los datos y utilizar análisis exploratorio de los datos para familiarizarse con ellos.

3. Preparar los datos: incluye limpieza, transformación, integración y reducción de datos. Se intenta mejorar la calidad de los datos a la vez que disminuir el tiempo requerido por el algoritmo de aprendizaje aplicado posteriormente.

4. Minería de datos: como se ha señalado anteriormente, este es la fase fundamental del proceso. Está constituido por una o más de las siguientes funciones, clasificación, regresión, clustering, resumen, recuperación de imágenes, extracción de reglas, etc.

5. Interpretación: explicar los patrones descubiertos, así como la posibilidad de visualizarlos.

6. Utilizar el conocimiento descubierto: hacer uso del modelo creado.

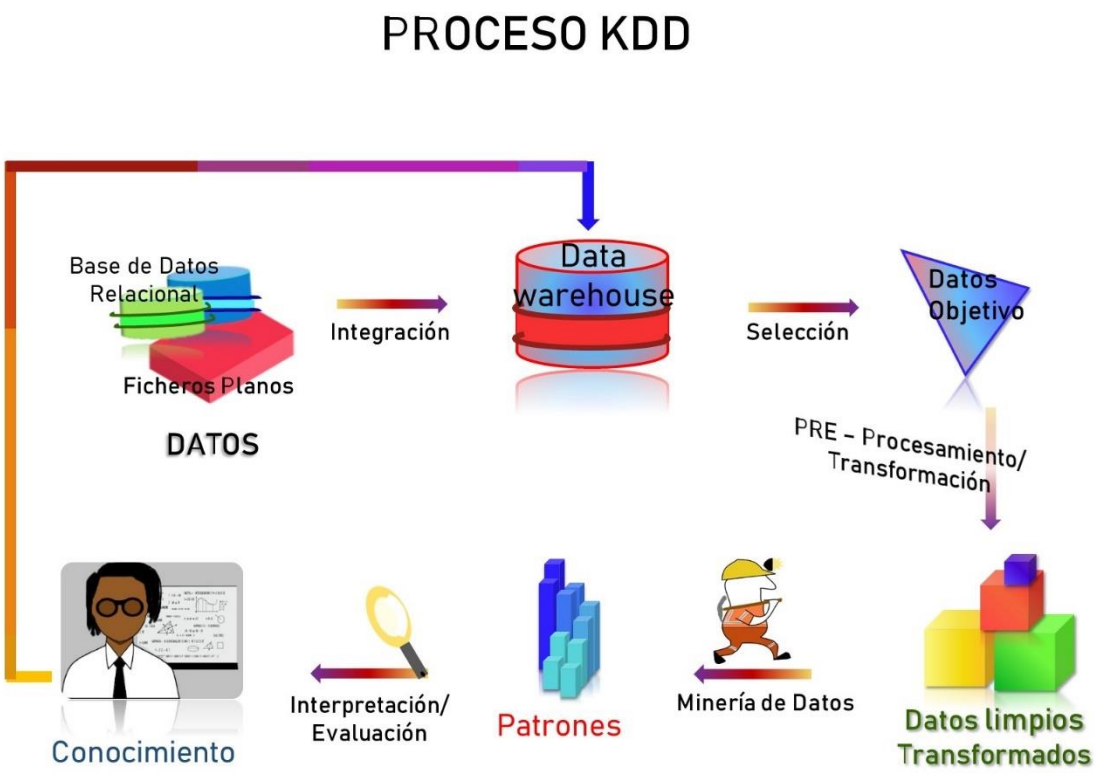



*Figura 6. Proceso de Descubrimiento de Conocimiento en Bases de Datos "KDD" (Elaboración propia)*

En conclusión, las fases que se seguirán en la investigación actual basadas en el proceso de KDD son la selección de las variables a utilizar, esta tarea se realizará en base a un análisis previo de los datos disponibles en el Data Warehouse, luego se realizará un preprocesamiento de los datos a través de consultas SQL. En la etapa de codificación en Python se realizan algunas transformaciones a los datos para que estos puedan ser utilizados por los diferentes algoritmos de Machine Learning Seleccionados. Por último, se estudiará la calidad de los modelos entrenados y se determinará si fuese necesario repetir el proceso de KDD o se puede avanzar a etapas posteriores como ser la utilización de los modelos entrenados en las tareas de colocación y retención de productos bancarios.

## 3.3.2. Definición de Recursos y Herramientas a Utilizar

En este apartado se explicará el contexto actual sobre el cual se pretende realizar el análisis de datos, lo cual implica la explicación de los programas de software actualmente utilizados para almacenar la información necesaria que será utilizada de cara a la implementación de

las técnicas de aprendizaje automatizado, así como las diferentes herramientas y lenguajes de programación que se desean emplear en la implementación de los algoritmos predictivos.

**Explicación del Contexto:** Dentro de la institución bancaria se posee un sistema de CRM, a través del cual se guarda toda la información que es generada por la interacción entre el banco con los clientes, como ser el registro de nuevos clientes y la adquisición de nuevos productos bancarios por parte de estos clientes nuevos y ya existentes, esta información es extraída por procesos ETL (Extract, transform and load) en horas no transaccionales del sistema CRM, dicha información extraída por los diversos procesos ETLs es guardada en el data warehouse empresarial, el cual se compone de diversas diversas tablas que almacenan información de propósitos específicos, al final del proceso de extraer la información del sistema fuente y guardarla en el data warehouse, se procede a crear consultas SQL, las cuales toman los campos necesarios según se requiera la información en cada una de las tablas, dicho flujo se puede apreciar en la figura 7.

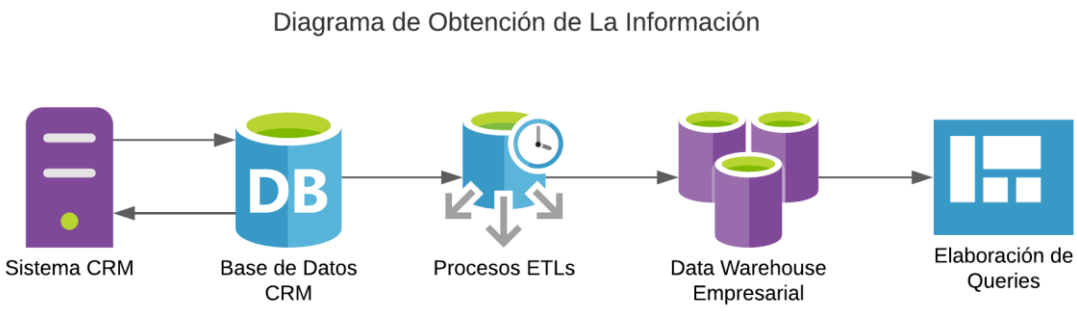



*Figura 7. Proceso de obtención de la información (Elaboración propia)*

A continuación, se definen los componentes principales que tienen relevancia en el trabajo actual:

**Sistema CRM:** Sistema que posee información descriptiva de cada uno de los clientes de la institución, este sistema es de gran importancia para la elaboración de la herramienta de software final, ya que del él se extraerán las variables demográficas de los clientes, las cuales serán utilizadas en las consultas SQL para construir los diferentes datasets a utilizar.

**Data Warehouse:** Es una base de Datos relacional SAP Hana la cuál posee capacidad para almacenar una gran cantidad de información proveniente de diversos sistemas fuente que son utilizados por la institución. Utilizando la información disponible en el data warehouse se puede obtener a través de consultas SQL, un dataset que especifique las características demográficas de los clientes junto a los productos bancarios que estos mismos disponen, así como el historial de tenencia de cada producto.

**Herramientas a utilizar:** Para poder realizar el análisis propuesto en el trabajo actual, se hará uso de lenguajes de programación, frameworks y demás herramientas tecnológicas las cuales se listan en la Tabla 1:

| Nombre de la herramienta | Versión | Relevancia |
|---|---|---|
| **Interactive SQL** | 17.0.0 build 1062 | Herramienta que permite realizar conexiones a bases de datos y ejecutar consultas SQL de forma interactiva. |
| **Anaconda3** | 2020.11 | Programa se podrá acceder a varias herramientas de desarrollo de software y algoritmos de machine learning, las cuales serán utilizas a lo largo del proceso de análisis. |
| **Jupyter NoteBooks** | 6.4.5 | Aplicación utilizada para ejecutar el código Python y visualizar los resultados por medio de cuadernos. |
| **Python** | 3.8.5 | Lenguaje de programación con el cuál se escribirá el código para implementar los algoritmos de Machine Learning |
| **hdbcli** | 2.4.171 | Paquete que contiene las funciones necesarias para conectarse a una base de datos relacional SAP Hana |
| **pandas** | 1.3.4 | Biblioteca de software escrita como extensión de NumPy para manipulación y análisis de datos |
| **numpy** | 1.21.2 | biblioteca para el lenguaje de programación Python que da soporte para crear vectores y matrices grandes multidimensionales |
| **matplotlib** | 3.4.3 | Librería que se utilizará para generar visualizaciones sobre el comportamiento de los datos |
| **seaborn** | 0.11.2 | Librería basada en matplotlib para crear visualizaciones más potentes y personalizables |

| | | |
|---|---|---|
| **sklearn** | 1.0.2 | Librería que contiene la implementación de los algoritmos de inteligencia artificial y calculo de métricas de evaluación de dichos algoritmos |
| **imblearn** | 0.9.0 | Librería que posee funciones que permite aplicar técnicas de balanceo de clases a conjunto de datos desbalanceados |
| **pycm** | 3.1 | Librería de matrices de confusión multi clase que muestra métricas de evaluación de forma sencilla y rápida |

*Tabla 1.Herramientas y Frameworks Utilizados (Elaboración propia)*

### 3.3.3. Técnicas de Machine Learning a implementar

Este paso constituye el más importante del proceso metodológico a emplear ya que es en este punto donde se implementará cada una de las técnicas de tratamiento de datos para poder arrojar resultados sobre los diferentes productos bancarios y su relación con los clientes de la institución. Se seleccionaron 3 algoritmos para las técnicas de agrupamiento y 4 para las técnicas de clasificación los cuales serán comparados entre sí para determinar cuál es el más apropiado para obtener información relevante del dataset generado según sea el caso. Dichos algoritmos fueron explicados en el apartado 2.3. y se listan a continuación:

**KMeans:** Algoritmo de agrupamiento el cual utiliza la distancia euclidiana para generar el número de clústeres especificado previamente en los parámetros del algoritmo. Este algoritmo realiza un proceso interactivo en el cual se establecen centroides los cuales se van afinando en cada iteración hasta obtener los clústeres deseados.

**Agrupamiento de Mezcla Gausiana:** Algoritmo que se basa en modelos probabilísticos que poseen una distribución gaussiana (Distribución normal), donde se calculan parámetros como la covarianza para así, poder determinar los clústeres a formar.

**Algoritmo DBSCAN:** Algoritmo que utiliza como media de agrupamiento la densidad, la densidad se mide a través de la vecindad de puntos establecidos en algoritmo, un punto se considera lo suficientemente denso si posee una vecindad minina de n puntos vecinos, separados a una distancia mínima épsilon. Este algoritmo posee la desventaja que puede generar puntos sin clasificar, los cuales son llamados ruido.

**CART:** Algoritmo utilizado en la construcción de árboles de clasificación y regresión mediante la construcción de árboles binarios. Este algoritmo se utilizará en la fase de predicción de abandono de productos.

**Random Forest:** Algoritmo de clasificación de aprendizaje supervisado que utiliza varios algoritmos de árboles de decisión, cuyos resultados son combinados para poder obtener un mejor nivel de predicción.

**Regresión Logística:** Algoritmo utilizado para problemas de clasificación donde una función sigmoidea estima la probabilidad de que un ejemplo dado pertenezca a una clase o a otra en base a una serie de variables independientes.

**K-Nearest Neighbors:** Este algoritmo utiliza la distancia para definir que ejemplos son más cercanos o parecidos al caso a predecir, en esta técnica se asume que el ítem a predecir tendrá un comportamiento similar a los ejemplos mas cercanos a dicho ítem.

## 3.3.4. Métricas de Evaluación de Algoritmos

Al finalizar la fase de desarrollo de algoritmos de machine learning, se obtendrán resultados provenientes de dichos algoritmos, es en esta fase se determinará si los datos obtenidos de los diferentes algoritmos y técnicas poseen relevancia suficiente para ser tomados en cuenta en la toma de decisiones referente a la colocación y retención de productos.

### *3.3.4.1. Matriz de Confusión*

En machine learning una matriz de confusión es una herramienta que permite visualizar el desempeño de un algoritmo. Toma el nombre de matriz de confusión en aprendizaje supervisado, pero en aprendizaje no supervisado se le conoce por matriz de coincidencia, otro nombre más general es matriz de error (Stehman, S., 1997). La matriz emplea la columnas y filas para mostrar las clases predichas y las clases reales, con ello nos permite visualizar qué tipos de aciertos y errores está teniendo nuestro modelo a la hora de pasar por el proceso de aprendizaje con los datos o si el sistema está confundiendo dos clases. Con la matriz de confusión (Figura 8) podemos establecer métricas para evaluar la efectividad de un modelo de machine learning, particularmente para fines del presente estudio nos interesan los clasificadores de tipo binario.

| | | Valores reales u observados | |
|---|---|---|---|
| | | **Positivos** | **Negativos** |
| **Valores predichos** | **Positivos** | Verdaderos Positivos (VP) True Positives (TP) | Falsos Positivos False Positives (FP) ERROR tipo 1 |
| | **Negativos** | Falsos Negativos False Negatives (FN) ERROR tipo 2 | Verdaderos Negativos (VN) True Nagative (TN) |

$$\text{Accuracy} = \frac{VP+VN}{P+N} = \frac{TP+TN}{P+N}$$

$$\text{Precision} = \frac{VP}{VP+FP} = \frac{TP}{TP+FP}$$

$$\text{Recall} = \frac{VP}{P} = \frac{TP}{P}$$

$$\text{F1 Score} = \frac{2(\text{Recall})(\text{Precision})}{\text{Recall} + \text{Precision}}$$

*Figura 8. Matriz de Confusión y algunas de sus Métricas (Elaboración propia)*

### *3.3.4.2. Exactitud (Accuracy)*

Se refiere a lo cerca que está el resultado de una medición del valor verdadero. En términos estadísticos, la exactitud está relacionada con el sesgo de una estimación. Se representa como la proporción de resultados verdaderos tanto verdaderos positivos (VP o TP) como verdaderos negativos (VN) dividido entre el número total de casos examinados, P (verdaderos positivos, falsos positivos) y N (verdaderos negativos, falsos negativos). En forma práctica, la Exactitud es la cantidad de predicciones positivas que fueron correctas.

### *3.3.4.3. Precisión*

Se refiere a la dispersión del conjunto de valores obtenidos a partir de mediciones repetidas de una magnitud. Cuanto menor es la dispersión mayor la precisión. Se representa por la proporción de verdaderos positivos dividido entre todos los resultados positivos (tanto verdaderos positivos, como falsos positivos). En resumidas cuentas, se trata del porcentaje de casos positivos detectados. Como dato estadístico, la precisión es útil únicamente cuando se tienen datasets simétricos. Es decir que la cantidad de casos de la clase 1 y de la clase 2 tienen magnitudes similares.

### *3.3.4.4. Sensibilidad (Recall o Sensitivity)*

También se conoce como Tasa de Verdaderos Positivos TVP ó TPR (True Positive Rate). Es la proporción de casos positivos que fueron correctamente identificadas por el algoritmo. La sensibilidad es semejante a otra métrica conocida como especificidad, ambos valores nos indican la capacidad de nuestro estimador para discriminar los casos positivos, de los negativos. La sensibilidad se representa como la fracción de verdaderos positivos, mientras que la especificidad, es la fracción de verdaderos negativos según; VP/(VP+FN) & VN/(VN+FN) respectivamente. Semejante a estos cálculos se encuentran la tasa de falsos negativos, tasa de falsos positivos, mientras podemos intuir que especificidad es la tasa de verdaderos negativos. Con todos estos valores se determina el valor predictivo positivo y el valor predictivo negativo, lo que viene estando detallado en el teorema matemático de Bayes

sobre probabilidades. La precisión y el recall ayudan a determinar si existe una pérdida de positivos en el proceso de aprendizaje.

#### *3.3.4.5. F1 Score*

Esta métrica resume precisión y sensibilidad por lo que es muy empleada. Resulta de gran utilidad cuando la distribución de las clases es desigual. Se calcula como dos veces el producto de la precisión por el recall entre la suma de la precisión y el recall, generando 4 posibles casos; Alta precisión y alto recall, dado un modelo de machine learning que maneja perfectamente los datos; Alta precisió y bajo recall, donde el modelo de machine learning no detecta muy bien las clases, pero cuando lo hace es altamente confiable; Baja precisión y alto recall, dado un modelo de machine learning que detecta bien las clases, pero también incluye muestras de otra clase; finalmente baja precisión y bajo recall, donde el modelo de machine learning no logra clasificar la clase correctamente.

#### *3.3.4.6. Curva ROC*

Un gráfico de características operativas del receptor (receiver operating characteristic, ROC) es una técnica para visualizar, organizar y seleccionar clasificadores en función de su rendimiento. Los gráficos ROC se han utilizado durante mucho tiempo en la teoría de detección de señales para representar el equilibrio entre las tasas de aciertos y las tasas de falsas alarmas de los clasificadores (Egan, 1975; Swets et al., 2000). Los gráficos ROC fueron adoptados en el aprendizaje automático gracias a Spackman (1989), quien demostró el valor de las curvas ROC en la evaluación y comparación de algoritmos. En los últimos años se ha visto un aumento en el uso de gráficos ROC en la comunidad de aprendizaje automático, debido en parte a la comprensión de que la precisión de clasificación simple suele ser una métrica deficiente para medir el rendimiento (Provost et al., 1998). Además de ser un método de representación gráfica de rendimiento generalmente útil, tienen propiedades que los hacen especialmente útiles para dominios con distribución de clases sesgada y costos de error de clasificación desiguales. Estas características se han vuelto cada vez más importantes a medida que continúa la investigación en las áreas del aprendizaje sensible y el aprendizaje en presencia de clases desequilibradas.

La curva AUC - ROC es una medida de rendimiento para los problemas de clasificación en varias configuraciones de umbral. ROC es una curva de probabilidad y AUC (Area Under the Curve o área bajo la curva) representa el grado o medida de separabilidad. Indicado que tan capacitado está el modelo para distinguir clases. Cuanto mayor sea el AUC, mejor será el modelo para predecir que 0 pertenece a la clase como 0 y que 1 pertenece a la clase 1. Por analogía, cuanto mayor sea el AUC, mejor será el modelo para distinguir entre los casos

binarios. En la curva ROC (Figura 9) se mide TPR (True Positive Rates es decir Recall) contra FPR (False Positive Rates) en los ejes Y & X respectivamente.

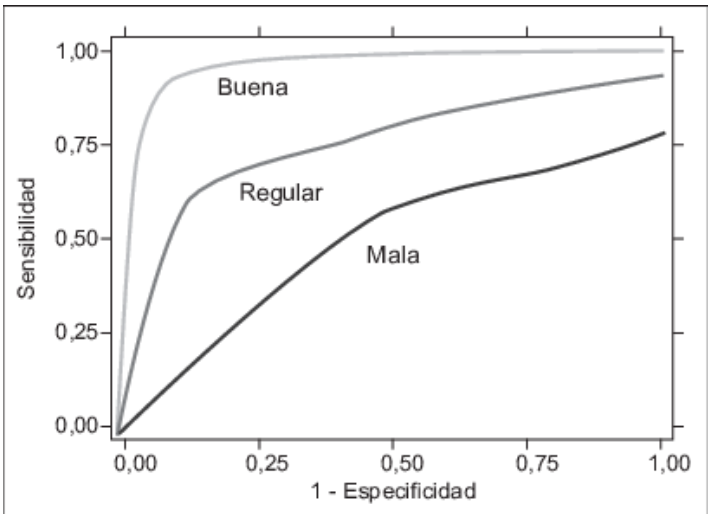



*Figura 9. ejemplo de Curva ROC, Fuente: Me, Burgos & Manterola, Carlos. (2010).*

## 3.3.5. Técnicas de Optimización de Algoritmos

**SMOTE:** Las teorías de machine learning contemplan el balanceo de clases cuando los datasets poseen clases desbalanceadas. Las estrategias empleadas son conocidas como under sampling y over sampling (Abdul J., 1977), en español estas técnicas son traducidas como submuestreo y sobre muestreo. Ejemplificando un antecedente lamentable del problema, un algoritmo usado para calcular la reincidencia de casos criminales en la justicia en EE UU tenía un notable sesgo por reincidir a la gente negra en delitos. Muchos análisis determinaron que el problema esencial estaba en el desbalanceo de casos, hay al menos 200 años de registros donde la gente negra era incriminada en casos únicamente por su color de piel y tenían demasiadas prohibiciones (Benjamin R. 2019). Estos algoritmos particulares tienen un sesgo irreparable con técnicas como el under sampling o el over sampling, dado que estas técnicas se basan en nivelar los porcentajes de muestreo de casos para corregir el sesgo, corresponde a una de las tantas técnicas que existen pasa solucionar el problema de clases desbalanceadas.

SMOTE se Propone como un enfoque de sobre-muestreo en el que la clase minoritaria está sobre-muestreada por ejemplos creados “sintéticamente” en lugar de sobre-muestrear con reemplazo (Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P., 2002). En este trabajo SMOTE se aplicará a través de la librería de pyhton imblearn.

**Análisis de Componentes principales:** llamada ACP o PCA por sus siglas en inglés (Principal Component Analysis) es una técnica que se utiliza para reducir las variables de un

conjunto de datos en un número de variables o componentes más pequeños. El PCA busca combinaciones lineales de las variables que mejor describen la tendencia del proceso Matemáticamente, el PCA se basa en una descomposición de la matriz de covarianza de las variables del proceso a lo largo de las direcciones que mejor explican las principales causas de variabilidad de la información analizada (Garcia-Alvarez, D., & Fuente, M. J., 2011). El PCA se utilizará en este trabajo para poder analizar el comportamiento del conjunto de datos a nivel visual y de esta forma poder identificar el agrupamiento de las instancias analizadas.

**Método de la Silueta:** El análisis de silueta se utiliza para estudiar y comprender la distancia de separación entre los grupos resultantes. Este análisis se utiliza para medir qué tan cerca cada objeto dentro de un grupo está de otro objeto perteneciente a otro grupo. Los valores de puntuación de silueta se encuentran entre -1 y +1. El valor de +1 indica un correcto agrupamiento de objetos, mientras que el valor de -1 muestra que los objetos no están agrupados correctamente (Ogbuabor, G., & Ugwoke, F. N., 2018). En el trabajo actual esta técnica se utilizará para evaluar los valores óptimos de parámetros a utilizar en la implementación de los algoritmos de clustering.

# 4. Desarrollo Específico de la Contribución

En este apartado se describirá todo el proceso de minería de datos llevado a cabo para implementar los algoritmos de machine learning pensados para poder cumplir los objetivos del presente trabajo.

La primera fase será el análisis de selección de las variables disponibles en el Data Warehouse, seguidamente se explicará la creación de las consultas SQL desarrolladas para generar los datasets finales. Culminada la explicación de la obtención de los datos, se procederá a transformar los datos y entrenar cada uno de los algoritmos de Machine Learning, finalmente se procederá a medir los resultados obtenidos por los algoritmos de clasificación a través de las distintas métricas de evaluación.

## 4.1. Creación de Datasets

### 4.1.1. Análisis de Variables

Inicialmente se observan el total de campos extraídos del sistema CRM que se tienen disponibles dentro del data Warehouse. Se poseen dos sistemas CRM, y una tabla para cada sistema. Dentro de ambas tablas existe información que se encuentra en ambos sistemas CRM, así como información que sólo se encuentra en un solo sistema. En conclusión, para cada tabla se posee la siguiente cantidad de variables a analizar:

- **Tabla Sistema CRM 1:** 112 campos
- **Tabla Sistema CRM 2:** 122 campos

De todas las variables analizadas se omitieron las variables que poseen información personal de los individuos bajo análisis como ser nombre, identificación, dirección de residencia, esto con el objetivo de cumplir con la ley RGPD, también se omitieron las variables que establecían si los usuarios poseían algún servicio específico como ser servicio de envío de mensajes por transacción o si poseían la banca móvil habilitada. Otras variables que se omitieron fueron las variables que especificaban si los clientes entraban en la categoría de leyes especiales de otros países diferentes a Honduras.

Al final de todo el análisis a los campos obtenidos de los sistemas CRM se seleccionaron las variables que tuvieran información generalista de los clientes, de forma tal que eviten que sean identificables pero que permitan tener una idea inicial del perfil del usuario. Los campos seleccionados de los sistemas CRM fueron los siguientes:

- Genero
- Estado Civil
- Nivel Educativo
- Ocupación
- Generación
- Región

Adicionalmente a las variables obtenidas de CRM que pretenden proporcionar información sobre el perfil del cliente, se decidió añadir información que daría una idea del comportamiento general del cliente, para lograr este cometido se consultaron las tablas transaccionales de los clientes y se realizó un cálculo de promedio de los créditos, débitos y número de transacciones de débito que se registran para cada cliente bajo estudio en los últimos 6 meses, estos cálculos se consideran simplistas si se toma en cuenta que cada transacción bancaria se puede categorizar según su naturaleza, canal, tipo de servicio, tipo de pago y demás factores que permiten categorizar las transacciones de forma más específica, en el caso de los cálculos efectuados sólo se tomó en cuenta si la transacción correspondía a un débito o a un crédito de dinero.

Las variables mencionadas anteriormente serán las utilizadas para realizar la fase de agrupamiento de los clientes, a continuación, se detalla la selección de los productos financieros a analizar para cada uno de los usuarios bajo estudio, con la información de dichos productos se pretende resolver la parte de cálculo estadísticos de la tenencia de productos y modelos predictivos de abandono de productos.

De la gran cartera de productos bancarios, se seleccionaron algunos productos que puedan ser adquiridos por personas naturales, entre estos productos se escogieron productos de tres tipos: cuentas de depósitos, financiamientos y las tarjetas de crédito, a continuación, se detalla la cantidad de productos seleccionados de cada tipo:

- **Cuentas de Depósito:** 4 productos seleccionados
- **Financiamientos:** 4 productos seleccionados
- **Tarjetas de Crédito:** 8 productos seleccionados

Con las variables demográficas extraídas de CRM, las variables transaccionales y los productos seleccionados a estudiar, se crearán los datasets que servirán para alimentar los algoritmos de Machine Learning a implementar.

## 4.1.2. Desarrollo de Consultas SQL

Como se mencionó anteriormente, los datasets utilizados en este trabajo se generan a través de la utilización de consultas SQL, las cuales toman los campos seleccionados bajo previo análisis del Data Warehouse empresarial y los recupera por medio de dichas consultas. Cada consulta está conformada por subconsultas, las cuales se relacionan a través de joins, utilizando como campo relacionador el código del cliente (Este campo no se muestra en el dataset final). Para efectos del trabajo actual se desarrollaron tres subconsultas principales, las cuales se relacionan entre ellas para construir los datasets finales que serán utilizados en los algoritmos de Machine Learning. Dependiendo del dataset deseado se establecieron una serie de filtros o condicionales por medio de la cláusula *WHERE* en cada una de las subconsultas las cuales pueden variar de un dataset a otro, estos parámetros serán explicados a continuación.

## 4.1.3. Consulta Principal

La consulta principal es el código SQL utilizado para recuperar los campos que fueron seleccionados en el análisis previo y que se encuentran en las tablas que corresponden a los sistemas CRM, en dicha consulta también se consolidan a través de un *LEFT JOIN* las variables establecidas para determinar el comportamiento del cliente que se calculan de las tablas transaccionales, las cuales toman el promedio de las transacciones efectuadas por los clientes activos en el último año.

**Condicionales de la Cláusula WHERE**

Los condicionales agregados para esta subconsulta fueron no recuperar registros de clientes que poseyeran campos demográficos con valor nulo, en blanco o con valor “sin definir”, con este filtro se puede asegurar que el dataset final no poseerá valores nulos, ya que todas las instancias del dataset tendrán todos los valores de los atributos completos, de esta forma se facilita la parte de transformación de datos en Python.

Con el caso de las variables que poseen información transaccional se especificó que para los clientes cuyo cálculo final sea nulo, es decir no se encontraron transacciones efectuadas para dicho cliente, se coloque cero como valor por defecto.

Los atributos que se obtienen de esta consulta se describen a continuación en la tabla 2:

| Nombre del atributo | Tipo de Dato | Valores | Explicación de Cada Valor |
|---|---|---|---|
| **GENERO** | Cualitativo | FEMENINO | |
| | | MASCULINO | |
| **ESTADO_CIVIL** | Cualitativa | CASADO/A | |
| | | DIVORCIADO/A | |
| | | SEPARADO/A | |
| | | SOLTERO/A | |
| | | VIUDO/A | |
| | | UNIÓN LIBRE | |
| **NIVEL_EDUCATIVO** | Cualitativa | PRIMARIA | |
| | | SECUNDARIA | |
| | | UNIVERSITARIA | |
| | | MAESTRÍA | |
| | | DOCTORADO | |
| | | SIN ESCOLARIDAD | |
| **OCUPACION** | Cualitativa | AMA DE CASA | |
| | | COMERCIANTE INDIVIDU | |
| | | COMERCIANTE INFORMAL | |
| | | EMPLEADO PRIVADO | |
| | | EMPLEADO PÚBLICO | |
| | | ESTUDIANTE/MENOR E | |
| | | JUBILADO | |
| | | PROFESIONAL INDEPEND | |
| **GENERACION** | Cualitativa | GI | Nacidos entre 1901 y 1924 |
| | | GS | Nacidos entre 1925 y 1942 |
| | | B | Nacidos entre 1943 y |

| | | | |
|---|---|---|---|
| | | | 1960 |
| | | X | Nacidos entre 1961 y 1980 |
| | | Y | Nacidos entre 1981 y 1995 |
| | | Z | Nacidos en 1996 en adelante |
| **REGION** | Cualitativa | CENTRO-SUR | |
| | | NORTE | |
| | | ATLANTICO | |
| **CREDITOS** | Cuantitativa Continua | | Promedio de créditos recibidos por mes |
| **DEBITOS** | Cuantitativa Continua | | Promedio de débitos realizados por mes |
| **TRANSACCIONES_PROMEDIO** | Cuantitativa Continua | | Promedio de transacciones de débito realizadas por mes |

*Tabla 2.Descripción de Atributos de la Consulta Principal (Elaboración propia)*

## 4.1.4. Consulta de Tenencia de Productos

Esta subconsulta SQL hace referencia al código escrito para obtener la matriz de tenencia de productos, la cual es básicamente una serie de campos que corresponden a cada producto bajo estudio con valor de 0 o 1 de tipo entero, estos valores se obtienen de las diferentes tablas donde se guarda la información general de cada producto por cliente. Se relaciona con la consulta principal por medio de un *LET JOIN,* utilizando como campo relacionador el código de cliente, siendo dicha consulta la consulta secundaria, es decir si en dicha consulta no se encuentra registro de producto para cada cliente bajo estudio, se asigna el valor por defecto de 0.

**Condicionales de la Cláusula WHERE**

Para poder identificar cada producto en la tabla se filtra por código o serie de códigos los productos cuyo código de producto sea igual a los especificados, también se especifica que el periodo de estudio sea igual al día de ayer (función SQL ADD_DAYS(CURRENT_DATE, -

1)) para obtener la información más actual de los productos, por último se especifica que el estado actual del producto sea de activo o vigente.

Los atributos que se obtienen de esta consulta se pueden observar en la tabla 3:

| Nombre del Atributo | Tipo de Dato | Valores | Explicación de Cada Valor |
|---|---|---|---|
| **CUENTA_CHEQUES** | Cuantitativa | 0 | No posee el producto |
| | | 1 | Si posee el producto |
| **CUENTA_AHORROS** | Cuantitativa | 0 | No posee el producto |
| | | 1 | Si posee el producto |
| **DEPOSITOS_A_PLAZO** | Cuantitativa | 0 | No posee el producto |
| | | 1 | Si posee el producto |
| **BONOS_DE_CAJA** | Cuantitativa | 0 | No posee el producto |
| | | 1 | Si posee el producto |
| **PRESTAMOS_CONSUMO** | Cuantitativa | 0 | No posee el producto |
| | | 1 | Si posee el producto |
| **PRESTAMOS_VIVIENDA** | Cuantitativa | 0 | No posee el producto |
| | | 1 | Si posee el producto |
| **ADELANTO_PAGO_PLUS** | Cuantitativa | 0 | No posee el producto |
| | | 1 | Si posee el producto |
| **EXTRAFINANCIAMIENTO** | Cuantitativa | 0 | No posee el producto |
| | | 1 | Si posee el producto |
| **TC_VISA** | Cuantitativa | 0 | No posee el producto |
| | | 1 | Si posee el producto |
| **TC_CASHBACK** | Cuantitativa | 0 | No posee el producto |
| | | 1 | Si posee el producto |
| **TC_OLIMPIA** | Cuantitativa | 0 | No posee el producto |
| | | 1 | Si posee el producto |
| **TC_HMC** | Cuantitativa | 0 | No posee el producto |
| | | 1 | Si posee el producto |
| **TC_LADY_LEE** | Cuantitativa | 0 | No posee el producto |
| | | 1 | Si posee el producto |
| **TC_PUMA** | Cuantitativa | 0 | No posee el producto |
| | | 1 | Si posee el producto |

| | | | |
|---|---|---|---|
| **TC_ANTORCHA** | Cuantitativa | 0 | No posee el producto |
| | | 1 | Si posee el producto |
| **TC_CELEBRA** | Cuantitativa | 0 | No posee el producto |
| | | 1 | Si posee el producto |

*Tabla 3. Descripción de Atributos de la Consulta Tenencia de Productos (Elaboración propia)*

### 4.1.5. Consulta de Abandono de Productos

Esta última subconsulta es la consulta utilizada para crear los diferentes datasets de abandono de productos, aquí se recupera información de las tablas que guardan la información de cada producto de forma diaria, se toman los productos adquiridos en el periodo de un año y se revisa 6 meses después el estado del producto, de esta forma se determina por medio de un *CASE* si el producto se considera vigente o abandonado. Esta consulta se relaciona con la consulta principal a través de un *LEFT JOIN* utilizando como campo relacionador el código del cliente.

**Condicionales de la Cláusula WHERE**

Los condicionales que se establecen en este caso es que los códigos de productos recuperados sean igual a los códigos o conjunto de códigos del producto bajo análisis. Para obtener el estado del producto que determinará si el producto fue abandonado o no, se establece que la fecha de adquisición del producto sea igual a su equivalente seis meses después y se utiliza el estado del producto que corresponde a la fecha de 6 meses después de su fecha de adquisición en la cláusula CASE.

Los atributos que se obtienen de esta consulta se describen a continuación en la tabla 4:

| Nombre del Atributo | Tipo de Dato | Clases | Explicación de Cada Clase |
|---|---|---|---|
| **ESTADO_PRODUCTO** | Cualitativo | Abandono | |
| | | Vigente | |

*Tabla 4. Descripción de los Atributos de la Consulta Abandono de Productos (Elaboración propia)*

## 4.2. Técnicas de Agrupamiento para Segmentación de Clientes

En esta fase se utilizarán diferentes algoritmos de clustering sobre el conjunto de clientes bajo análisis, luego se procederá a calcular el porcentaje de tenencia de productos por cada segmento de cliente obtenido y se analizará las diferencias y similitudes de los resultados obtenidos. De los resultados de esta fase se determinará que productos serán analizados en la etapa de creación de modelos predictivos para abandono de productos bancarios.

### 4.2.1. Exploración y Transformación del Dataset Segmentación

Para poder llevar a cabo esta etapa de agrupamiento se utilizará el dataset de Segmentación de Clientes, el cual está compuesto de la consulta principal y la consulta de tenencia de productos, los atributos obtenidos de la consulta principal constituyen las variables utilizadas para realizar la parte de agrupamiento, las variables obtenidas del dataset de tenencia de producto serán utilizadas para calcular el porcentaje de tenencia de productos por segmento. El tipo de datos y valores de las de los atributos se encuentran especificados en el apartado anterior.

- **Periodo de Análisis:** Clientes creado en el sistema CRM a partir del 01/09/2019 hasta el 01/03/2021
- **Total de Instancias:** 127,208 instancias
- **Instancias con Valores Nulos:** 0 instancias
- **Total Atributos:** 22 atributos, 6 atributos demográficos para la fase de agrupamiento y 16 atributos para calcular los porcentajes de tenencia de productos por segmento.

En la figura 10 se muestra el histograma para observar la distribución de cada atributo demográfico a utilizar:

Como se puede observar en los histogramas de frecuencias, para el caso de las variables cualitativas, la mayoría de las variables presenta datos no balanceados en sus valores, pero en general se podría decir que poseen un balanceo aceptable ya que observa una disminución gradual en la frecuencia de sus valores.

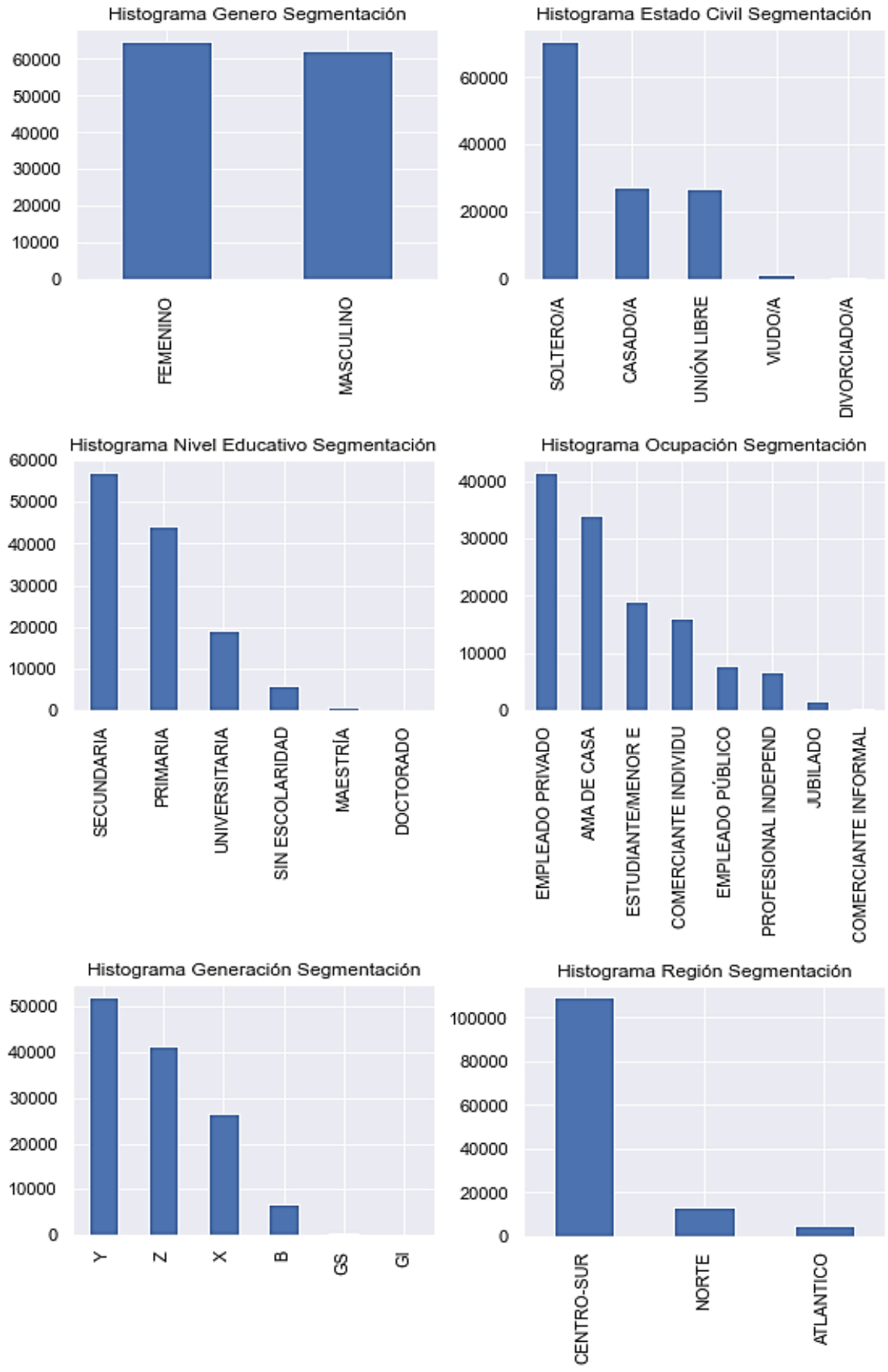



*Figura 10.Histogramas de frecuencia para los atributos de entrada Dataset Segmentación (Elaboración propia)*

**Transformación de Datos:** Para preparar los datos y poder ser utilizados en los distintos algoritmos de agrupamiento se aplicaron transformaciones donde se codificaron como variables dummy los atributos cualitativos, al final se obtuvo un dataset con 30 variables, el cuál será utilizado para entrenar los algoritmos de clustering. Al final de la transformación de los datos se utilizó la técnica PCA descrita en el apartado 3.3.5 para reducir las variables a dos componentes y poder visualizar la distribución de los datos la cual se muestra en la Figura 11.

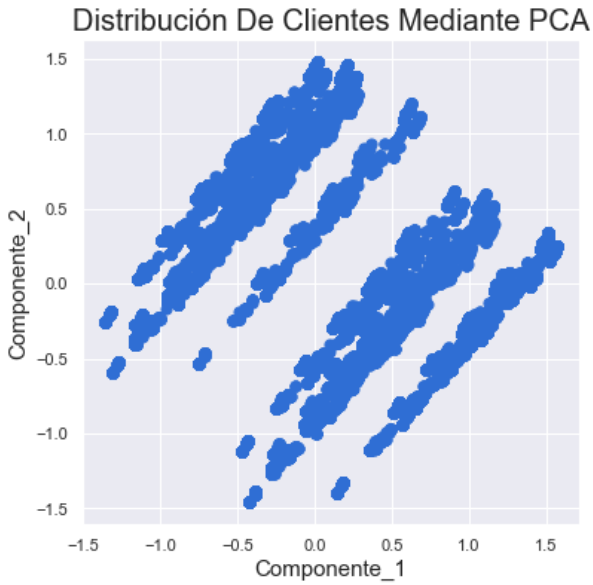



*Figura 11. Distribución gráfica a través de PCA de los clientes a segmentar (Elaboración propia)*

Esta grafica será de gran utilidad en la fase de selección del algoritmo de agrupamiento ya que desde ahora se puede hacer una idea de cómo el algoritmo más optimo debería agrupar los datos en base a como se encuentran distribuido gráficamente.

## 4.2.2. Selección de Algoritmos de Clustering

Para esta fase se inició comparando los diferentes algoritmos de agrupamiento que pueden ser encontrados en la librería scikit-learn de python. Algunos algoritmos fueron descartados ya que no respondieron bien ante un conjunto extenso de datos y la causa de su omisión se puede encontrar en la Tabla 5:

| Nombre de la técnica de Clustering | Razón de su omisión |
|---|---|
| **AffinityPropagation** | Error de memoria y tipo de datos. Los mensajes que se obtuvieron fueron que el algoritmo generaba varias gigas de memoria los cuales no podían ser guardados en caché, aparte |

| | |
|---|---|
| | el dataset a utilizar posee el tipo de datos en pandas float64, el cual, según los mensajes obtenidos no era procesable por el algoritmo. |
| **MeanShift** | Problemas de procesamiento con datos muy grandes. La memoria cache se desbordaba, con lo cual era necesario reiniciar el sistema ya que provocaba congelamiento del sistema operativo |
| **Agglomerative Clustering** | Problemas de procesamiento con datos muy grandes. La memoria cache se desbordaba, con lo cual era necesario reiniciar el sistema ya que provocaba congelamiento del sistema operativo |

*Tabla 5. Descripción de omisión de algoritmos de clustering (Elaboración propia)*

Al final de las pruebas iniciales fueron seleccionados los algoritmos de *DBSCAN*, *KMeans* y *Gaussian Mixture* por sus relativos bajos requerimientos de procesamiento y tiempo de ejecución. A continuación, se explicará cada uno de los algoritmos de agrupamientos seleccionados para realizar el análisis de resultado final.

## 4.2.3. Implementación de Algoritmos de Clustering

### *4.2.3.1. Algoritmo DBSCAN*

Para poder utilizar este algoritmo se determinaron valores apropiados para los argumentos épsilon y número mínimo de elementos por clúster (n), la configuración de valores a analizar se determinó en base a los requerimientos de memoria y que los diferentes valores de épsilon requerían, con valores de épsilon mayores a 1.5 el algoritmo se volvía difícil de computar. Para las configuraciones aceptables de épsilon y n del algoritmo, se calculó el puntaje de la silueta, el cual se define en el apartado 3.3.5 y nos permite hacernos una idea de los parámetros óptimos a utilizar, mediante este puntaje se determinó la configuración a utilizar al momento de implementar el algoritmo, también se tomó en cuenta que valores muy pequeños para épsilon generaban un mayor número de clústeres y valores muy grandes para n generaba un número reducido de clústeres.

Como se puede observar en la Figura 12, los mejores valores de silueta se obtienen con diferentes valores de épsilon y con un mínimo de muestras de 1,000, el segundo puntaje de la silueta aceptable corresponde para la configuración con n = 1,500, pero esta configuración decae casi 0.04 puntos con respecto a la configuración anterior.

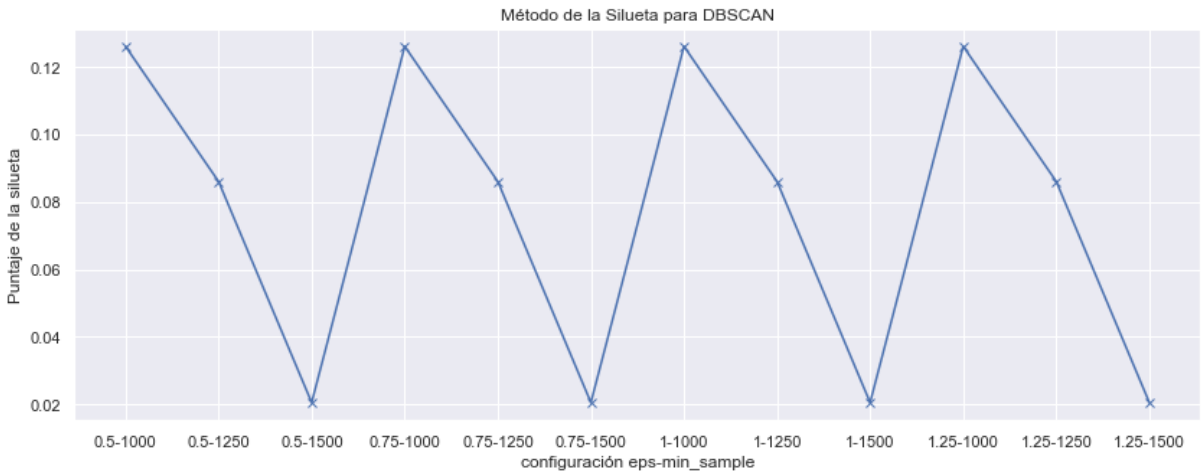



*Figura 12. Puntaje de la Silueta para diferentes configuraciones de DBSCAN (Elaboración propia)*

En base a lo anterior se eligió la siguiente configuración:

- Épsilon: 0.5
- n: 1000

Los clústeres obtenidos fueron los siguientes:

- Puntos ruidosos: 80,345
- Clúster 1: 3,948
- Clúster 2: 1,316
- Clúster 3: 1,353
- Clúster 4: 1,288
- Clúster 5: 1,971
- Clúster 6: 2,175
- Clúster 7: 1,760
- Clúster 8: 1,355
- Clúster 9: 1,300
- Clúster 10: 4,796
- Clúster 11: 3,404
- Clúster 12: 1,298
- Clúster 13: 1,196
- Clúster 14: 2,356
- Clúster 15: 1,815
- Clúster 16: 1,756
- Clúster 17: 1,151
- Clúster 18: 2,461
- Clúster 19: 1,678
- Clúster 20: 1,134
- Clúster 21: 1,579
- Clúster 22: 1,704
- Clúster 23: 1,540
- Clúster 24: 1,182
- Clúster 25: 1,347

### *4.2.3.2. Algoritmo KMeans*

Para poder determinar un número de clústeres para este algoritmo, primero se hizo uso del método de la silueta, los puntajes de la silueta para las diferentes configuraciones de KMeans se observan en la Figura 13:

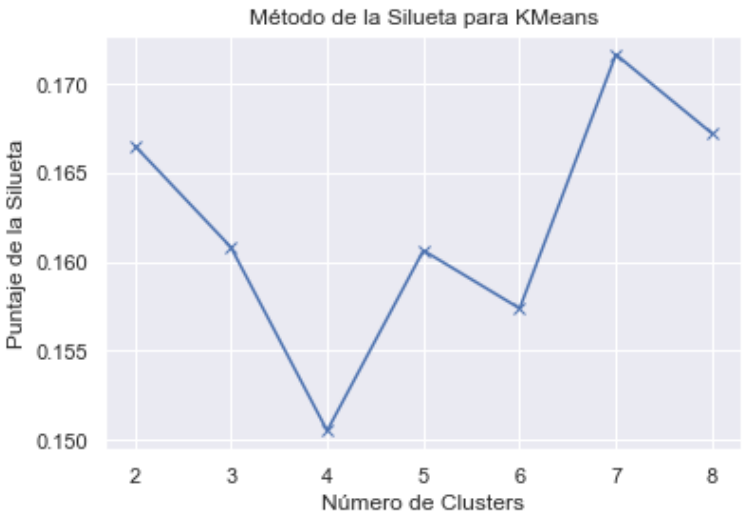



*Figura 13. Puntaje de la Silueta para diferentes números de clústeres de KMeans (Elaboración propia)*

Como se puede observar la gráfica los mejores puntajes de la silueta fueron obtenidos con la configuración de 2 clústeres, con un puntaje de silueta de 0.1665, y con una configuración de 7 clústeres, con un puntaje de silueta de 0.1797, siendo la diferencia entre valores aceptable. Si se toma en cuenta la figura sobre distribución de clientes utilizando PCA, se puede observar que es muy improbable que el conjunto de datos se pueda dividir en 7 grupos distinguibles, por lo tanto, se usará la configuración de 2 clústeres para entrenar el algoritmo.

- n_clusters=2

Los clústeres obtenidos por este algoritmo son los siguientes:

- Clúster 1: 64,848
- Clúster 2: 62,360

### *4.2.3.3. Algoritmo Mezcla Gaussiana*

Este algoritmo también necesita como argumento un valor especifico de número de clústeres a encontrar, para realizar esta tarea se aplicará el método de la silueta a las diferentes configuraciones de número de clúster que puede recibir dicho algoritmo.

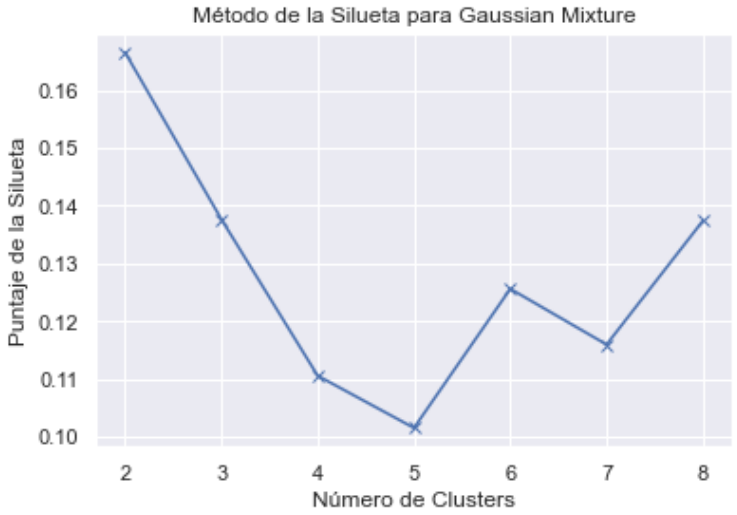



*Figura 14. Puntaje de la Silueta para diferentes números de clústeres de Mezcla Gaussiana (Elaboración propia)*

Como se puede observar en la Figura 14, indiscutiblemente la mejor configuración de clústeres para este algoritmo es 2, con un puntaje de la silueta para dicha configuración de 0.1665, el cual se encuentra casi 0.03 arriba del puntaje más próximo, el cual corresponde a la configuración de 8 clústeres, que posee un puntaje de 0.1374.

- n_components=2

Los clústeres obtenidos por este algoritmo para la configuración anterior son los siguientes:

- Clúster 1: 62,360
- Clúster 2: 64,848

## 4.2.4. Comparación de Resultados de Clustering

Con la generación de los clústeres con cada algoritmo ejecutado, se graficó, a través de la técnica de análisis de componentes, cada uno de los clústeres obtenidos por los algoritmos de machine learning, resultando las siguientes graficas de las Figuras 15, 16 y 17 para cada algoritmo.

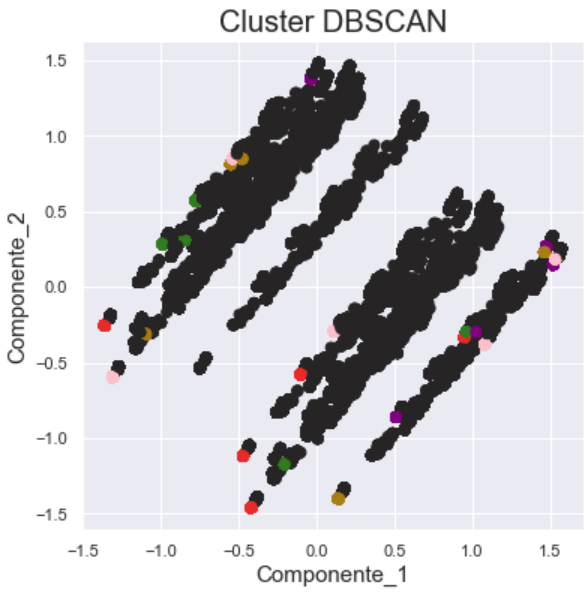



*Figura 15. Segmentación de clientes algoritmo DBSCAN (Elaboración propia)*

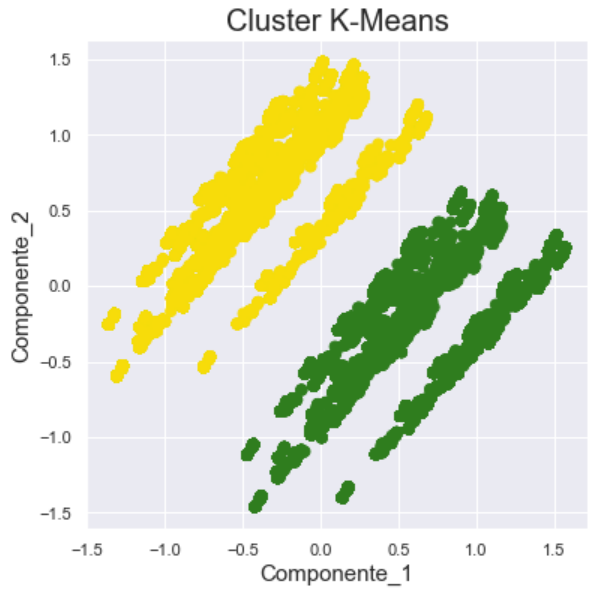



*Figura 16. Segmentación de clientes algoritmo KMeans (Elaboración propia)*


*Figura 17. Segmentación de clientes algoritmo Gaussian Mixture (Elaboración propia)*

A continuación, se muestra en la Tabla 6 una comparativa de los aspectos esenciales de cada algoritmo

| Algoritmo | Configuración | Puntaje de Silueta | Clusters Generados | Puntos Sin Clasicar |
|---|---|---|---|---|
| **DBSCAN** | Eps: 0.5, nim_samples: 1000 | 0.1260 | 25 | 80,345 |
| **KMeans** | n_clusters=2 | 0.1665 | 2 | 0 |
| **Gaussian Mixture** | n_components=2 | 0.1665 | 2 | 0 |

*Tabla 6. Comparativa de resultados algoritmos de agrupación (Elaboración propia)*

Como se puede observar en las Figuras 15, 16 y 17, y en base a los resultados obtenidos de cada técnica de clustering, el algoritmo DBSCAN fue el que presentó el valor de silueta más bajo y el que agrupó a los clientes de forma menos optima, según la gráfica obtenida, también se puede destacar que, con la configuración aplicada al algoritmo, este, generó muchos puntos sin clasificar, lo cual a nivel de segmentación de clientes para el caso actual, lo convierte en un algoritmo ineficaz.

Para el caso del algoritmo de los algoritmos de KMeans y Mezcla Gaussiana. Se pudo observar que ambos algoritmos presentaron el mismo puntaje de silueta al generar ambos algoritmos la cantidad de dos clústeres, los cuales a nivel gráfico son distinguibles el uno del otro.

En base a lo anterior se puede concluir que tanto el algoritmo de KMeans como el Algoritmo de Mezcla Gaussiana son algoritmos óptimos para realizar la tarea de segmentación de clientes que se posee actualmente, identificando cada algoritmo dos grandes grupos de clientes perfectamente distinguibles a nivel visual por medio de PCA, lo cual significa que poseen características demográficas diferentes en comparación de un grupo de clientes a otro.

Para la siguiente actividad de obtener el porcentaje de tenencia de productos por grupo de clientes se utilizarán los clústeres generados por Gaussian Mixture, aunque perfectamente se puede utilizar los clústeres generados por KMeans puesto que ambos segmentaron a los clientes de forma idéntica.

### 4.2.5. Análisis de Tenencia de Productos Por Grupos Demográficos de Clientes

Para esta fase se obtendrá para cada uno de los 16 productos seleccionados el porcentaje de clientes que posee dicho producto en cada clúster generado, esta métrica se calcula obteniendo la media de tenencia del producto, como este campo sólo puede tomar valores entre 0 y 1, calcular la media para cada variable es equivalente a calcular el porcentaje de tenencia, el cual es multiplicado posteriormente por 100. Con este cálculo se pretende encontrar si el porcentaje de tenencia de cada producto posee diferencias marcadas de un grupo demográfico de clientes a otro, de esta forma se podrían desarrollar campañas de colocación de productos según el grupo demográfico de interés

| Producto | % de Tenencia Grupo 1 | % de Tenencia Grupo 2 |
|---|---|---|
| **CUENTA_AHORROS** | 78.1815 | 83.6834 |
| **CUENTA_CHEQUES** | 14.8172 | 6.5075 |
| **PRESTAMOS_CONSUMO** | 3.1398 | 1.3262 |
| **TC_PUMA** | 1.9949 | 0.879 |
| **EXTRAFINANCIAMIENTO** | 1.3647 | 0.8774 |
| **TC_CELEBRA** | 0.7858 | 0.586 |
| **TC_VISA** | 0.5484 | 0.5582 |
| **TC_CASHBACK** | 0.4362 | 0.3038 |
| **TC_OLIMPIA** | 0.2085 | 0.0401 |
| **DEPOSITOS_A_PLAZO** | 0.1876 | 0.2884 |
| **PRESTAMOS_VIVIENDA** | 0.1523 | 0.0802 |

| | | |
|---|---|---|
| **BONOS_DE_CAJA** | 0.0305 | 0.0386 |
| **ADELANTO_PAGO_PLUS** | 0.0257 | 0.0077 |
| **TC_HMC** | 0.0048 | 0 |
| **TC_LADY_LEE** | 0 | 0 |
| **TC_ANTORCHA** | 0 | 0 |

*Tabla 7. Análisis de tenencia de productos por grupo identificado (Elaboración propia)*

De la Tabla 7 se puede observar que no existen diferencias significativas entre el porcentaje de tenencia de cada grupo bancario con respecto a los grupos demográficos identificados en la etapa de agrupamiento, por otro lado, si es posible establecer variaciones perceptibles entre los porcentajes de tenencia.

**Para el grupo # 1** se puede apreciar que poseen mayor disposición a poseer productos del tipo financiamiento como ser préstamos de consumo, extra financiamiento y préstamos de vivienda, también se puede observar que poseen un mayor porcentaje de tenencia de tarjetas de crédito, con lo cual se puede inferir que este grupo tiene mayor disposición a adquirir productos que le permitan hacer gastos de mayor índole y pagar posteriormente a tiempo inmediato a través de tarjetas de crédito o pagar a tiempo futuro por medio de préstamos y extra financiamientos.

**Para el grupo # 2** se puede observar que posee mayor inclinación a poseer productos del tipo cuentas de depósito, ya que posee un mayor porcentaje de tenencia de cuentas de ahorro, bonos de caja y depósitos a plazo, lo cual se puede traducir en que este grupo tiene mayor disposición a productos que le permitan ahorrar su propio dinero y obtener utilidades de dicho dinero.

## 4.3. Técnicas de Clasificación para Abandono de Productos

En esta fase se tratará de crear modelos predictivos para los productos que se reconocieron como más significativos en la etapa de análisis de la tenencia de productos, Lo que se pretende lograr es que cuando un cliente adquiera un producto en específico este mismo permanezca con el producto por un periodo de tiempo mayor a 6 meses.

De la etapa anterior se pudo observar que los productos con mayor tenencia los representan las cuentas de depósitos, de este tipo de productos se ha seleccionado las cuentas de cheque, de las tarjetas de crédito se han seleccionado dos de las tarjetas de crédito con mayor porcentaje de tenencia, estas son la tarjeta de crédito Puma, y Visa. Para el caso de los productos de financiamiento como ser préstamos, se han omitido de este análisis ya que dichos productos no pueden ser abandonados dado que los clientes firman acuerdos legales

donde se comprometen a permanecer con dicho producto hasta que cumplan con las obligaciones que estos representan.

Para cada producto en específico se ha generado un dataset atributos de entrada y su respectiva clase de salida, por cada producto se probarán 4 algoritmos de clasificación los cuales son *CART*, *Random Forest*, *KNeighbors* y *Logistic Regression* los cuales han sido descritos en apartados anteriores, de estos algoritmos se comprarán sus resultados para conocer cual algoritmo es óptimo para cada producto. Al final se hará un análisis global de los resultados obtenidos en cada caso utilizando como parámetros de medición las métricas obtenidas de la matriz de confusión, dichas métricas de explican en el apartado 3.4.4 de la actual investigación.

Los datasets que se utilizarán para entrenar los algoritmos predictivos se han formado de la unión de la consulta principal con la unión de la consulta de abandono de productos, esta última consulta se ha personalizado para cada producto de forma que obtenga sólo la información de abandono o vigencia del producto específico que se analizará.

## 4.3.1. Predicción de abandono de Productos para Cuentas de Cheque

### *4.3.1.1. Exploración y Transformación de Dataset*

A continuación, se explorará el dataset que se generó para desarrollar los modelos predictivos del producto Cuenta de Cheques.

- **Periodo de Análisis:** Clientes adquirieron el producto cuenta de cheques partir del 01/03/2020 hasta el 01/03/2021
- **Total de Instancias:** 11,279 instancias
- **Instancias con Valores Nulos:** 0 instancias
- **Total Atributos:** 10 atributos. 9 atributos de entrada 1 atributo de salida (Estado Producto)

Se puede observar en la Figura 18, para todos los casos de los atributos de entrada, siempre se encuentra un valor predominante sobre el resto de los valores. También se observa en la figura 19 que para los atributos créditos, débitos y transacciones promedio, los valores se concentran en un rango relativamente pequeño y poseyendo los tres atributos muchos valores outliers. Para el atributo de salida se observa en la Figura 20 que existe un desbalance sustancial de las dos clases a predecir, siendo la clase de interés la clase minoritaria.

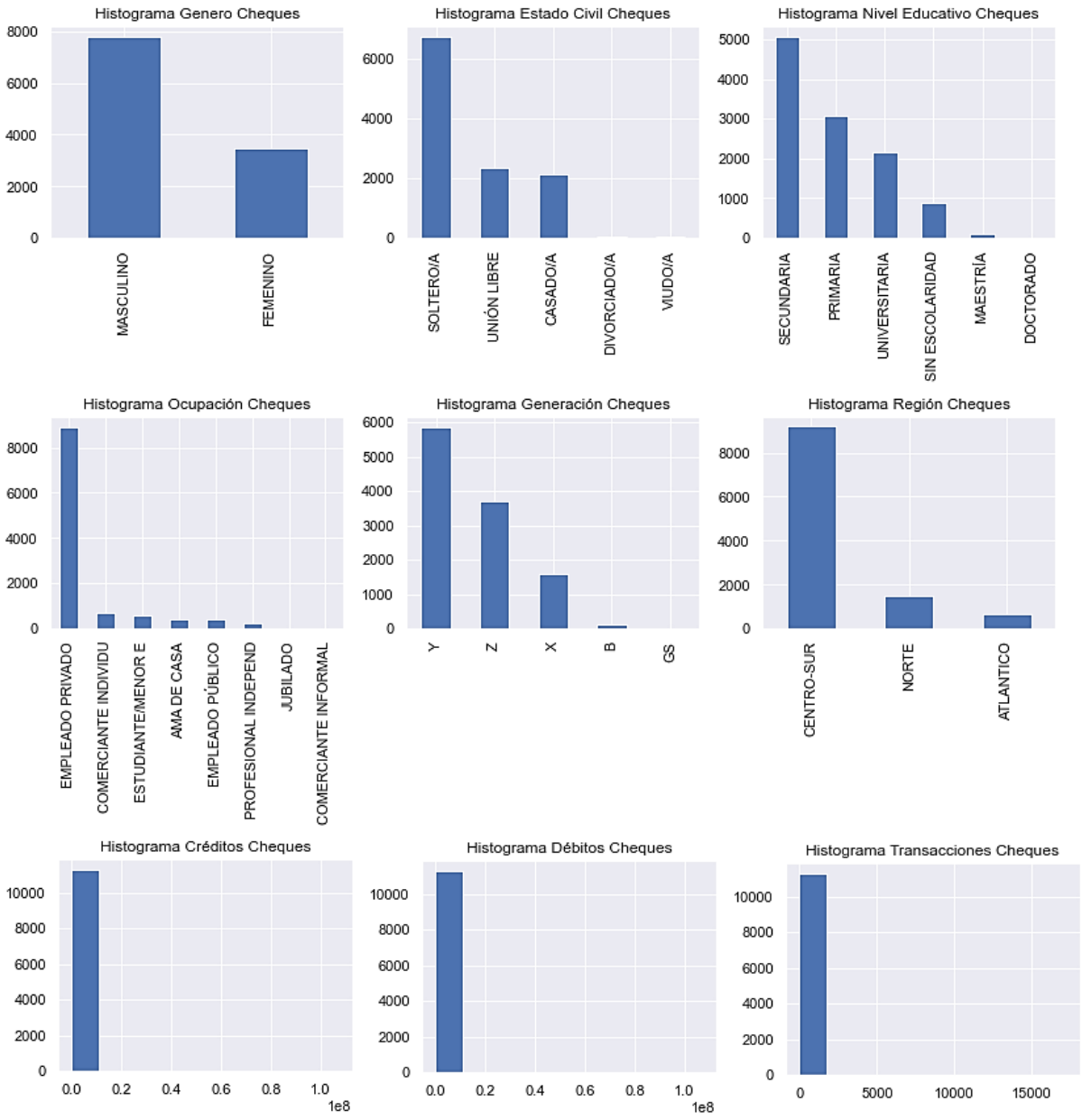



*Figura 18. Histogramas de Frecuencia para atributos de entrada Dataset Abandono Cuenta de Cheques (Elaboración propia)*

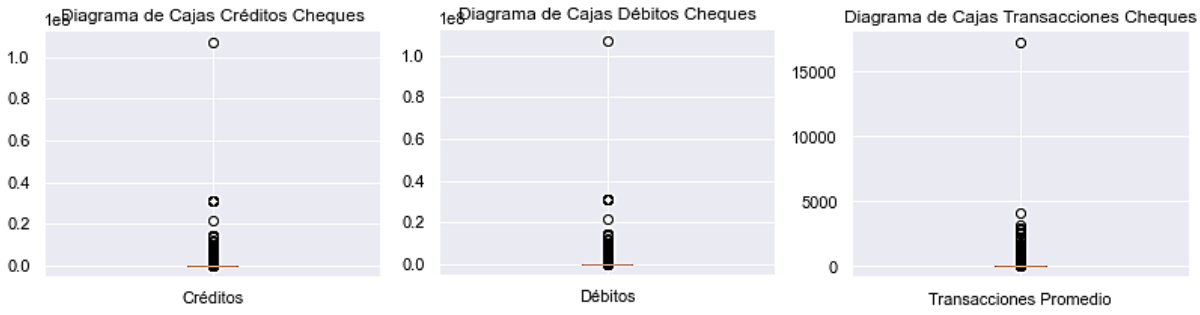



*Figura 19. Diagramas de Caja para atributos numéricos Dataset Abandono Cuenta de Cheques (Elaboración propia)*

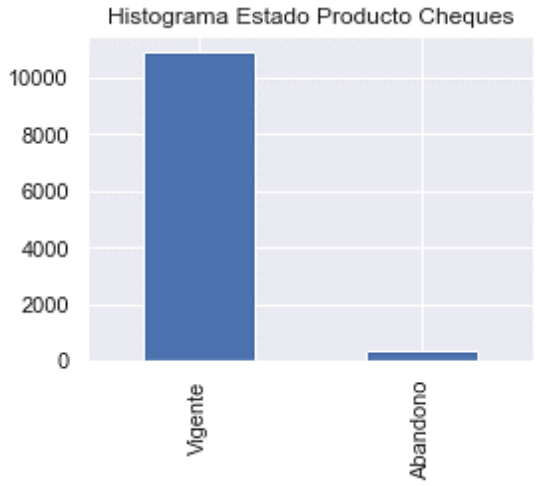



*Figura 20. Histograma de Frecuencia para la clase objetivo en Dataset Abandono Cuenta de Cheques (Elaboración propia)*

**Transformación de Datos:** Las trasformaciones aplicadas a los atributos de tipo cualitativo fueron la codificación de dichos atributos en variables dummy, para los atributos de tipo cuantitativo se aplicó normalización de datos. Al final se obtuvo un dataset con 33 variables

de tipo dummy y normalizadas y una variable que representa la clase a predecir.

### *4.3.1.2. Entrenamiento de Modelos de Clasificación*

**Preparación de Datos de Entrenamiento y Validación:** para entrenar los modelos seleccionados se utilizó una distribución de 80-20, la cual corresponde al 80% de las instancias disponibles para entrenar los modelos y el 20% para validarlos. Para lidiar con el problema de clases desbalanceadas se aplicó la técnica de SMOTE, descrita anteriormente en el apartado 3.3.5 al conjunto de datos de entrenamiento, con esta técnica multiplicaremos las instancias que pertenecen a la clase de interés abandono, obteniendo los siguientes resultados:

- Antes de SMOTE : Clase 'Vigente': 8,712, Clase 'Abandono': 3,11
- Después de SMOTE : Clase 'Vigente': 8,712, Clase 'Abandono': 8,712

Para el algoritmo de KNeighbors se estableció un parámetro de n_neighbors=14, el cuál mostró arrojar un mejor resultado para la sensibilidad del modelo (gráfica en anexos).

Para el algoritmo de Logistic Regression se estableció el parámetro solver= ‘sag’ ya que con este parámetro se obtuvo un mejor valor de sensibilidad (gráfica en anexos).

### *4.3.1.3. Evaluación de Resultados*

| Cta. Cheques | Clase | Precision | Recall | F1 score | AUC score |
|---|---|---|---|---|---|
| **CART** | Vigente | 0.97 | 0.89 | 0.93 | 0.52 |
| | Abandono | 0.03 | 0.11 | 0.04 | |
| **Random Forest** | Vigente | 0.98 | 0.91 | 0.94 | 0.57 |
| | Abandono | 0.03 | 0.11 | 0.04 | |
| **KNeighbors** | Vigente | 0.98 | 0.74 | 0.84 | 0.57 |
| | Abandono | 0.04 | 0.37 | 0.07 | |
| **Logistic Regression** | Vigente | 0.98 | 0.60 | 0.74 | 0.62 |
| | Abandono | 0.04 | 0.61 | 0.07 | |

*Tabla 8. Comparativa de Resultados Algoritmos de Clasificación Dataset Abandono Cuenta de Cheques (Elaboración propia)*

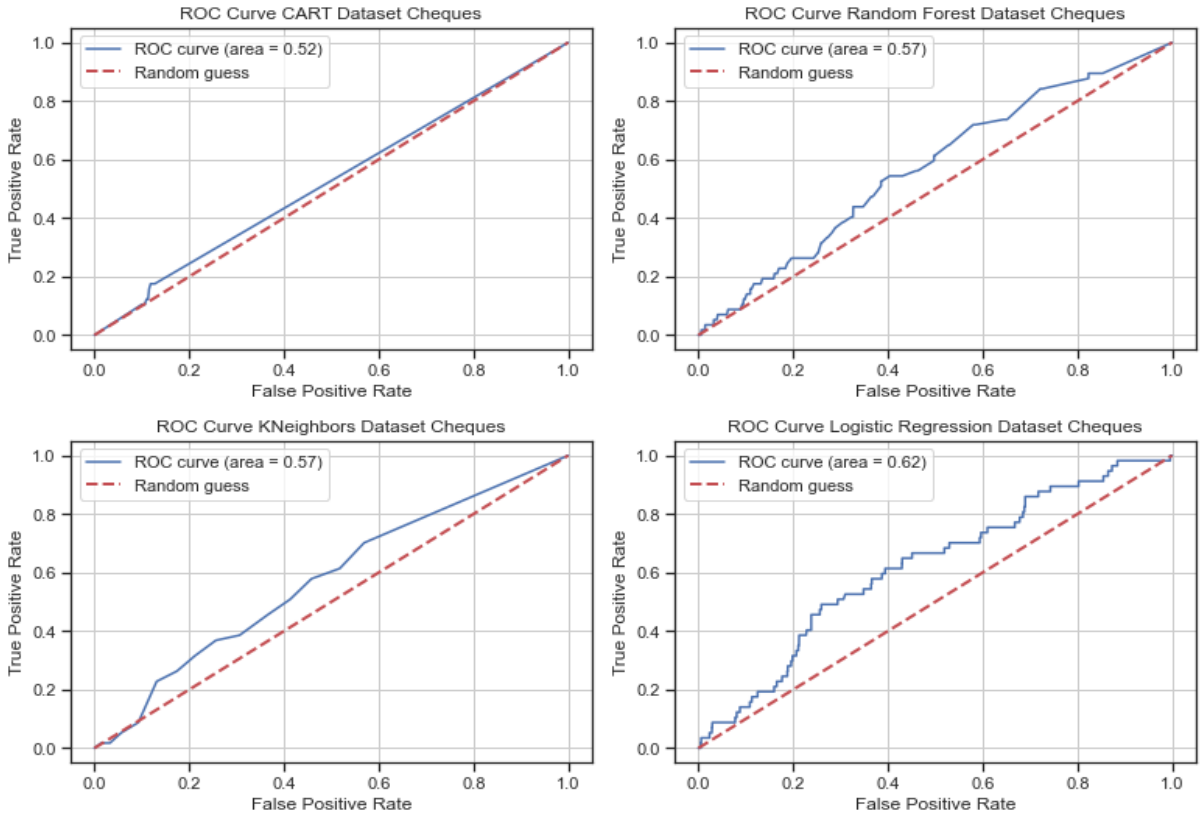



*Figura 21. Comparativa de Curvas ROC para Abandono de Cuentas de Cheque (Elaboración propia)*

Como se puede observar por medio de la tabla 8 y a través de las gráficas expuestas en la Figura 21, el algoritmo de regresión logística obtuvo un mejor desempeño en la fase de evaluación, ya que presentó el mejor resultado para el área bajo la curva, lo cual implica que en general es el algoritmo que mejor clasificará las instancias. Para los algoritmos de CART

y Random Forest, se puede observar que obtuvieron los mismos valores de precisión y sensibilidad (recall), pero Random Forest obtuvo un mejor puntaje de área bajo la curva ya que también obtuvo mejores valores de precisión y sensibilidad para la clase vigente. Con KNeighbors se puede observar un aumento en la sensibilidad con Respecto a Random Forest, pero a costa de sacrificar la precisión de la clase abandono. También se puede deducir que a nivel general el algoritmo de CART es el que se considera clasificará de peor forma las instancias.

## 4.3.2. Predicción de Abandono de Productos para Tarjeta de Crédito Puma

### *4.3.2.1. Exploración y Transformación de Dataset*

A continuación, se explorará el dataset que se generó para desarrollar los modelos predictivos del producto Tarjeta de Crédito Puma

- **Periodo de Análisis:** Clientes adquirieron el producto tarjeta de crédito puma a partir del 01/03/2020 hasta el 01/03/2021
- **Total de Instancias:** 2,223 instancias
- **Instancias con Valores Nulos:** 0 instancias
- **Total Atributos:** 10 atributos. 9 atributos de entrada 1 atributo de salida (Estado Producto)

Se puede observar en la Figura 22 y 23 que los atributos de entrada de este dataset poseen una distribución similar al dataset anterior donde los atributos cualitativos poseen una clase mayoritaria fuertemente marcada y los atributos cuantitativos se distribuyen en un rango de valores pequeños presentando un gran número de outliers. Para el atributo de salida se observa en la Figura 24 que existe un desbalance de clases, que se puede decir, es menor al desbalanceo que presentaba el dataset anterior.

**Transformación de Datos:** Las trasformaciones aplicadas a los atributos de tipo cualitativo fueron la codificación de dichos atributos en variables dummy, para los atributos de tipo cuantitativo se aplicó normalización de datos. Al final se obtuvo un dataset con 32 variables de tipo dummy, variables normalizadas para los atributos numéricos y una variable que representa la clase a predecir.

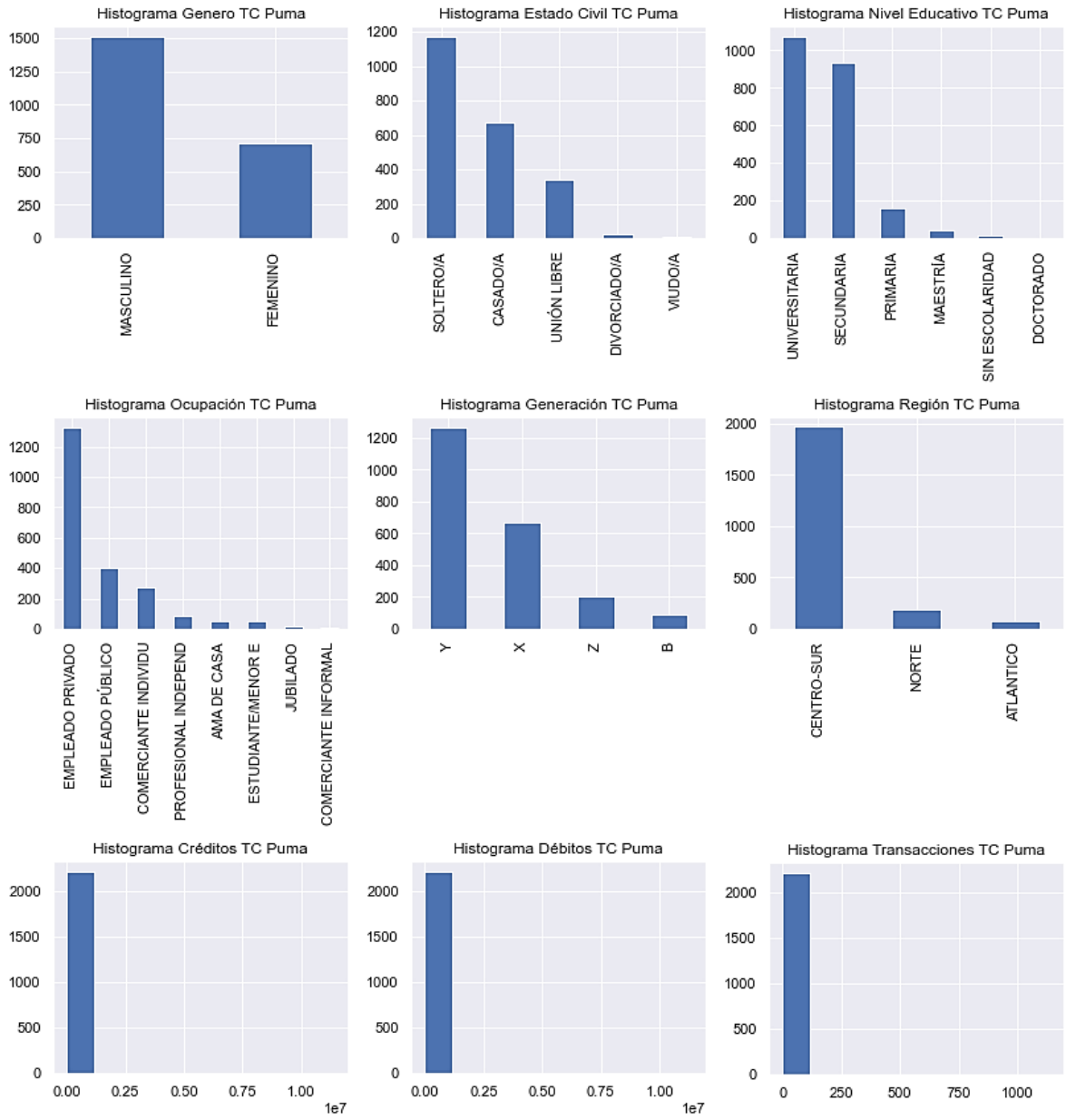



*Figura 22. Histogramas de Frecuencia para atributos de entrada Dataset Abandono TC Puma (Elaboración propia)*

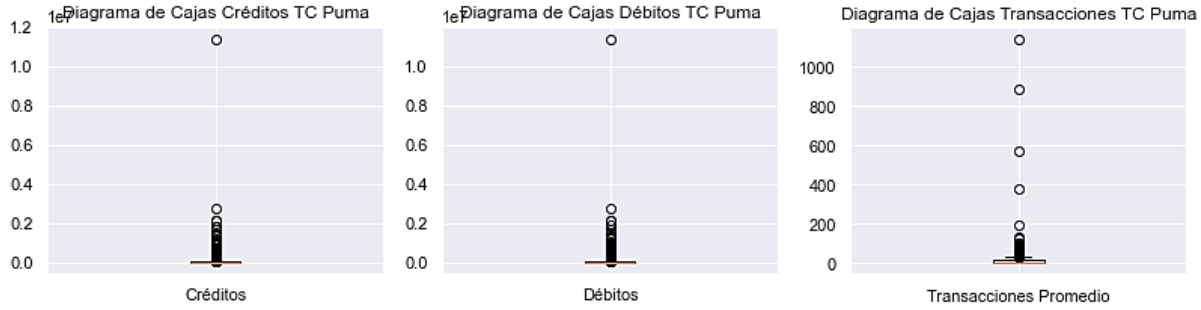



*Figura 23. Diagramas de Caja para atributos numéricos Dataset Abandono TC Puma (Elaboración propia)*

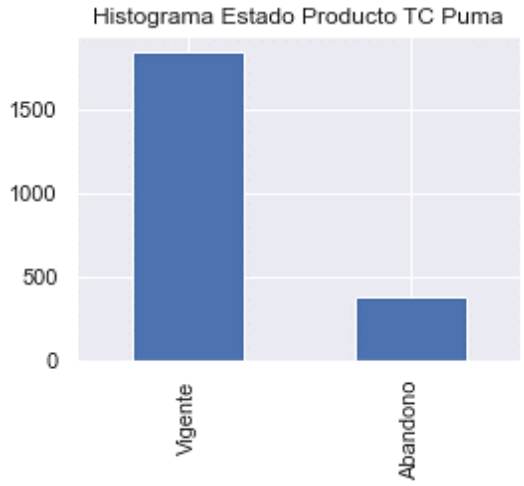



*Figura 24. Histograma de Frecuencia para la clase objetivo en Dataset Abandono TC Puma (Elaboración propia)*

### *4.3.2.2. Entrenamiento de Modelos de Clasificación*

**Preparación de Datos de Entrenamiento y Validación:** para entrenar los modelos seleccionados se utilizó una distribución de 80-20, la cual corresponde al 80% de las instancias disponibles para entrenar los modelos y el 20% para validarlos. Para mitigar el problema de clases desbalanceadas se aplicó la técnica de SMOTE al conjunto de datos de entrenamiento para obtener clases balanceadas, obteniendo los siguientes resultados:

- Antes de SMOTE : Clase 'Vigente': 1,480, Clase 'Abandono': 298
- Después de SMOTE : Clase 'Vigente': 1480, Clase 'Abandono': 1480

Para el algoritmo de KNeighbors se estableció un parámetro de n_neighbors=4, el cuál mostró arrojar un mejor resultado para la sensibilidad del modelo (gráfica en anexos).

Para el algoritmo de Logistic Regression se estableció el parámetro solver= 'liblinear' ya que con este parámetro se obtuvo un mejor valor de sensibilidad (gráfica en anexos).

### *4.3.2.3. Evaluación de Métricas*

| TC Puma | Clase | Precision | Recall | F1 score | AUC score |
|---|---|---|---|---|---|
| **CART** | Vigente | 0.86 | 0.72 | 0.78 | 0.57 |
| | Abandono | 0.25 | 0.42 | 0.31 | |
| **Random Forest** | Vigente | 0.87 | 0.80 | 0.83 | 0.61 |
| | Abandono | 0.31 | 0.42 | 0.36 | |
| **KNeighbors** | Vigente | 0.86 | 0.56 | 0.67 | 0.60 |
| | Abandono | 0.21 | 0.56 | 0.31 | |

| | | | | | |
|---|---|---|---|---|---|
| **Logistic Regression** | Vigente | 0.86 | 0.48 | 0.62 | 0.60 |
| | Abandono | 0.21 | 0.63 | 0.31 | |

*Tabla 9. Comparativa de Resultados Algoritmos de Clasificación Dataset Abandono TC Puma (Elaboración propia)*

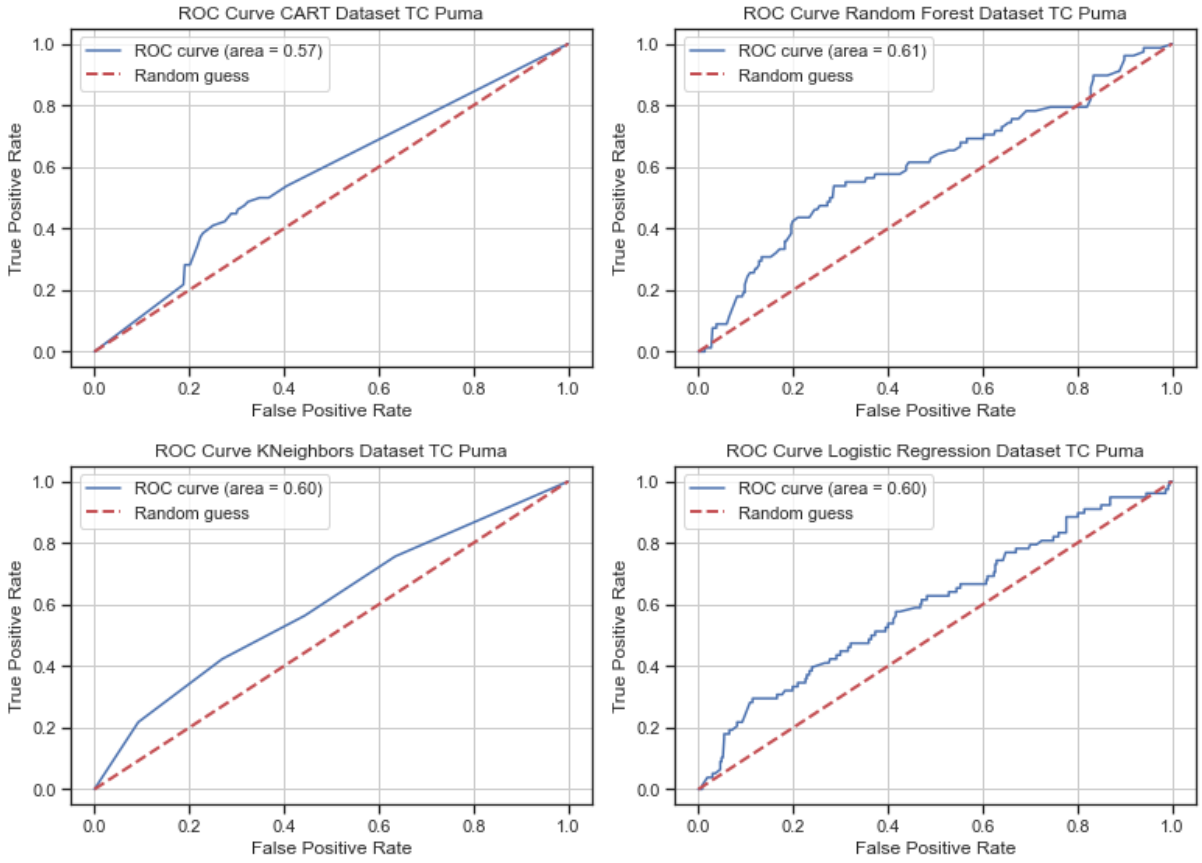



*Figura 25. Comparativa de Curvas ROC para Abandono Tarjeta de Crédito Puma (Elaboración propia)*

Según los datos obtenidos en la Tabla 9 para los resultados de los modelos predictivos entrenados y las gráficas de Área Bajo la Curva de la Figura 25, se puede apreciar que el algoritmo de Random Forest tuvo un mejor desempeño a nivel general, obteniendo un puntaje de área bajo la curva de 0.61, sim embargo los algoritmos de KNeighbors y Regresión Logística obtuvieron mejores puntajes de sensibilidad para la clase abandono. El algoritmo que obtuvo el desempeño más precario fue el algoritmo CART ya que obtuvo el menor puntaje de área bajo la curva.

## 4.3.3. Predicción de Abandono de Producto Tarjeta de Crédito Visa

### *4.3.3.1. Exploración y Transformación de Dataset*

A continuación, se explorará el dataset que se generó para desarrollar los modelos predictivos del producto Tarjeta de Crédito Visa

- **Periodo de Análisis:** Clientes adquirieron el producto tarjeta de crédito visa a partir del 01/03/2020 hasta el 01/03/2021
- **Total de Instancias:** 1,047instancias
- **Instancias con Valores Nulos:** 0 instancias
- **Total Atributos:** 10 atributos. 9 atributos de entrada 1 atributo de salida (Estado Producto)

En este caso se pueden observar algunas diferencias en comparación a los datasets anteriores. En la Figura 26 El atributo “Genero” presenta un mejor balanceo en sus clases. En el caso de los atributos numéricos se observa en la Figura 27 que existe un menor aglutinamiento de los valores, distribuyéndose estos en rangos ligeramente más uniformes, aún se puede observar la presencia de datos atípicos para estos atributos, pero se observan en menor medida. Para la clase a predecir también se puede contemplar en la Figura 28 que este dataset presenta el menor nivel de desbalanceo de clases.

**Transformación de Datos:** Las trasformaciones aplicadas a los atributos de tipo cualitativo fueron la codificación de dichos atributos en variables dummy, para los atributos de tipo numérico se realizó la normalización de sus datos. Al final se obtuvo un dataset con 32 variables de tipo dummy, variables normalizadas para los atributos numéricos y una variable que representa la clase a predecir.

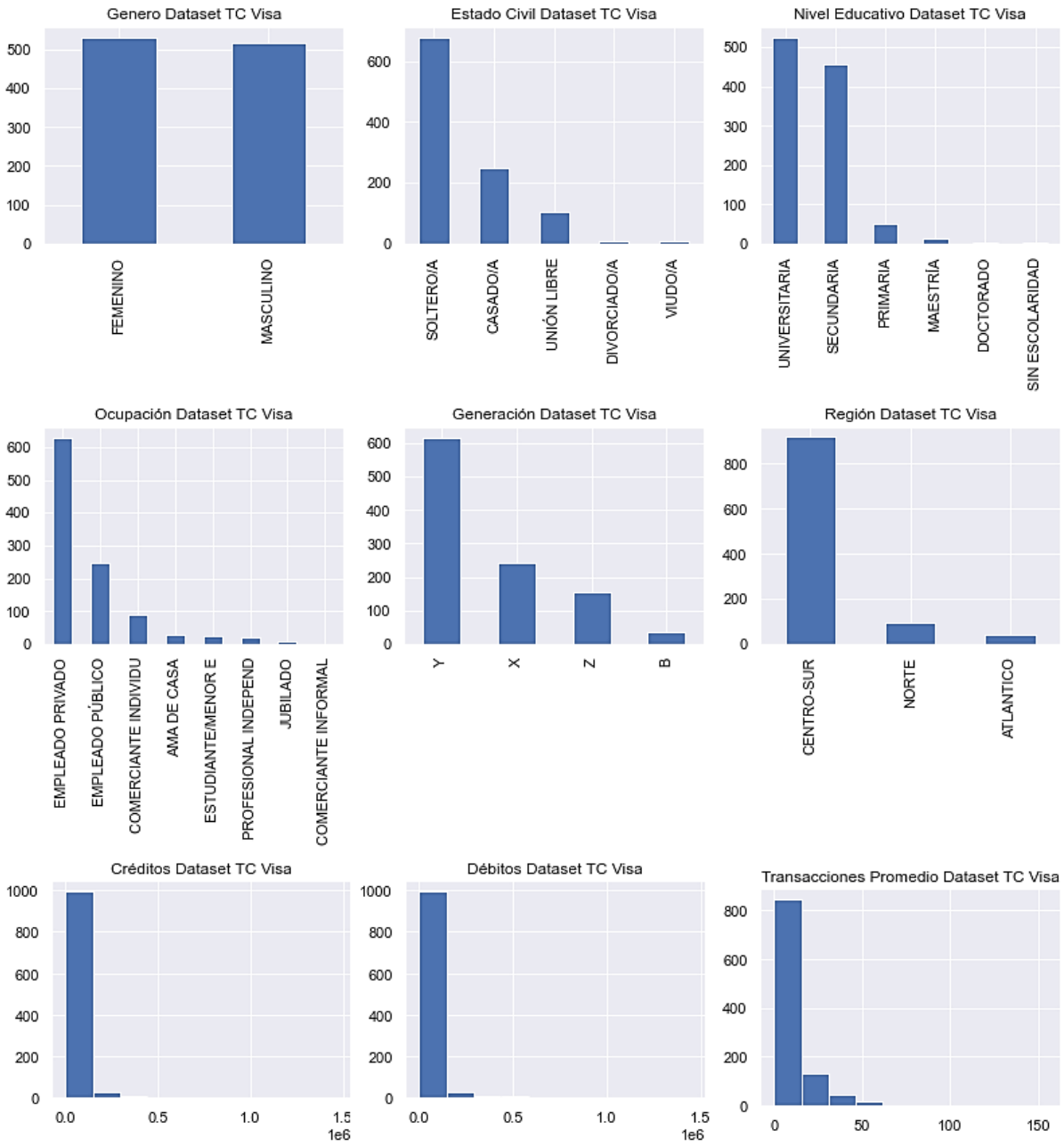



*Figura 26. Histogramas de Frecuencia para atributos de entrada Dataset Abandono TC Visa (Elaboración propia)*

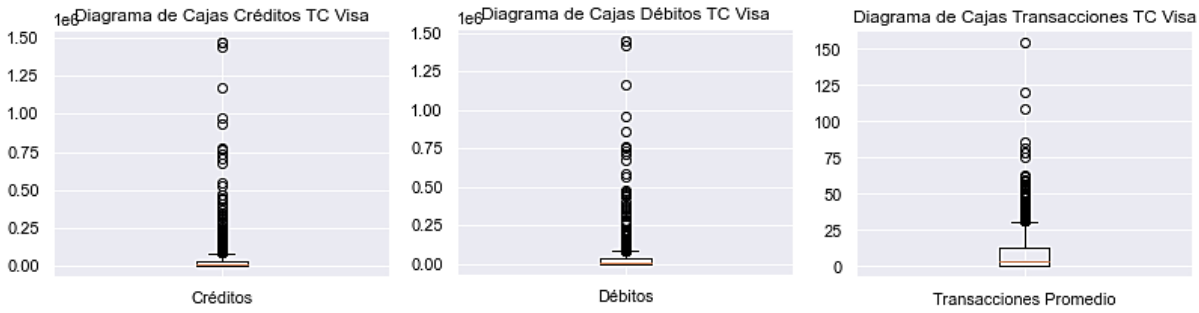



*Figura 27. Diagramas de Caja para atributos numéricos Dataset Abandono TC Visa (Elaboración propia)*

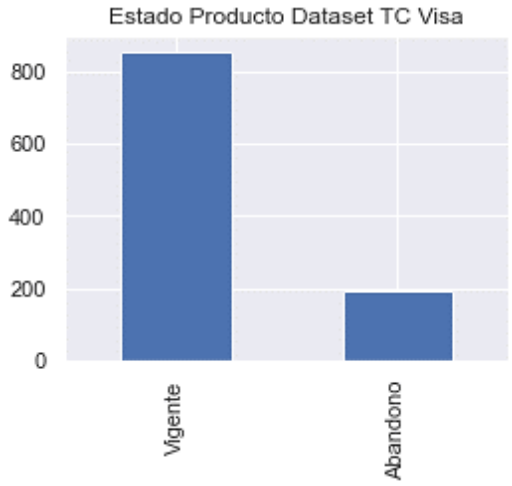



*Figura 28. Histograma de Frecuencia para la clase objetivo en Dataset Abandono TC Visa (Elaboración propia)*

### *4.3.3.2. Entrenamiento de Modelos de Clasificación*

**Preparación de Datos de Entrenamiento y Validación:** para entrenar los modelos seleccionados se utilizó una distribución de 80-20, la cual corresponde al 80% de las instancias disponibles para entrenar los modelos y el 20% para validarlos. Para reducir el inconveniente de clases desbalanceadas se utilizó SMOTE como técnica de sobre-muestreo al conjunto de datos de entrenamiento, al final se obtuvo la siguiente distribución del atributo a predecir:

- Antes de SMOTE : Clase 'Vigente': 691, Clase 'Abandono': 146
- Después de SMOTE : 'Abandono': 691, 'Vigente': 691

Para el algoritmo de KNeighbors se estableció un parámetro de n_neighbors=14, el cuál mostró arrojar un mejor resultado para la sensibilidad del modelo (gráfica en anexos).

Para el algoritmo de Logistic Regression se estableció el parámetro solver= ‘saga’ ya que con este parámetro se obtuvo un mejor valor de sensibilidad (gráfica en anexos).

### *4.3.3.3. Evaluación de Métricas*

| TC Visa | Clase | Precision | Recall | F1 score | AUC score |
|---|---|---|---|---|---|
| **CART** | Vigente | 0.81 | 0.76 | 0.78 | 0.56 |
| | Abandono | 0.29 | 0.35 | 0.31 | |
| **Random Forest** | Vigente | 0.82 | 0.82 | 0.82 | 0.64 |
| | Abandono | 0.36 | 0.35 | 0.35 | |
| **KNeighbors** | Vigente | 0.84 | 0.49 | 0.62 | 0.62 |

| | | | | | |
|---|---|---|---|---|---|
| | Abandono | 0.27 | 0.67 | 0.39 | |
| **Logistic Regression** | Vigente | 0.85 | 0.58 | 0.69 | 0.60 |
| | Abandono | 0.30 | 0.63 | 0.40 | |

*Tabla 10. Comparativa de Resultados Algoritmos de Clasificación Dataset Abandono TC Visa (Elaboración propia)*

A continuación, se muestran a nivel gráfico la Curva ROC para cada uno de los algoritmos:

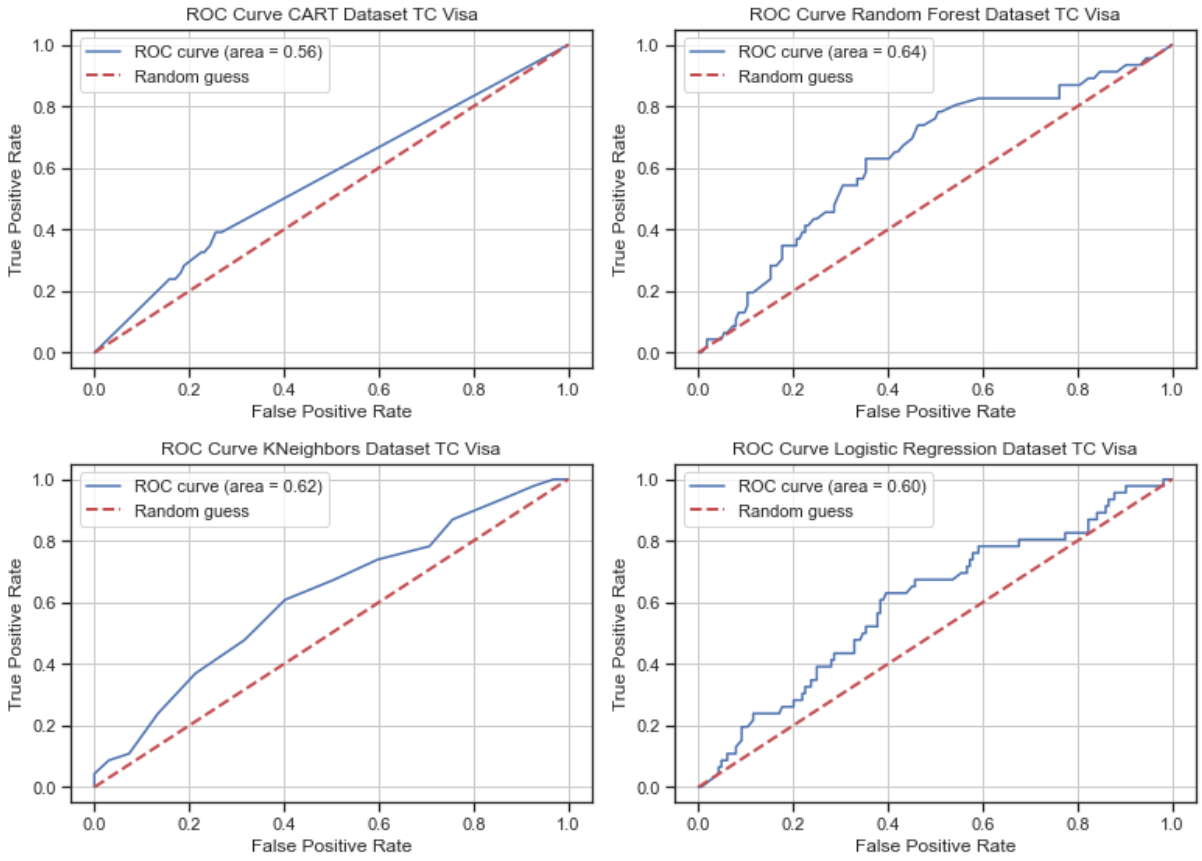



*Figura 29. Comparativa de Curvas ROC para Abandono Tarjeta de Crédito Visa (Elaboración propia)*

De los resultados obtenidos de la Tabla 10 y las gráficas de Área Bajo la Curva de la Figura 29, se observa que el algoritmo Random Forest obtuvo el puntaje de área bajo la curva más alto, llegando a ser de 0.64, esto indica que en términos generales es el algoritmo que clasificará de mejor manera las dos clases a predecir. Sin embargo, los algoritmos de K-Vecinos y Regresión Logística obtuvieron puntajes más altos de sensibilidad y F1 para la clase abandono, los cual indica que estos algoritmos acertaran más al clasificar correctamente las instancias que pertenezcan a esta clase. También cabe resaltar que el algoritmo CART obtuvo el puntaje más bajo para el área bajo la curva, lo cual se puede traducir en que tendrá el peor desempeño entre los cuatro algoritmos bajo estudio para este producto en específico.

### 4.3.4. Análisis de Resultados de los Modelos de Clasificación

Tomando en cuenta los resultados de los modelos de clasificación entrenados para predecir el abandono de los tres productos financieros bajo estudio, se puede concluir de forma precisa que el algoritmo CART tuvo el desempeño más deficiente en los tres casos de estudio.

De forma diferente, la conclusión del algoritmo que tuvo el mejor desempeño no es tan clara y su respuesta debe basarse en como cada conjunto de datos está distribuido y cómo se comportan con cada algoritmo utilizado y en base a las necesidades que el negocio disponga en un momento y contexto determinado. Por ejemplo, se puede apreciar que el Algoritmo de Random Forest, en comparación a los demás algoritmos analizados tuvo los mejores puntajes de F1 para la clase 'Vigente' y resultados promedio de F1 para la clase de interés 'Abandono', estos resultados le permitieron tener los mejores resultados de área la curva lo cual se traduce en que es el algoritmo que presentó una mejor relación de especificidad y sensibilidad. Por otra parte, si el estudio se centra en la clase de interés 'Abandono', el algoritmo de Regresión logística demostró tener los mejores puntajes para la sensibilidad, a costa de sacrificar la precisión del modelo, esto indica que habrá un mayor número de falsos negativos, pero se elevará el número de verdaderos positivos.

De lo anterior se puede decir que si las necesidades del negocio se centran en obtener un modelo que prediga de forma aceptable las instancia que pertenecen a ambas clases, entonces se recomienda utilizar Random Forest como algoritmo de clasificación. Por otra parta si el objetivo es predecir el mayor número de clientes que abandonará un producto de forma correcta el algoritmo a emplear deberá ser la regresión logística.

Cabe recalcar que si se toma en cuenta que el conjunto de datos que presentó una distribución más uniforme en sus valores y cuyos atributos numéricos presentaron menos valores atípicos fue el conjunto de datos utilizado para entrenar y evaluar los modelos predictivos de abandono para el producto Tarjeta de Crédito Visa y que dichos modelos también arrojaron los mejores resultados para las métricas evaluadas, se puede concluir que el desbalanceo de las clases y los valores atípicos en los conjuntos de datos utilizados merman la calidad de los modelos entrenados.

Por último, es importante mencionar que los resultados obtenidos para todos los modelos predictivos desarrollados se consideran insuficientes para poder utilizar dichos modelos en un ambiente productivo, por lo cual se deberá buscar la forma de mejorar las métricas de evaluación por medio del mejoramiento de los conjuntos de datos utilizados y pruebas integrales de análisis continuas hasta obtener los modelos de clasificación óptimos para ser utilizados en un contexto real.

# 5. Conclusiones y trabajo futuro

## 5.1. Conclusiones

El objetivo principal del trabajo actual pretendía realizar un análisis financiero de los productos de Banco Atlántida con el fin de reconocer patrones demográficos en los clientes y variables de comportamiento que permitan gestionar de mejor forma la colocación y retención de productos. Para lograr este cometido se planteó el uso de técnicas de inteligencia artificial de aprendizaje supervisado y no supervisado para realizar el desarrollo del análisis.

Para poder determinar la segmentación de los clientes a nivel demográfico se planteó comparar diferentes técnicas de agrupamiento y analizar el agrupamiento efectuado por cada una de ellas para seleccionar la segmentación de clientes más idónea, del resultado anterior se identificaron dos segmentos de clientes a nivel demográfico, los cuales presentaron tener afinidad a tipos de productos diferentes según el grupo demográfico al cual pertenecían, siendo este margen de afinidad no sustancial.

Del análisis anterior se identificaron los productos con mayor tenencia y se seleccionaron tres de ellos para predecir su pronto abandono por parte de los clientes que adquieren dichos productos, para lograr este cometido se emplearon diferentes algoritmos de clasificación los cuales se compararon entre si por medio de diferentes métricas de evaluación y se determinó que el algoritmo de Random Forest sería el algoritmo adecuado a utilizar si se desea obtener clasificaciones aceptables para ambas clases mientras que el algoritmo de regresión logística se identificó como el algoritmo a preferir si se desea obtener la mejor predicción de productos abandonados. De igual forma los resultados obtenidos en esta fase se pueden mejorar si se realizan las modificaciones necesarias a las variables a utilizar para entrenar los modelos a desarrollar.

En base a lo expuesto anteriormente se determina que los resultados obtenidos son aceptables como primer vistazo en el proceso de descubrimiento del conocimiento aplicado a un Data Warehouse con una estructura compleja y volumen de datos considerados como datos masivos. De este análisis inicial se pueden tomar las pautas necesarias para realizar las mejoras a los modelos entrenados hasta obtener modelos que puedan ser utilizados en diferentes aplicaciones que la entidad bancaria considere oportunas.

## 5.2. Líneas de trabajo futuro

Partiendo del estudio efectuado en el presente trabajo, se pueden establecer como líneas de trabajo futuro los siguientes puntos:

- Dado que el proceso de KDD es un proceso iterativo, se considera como paso futuro iterar este proceso sobre las demás variables disponibles en el Data Warehouse para hasta encontrar las variables que combinadas unas con otras, arrojen los mejores resultados al ser empleadas en los distintos algoritmos de Machine Learning.

- Analizar los resultados obtenidos en los algoritmos de clasificación de abandono de productos con algoritmos más complejos como ser las redes neuronales y determinar si tomando en cuenta el mayor uso de recursos de estos algoritmos es conveniente su implementación sobre los algoritmos de clasificación.

- Afinar las variables utilizadas para describir el comportamiento de los clientes en base a la metodóloga RFM, y clasificación de las variables en categorías más específicas como ser el canal por el cual se realizan las transacciones o el tipo de servicio que genera la transacción, de esta forma se aumentara la precisión de los modelos entrenados.

- Analizar si es factible generalizar los resultados obtenidos del presente análisis a demás entidades bancarias dentro del territorio hondureño y desarrollar una metodología de que permita seleccionar de forma rápida las variables más significativas a nivel demográfico y de comportamiento de clientes.

- Crear una matriz de puntuación de los productos estudiados en base a las variables de comportamiento del cliente como ser transacciones efectuadas con dicho producto, saldo disponible, débitos, créditos. De esta matriz, y aplicando el agrupamiento demográfico especifico se puede desarrollar un sistema de recomendación para los productos financieros de la institución bancaria.

# 6. Bibliografía


- Purdy, M., & Daugherty, P. (2016). Why artificial intelligence is the future of growth, Accenture.
- Parasuraman, A., Zeithaml, V. A., & Berry, L. L. (1985). A conceptual model of service quality and its implications for future research. *Journal of marketing*, *49*(4), 41-50.
- Czaplewski, A. J.; Olson, E. M.; Slater, S. F. (2002). Applying the RATER model for service success. Marketing Management, 11(1), 14.
- Wei, J. T., Lin, S. Y., & Wu, H. H. (2010). A review of the application of RFM model. African Journal of Business Management, 4(19), 4199-4206. Recuperado de https://academicjournals.org/journal/AJBM/article-abstract/EB3418D18198
- Samuel, Arthur L. (1959). "Some Studies in Machine Learning Using the Game of Checkers". IBM Journal of Research and Development. 44: 206–226.
- Richard Sutton (1990). "Samuel's Checkers Player". Reinforcement Learning: An Introduction. Prensa del MIT. http://incompleteideas.net/book/11/node3.html consultado por última vez el 20 de enero del 2022
- Hsu Feng-hsiung; Hoane A. Joseph. Campbell Murray (1995). "Deep Thought (Chess)". CGA Tournaments, https://www.game-ai-forum.org/icga-tournaments/program.php?id=349 consultado por última vez el 20 de enero del 2022.
- *Campbell, Murray (1998). "An Enjoyable Game". In Stork, D. G. (ed.). HAL's Legacy: 2001's Computer as Dream and Reality. Cambridge, Mass: MIT Press.*
- Bill Wall (2008). «Who is the Strongest Chess Player?». Chess.com. https://www.chess.com/article/view/who-is-the-strongest-chess-player consultado por última vez el 20 de enero del 2022.
- Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., ... & Hassabis, D. (2017). Mastering chess and shogi by self-play with a general reinforcement learning algorithm. *arXiv preprint arXiv:1712.01815*.
- Breiman, Leo; Friedman, J. H.; Olshen, R. A.; Stone, C. J. (1984). *Classification and regression trees*. Monterey, CA: Wadsworth & Brooks/Cole Advanced Books & Software.
- Kassambara, Alboukadel. (2017) Machine Learning Essentials: Practical Guide in R. STHDA, Primera Edición.
- Kearns, Michael; Valiant, Leslie (1989); “Crytographic limitations on learning Boolean formulae and finite automata”. *Symposium on Theory of computing (ACM) 21: 433-444*. Recuperado de https://dl.acm.org/doi/10.1145/73007.73049

- Breiman, L. (2001). Random forests. *Machine learning*, *45*(1), 5-32.
- Kononenko, I., & Kukar, M. (2007). *Machine learning and data mining: Introduction to principles and algorithms*. Chichester, UK: Horwood Publishing.
- Pascual, D., Pla, F., & Sánchez, S. (2007). Algoritmos de agrupamiento. *Método Informáticos Avanzados*, 164-174.
- Tsiptsis, K. (2009). AntoniosChorianopoulos,". Data Mining Techniques In CRM.
- Ríos Carrillo, D. A. (2022). *Implementación de un método de agrupación de señales sísmicas generadas por el volcán Cotopaxi basado en aprendizaje automático no supervisado utilizando el modelo de mezcla gaussiana* (Bachelor's thesis, Quito, 2022).
- Donepudi, P (2017). AI and Machine Learning in Banking: A Systematic Literature Review. Asian Journal of Applied Science and Engineering, 6(3), 157-162. Recuperado de https://www.abc.us.org/ojs/index.php/ei/article/view/490/973
- Patidar, R., & Sharma, L. (2011). Credit card fraud detection using neural network. *International Journal of Soft Computing and Engineering (IJSCE)*, *1*(32-38). Recuperado de http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.301.8231&rep=rep1&type=pdf
- Aziz, S., & Dowling, M. (2019). Machine learning and AI for risk management. In *Disrupting finance* (pp. 33-50). Palgrave Pivot, Cham. Recuperado de https://library.oapen.org/bitstream/handle/20.500.12657/23126/1007030.pdf?sequence=1#page=54
- Kaya, O., Schildbach, J., AG, D. B., & Schneider, S. (2019). Artificial intelligence in banking. *Artificial intelligence*.
- Reglamento (UE) 2016/679 del Parlamento Europeo y del Consejo de 27 de abril de 2016 relativo a la protección de las personas físicas en lo que respecta al tratamiento de datos personales y a la libre circulación de estos datos y por el que se deroga la Directiva 95/46/CE (Reglamento General de Protección de Datos). Diario Oficial de la Unión Europea L 119, 4 de mayo de 2016, pp. 1-88. Recuperado de https://www.boe.es/doue/2016/119/L00001-00088.pdf

- Bach, M. P., Juković, S., Dumičić, K., & Šarlija, N. (2013). Business client segmentation in banking using self-organizing maps. *South East European Journal of Economics and Business*, *8*(2), 32-41. Recuperado de https://sciendo.com/pdf/10.2478/jeb-2013-0007

- Zakrzewska, D., & Murlewski, J. (2005, September). Clustering algorithms for bank customer segmentation. In *5th International Conference on Intelligent Systems Design and Applications (ISDA'05)* (pp. 197-202). IEEE.

- Popović, D., & Bašić, B. D. (2009). Churn prediction model in retail banking using fuzzy C-means algorithm. *Informatica*, *33*(2).

- Bilal Zorić, A. (2016). Predicting customer churn in banking industry using neural networks. *Interdisciplinary Description of Complex Systems: INDECS*, *14*(2), 116-124.

- Stehman, Stephen V. (1997). "Selecting and interpreting measures of thematic classification accuracy". Remote Sensing of Environment. 62 (1): 77–89.

- *Egan, J.P., (1975). "Signal detection theory and ROC analysis, Series in Cognition and Perception". Academic Press, New York*

- *Swets, J.A., Dawes, R.M., Monahan, J., (2000).* "Better decisions through science"*. Scientific American 283, 82–87.*

- *Spackman, K.A., (1989).* "Signal detection theory: Valuable tools for evaluating inductive learning. In: Proc. Sixth Internat". Workshop on Machine Learning. Morgan Kaufman, San Mateo, CA, pp. 160–163.

- *Provost, F., Fawcett, T., (1998).* "Robust classification systems for imprecise environments"*. AAAI Press, Menlo Park, CA, pp. 706–713.*

- Me, Burgos & Manterola, Carlos. (2010). Cómo interpretar un artículo sobre pruebas diagnósticas. Revista Chilena de Cirugia. 62. 301-308. 10.4067/S0718-40262010000300018.

- Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, *16*, 321-357. Recuperado de https://www.jair.org/index.php/jair/article/view/10302

- Martínez, G. (2001). Minería de datos. *Cómo hallar una aguja en un pajar. Ingenierías*, *14*(53), 53-66. Recuperado de https://www.cs.buap.mx/~bbeltran/NotasMD.pdf

- Riquelme Santos, J. C., Ruiz, R., & Gilbert, K. (2006). Minería de datos: Conceptos y tendencias. *Inteligencia Artificial: Revista Iberoamericana de Inteligencia Artificial, 10 (29), 11-18.* Recuperado de https://idus.us.es/handle/11441/43290

- Garcia-Alvarez, D., & Fuente, M. J. (2011). Estudio comparativo de técnicas de detección de fallos basadas en el Análisis de Componentes Principales (PCA). *Revista Iberoamericana de Automática e Informática Industrial RIAI*, *8*(3), 182-195. Recuperado de https://www.sciencedirect.com/science/article/pii/S1697791211000070

- Ogbuabor, G., & Ugwoke, F. N. (2018). Clustering algorithm for a healthcare dataset using silhouette score value. *AIRCC's International Journal of Computer Science and Information Technology*, *10*(2), 27-37. Recuperado de http://ischolar.info/index.php/IJCSIT/article/view/172096

# Anexos

## Anexo I. Consultas SQL para Generación de Datasets

Las consultas mostradas en este apartado han sido seudonimizadas para conservar la Seguridad de la Fuente de datos sobre la cual fueron construidas

### I.I Consulta SQL para Generación de Dataset Segmentación

```
SELECT DISTINCT A.codigo_de_cliente,A.campo_genero,A.campo_estado_civil,
                A.campo_nivel_educativo,
                A.campo_ocupacion,A.campo_generacion,A.campo_region,
                A.creditos_de_cliente,A.debitos_de_cliente,
                A.transacciones_promedio_cliente,( CASE
                                WHEN LEFT(B1.codigo_del_producto, 2) LIKE
                                     'CODIGO PRODUCTO' THEN
                                1
                                ELSE 0
                                END ) AS CUENTA_CHEQUES,
                ( CASE
                    WHEN
                                LEFT(B2.codigo_del_producto, 2) LIKE
                                'CODIGO PRODUCTO' THEN 1
                                                              ELSE 0
                                                            END ) AS
                CUENTA_AHORROS
                ,(
                                CASE
                                  WHEN B3.codigo_del_producto LIKE
                                       'CODIGO PRODUCTO' THEN 1
                                  ELSE 0
                                END ) AS DEPOSITOS_A_PLAZO,
                (
                CASE
                  WHEN
                                B4.codigo_del_producto LIKE 'CODIGO PRODUCT
O'
                THEN
                  1
                                                                  ELSE 0
```

```
                                                    END ) AS
BONOS_DE_CAJA
,(
                CASE
                  WHEN C1.codigo_del_producto LIKE
                       'CODIGO PRODUCTO' THEN 1
                  ELSE 0
                END ) AS PRESTAMOS_CONSUMO,
(
CASE
  WHEN
                C2.codigo_del_producto LIKE 'CODIGO PRODUCT
O'
THEN
  1
                                                      ELSE 0
                                                    END ) AS
PRESTAMOS_VIVIENDA
,( CASE
                WHEN C3.codigo_del_producto LIKE
                     'CODIGO PRODUCTO' THEN 1
                ELSE 0
   END )
                AS ADELANTO_PAGO_PLUS,
( CASE
WHEN LEFT(C4.codigo_del_producto, 2)
     LIKE 'CODIGO PRODUCTO' THEN 1
ELSE 0
                                          END ) AS
EXTRAFINANCIAMIENTO
,(
                CASE
                  WHEN D1.codigo_del_producto IN (
                       'CODIGOS DEL PRODUCTO' ) THEN 1
                  ELSE 0
                END ) AS TC_VISA,( CASE
                                     WHEN
D2.codigo_del_producto IN (
'CODIGOS DEL PRODUCTO' ) THEN 1
                                     ELSE 0
                                   END ) AS TC_CASHBACK,(
```

```
                                CASE
                                  WHEN D3.codigo_del_producto IN (
                                       'CODIGOS DEL PRODUCTO' ) THEN 1
                                  ELSE 0
                                END ) AS TC_OLIMPIA,( CASE
                                                        WHEN
                                D4.codigo_del_producto IN (
                                'CODIGOS DEL PRODUCTO' ) THEN 1
                                                        ELSE 0
                                                      END ) AS TC_HMC,(
                                CASE
                                  WHEN D5.codigo_del_producto IN (
                                       'CODIGOS DEL PRODUCTO' ) THEN 1
                                  ELSE 0
                                END ) AS TC_LADY_LEE,( CASE
                                                         WHEN
                                D6.codigo_del_producto IN (
                                'CODIGOS DEL PRODUCTO' ) THEN 1
                                                         ELSE 0
                                                       END ) AS TC_PUMA,(
                                CASE
                                  WHEN D7.codigo_del_producto IN (
                                       'CODIGOS DEL PRODUCTO' ) THEN 1
                                  ELSE 0
                                END ) AS TC_ANTORCHA,( CASE
                                                         WHEN
                                D8.codigo_del_producto IN (
                                'CODIGOS DEL PRODUCTO' ) THEN 1
                                                         ELSE 0
                                                       END ) AS TC_CELEBRA
FROM    (SELECT A.codigo_de_cliente,A.campo_genero,(
                               CASE
                                                      WHEN
                               A.campo_estado_civil IN
                               (
                               'CASADO/A', 'MARRIED', 'VERHEIRATET' ) THEN
                                                      'CASADO/A'
                                                      WHEN
                               A.campo_estado_civil IN (
                               'DIVORCED', 'DIVORCIADO/A', 'GESCHIEDEN' )
                                    THEN
```

```
                                        'DIVORCIADO/A'
                                        WHEN
                                A.campo_estado_civil IN (
                                'GETRENNT LEBEND', 'SEPARATED'
                                        ) THEN
                                        'SEPARADO/A'
                                        WHEN
                                A.campo_estado_civil IN
                                (
                                'LEDIG', 'SINGLE', 'SOLTERO/A' )
                                THEN
                                        'SOLTERO/A'
                                        WHEN
                                A.campo_estado_civil IN (
                                'VERWITWET', 'VIUDO', 'WIDOWED'
                                        ) THEN
                                        'VIUDO/A'
                                WHEN
                                A.campo_estado_civil LIKE 'UNIÓN LIBRE' THEN
                                'UNIÓN LIBRE'
                                        ELSE ''
                                    END ) AS CAMPO_ESTADO_C
IVIL,
                                ( CASE
                                    WHEN A.campo_nivel_educativo IN (
                                         'DOCTORADO', 'DOKTOR', 'DR' ) THEN
                                    'DOCTORADO'
                                    ELSE A.campo_nivel_educativo
                                  END ) AS CAMPO_NIVEL_EDUCATIVO,
                A.campo_ocupacion,
                        B.edad_segmentacion AS CAMPO_GENERACION,B.campo_regio
n,
                        Round(Ifnull(T.credito_ultimo_anio, 0), 2) AS
                        CREDITOS_DE_CLIENTE
                        ,
                        Round(Ifnull(T.debito_ultimo_anio, 0), 2) AS
                        DEBITOS_DE_CLIENTE,
                                Round(Ifnull(T.num_transacciones_promedio_cli
ente,
                                      0), 2)
                                AS TRANSACCIONES_PROMEDIO_CLIENTE
```

```
        FROM    esquema.tabla_crm_principal AS A
                INNER JOIN esquema.tabla_crm_secundario AS B
                        ON A.codigo_de_cliente = B.codigo_de_cliente
                LEFT JOIN (SELECT
                        To_char(codigo_de_cliente) AS CODIGO_DE_CLIENTE,
                        To_double(Sum(
                        CASE
                                  WHEN
                                 tipo_transaccion = 'CREDITO' THEN
                                  monto_transaccion
                        ELSE
                        0
                        END
                                 ) / 12
                        )
                                 CREDITO_ULTIMO_ANIO,To_double(Abs(
                                                      Sum(CASE
                                                            WHEN tipo_transa
ccion
                                                                 = 'DEBITO'
                                                          THEN
                                                            monto_transaccio
n
                                                            ELSE 0
                                                          END)) / 12)
                                                     DEBITO_ULTIMO_ANIO,
                                        To_double(Sum(CASE
                                        WHEN
                                 codigo_categoria_transaccion IN (
                                 'CODIGOS PARA DEBITOS' )
                                 AND tipo_transaccion = 'DEBITO' THEN 1
                                                       ELSE 0
                                                                    END
                                                  ) / 12)
                                        AS NUM_TRANSACCIONES_PROMEDIO_CLIE
NTE
                           FROM    esquema.tabla_transaccional
                           WHERE   ( periodo_de_analisis BETWEEN
                                     To_int(To_char(
                                     Add_months(CURRENT_DATE,
                                     -12),
```

```
                                    'YYYYMMDD')) AND
                                             To_int(To_char
                                             (
                                             Add_days(
                                                    CURRENT_DATE,
                                             -1
                                                           ),
                                                    'YYYYMMDD')) )
                                    AND LEFT(campo_numero_cuenta_contable, 3)
IN (
                                        'CODIGOS_PARA_CUENTAS_DE_DEPOSITO' )
                                    AND numero_cuenta_detalle != 0
                             GROUP  BY codigo_de_cliente) AS T
                         ON A.codigo_de_cliente = T.codigo_de_cliente
          WHERE  A.campo_estado_cliente LIKE 'ACTIVO'
                 AND A.campo_banca_de_cliente LIKE 'SEGMENTO PERSONAS'
                 AND To_int(A.campo_fecha_creacion_del_cliente) BETWEEN
                     To_int('20190901') AND To_int(
                         To_char(Add_months(To_date('20190901', 'YYYYMMDD')
                                 , 18), 'YYYYMMDD'))
                 AND A.campo_genero NOT LIKE 'N/A'
                 AND A.campo_estado_civil NOT LIKE ''
                 AND A.campo_nivel_educativo NOT LIKE ''
                 AND A.campo_ocupacion NOT LIKE 'N/A'
                 AND B.edad_segmentacion NOT LIKE 'O'
                 AND B.campo_region NOT LIKE 'SIN DEFINIR') AS A
        LEFT JOIN (SELECT DISTINCT codigo_de_cliente,codigo_del_producto
                   FROM   esquema.tabla_productos_de_deposito
                   WHERE  LEFT(codigo_del_producto, 2) LIKE 'CODIGO PRODUCTO
'
                          AND periodo_de_analisis = Add_days(CURRENT_DATE, -
1)
                          AND campo_estado_del_producto NOT LIKE 'CERRADO')
AS B1
                ON A.codigo_de_cliente = B1.codigo_de_cliente
        LEFT JOIN (SELECT DISTINCT codigo_de_cliente,codigo_del_producto
                   FROM   esquema.tabla_productos_de_deposito
                   WHERE  LEFT(codigo_del_producto, 2) LIKE 'CODIGO PRODUCTO
'
                          AND periodo_de_analisis = Add_days(CURRENT_DATE, -
1)
```

```
                        AND campo_estado_del_producto NOT LIKE 'CERRADO')
AS B2
             ON A.codigo_de_cliente = B2.codigo_de_cliente
      LEFT JOIN (SELECT DISTINCT codigo_de_cliente,codigo_del_producto
                 FROM   esquema.tabla_productos_de_deposito
                 WHERE  codigo_del_producto LIKE 'CODIGO PRODUCTO'
                        AND periodo_de_analisis = Add_days(CURRENT_DATE, -
1)
                        AND campo_estado_del_producto NOT LIKE 'CERRADO')
AS B3
             ON A.codigo_de_cliente = B3.codigo_de_cliente
      LEFT JOIN (SELECT DISTINCT codigo_de_cliente,codigo_del_producto
                 FROM   esquema.tabla_productos_de_deposito
                 WHERE  codigo_del_producto LIKE 'CODIGO PRODUCTO'
                        AND periodo_de_analisis = Add_days(CURRENT_DATE, -
1)
                        AND campo_estado_del_producto NOT LIKE 'CERRADO')
AS B4
             ON A.codigo_de_cliente = B4.codigo_de_cliente
      LEFT JOIN (SELECT DISTINCT codigo_de_cliente,codigo_del_producto
                 FROM   esquema.tabla_productos_de_financiamiento
                 WHERE  codigo_del_producto LIKE 'CODIGO PRODUCTO'
                        AND periodo_de_analisis = Add_days(CURRENT_DATE, -
1)
                        AND ( campo_estado_del_producto NOT LIKE 'CERRADO'
                              OR categoria_del_producto NOT LIKE
                                 'PRODUCTO PERDIDO' )
                                    ) AS C1
             ON A.codigo_de_cliente = C1.codigo_de_cliente
      LEFT JOIN (SELECT DISTINCT codigo_de_cliente,codigo_del_producto
                 FROM   esquema.tabla_productos_de_financiamiento
                 WHERE  codigo_del_producto LIKE 'CODIGO PRODUCTO'
                        AND periodo_de_analisis = Add_days(CURRENT_DATE, -
1)
                        AND ( campo_estado_del_producto NOT LIKE 'CERRADO'
                              OR categoria_del_producto NOT LIKE
                                 'PRODUCTO PERDIDO' )
                                    ) AS C2
             ON A.codigo_de_cliente = C2.codigo_de_cliente
      LEFT JOIN (SELECT DISTINCT codigo_de_cliente,codigo_del_producto
                 FROM   esquema.tabla_productos_de_financiamiento
```

```
                WHERE  codigo_del_producto LIKE 'CODIGO PRODUCTO'
                       AND periodo_de_analisis = Add_days(CURRENT_DATE, -
1)
                       AND ( campo_estado_del_producto NOT LIKE 'CERRADO'
                              OR categoria_del_producto NOT LIKE
                                 'PRODUCTO PERDIDO' )
                                    ) AS C3
            ON A.codigo_de_cliente = C3.codigo_de_cliente
      LEFT JOIN (SELECT DISTINCT codigo_de_cliente,codigo_del_producto
                 FROM   esquema.tabla_productos_de_financiamiento
                 WHERE  LEFT(codigo_del_producto, 2) LIKE 'CODIGO PRODUCTO
'
                       AND periodo_de_analisis = Add_days(CURRENT_DATE, -
1)
                       AND ( campo_estado_del_producto NOT LIKE 'CERRADO'
                              OR categoria_del_producto NOT LIKE
                                 'PRODUCTO PERDIDO' )
                                    ) AS C4
            ON A.codigo_de_cliente = C4.codigo_de_cliente
      LEFT JOIN (SELECT DISTINCT To_char(codigo_de_cliente) AS
                                   CODIGO_DE_CLIENTE,
                 codigo_producto_tarejta AS CODIGO_DEL_PRODUCTO
                 FROM   esquema.tabla_productos_tarjetas_de_credito
                 WHERE  codigo_producto_tarejta IN ( 'CODIGOS DEL PRODUCTO'
 )
                 AND periodo_de_analisis = Add_days(CURRENT_DATE, -1)
                 AND campo_estado_de_la_tarjeta IN (
                     'CODIGOS PARA TARJETAS ACTIVAS' ))
                                    AS D1
            ON A.codigo_de_cliente = D1.codigo_de_cliente
      LEFT JOIN (SELECT DISTINCT To_char(codigo_de_cliente) AS
                                   CODIGO_DE_CLIENTE,
                 codigo_producto_tarejta AS CODIGO_DEL_PRODUCTO
                 FROM   esquema.tabla_productos_tarjetas_de_credito
                 WHERE  codigo_producto_tarejta IN ( 'CODIGOS DEL PRODUCTO'
 )
                 AND periodo_de_analisis = Add_days(CURRENT_DATE, -1)
                 AND campo_estado_de_la_tarjeta IN (
                     'CODIGOS PARA TARJETAS ACTIVAS' ))
                                    AS D2
            ON A.codigo_de_cliente = D2.codigo_de_cliente
```

```
        LEFT JOIN (SELECT DISTINCT To_char(codigo_de_cliente) AS
                                   CODIGO_DE_CLIENTE,
                   codigo_producto_tarejta AS CODIGO_DEL_PRODUCTO
                   FROM   esquema.tabla_productos_tarjetas_de_credito
                   WHERE  codigo_producto_tarejta IN ( 'CODIGOS DEL PRODUCTO'
)
                   AND periodo_de_analisis = Add_days(CURRENT_DATE, -1)
                   AND campo_estado_de_la_tarjeta IN (
                       'CODIGOS PARA TARJETAS ACTIVAS' ))
                                   AS D3
               ON A.codigo_de_cliente = D3.codigo_de_cliente
        LEFT JOIN (SELECT DISTINCT To_char(codigo_de_cliente) AS
                                   CODIGO_DE_CLIENTE,
                   codigo_producto_tarejta AS CODIGO_DEL_PRODUCTO
                   FROM   esquema.tabla_productos_tarjetas_de_credito
                   WHERE  codigo_producto_tarejta IN ( 'CODIGOS DEL PRODUCTO'
)
                   AND periodo_de_analisis = Add_days(CURRENT_DATE, -1)
                   AND campo_estado_de_la_tarjeta IN (
                       'CODIGOS PARA TARJETAS ACTIVAS' ))
                                   AS D4
               ON A.codigo_de_cliente = D4.codigo_de_cliente
        LEFT JOIN (SELECT DISTINCT To_char(codigo_de_cliente) AS
                                   CODIGO_DE_CLIENTE,
                   codigo_producto_tarejta AS CODIGO_DEL_PRODUCTO
                   FROM   esquema.tabla_productos_tarjetas_de_credito
                   WHERE  codigo_producto_tarejta IN ( 'CODIGOS DEL PRODUCTO'
)
                   AND periodo_de_analisis = Add_days(CURRENT_DATE, -1)
                   AND campo_estado_de_la_tarjeta IN (
                       'CODIGOS PARA TARJETAS ACTIVAS' ))
                                   AS D5
               ON A.codigo_de_cliente = D5.codigo_de_cliente
        LEFT JOIN (SELECT DISTINCT To_char(codigo_de_cliente) AS
                                   CODIGO_DE_CLIENTE,
                   codigo_producto_tarejta AS CODIGO_DEL_PRODUCTO
                   FROM   esquema.tabla_productos_tarjetas_de_credito
                   WHERE  codigo_producto_tarejta IN ( 'CODIGOS DEL PRODUCTO'
)
                   AND periodo_de_analisis = Add_days(CURRENT_DATE, -1)
                   AND campo_estado_de_la_tarjeta IN (
```

```
                    'CODIGOS PARA TARJETAS ACTIVAS' ))
                                        AS D6
              ON A.codigo_de_cliente = D6.codigo_de_cliente
       LEFT JOIN (SELECT DISTINCT To_char(codigo_de_cliente) AS
                                        CODIGO_DE_CLIENTE,
                  codigo_producto_tarejta AS CODIGO_DEL_PRODUCTO
                  FROM   esquema.tabla_productos_tarjetas_de_credito
                  WHERE  codigo_producto_tarejta IN ( 'CODIGOS DEL PRODUCTO'
)
                  AND periodo_de_analisis = Add_days(CURRENT_DATE, -1)
                  AND campo_estado_de_la_tarjeta IN (
                      'CODIGOS PARA TARJETAS ACTIVAS' ))
                                        AS D7
              ON A.codigo_de_cliente = D7.codigo_de_cliente
       LEFT JOIN (SELECT DISTINCT To_char(codigo_de_cliente) AS
                                        CODIGO_DE_CLIENTE,
                  codigo_producto_tarejta AS CODIGO_DEL_PRODUCTO
                  FROM   esquema.tabla_productos_tarjetas_de_credito
                  WHERE  codigo_producto_tarejta IN ( 'CODIGOS DEL PRODUCTO'
)
                  AND periodo_de_analisis = Add_days(CURRENT_DATE, -1)
                  AND campo_estado_de_la_tarjeta IN (
                      'CODIGOS PARA TARJETAS ACTIVAS' ))
                                        AS D8
              ON A.codigo_de_cliente = D8.codigo_de_cliente
```

## I.II Consulta SQL para Generación de Dataset Abandono Cuenta de Cheques

```
SELECT A.codigo_de_cliente,B.campo_genero,B.campo_estado_civil,
       B.campo_nivel_educativo,
       B.campo_ocupacion,B.campo_generacion,B.campo_region,
       B.creditos_de_cliente,B.debitos_de_cliente,
       B.transacciones_promedio_cliente,
       A.estado_producto
FROM   (SELECT A.codigo_de_cliente,To_char(A.numero_de_cuenta_del_producto)
 AS
                                    CUENTA,(
                     CASE
                     WHEN
                             B.campo_estado_del_producto IS NULL THEN
```

```
                    'Abandono'
                    WHEN
                            B.campo_estado_del_producto NOT LIKE 'ACTIVO'
THEN
                    'Abandono'
                             ELSE
                            'Vigente'
                    END )
                            AS ESTADO_PRODUCTO
        FROM    (SELECT *
                 FROM   esquema.tabla_productos_de_deposito
                 WHERE  periodo_de_analisis BETWEEN
                        To_date('20200301', 'YYYYMMDD') AND
                        Add_months(To_date('20200301',
                                   'YYYYMMDD'),
                        12)
                        AND fecha_adquisicion_producto = periodo_de_analisis
                        AND LEFT(codigo_del_producto, 2) LIKE 'CODIGO PRODUC
TO')
               AS A
               LEFT JOIN (SELECT *
                          FROM   esquema.tabla_productos_de_deposito
                          WHERE  periodo_de_analisis BETWEEN
                                 Add_months(To_date('20200301',
                                            'YYYYMMDD'),
                                 6) AND
                                         Add_months(Add_months(
                                                    To_date
                                                    ('20200301',
                                                    'YYYYMMDD'),
                                                    12), 6)) AS B
                       ON A.numero_de_cuenta_del_producto =
                          B.numero_de_cuenta_del_producto
                          AND A.fecha_adquisicion_producto =
                              Add_months(B.periodo_de_analisis,
                              -6)) AS A
       INNER JOIN (SELECT
                   A.codigo_de_cliente,A.campo_genero,(
                   CASE
                                 WHEN
                   A.campo_estado_civil IN
```

```
                    (
                    'CASADO/A', 'MARRIED',
                    'VERHEIRATET' )
                           THEN
                                    'CASADO/A'
                                    WHEN
                    A.campo_estado_civil IN (
                    'DIVORCED', 'DIVORCIADO/A',
                    'GESCHIEDEN' )
                                     THEN
                                    'DIVORCIADO/A'
                                    WHEN
                    A.campo_estado_civil IN (
                    'GETRENNT LEBEND', 'SEPARATED' )
                           THEN
                                    'SEPARADO/A'
                                    WHEN
                    A.campo_estado_civil IN
                    ( 'LEDIG', 'SINGLE',
                      'SOLTERO/A' )
                           THEN
                                    'SOLTERO/A'
                                    WHEN
                    A.campo_estado_civil IN (
                    'VERWITWET', 'VIUDO',
                    'WIDOWED' )
                           THEN
                                    'VIUDO/A'
                                    WHEN
                    A.campo_estado_civil LIKE
                    'UNIÓN LIBRE' THEN
                                    'UNIÓN LIBRE'
                                    ELSE ''
                                END ) AS
                    CAMPO_ESTADO_CIVIL,
                                         ( CASE
                                             WHEN A.campo_nivel_educativo IN
(
                                                  'DOCTORADO', 'DOKTOR'
                                                  , 'DR' ) THEN
                                             'DOCTORADO'
```

```
                                        ELSE A.campo_nivel_educativo
                                    END ) AS CAMPO_NIVEL_EDUCATIVO,
                                             A.campo_ocupacion,
                                             B.edad_segmentacion AS
                                             CAMPO_GENERACION
                                                                  ,
                        B.campo_region,
                        Round(Ifnull(T.credito_ultimo_anio, 0), 2) AS
                        CREDITOS_DE_CLIENTE,
Round(Ifnull(T.debito_ultimo_anio, 0)
, 2)
AS DEBITOS_DE_CLIENTE
,
Round(Ifnull(T.num_transacciones_promedio_cliente, 0), 2) AS
TRANSACCIONES_PROMEDIO_CLIENTE
FROM   esquema.tabla_crm_principal AS A
INNER JOIN esquema.tabla_crm_secundario AS B
ON A.codigo_de_cliente = B.codigo_de_cliente
LEFT JOIN (SELECT
To_char(codigo_de_cliente) AS CODIGO_DE_CLIENTE,
To_double(Sum(
CASE
WHEN
tipo_transaccion = 'CREDITO' THEN
monto_transaccion
ELSE
0
END
) / 12
)
CREDITO_ULTIMO_ANIO,To_double(Abs(
Sum(CASE
WHEN tipo_transaccion
= 'DEBITO'
THEN
monto_transaccion
ELSE 0
END)) / 12)
DEBITO_ULTIMO_ANIO,
  To_double(Sum(CASE
  WHEN
```

```
codigo_categoria_transaccion IN (
'CODIGOS PARA DEBITOS' )
AND tipo_transaccion = 'DEBITO' THEN 1
                ELSE 0
                                END
            ) / 12)
  AS
NUM_TRANSACCIONES_PROMEDIO_CLIENTE
FROM   esquema.tabla_transaccional
WHERE  ( periodo_de_analisis BETWEEN
To_int(To_char(
Add_months(CURRENT_DATE,
-12),
'YYYYMMDD')) AND
      To_int(To_char
      (
      Add_days(
            CURRENT_DATE,
      -1
                  ),
            'YYYYMMDD')) )
AND LEFT(campo_numero_cuenta_contable, 3)
IN (
'CODIGOS_PARA_CUENTAS_DE_DEPOSITO' )
AND numero_cuenta_detalle != 0
GROUP  BY codigo_de_cliente) AS T
ON A.codigo_de_cliente = T.codigo_de_cliente
WHERE  A.campo_estado_cliente LIKE 'ACTIVO'
AND A.campo_banca_de_cliente LIKE 'SEGMENTO PERSONAS'
AND A.campo_genero NOT LIKE 'N/A'
AND A.campo_estado_civil NOT LIKE ''
AND A.campo_nivel_educativo NOT LIKE ''
AND A.campo_ocupacion NOT LIKE 'N/A'
AND B.edad_segmentacion NOT LIKE 'O'
AND B.campo_region NOT LIKE 'SIN DEFINIR') AS B
ON A.codigo_de_cliente = B.codigo_de_cliente
```

## I.III Consulta SQL para Generación de Dataset Abandono Tarjeta de Crédito Puma y Visa

```
SELECT A.codigo_de_cliente,B.campo_genero,B.campo_estado_civil,
       B.campo_nivel_educativo,
       B.campo_ocupacion,B.campo_generacion,B.campo_region,
       B.creditos_de_cliente,B.debitos_de_cliente,
       B.transacciones_promedio_cliente,A.estado_producto
FROM   (SELECT To_char(A.codigo_de_cliente) AS CODIGO_DE_CLIENTE,
                    A.numero_de_cuenta_del_producto,( CASE
                            WHEN B.campo_estado_de_la_tarjeta IS NULL THE
N
                            'Abandono'
                            WHEN B.campo_estado_de_la_tarjeta IN (
                                 'CODIGOS PARA TARJETAS ACTIVAS' )
                                      THEN
                            'Vigente'
                            ELSE 'Abandono'
                            END ) AS ESTADO_PRODUCTO
        FROM   (SELECT *
                FROM   esquema.tabla_productos_tarjetas_de_credito
                WHERE  periodo_de_analisis BETWEEN
                       To_date('20200301', 'YYYYMMDD') AND
                       Add_months(To_date('20200301',
                                  'YYYYMMDD'),
                       12)
                       AND To_date(fecha_adquisicion_producto, 'YYYYMMDD')
=
                           periodo_de_analisis
                       AND codigo_producto_tarejta IN ( 'CODIGOS DEL PRODUC
TO' )
               ) AS A
               LEFT JOIN (SELECT *
                          FROM   esquema.tabla_productos_tarjetas_de_credit
o
                          WHERE  periodo_de_analisis BETWEEN
                                 Add_months(To_date('20200301',
                                            'YYYYMMDD'),
                                 6) AND
                                        Add_months(Add_months(
```

```
                                                    To_date
                                                    ('20200301',
                                                    'YYYYMMDD'),
                                                    12), 6)) AS B
                        ON A.numero_de_cuenta_del_producto =
                           B.numero_de_cuenta_del_producto
                           AND To_date(A.fecha_adquisicion_producto, 'YYYYMMD
D') =
                               Add_months(B.periodo_de_analisis, -6)) AS A
       INNER JOIN (SELECT A.codigo_de_cliente,A.campo_genero,( CASE
                                                                WHEN
                A.campo_estado_civil IN
                (
                           'CASADO/A', 'MARRIED', 'VERHEIRATET' )
                            THEN
                                                                'CASADO/A'
                                                                WHEN
                A.campo_estado_civil IN (
                                    'DIVORCED', 'DIVORCIADO/A', 'GESCHIEDEN' )
                                                                    THEN
                                                                        'DIVORCIAD
O/A'
                                                                WHEN
                                    A.campo_estado_civil IN (
                                    'GETRENNT LEBEND', 'SEPARATED' ) THEN
                                          'SEPARADO/A'
                                          WHEN A.campo_estado_civil IN
                                               (
                           'LEDIG', 'SINGLE', 'SOLTERO/A' )
                            THEN
                                                                'SOLTERO/A
'
                                                                WHEN
                                    A.campo_estado_civil IN (
                                    'VERWITWET', 'VIUDO', 'WIDOWED' ) THEN
                                          'VIUDO/A'
                                          WHEN
                                    A.campo_estado_civil LIKE 'UNIÓN LIBRE' TH
EN
                                          'UNIÓN LIBRE'
                                                                ELSE ''
```

```
                                                                END ) AS
                CAMPO_ESTADO_CIVIL,
                                        ( CASE
                                            WHEN A.campo_nivel_educativo IN (
                                                 'DOCTORADO', 'DOKTOR', 'DR'
                                                                            ) THEN
                                            'DOCTORADO'
                                            ELSE A.campo_nivel_educativo
                                          END ) AS CAMPO_NIVEL_EDUCATIVO,
                                 A.campo_ocupacion,
                                B.edad_segmentacion AS CAMPO_GENERACION,B.campo_re
gion,
                                 Round(
                                Ifnull(T.credito_ultimo_anio, 0), 2) AS
                                 CREDITOS_DE_CLIENTE,
                                Round(Ifnull(T.debito_ultimo_anio, 0), 2) AS
                                DEBITOS_DE_CLIENTE,
                                Round(Ifnull(T.num_transacciones_promedio_cliente,
 0),
                                2) AS
                                TRANSACCIONES_PROMEDIO_CLIENTE
                         FROM   esquema.tabla_crm_principal AS A
                                INNER JOIN esquema.tabla_crm_secundario AS B
                                        ON A.codigo_de_cliente = B.codigo_de_clie
nte
                                 LEFT JOIN (SELECT
                                        To_char(codigo_de_cliente) AS
                                        CODIGO_DE_CLIENTE,
                                        To_double(Sum(
                                        CASE
                                                    WHEN
                                                   tipo_transaccion = 'CREDITO' TH
EN
                                                    monto_transaccion
                                        ELSE
                                        0
                                        END
                                                   ) / 12
                                        )
                                                   CREDITO_ULTIMO_ANIO,To_double(A
bs(
```

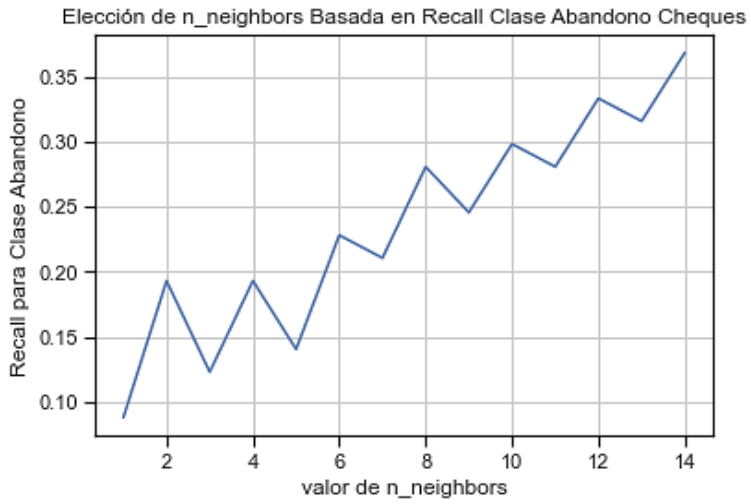
```
Sum(CASE
      WHEN tipo_transaccion
           = 'DEBITO'
    THEN
      monto_transaccion
      ELSE 0
    END)) / 12)
DEBITO_ULTIMO_ANIO,
                To_double(Sum(CASE
                WHEN
          codigo_categoria_transaccion IN
(
          'CODIGOS PARA DEBITOS' )
          AND tipo_transaccion = 'DEBITO'
THEN
                1
                              ELSE 0
                              END
                         ) / 12)
                AS
NUM_TRANSACCIONES_PROMEDIO_CLIENTE
   FROM   esquema.tabla_transaccional
   WHERE  ( periodo_de_analisis BETWEEN
            To_int(To_char(
            Add_months(CURRENT_DATE,
            -12),
            'YYYYMMDD')) AND
                     To_int(To_char
                     (
                     Add_days(
                            CURRENT_DATE
,
                     -1
                                  ),
                            'YYYYMMDD'))
)
          AND LEFT(
  campo_numero_cuenta_contable, 3)
              IN (
  'CODIGOS_PARA_CUENTAS_DE_DEPOSITO' )
          AND numero_cuenta_detalle != 0
```

```
                                    GROUP  BY codigo_de_cliente) AS T
                                ON A.codigo_de_cliente = T.codigo_de_clien
te
                    WHERE  A.campo_estado_cliente LIKE 'ACTIVO'
                           AND A.campo_banca_de_cliente LIKE 'SEGMENTO PERSO
NAS'
                           AND A.campo_genero NOT LIKE 'N/A'
                           AND A.campo_estado_civil NOT LIKE ''
                           AND A.campo_nivel_educativo NOT LIKE ''
                           AND A.campo_ocupacion NOT LIKE 'N/A'
                           AND B.edad_segmentacion NOT LIKE 'O'
                           AND B.campo_region NOT LIKE 'SIN DEFINIR') AS B
                ON A.codigo_de_cliente = B.codigo_de_cliente
```

# Anexo II. Selección de Parámetros Para Algoritmos de Clasificación

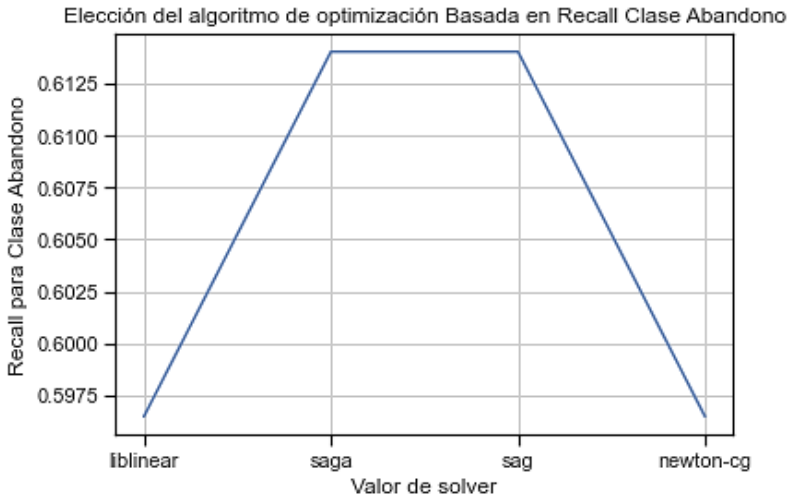
Para seleccionar los valores de n_neighbors en los clasificadores de KNeighbors y solver en los clasificadores de regresión logística se realizaron pruebas donde se calculó la sensibilidad que presentaba cada uno de los clasificadores con diferentes parámetros en cada uno de los casos donde se ejecutaron algoritmos que generan dichos modelos de clasificación, los resultados obtenidos son los siguientes:

### Predicción de abandono para Cuenta de Cheques




Valor seleccionado para n_neighbors: 14

*Figura 30. Elección de parámetro n_neighbors para algoritmo KNeighbors Cuentas de Cheque (Elaboración propia)*



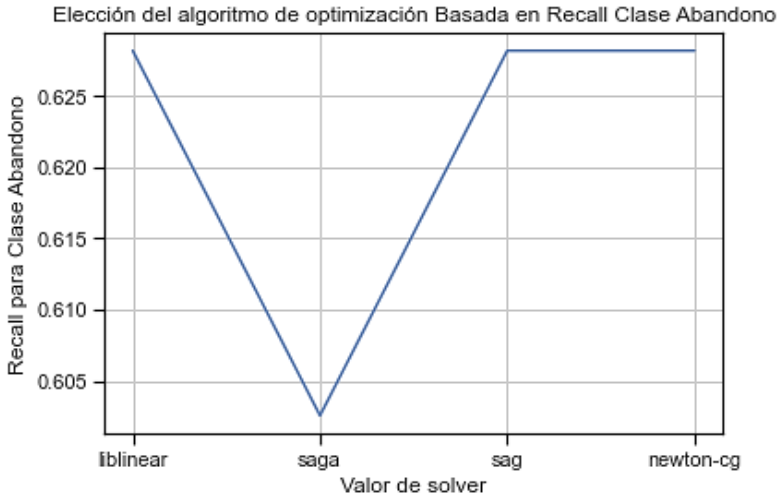

Valor seleccionado para solver: ‘sag’

*Figura 31. Elección de parámetro solver para algoritmo Logistic Regression Cuentas de Cheque (Elaboración propia)*

**Predicción de abandono para Tarjeta de Crédito Puma**


Valor seleccionado para n_neighbors: 4

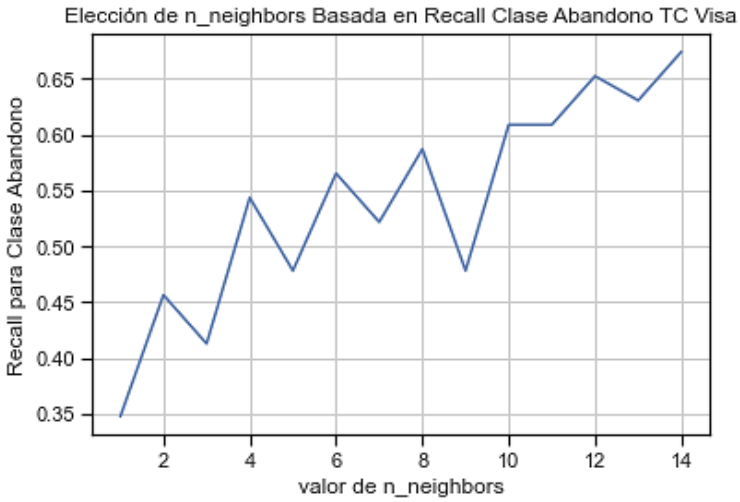
*Figura 32. Elección de parámetro n_neighbors para algoritmo KNeighbors TC Puma (Elaboración propia)*




Valor seleccionado para solver: 'liblinear'

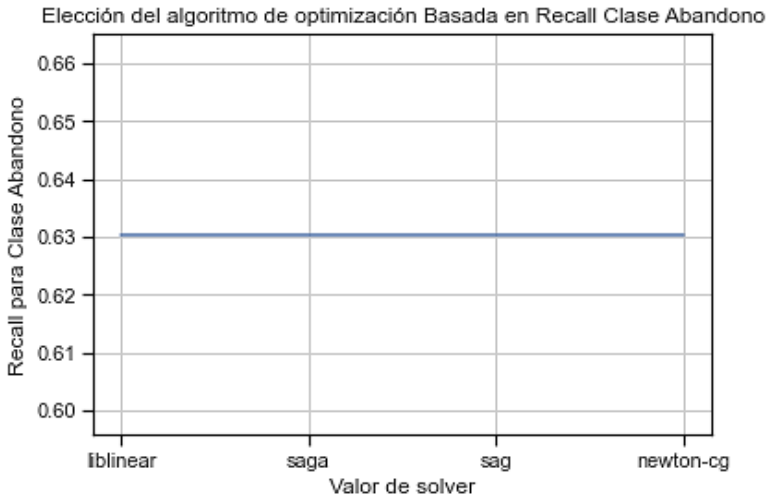
*Figura 33. Elección de parámetro solver para algoritmo Logistic Regression TC Puma (Elaboración propia)*

**Predicción de abandono para Tarjeta de Crédito Visa**




Valor seleccionado para n_neighbors: 14

*Figura 34. Elección de parámetro n_neighbors para algoritmo KNeighbors TC Visa (Elaboración propia)*




Valor seleccionado para solver: ‘saga’

*Figura 35. Elección de parámetro solver para algoritmo Logistic Regression TC Visa (Elaboración propia)*

# Anexo III. Matrices de Confusión Generadas para Cada uno de los Modelos Entrenados

A continuación, se muestran las matrices de confusión que se obtuvieron después de entrenar cada uno de los modelos predictivos de clasificación, en base a estas matrices se calcularon los valores de precisión, sensibilidad (recall), puntaje f1 (f1 score) y el valor final de AUC (área bajo la curva).

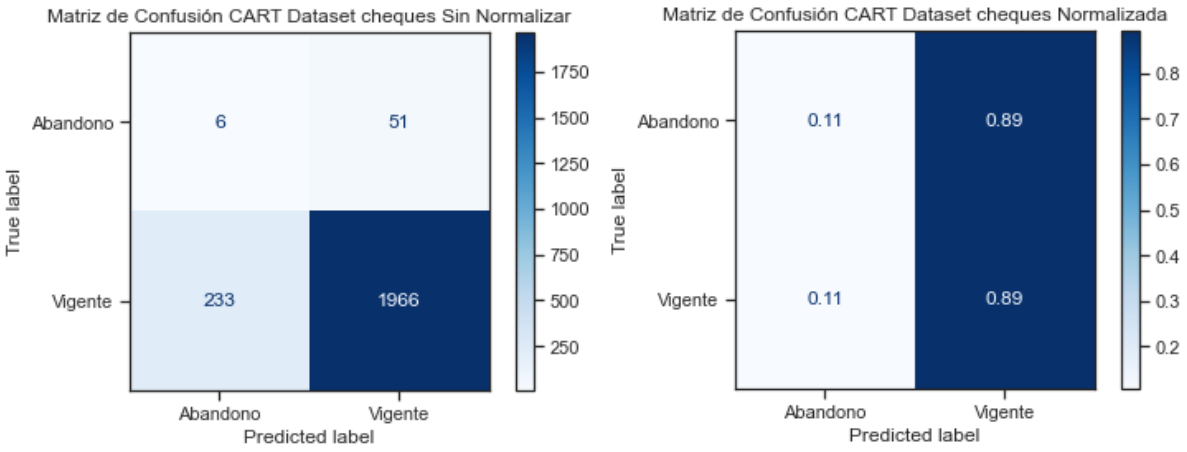



*Figura 36. Matriz de Confusión para algoritmo CART Abandono Cta. Cheques (Elaboración propia)*

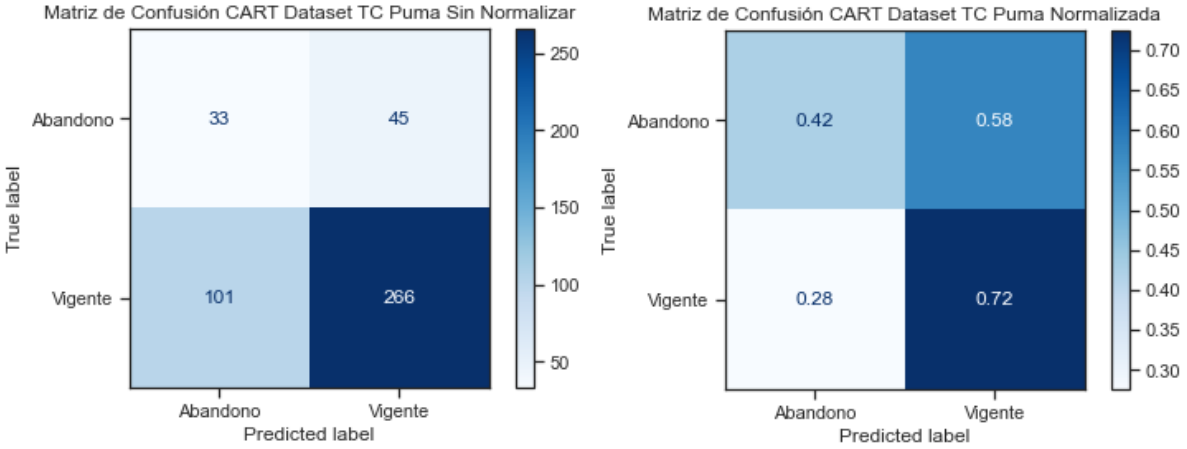



*Figura 37. Matriz de Confusión para algoritmo CART Abandono TC Puma (Elaboración propia)*

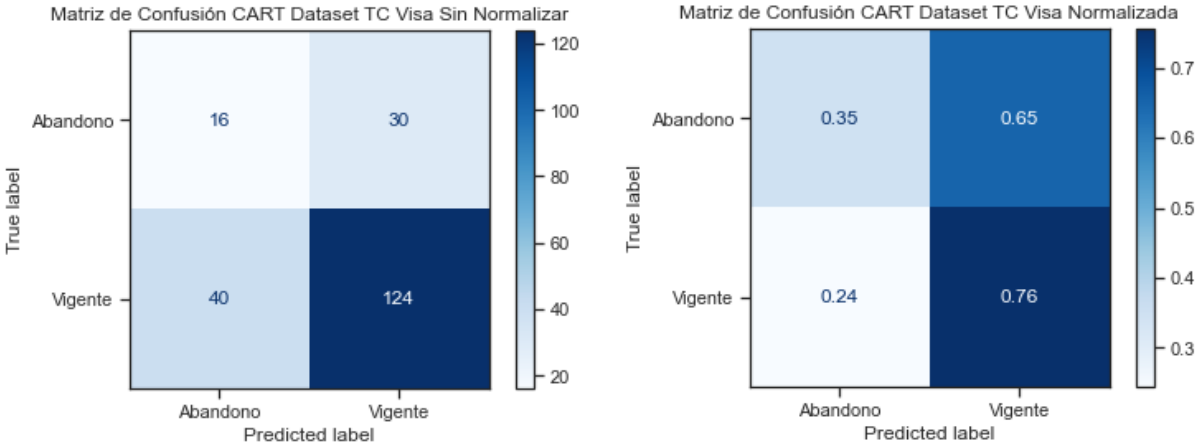



*Figura 38. Matriz de Confusión para algoritmo CART Abandono TC Visa (Elaboración propia)*

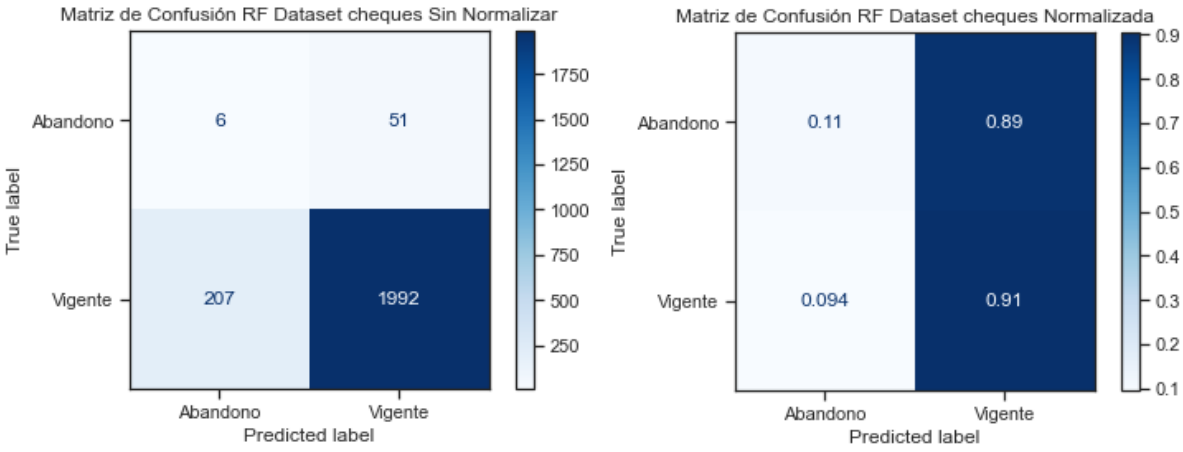



*Figura 39. Matriz de Confusión para algoritmo Random Forest Abandono Cta. Cheques (Elaboración propia)*

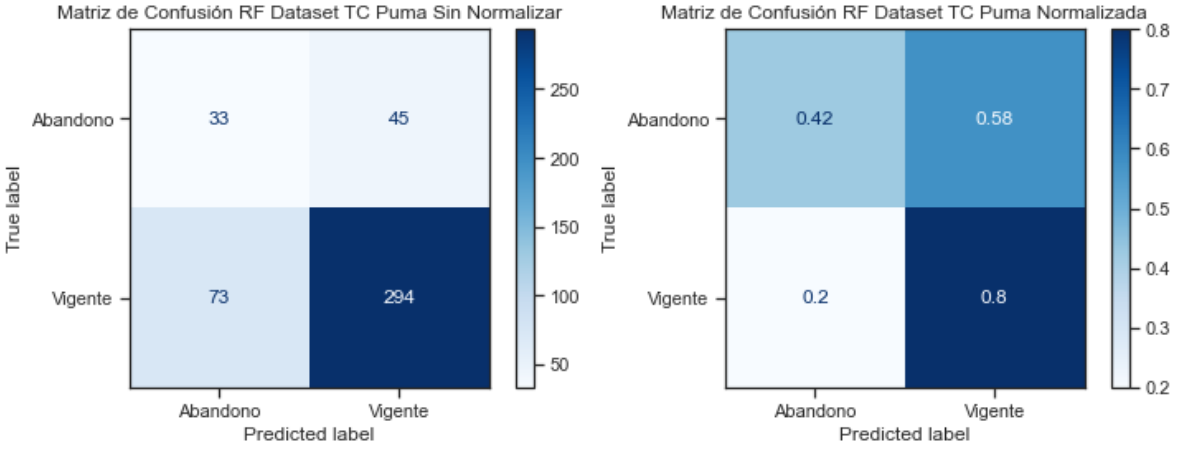



*Figura 40. Matriz de Confusión para algoritmo Random Forest Abandono TC Puma (Elaboración propia)*

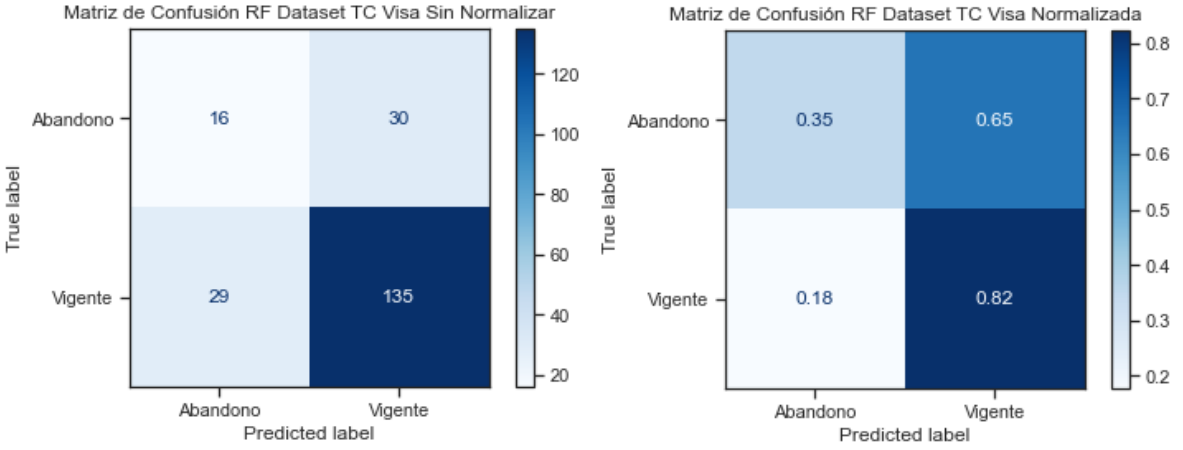



*Figura 41. Matriz de Confusión para algoritmo Random Forest Abandono TC Visa (Elaboración propia)*

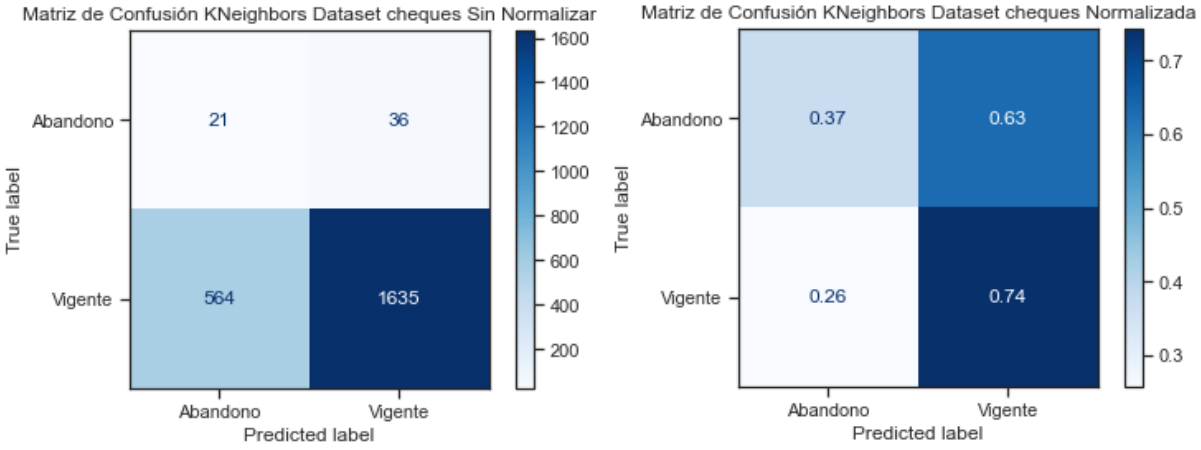



*Figura 42. Matriz de Confusión para algoritmo KNeighbors Abandono Cta. Cheques (Elaboración propia)*

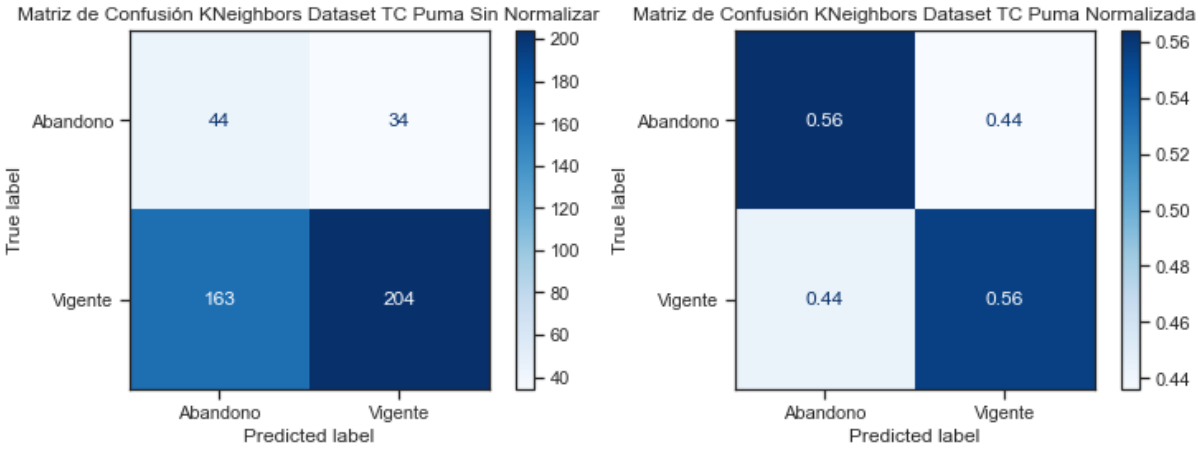



*Figura 43. Matriz de Confusión para algoritmo KNeighbors Abandono TC Puma (Elaboración propia)*

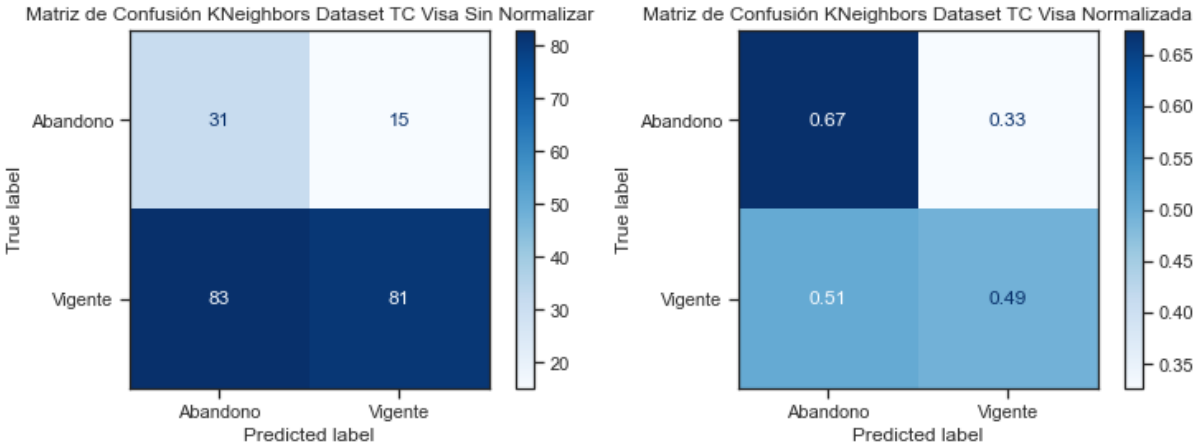



*Figura 44. Matriz de Confusión para algoritmo KNeighbors Abandono TC Visa (Elaboración propia)*

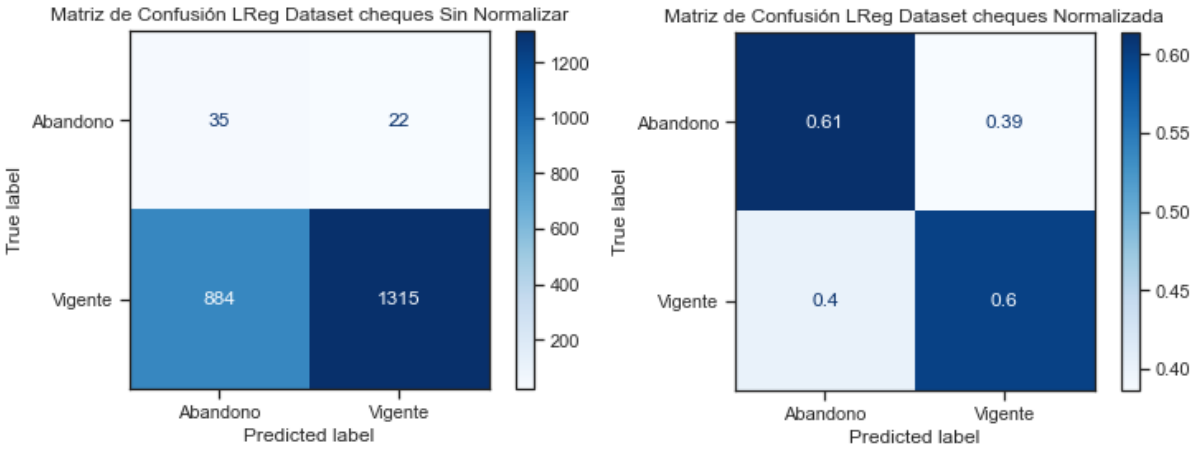



*Figura 45. Matriz de Confusión para algoritmo Logistic Regression Abandono Cta. Cheques (Elaboración propia)*

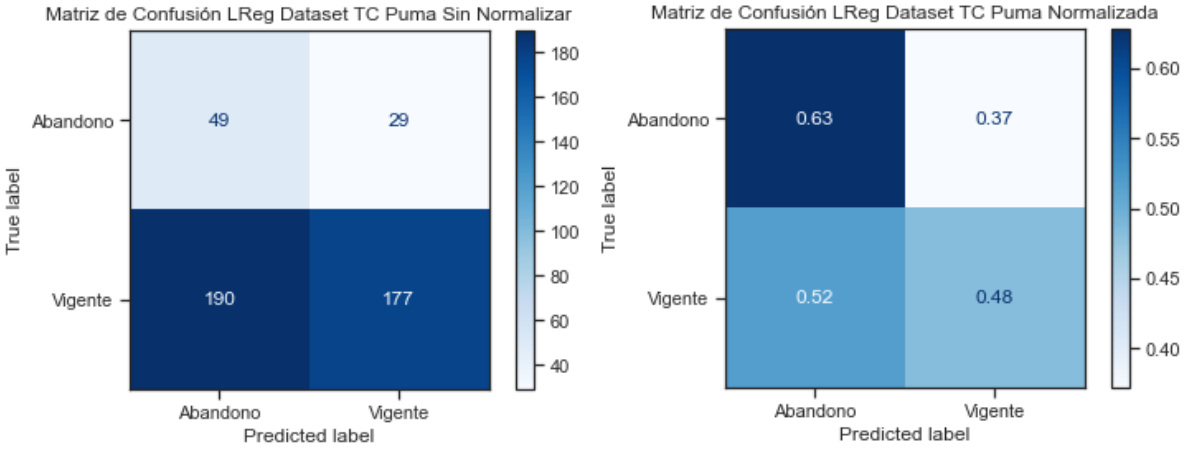



*Figura 46. Matriz de Confusión para algoritmo Logistic Regression Abandono TC Puma (Elaboración propia)*

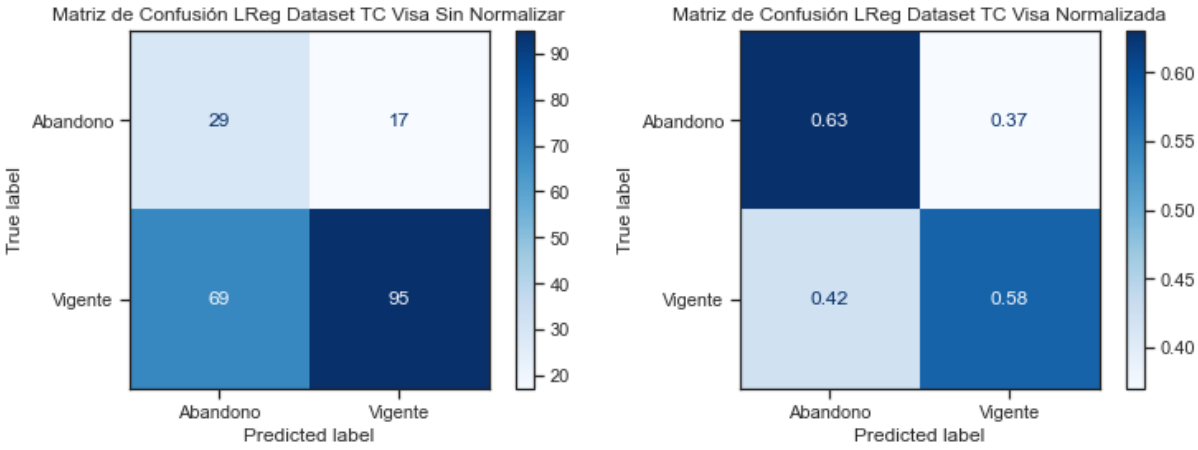



*Figura 47. Matriz de Confusión para algoritmo Logistic Regression Abandono TC Visa (Elaboración propia)*