MODELO PREDICTIVO PARA CLASIFICACIÓN DE CLIENTES EN LA ADQUISICIÓN DE PRODUCTOS FINANCIEROS BANCARIOS

Jonatan Andrey Rodriguez Ardila

Maestría en Ingeniería y Analítica de Datos
MIAD

Universidad Jorge Tadeo Lozano
Facultad de Ciencias Naturales e Ingeniería
Bogotá D.C., Colombia
2022

MODELO PREDICTIVO PARA CLASIFICACIÓN DE CLIENTES EN LA ADQUISICIÓN DE PRODUCTOS FINANCIEROS BANCARIOS

Jonatan Andrey Rodriguez Ardila

Trabajo de grado presentado como requisito para optar al título de

Magister en Ingeniería y analítica de datos

Director:
Jorge Herrera

Universidad Jorge Tadeo Lozano
Facultad de Ciencias Naturales e Ingeniería
Bogotá D.C., Colombia
2022

DEDICATORIA

Inicialmente dedico este trabajo de grado a Dios por permitirme culminar una etapa más en mi formación profesional, llena de satisfacción y del deber cumplido con todo lo aprendido en esta maestría.

A mis padres y mi hermana por ser ese motor de energía incondicional que siempre están ahí apoyándome en este proceso de formación para lograr estos grandes retos y objetivos propuestos a nivel personal.

A toda mi familia por darme ese ánimo y apoyo para nunca desistir y ser una guía para todos ellos y sus hijos.

A mis amigos y compañeros de trabajo por el apoyo recibido y el conocimiento compartido durante el proceso de formación en cada una de las asignaturas en las que compartimos.

A cada una de las personas que con sus consejos formaron en mí una gran persona y profesional del que hoy en día me siento muy orgulloso por alcanzar este gran éxito y logro a nivel personal y profesional.

Por último, a los profesores por su dedicación y esfuerzo, por brindarme sus conocimientos y sabiduría ya que fue esto lo que me motivó a culminar esta maestría.

AGRADECIMIENTOS

Principalmente agradezco el apoyo y trabajo de todas las personas que de manera directa o indirecta hicieron posible la elaboración y terminación de este trabajo de grado por lo que les doy infinitas gracias por estar ahí cuando más lo necesite, cuando no se tenía mucha inspiración y con una palabra cambiaron el rumbo de mis pensamientos llegando a la meta dispuesta.

Agradezco a mí director de trabajo Jorge Herrera por guiarme en la elaboración y terminación de este trabajo de grado, alcanzando así los objetivos planteados en este proyecto.

A todos los docentes de la Universidad Jorge Tadeo Lozano por su disciplina, exigencia, compromiso y dedicación por enseñarme correctamente, ya que esas fueron las bases para la creación de este proyecto de grado, sin ustedes de seguro esto no hubiera sido posible.

Tabla de contenido

LISTA DE TABLAS

LISTA DE FIGURAS

1. INTRODUCCIÓN

Tomando como base que las campañas de marketing actuales que realizan las entidades financieras del sector bancario para la captación de futuros clientes son una situación de orden no controlado que se presenta en la gran mayoría de las empresas y que conlleva una defectuosa gestión de los recursos disponibles en el área (ya sea esfuerzo humano, llamadas telefónicas, tiempo, dinero, etc.), este proyecto propone una solución desde el punto de vista tecnológico, de cómo puede la entidad financiera tener una mayor efectividad para futuras campañas de marketing.

El proyecto se pretende desarrollar para una entidad financiera exactamente del sector bancario. Este proyecto se llevará a cabo en el área comercial de la entidad referente a las campañas de marketing para impulsar la adquisición de productos financieros bancarios a largo plazo. El captar personas que puedan adquirir un producto financiero bancario es una tarea tediosa que podría conllevar bastante esfuerzo y jornadas exhaustivas debido a la cantidad de contactos que se le suministran a cada uno de los agentes (empleados de la entidad) para que puedan contactar a dichas personas a las cuales se le ofertan los productos.

Este proyecto no se ha resuelto antes, ya que las campañas que se realizan para captar posibles clientes las hacen por medio de bases de datos propias que ya tiene la entidad en su poder, sin realizar una segmentación que permita identificar un público objetivo.

Con este proyecto lo que se busca es poder construir un modelo predictivo de aprendizaje supervisado que permita determinar cuáles serán las posibles personas que puedan adquirir los productos financieros bancarios a largo plazo.

Teniendo en cuenta que el objetivo del modelado es predecir las posibles personas que deseen adquirir el instrumento financiero ofrecido, se entrenarán algunos modelos supervisados referentes a la clasificación de negocio mediante técnicas de machine learning.

El proyecto y los resultados mostrarán que la utilización de los modelos construidos para solucionar la ineficacia de las campañas tendrá gran valía. Esto a su vez supone tener una guía para direccionar las nuevas campañas mediante la implementación de marketing directo.

2. MARCO TEÓRICO

Con el objetivo de tener un entendimiento global del contexto en el cual se realiza el proyecto, a continuación se presentan una serie de definiciones y conceptos con el objetivo de establecer una apropiada contextualización sobre el desarrollo de este proyecto.

2.1. MARKETING DIRECTO

El marketing directo es un tipo de campaña que consiste en tener una comunicación promocional directa con el público objetivo [1]. Es decir, se enfoca en un segmento de clientes en particular que cumplen con unas ciertas características en específico y que probablemente puedan llegar a adquirir los productos que se le oferten.

2.2. PRODUCTOS FINANCIEROS BANCARIOS A LARGO PLAZO

Son un instrumento financiero que consiste en que un cliente suministra al banco un dinero por un tiempo determinado y luego de que este tiempo transcurra poder recibirlo de vuelta con un monto adicional fruto de la rentabilidad.

2.3. MACHINE LEARNING

También conocido como aprendizaje automático, es una rama de la inteligencia artificial que tiene como propósito el diseño y desarrollo de algoritmos que permitan que las máquinas aprendan a partir de los datos y no como tal de una programación previa [2]. Este proceso de aprendizaje facilita a los sistemas la adquisición de conocimiento progresivo, con mejoras a las tareas y análisis de los datos continuo [3].

Una de las características del aprendizaje automático es que debe ser entrenado y no como los sistemas tradicionales que debe ser completamente programado. Con estos algoritmos de aprendizaje automático se pueden elaborar modelos predictivos, para las posteriores tomas de decisiones, con un gran nivel de eficiencia en los resultados [4].

Los algoritmos de aprendizaje automático suelen clasificarse como supervisados o no supervisados, aunque se debe considerar que existen otros que no serán enunciados en el marco de este trabajo [5].

2.3.1. APRENDIZAJE SUPERVISADO

Se define de esta manera, ya que depende de un humano que interactúa con el modelo para enseñarle las conclusiones a las que debe llegar, es decir, aplica lo aprendido de

los datos históricos a nuevos datos utilizando variables etiquetadas para predecir eventos futuros, la salida del algoritmo ya es conocida en este caso [6].

Este tipo de algoritmo suele recibir un conjunto de datos de entrada con resultados correctos para que sean ingeridos al modelo con el objetivo de que este pueda aprender de esos datos de "entrenamiento" y así producir predicciones sobre los valores de salida [7].

2.3.1.1. ÁRBOLES DE DECISIÓN (DECISION TREE)

Es un algoritmo de tipo supervisado que generalmente se usa en problemas de clasificación. Están diseñados para trabajar con variables de tipo binaria ya que maneja un conjunto de árboles de decisión y tienen una distribución de ramificación que está conformada por unos nodos que representan los resultados para cada nodo del árbol [8].

2.3.1.2. BOSQUE ALEATORIO (RANDOM FOREST)

Es un algoritmo de tipo supervisado que está basado en la utilización de un conjunto de árboles de poca profundidad con el objetivo de mejorar el algoritmo univariado de árbol de decisión obteniendo predicciones de cada árbol y seleccionando la mejor solución mediante votación [9].

2.3.1.3. REGRESIÓN LOGÍSTICA (LOGISTIC REGRESSION)

Es un algoritmo comúnmente usado para el análisis estadístico y suele utilizarse para ver asociaciones y correlaciones entre variables con el objetivo de predecir una variable dependiente en relación con sus variables descriptoras y/o independientes [10].

2.3.1.4. EXTREME GRADIENT BOOSTING (XGBOOST)

Es un algoritmo que está diseñado principalmente para mejorar la velocidad de ejecución de las predicciones realizadas sobre los datos y así mismo para maximizar el rendimiento. Está basado bajo árboles de decisiones con gradient boosting [11].

2.3.1.5. MÁQUINA DE VECTORES DE SOPORTE (SVM)

Es un algoritmo de clasificación discriminatorio que está diseñado principalmente para dividir los datos etiquetados a través de un hiperplano para clasificar los nuevos datos ingeridos a través de espacios dimensionales trazados por unas líneas divisorias (hiperplanos) [12].

2.3.1.6. NAÏVE BAYES

Es un algoritmo de clasificación que está basado en el teorema de bayes que brinda una forma fácil y rápida de construir modelos con comportamientos buenos debido a su simplicidad. Es práctico de utilizar con conjuntos de datos de entrenamientos pequeños y donde sus atributos predictores no contengan un alto grado de correlación entre ellos [13].

2.3.2. APRENDIZAJE NO SUPERVISADO

Es usado cuando los datos no están clasificados ni etiquetados y no depende de un humano. El objetivo de este tipo de algoritmo es buscar estructuras desconocidas que permita que la máquina aprenda a clasificar de acuerdo con los datos que obtiene y genere el modelo por su propia cuenta, esperando que se arrojen resultados esperados [6].

2.4. GOOGLE COLLABORATORY

Es una herramienta que nos permitirá trabajar sobre el navegador web sin necesidad de realizar configuraciones previas de una instalación. Es muy flexible ya que permite desarrollar código en python y realizar ejecuciones propiamente dichas sobre el navegador [14].

2.5. MATRIZ DE CONFUSIÓN

Es una de las técnicas o herramientas de mayor relevancia e importancia, ya que permite evaluar cada uno de los modelos con base a un conteo de aciertos y errores de la clase predicha en los algoritmos de clasificación [15].

2.6. MATRIZ DE CORRELACIÓN

Una matriz de correlación permite identificar posibles relaciones, asociaciones o correlaciones entre las variables presentes en un conjunto de datos. Estas matrices definen las relaciones por medio de un puntaje entre -1 y 1 para identificar correlaciones extremadamente fuertes o extremadamente débiles [16].

2.7. CURVA DE ROC

Es una técnica que permite identificar el equilibrio que puede existir entre la sensibilidad (Verdaderos Positivos) y la especificidad (Falsos Positivos) de un modelo de machine learning para evaluar su rendimiento [17].

2.8. ÁREA BAJO LA CURVA AUC

Es una representación gráfica del área bajo la curva ROC, que permite identificar que tan bueno es el modelo a evaluar. Aquí se evalúa la probabilidad que tiene el modelo para distinguir entre los verdaderos positivos y falsos positivos [18].

2.9. INDICADORES DE DESEMPEÑO EN MACHINE LEARNING

Son métricas o indicadores que permiten evaluar a cada uno de los algoritmos o modelos de machine learning para medir precisiones, sensibilidad, exhaustividad, especificidad, balances, promedios, etc y tomar decisiones con base a estos indicadores medibles [19].

2.10. LABEL ENCODER

Es una técnica de machine learning que permite transformar información categórica a numérica asignando a cada dato categórico un número entero único para su representación [20].

2.11. STANDARD SCALER

Es una técnica de normalización de los datos que permite eliminar el sesgo potencial que el modelo puede tener hacia características con magnitudes más altas [21].

2.12. SMOTE (SOBREMUESTREO)

Es una técnica la cual permite equilibrar en igual proporción y/o volumen de datos la clase minorista con respecto a la clase mayoritaria que se encuentra sobre la variable objetivo o predicha de un conjunto de datos [22].

2.13. GRID SEARCH CV

Es un paquete de python que permite recorrer los hiperparametros de cada uno de los modelos para obtener los mejores valores que se emplearán en el modelo final. Es de vital importancia en los modelos de machine learning para definir correctamente cuáles serán los mejores valores o parámetros de los hiperparametros [23].

2.14. CONJUNTO DE DATOS

El conjunto de datos a trabajar en este proyecto contiene 41.188 registros y fue extraído del repositorio UCI Machine Learning [24].

3. ESTADO DEL ARTE

En la Tabla 1 se presentarán algunos de los estudios relacionados previos a esta investigación.

Tabla 1. Trabajos de Investigación Campañas de Marketing Directo

Objetivos y Resultados	Herramientas y Técnicas Aplicadas	Limitaciones Investigación Vs Proyecto a Desarrollar
El objetivo de esta investigación fue utilizar técnicas de minería de datos basadas en el concepto de Business Intelligence para la toma de decisiones en campañas de marketing directo y obtener una visión de resultados concretos para el mejoramiento de las campañas. Gracias al business intelligence y a la implementación de técnicas de clasificación en esta investigación, se pudo demostrar que es posible mejorar futuras campañas de marketing en la organización [25].	- Weka Data Mining. - Técnicas de clasificación (Árboles de decisión), agrupamiento (Clustering) y reglas de asociación (Apriori),	- Se utilizó una submuestra de 4.521 registros con base a la muestra original. Esto puede representar desventajas para el modelo, debido a los pocos datos que se tienen para el entrenamiento de este, por lo cual se hace más complejo llegar a tener un modelo más robusto y de mejor predicción. - Solo aplicaron una técnica de clasificación.
El objetivo de esta investigación fue diseñar un pequeño Data Mart a partir de una Bodega de Datos centralizada, con la finalidad de orientar el proyecto a temas netamente de marketing directo, que puedan brindar información útil para la toma de decisiones y determinar específicamente qué tan eficaz es la contactabilidad con el usuario final. El resultado obtenido aquí, demostró cual es el medio por el cual se es más efectivo la contactabilidad con un usuario final siendo los medios de comunicación más efectivos el email y el teléfono [26].	- Data Mart - Herramienta de Integración de Datos - Lenguaje SQL	- No se utilizaron técnicas de aprendizaje predictivo. - Integración de más de diez (10) fuentes de datos de distintos sistemas, lo que conlleva una complejidad aún mayor en el análisis de la información contenida allí. - Para segmentar los clientes que se pretenden contactar, se realiza una lógica en la base de datos con reglas de negocio construidas a partir de querys SQL.
El objetivo de esta investigación fue el análisis que se obtuvo de las relaciones y/o asociaciones que llegaron a existir entre los chatbots de facebook y la eficacia de una campaña de marketing mediante este medio (chatbots-personas). El resultado obtenido en esta investigación demostró que los chatbots bien construidos tienden a tener gran impacto en la satisfacción de un cliente que por consecuencia lleva al mejoramiento de las campañas de marketing tratadas por este medio [27].	- Chatbot - SPSS Modeler	- No se utilizaron técnicas de aprendizaje predictivo. - La información es proveniente de encuestas, con una muestra muy pequeña de datos (250 registros). - Esta investigación está orientada a temas de correlación que pueda existir entre dos o más variables para identificar la satisfacción de un cliente.

El objetivo de esta investigación fue la construcción de un modelo de machine learning que permitiera medir la efectividad de las campañas de marketing digital que se llevan a cabo en una red social como YouTube bajo herramientas como Social Media Analytics. Los resultados obtenidos demostraron que mediante algoritmos de clasificación es posible efectuar mejoras y optimización de recursos con el fin de minimizar inversión a campañas de marketing que no generan valor a la entidad [28].	- Técnicas de clasificación (Naive Bayes, Regresión Logística, Árboles de decisión, Random Forest). - Social Media Analytics - Data Robot	- La información es proveniente de una campaña efectuada desde YouTube, donde se recolectó un conjunto de datos muy pequeño (106 registros). - Las evaluaciones aplicadas en los modelos de clasificación efectuados en esta investigación solo conllevan a temas de correlación en función con una variable objetivo.
El objetivo de esta investigación fue identificar cuáles eran las variables más representativas para la adquisición de un crédito financiero con base a la variable dependiente. En este caso los resultados reflejaron la variabilidad que puede representar las variables independientes cuando se toman las de mayor relevancia, ya que ello genera la identificación de potenciales clientes que posiblemente lleguen a adquirir un crédito de efectivo. [29].	- Modelos de machine learning (xgboost y lightgbm) - Teradata Studio - Python - Tableau - AWS SageMaker	- Con la construcción del modelo xgboost se pudo identificar la medición de la precisión que se obtuvo para adquirir un crédito de efectivo el cual conllevo al 70.4% y del otro lado con la construcción del modelo lightgbm se obtuvo una precisión del 69.3%, lo que conlleva a que se podría mejorar ese porcentaje si se hace un análisis más profundo con otros modelos de machine learning.
El objetivo de esta investigación fue la participación que se tiene con los productos o sistemas de administración de tarjetas de crédito en entidades financieras con respecto al marketing relacional y la relación con sus clientes. Los resultados obtenidos en esta investigación reflejaron que mediante el marketing relacional y las diferentes técnicas descriptivas es posible identificar la fidelización que se da entre los clientes y la entidad y la perduración de la relación contractual entre estos dos roles. [30].	-Estadística descriptiva - Tablas de frecuencia - Gráfico de barras y tortas	- No se utilizaron técnicas de aprendizaje predictivo. - La técnica aplicada en esta investigación está orientada a temas cuantitativos, análisis descriptivos agrupados y univariados.
El objetivo de esta investigación fue proporcionar un marco integral para guiar los esfuerzos de investigación centrados en la estrategia de marketing directo para lograr el éxito utilizando métodos de minería de datos. Los resultados demostraron que la minería de datos es una herramienta muy valiosa para el marketing directo que puede mejorar las campañas de marketing de los bancos en campañas futuras [31].	- Técnicas de clasificación (Árboles de decisión) - IBM SPSS Modeler	- Se eligió el algoritmo del árbol de decisión porque son herramientas potentes y populares para la clasificación y la predicción, pero no se tomó en cuenta ningún otro tipo de algoritmo que pueda mejorar aún más la predicción del modelo de esta investigación que finalizó con una precisión de 93.37%.

El objetivo de esta investigación fue utilizar técnicas de minería de datos para interpretar y definir las características importantes para aumentar la efectividad de la campaña, es decir, si el cliente suscribe el depósito a plazo. Los resultados experimentales mostraron que un conjunto reducido de características mejora el rendimiento de la clasificación [32].	- Técnicas de clasificación (Naive Bayes) - Métodos de selección de características (Ganancia de información y Chi-cuadrado) - Weka Data Mining	- Se eligió el algoritmo de Naive Bayes pero no se tomó en cuenta ningún otro tipo de algoritmo a evaluar en esta investigación, así mismo no se representó la precisión de este tipo de algoritmo en este trabajo de investigación. - Se orientó demasiado a métodos de selección, dejando atrás puntos relevantes para obtener un modelo de predicción ajustado a los objetivos del trabajo de investigación.
El objetivo de esta investigación fue presentar distintas técnicas de algoritmos de clasificación como árboles de decisión, algoritmo de injerto J48, etc., sobre un conjunto de datos de marketing directo de una entidad del sector bancario portugués. Los resultados de esta investigación mostraron mediciones muy buenas y precisas, así mismo demostró que el método propuesto tiene una mejor clasificación en comparación con la técnica de minería de reglas de clasificación y asociación [33].	- Técnicas de clasificación (Árboles de decisión). Algoritmo de injerto J48 y algoritmo de árbol LAD - Aprendizaje automático (red de función de base radial y una máquina de vectores de soporte) - Weka Data Mining	- A pesar de que se probaron diferentes técnicas de clasificación se pudo observar que es posible mejorar aún más la predicción del modelo de esta investigación ya que el método de clasificación con mejor exactitud (máquina de vectores de soporte (SVM)) finalizó con una precisión de 86.95%. - No se utilizaron técnicas o gráficos descriptivos que representan el conjunto de datos que se llevó a cabo en esta investigación.
El objetivo de esta investigación fue la construcción y puesta en marcha de técnicas de minería de árboles de decisión y teoría de conjuntos aproximados (RST), utilizando datos obtenidos de una campaña de marketing de depósitos bancarios. Los resultados mostraron que la RST produce un mejor resumen del conjunto de datos debido al proceso de reducción de características que logra el mejor conjunto mínimo de características lo que conlleva a tener un mejor modelo predictivo y mejorar la eficiencia de las futuras campañas de marketing directo en la entidad [34].	- Técnicas de clasificación (Árboles de decisión) - Teoría de conjuntos - Base de Datos Relacional - Weka Data Mining	- Se diseñó una base de datos relacional para almacenar y preprocesar la información del conjunto de datos con el objetivo de realizar la limpieza de datos allí. Esto implica que se genere una complejidad y esfuerzo mayor a la hora de relacionar la información para interpretar el resultado final. - Se eligió el algoritmo de árboles de decisión, pero no se tomó en cuenta ningún otro tipo de algoritmo que pueda mejorar aún más la predicción del modelo de esta investigación.

4. PLANTEAMIENTO DEL PROBLEMA

Existe un problema de dificultad y costo en las campañas de marketing actuales que se realizan en la entidad bancaria con los productos financieros bancarios a largo plazo.

Se necesita solucionar el problema planteado ya que el área comercial del banco requiere aumentar la eficiencia de las campañas relacionadas a la suscripción de productos financieros a largo plazo y de igual forma optimizar los recursos predispuestos para la promoción del activo financiero.

El principal problema del proyecto se centra en la depuración de los datos ya que, se cuenta con un volumen considerable de datos recopilados a partir de las campañas telefónicas realizadas hasta la fecha del proyecto que no se encuentra de manera segmentada o clasificada para llegar a un público objetivo.

Esto refleja una afectación en el incumplimiento de los objetivos de rentabilidad de los activos financieros y deficiencias en la captación de capital para la dirección general, una ineficiencia en las campañas efectuadas y exceso de recursos predispuesto para este rubro en el área comercial y jornadas exhaustivas debido a la cantidad de contactos que deben realizar los agentes de la entidad.

5. OBJETIVOS

5.1. OBJETIVO GENERAL

Implementar un modelo de aprendizaje automático que permita clasificar a posibles clientes que puedan adquirir los productos financieros bancarios a largo plazo con el objetivo de aumentar la eficiencia de las campañas futuras y optimizar los recursos disponibles en el área.

5.2. OBJETIVOS ESPECÍFICOS

- Depurar o estructurar el conjunto de datos manejado en este proyecto, para garantizar una visión segmentada de los clientes que puedan adquirir un producto financiero bancario a largo plazo.
- Probar y evaluar distintos modelos predictivos de aprendizaje automático supervisado que permita encontrar patrones de comportamiento en las diferentes variables, para establecer el más adecuado.
- Determinar las variables y factores de mayor incidencia que son significativas en la adquisición de un producto financiero bancario a largo plazo.
- Desplegar un modelo de aprendizaje automático con el mejor rendimiento en un entorno web.

6. METODOLOGÍA

Para el desarrollo de este proyecto se usará la metodología Cross-Industry Standard Process for Data Mining (CRISP-DM). En la Tabla 2 se establece las diferentes etapas para el desarrollo de los objetivos propuestos en el presente trabajo:

Tabla 2. Etapas metodología CRISP-DM

Etapa	Descripción
Entendimiento del negocio	En esta etapa se encuentran las tareas de adquirir los conceptos básicos del entorno de la situación a analizar, con el objetivo de comprender claramente el negocio a tratar [35].
Comprensión de los datos	Inicia con la recopilación y familiarización de los datos haciendo una exploración de estos, evaluando el conjunto de datos y revisando la calidad de los mismos [35].
Preparación de los datos	Contempla todas las actividades a realizar como el preprocesamiento de los datos, visualización y distribución de los datos, los análisis univariados de cada variable con el objetivo de obtener un conjunto de datos normalizado que será ingestado en etapas posteriores a un modelo en específico [35].
Modelamiento	Se escogen distintas técnicas de modelado y se selecciona la que mejor se ajuste y que cumplan con los objetivos propuestos de este proyecto [35].
Evaluación	En esta etapa se evaluaran varios aspectos del modelo elegido en la etapa anterior como la precisión, la exactitud, la sensibilidad, la puntuación F equilibrada (f1-micro), etc, con el objetivo de validar que tan bueno es nuestro modelo frente a nuevos datos que se le ingestaran a dicho modelo [35].
Despliegue	En esta etapa se explora la utilidad de los modelos construidos y así mismo se despliega en un ambiente productivo para llevar a cabo las predicciones requeridas frente a escenarios de datos reales [35].

7. DESARROLLO DE LA PROPUESTA

Dando inicio al desarrollo de este proyecto bajo la metodología CRISP-DM la cual es utilizada como punto de referencia para los proyectos de analítica y minería de datos y la cual será fundamental para la implementación de este proyecto, se iniciará describiendo cada una de las fases que conlleva a la realización de este:

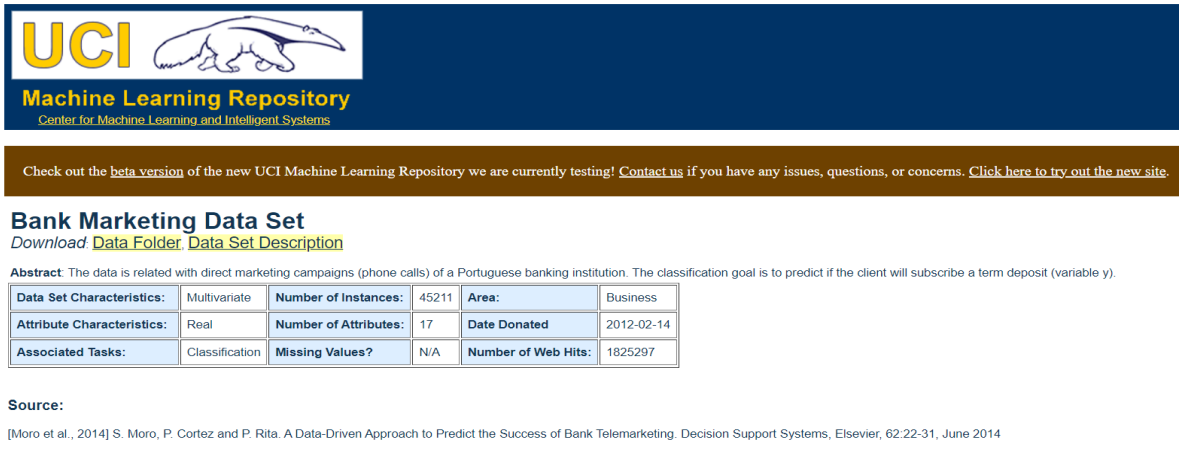
7.1. COMPRENSIÓN DE LOS DATOS

7.1.1. RECOLECCIÓN DE LOS DATOS

Para el desarrollo del presente trabajo de grado el conjunto de datos a trabajar en este proyecto contiene 41.188 registros y fue extraído del repositorio UCI Machine Learning.

Este conjunto de datos proviene de una entidad financiera del sector bancario en Portugal y contiene una serie de variables referentes a las campañas de marketing directo (en este caso llamadas telefónicas) con el objetivo de ofrecer a sus clientes un instrumento financiero como lo son productos bancarios a largo plazo (CDT, Cuentas de ahorro, Cuentas monederos, etc).

El objetivo de este conjunto de datos es predecir si un cliente de la entidad aceptará o no el producto financiero de acuerdo con una serie de características.


UCI Machine Learning Repository
Center for Machine Learning and Intelligent Systems

Check out the beta version of the new UCI Machine Learning Repository we are currently testing! Contact us if you have any issues, questions, or concerns. Click here to try out the new site.

Bank Marketing Data Set
Download: Data Folder, Data Set Description

Abstract: The data is related with direct marketing campaigns (phone calls) of a Portuguese banking institution. The classification goal is to predict if the client will subscribe a term deposit (variable y).

Data Set Characteristics:	Multivariate	**Number of Instances:**	45211	**Area:**	Business
Attribute Characteristics:	Real	**Number of Attributes:**	17	**Date Donated**	2012-02-14
Associated Tasks:	Classification	**Missing Values?**	N/A	**Number of Web Hits:**	1825297

Source:

[Moro et al., 2014] S. Moro, P. Cortez and P. Rita. A Data-Driven Approach to Predict the Success of Bank Telemarketing. Decision Support Systems, Elsevier, 62:22-31, June 2014

Ilustración 1. Conjunto de Datos Marketing Directo
Fuente: Imagen capturada desde el repositorio UCI Machine Learning

7.1.2. DESCRIPCIÓN DE LOS DATOS

Este conjunto de datos contiene una serie de atributos que se distribuyen en varios contextos. En las siguientes tablas se describirán cada uno de estos:

Tabla 3. Atributos información personal y financiera

Campo	Tipo de Dato	Descripción
Age (Edad)	Numérico	Edad de la persona
Job (Trabajo)	Categórico	Tipo de trabajo de la persona
Marital (Estado Civil)	Categórico	Estado civil de la persona
Education (Educación)	Categórico	Nivel de educación de la persona
Default (Crédito en Mora)	Categórico	La persona tiene crédito en mora
Housing (Préstamo Vivienda)	Categórico	La persona tiene préstamo de vivienda vigente
Loan (Préstamo Personal)	Categórico	La persona tiene préstamo de libre inversión

Tabla 4. Atributos campañas anterior y actual

Campo	Tipo de Dato	Descripción
Contact (Contacto)	Categórico	Tipo de contacto con la persona
Month (Mes)	Categórico	Nombre mes del último contacto con la persona
Day_Of_Week (Día Semana)	Categórico	Nombre del día del último contacto con la persona
Duration (Duración)	Numérico	Duración de la comunicación (segundos) con la persona
Campaign (Campaña)	Numérico	Cantidad de contactos durante la campaña vigente
PDays (Días Pasados)	Numérico	Cantidad días que pasaron desde el último contacto y campaña
Previous (Contactos Anterior)	Numérico	Cantidad de contactos realizados antes de la campaña vigente
Poutcome (Campaña Anterior)	Categórico	Resultado obtenido de la campaña anterior

Tabla 5. Atributos social y económico

Campo	Tipo de Dato	Descripción
Emp.Var.Rate (Tasa Variación Empleo)	Numérico	Indicador trimestral de la tasa de variación de empleo
Cons.Price.Idx (Índice Precio Consumidor)	Numérico	Indicador mensual del índice del precio al consumidor
Cons.Conf.Idx (Índice Confianza Consumidor)	Numérico	Indicador mensual del índice de confianza al consumidor
Euribor3m (Tasa Euribor)	Numérico	Indicador diario de la tasa euribor a 3 meses
Nr.Employed (Número Empleados)	Numérico	Indicador trimestral del número de empleados

Tabla 6. Atributo de salida (Objetivo deseado)

Campo	Tipo de Dato	Descripción
Y (Producto Financiero Bancario)	Binario	Variable de salida que indica si la persona adquiere o no el producto financiero bancario.

7.1.3. EXPLORACIÓN DE LOS DATOS

La exploración de los datos es uno de los puntos más importantes y base fundamental dentro de un proyecto de analítica de datos para tomar las mejores decisiones sobre el conjunto de datos.

Para hacer análisis exploratorios sobre el conjunto de datos, necesitamos configurar nuestro entorno de trabajo. En este caso se utilizará Google Collaboratory como herramienta de trabajo para el desarrollo de este proyecto. Esta es una herramienta que nos permitirá desarrollar código en Python y realizar ejecuciones sobre el navegador.

Para esto, primero importamos el conjunto de datos:

```
[6] url = "https://raw.githubusercontent.com/JonatanAndreyRodriguez/Trabajo-Grado-MIAD/master/Conjunto%20Datos/Marketing%20Bancario.csv"
    download = requests.get(url).content
    df = pd.read_csv(io.StringIO(download.decode('utf-8')))

    df.head(10)
```

	age	job	marital	education	default	housing	loan	contact	month	day_of_week	...	campaign	pdays	previous	poutcome	emp.var.rate	cons.price.idx	cons.conf.idx	euribor3m	nr.employed	y
0	56	housemaid	married	basic.4y	no	no	no	telephone	may	mon	...	1	999	0	nonexistent	1.1	93.994	-36.4	4.857	5191.0	no
1	57	services	married	high.school	unknown	no	no	telephone	may	mon	...	1	999	0	nonexistent	1.1	93.994	-36.4	4.857	5191.0	no
2	37	services	married	high.school	no	yes	no	telephone	may	mon	...	1	999	0	nonexistent	1.1	93.994	-36.4	4.857	5191.0	no
3	40	admin.	married	basic.6y	no	no	no	telephone	may	mon	...	1	999	0	nonexistent	1.1	93.994	-36.4	4.857	5191.0	no
4	56	services	married	high.school	no	no	yes	telephone	may	mon	...	1	999	0	nonexistent	1.1	93.994	-36.4	4.857	5191.0	no
5	45	services	married	basic.9y	unknown	no	no	telephone	may	mon	...	1	999	0	nonexistent	1.1	93.994	-36.4	4.857	5191.0	no
6	59	admin.	married	professional.course	no	no	no	telephone	may	mon	...	1	999	0	nonexistent	1.1	93.994	-36.4	4.857	5191.0	no
7	41	blue-collar	married	unknown	unknown	no	no	telephone	may	mon	...	1	999	0	nonexistent	1.1	93.994	-36.4	4.857	5191.0	no
8	24	technician	single	professional.course	no	yes	no	telephone	may	mon	...	1	999	0	nonexistent	1.1	93.994	-36.4	4.857	5191.0	no
9	25	services	single	high.school	no	yes	no	telephone	may	mon	...	1	999	0	nonexistent	1.1	93.994	-36.4	4.857	5191.0	no

10 rows × 21 columns

Ilustración 2. Importación conjunto de datos
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

Luego validamos el tipo de dato de cada uno de los atributos del conjunto de datos:

```
[8] df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 41188 entries, 0 to 41187
Data columns (total 21 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   age             41188 non-null  int64
 1   job             41188 non-null  object
 2   marital         41188 non-null  object
 3   education       41188 non-null  object
 4   default         41188 non-null  object
 5   housing         41188 non-null  object
 6   loan            41188 non-null  object
 7   contact         41188 non-null  object
 8   month           41188 non-null  object
 9   day_of_week     41188 non-null  object
 10  duration        41188 non-null  int64
 11  campaign        41188 non-null  int64
 12  pdays           41188 non-null  int64
 13  previous        41188 non-null  int64
 14  poutcome        41188 non-null  object
 15  emp.var.rate    41188 non-null  float64
 16  cons.price.idx  41188 non-null  float64
 17  cons.conf.idx   41188 non-null  float64
 18  euribor3m       41188 non-null  float64
 19  nr.employed     41188 non-null  float64
 20  y               41188 non-null  object
dtypes: float64(5), int64(5), object(11)
memory usage: 6.6+ MB
```

Ilustración 3. Tipo de dato para el conjunto de datos
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

Como podemos observar en la ilustración 3 los atributos contenidos en nuestro conjunto de datos tienen tres (3) tipos de datos que son: enteros, categóricos y flotantes.

Posteriormente realizaremos un análisis descriptivo de las variables numéricas:

```
df.describe(exclude = 'object')
```

	age	duration	campaign	pdays	previous	emp.var.rate	cons.price.idx	cons.conf.idx	euribor3m	nr.employed
count	41188.00000	41188.000000	41188.000000	41188.000000	41188.000000	41188.000000	41188.000000	41188.000000	41188.000000	41188.000000
mean	40.02406	258.285010	2.567593	962.475454	0.172963	0.081886	93.575664	-40.502600	3.621291	5167.035911
std	10.42125	259.279249	2.770014	186.910907	0.494901	1.570960	0.578840	4.628198	1.734447	72.251528
min	17.00000	0.000000	1.000000	0.000000	0.000000	-3.400000	92.201000	-50.800000	0.634000	4963.600000
25%	32.00000	102.000000	1.000000	999.000000	0.000000	-1.800000	93.075000	-42.700000	1.344000	5099.100000
50%	38.00000	180.000000	2.000000	999.000000	0.000000	1.100000	93.749000	-41.800000	4.857000	5191.000000
75%	47.00000	319.000000	3.000000	999.000000	0.000000	1.400000	93.994000	-36.400000	4.961000	5228.100000
max	98.00000	4918.000000	56.000000	999.000000	7.000000	1.400000	94.767000	-26.900000	5.045000	5228.100000

Ilustración 4. Análisis descriptivo variables numéricas
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 4 podemos ir observando información muy relevante como la siguiente:

- La edad promedio de los clientes es de aproximadamente 40 años. (Mínimo: 17 años y Máximo: 98 años).
- El promedio de duración de las llamadas es de 258 segundos (corresponden a 4 minutos). (Mínimo: 0 y Máximo: 4.918 (5 minutos)).
- La cantidad promedio de llamadas durante una campaña vigente es tres (3). (Mínimo: 1 y Máximo: 56). Aquí podemos intuir que el número máximo de llamadas podría ser un dato inusual dentro del conjunto de datos.
- Un dato muy importante para tener en cuenta en el atributo pdays (Días Pasados) es que el número 999 representa que el cliente no fue contactado previamente, por lo cual este análisis descriptivo para este atributo en la ilustración 4 no genera una descripción real para tomar decisiones sobre los datos contenidos en este atributo.
- La cantidad promedio de llamadas durante una campaña anterior a la vigente es una (1). (Mínimo: 0 y Máximo: 7).
- Para lo atributos Emp.Var.Rate (Tasa de variación del empleo), Cons.Price.Idx (Índice de precios al consumidor), Cons.Conf.Idx (Índice de confianza del consumidor), Euribor3m (Indicador diario de la tasa euribor a 3 meses) y Nr.Employed (Número de empleados) al ser variables de contexto social y económica a nivel poblacional dentro de un país no serán tenidas en cuenta dentro de los siguientes análisis ya que no son variables influyentes con la variable predictora para el objetivo final del proyecto.

Ahora realizaremos un análisis descriptivo de las variables categóricas:

```
df.describe(exclude = ['float', 'int64'])
```

	job	marital	education	default	housing	loan	contact	month	day_of_week	poutcome	y
count	41188	41188	41188	41188	41188	41188	41188	41188	41188	41188	41188
unique	12	4	8	3	3	3	2	10	5	3	2
top	admin.	married	university.degree	no	yes	no	cellular	may	thu	nonexistent	no
freq	10422	24928	12168	32588	21576	33950	26144	13769	8623	35563	36548

Ilustración 5. Análisis descriptivo variables categóricas
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 5 podemos ir observando información importante como la siguiente:

- Existen 12 tipos de trabajos distintos en el conjunto de datos.
- Existen 4 tipos de estado civil.
- Existen 8 niveles de educación.
- Existen 3 tipos de valores únicos en la variables default (Crédito en Mora), housing (Préstamo Vivienda) y loan (Préstamo Personal).
- Existen 2 medios de contacto de comunicación con el cliente.
- Hay únicamente 10 meses contenidos en el conjunto de datos.
- Hay únicamente 5 días de la semana contenidos en el conjunto de datos.
- Existen 3 tipos de resultados obtenidos en la campaña anterior.
- Existen únicamente 2 valores en nuestra variable objetivo.

Una vez realizado el análisis descriptivo de los atributos, se validará si estos contienen información nula o faltante:

```
df.isna().sum()

age               0
job               0
marital           0
education         0
default           0
housing           0
loan              0
contact           0
month             0
day_of_week       0
duration          0
campaign          0
pdays             0
previous          0
poutcome          0
emp.var.rate      0
cons.price.idx    0
cons.conf.idx     0
euribor3m         0
nr.employed       0
y                 0
dtype: int64
```

Ilustración 6. Información datos nulos o faltantes
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

Es importante observar en la ilustración 6 que este conjunto de datos no cuenta con información faltante o nula.

Ahora iniciaremos la exploración gráfica de los datos para alcanzar algunos resultados analíticos útiles. Realizaremos gráficos y examinaremos los datos con el objetivo de poder hacer análisis univariados y compuestos entre variables, detectar algunos posibles valores atípicos, observar las distribuciones que se dan en los datos y así mismo verificar posibles correlaciones entre las variables mediante matrices de correlación.

7.1.3.1. ANÁLISIS UNIVARIADO

Procederemos a graficar cada una de las variables del conjunto de datos para ir detectando el comportamiento de los datos que contiene cada una de estas.

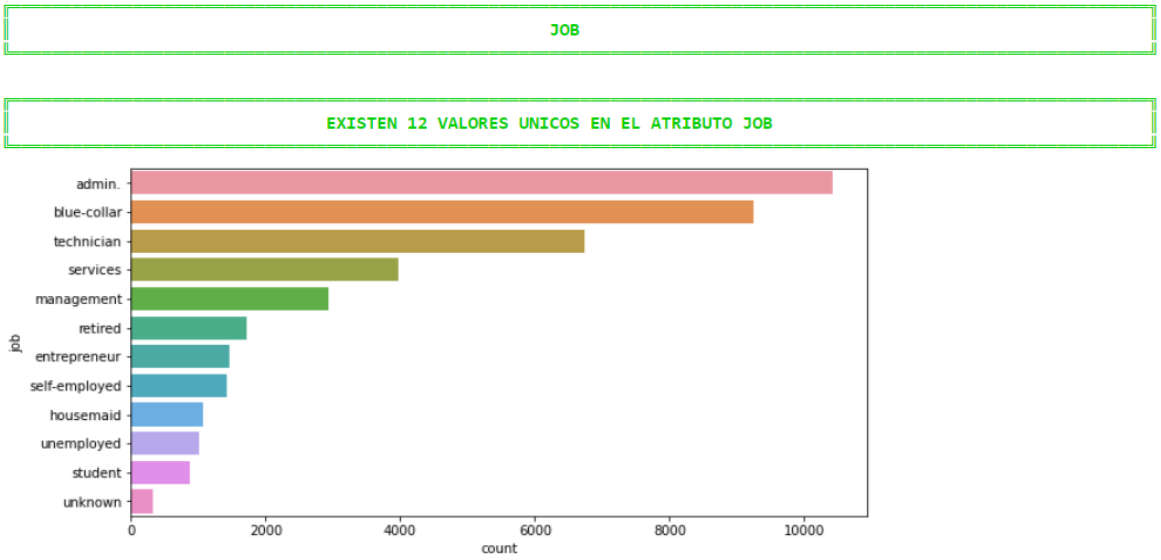


Ilustración 7. Análisis univariado variable Job (Trabajo)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 7 podemos observar que en la variable Job (Trabajo) existen 12 (doce) tipos de trabajo de los cuales los que más sobresalen en cantidades son los admin (administrativos), blue-collar (obreros) y los tehnician (técnicos). Así mismo la gráfica representa que hay un porcentaje mínimo de datos desconocidos (Unknown) los cuales se deberán tratar en la próxima fase de CRISP-DM.

```
MARITAL
```

```
EXISTEN 4 VALORES UNICOS EN EL ATRIBUTO MARITAL
```

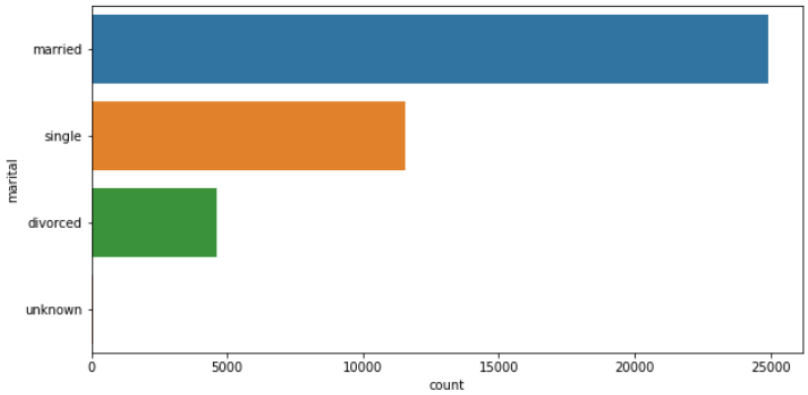


Ilustración 8. Análisis univariado variable Marital (Estado Civil)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 8 podemos observar que en la variable Marital (Estado Civil) existen 4 (cuatro) posibles valores de los cuales el que más sobresale en cantidades es el married (casado). Así mismo la gráfica representa que hay un porcentaje mínimo de datos desconocidos (Unknown) los cuales se deberán tratar en la próxima fase de CRISP-DM.

```
EDUCATION
```

```
EXISTEN 8 VALORES UNICOS EN EL ATRIBUTO EDUCATION
```

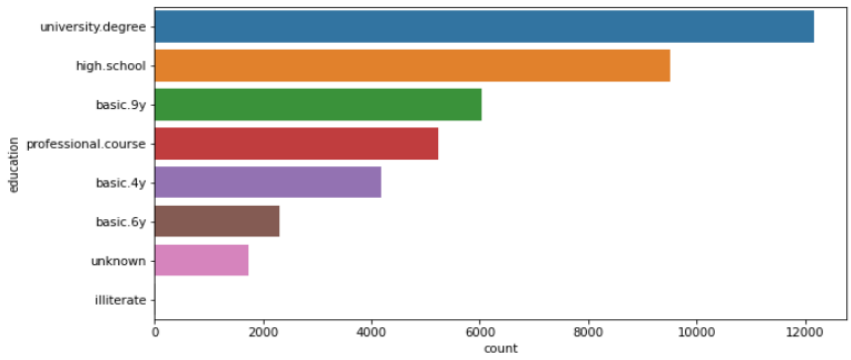


Ilustración 9. Análisis univariado variable Education (Educación)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 9 podemos observar que en la variable Education (Educación) existen 8 (ocho) niveles de educación de los cuales los que más sobresalen en cantidades son university degree (título universitario), high school (escuela secundaria) y basic 9y

(básico 9). La gráfica también representa que hay un porcentaje mínimo de datos desconocidos (Unknown) los cuales se deberán tratar en la próxima fase de CRISP-DM y así mismo hay un porcentaje alto de datos que están sobre el nivel básico los cuales deberán ser analizados si es una buena opción poderlos agrupar.

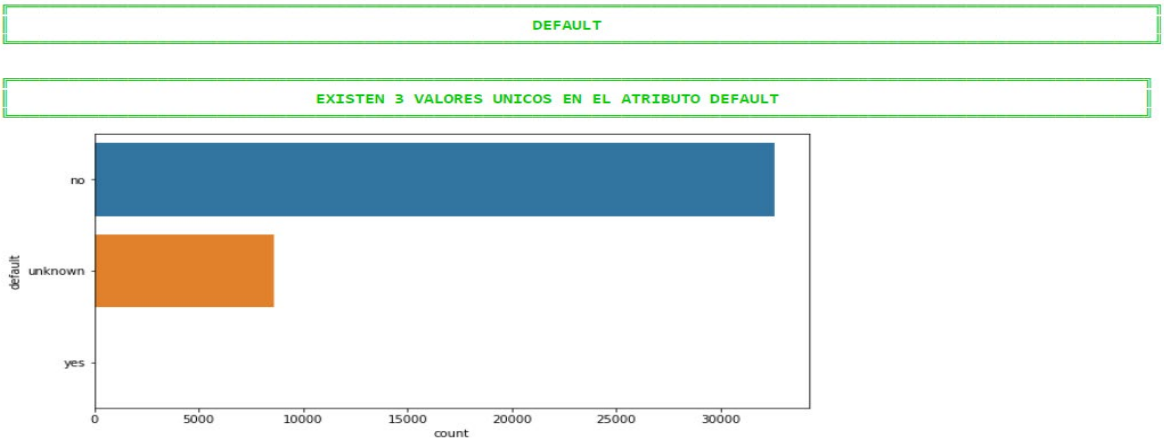


Ilustración 10. Análisis univariado variable Default (Crédito en mora)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 10 podemos observar que en la variable Default (Crédito en mora) existen 3 (tres) posibles valores de los cuales el que más sobresale en cantidades es el No (No tiene crédito en mora). Así mismo la gráfica representa que hay un porcentaje considerable de datos desconocidos (Unknown) los cuales se deberán tratar en la próxima fase de CRISP-DM.

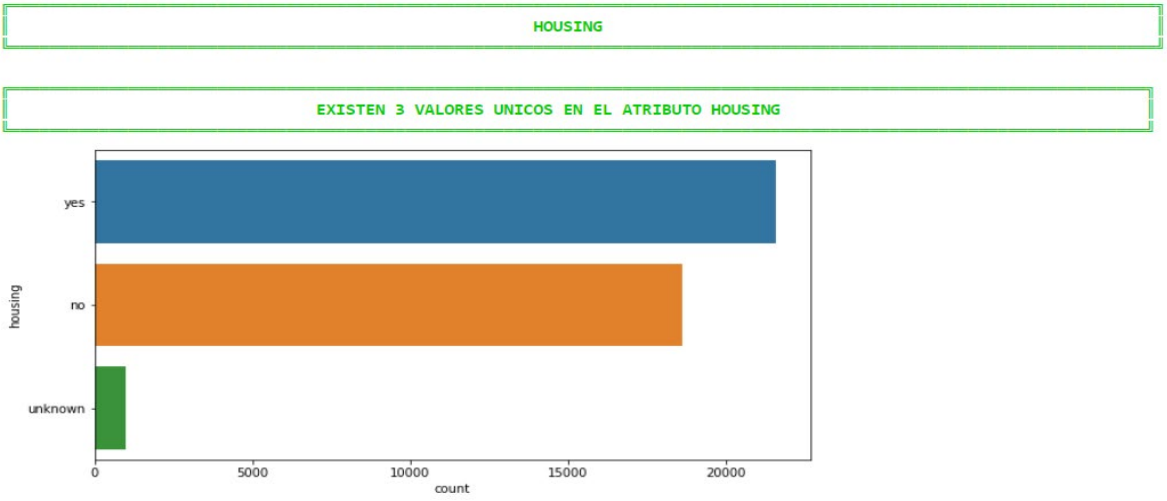


Ilustración 11. Análisis univariado variable Housing (Crédito de vivienda)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 11 podemos observar que en la variable Housing (Crédito de vivienda) existen 3 (tres) posibles valores que están distribuidos equitativamente entre el Sí y el

No. Así mismo la gráfica representa que hay un porcentaje mínimo de datos desconocidos (Unknown) los cuales se deberán tratar en la próxima fase de CRISP-DM.

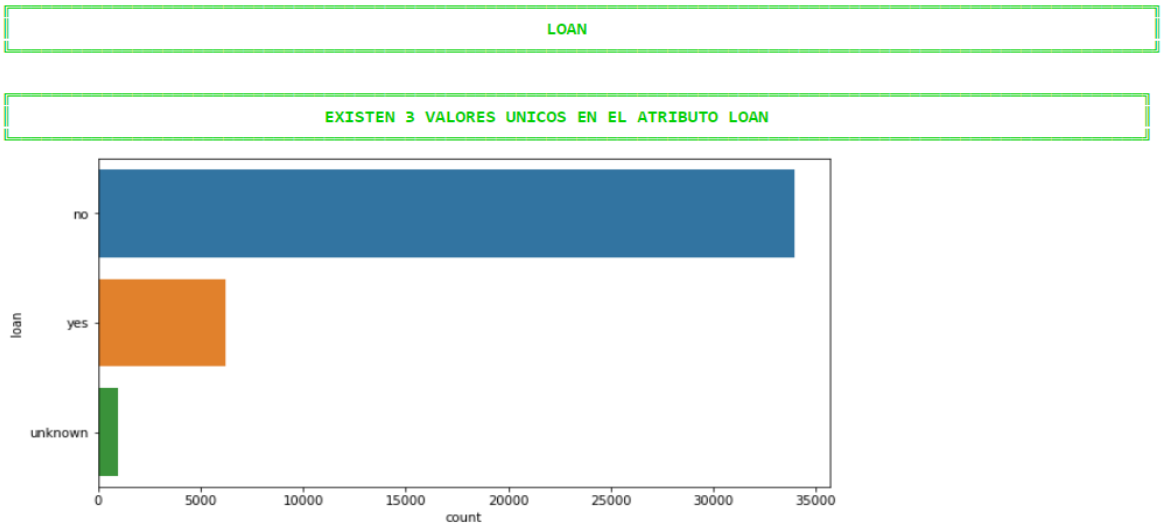


Ilustración 12. Análisis univariado variable Loan (Préstamo personal)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 12 podemos observar que en la variable Loan (Préstamo personal) existen 3 (tres) posibles valores de los cuales el que más sobresale en cantidades es el No (No tiene préstamo personal). Así mismo la gráfica representa que hay un porcentaje considerable de datos desconocidos (Unknown) los cuales se deberán tratar en la próxima fase de CRISP-DM.

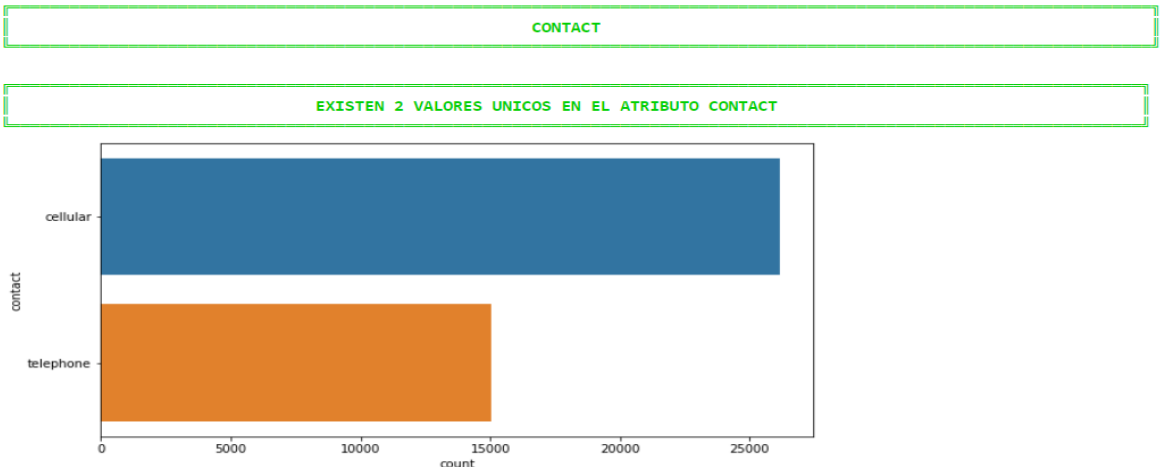


Ilustración 13. Análisis univariado variable Contact (Contacto)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 13 podemos observar que en la variable Contact (Contacto) existen 2 (dos) posibles valores de los cuales el que más sobresale en cantidades es la comunicación por celular.

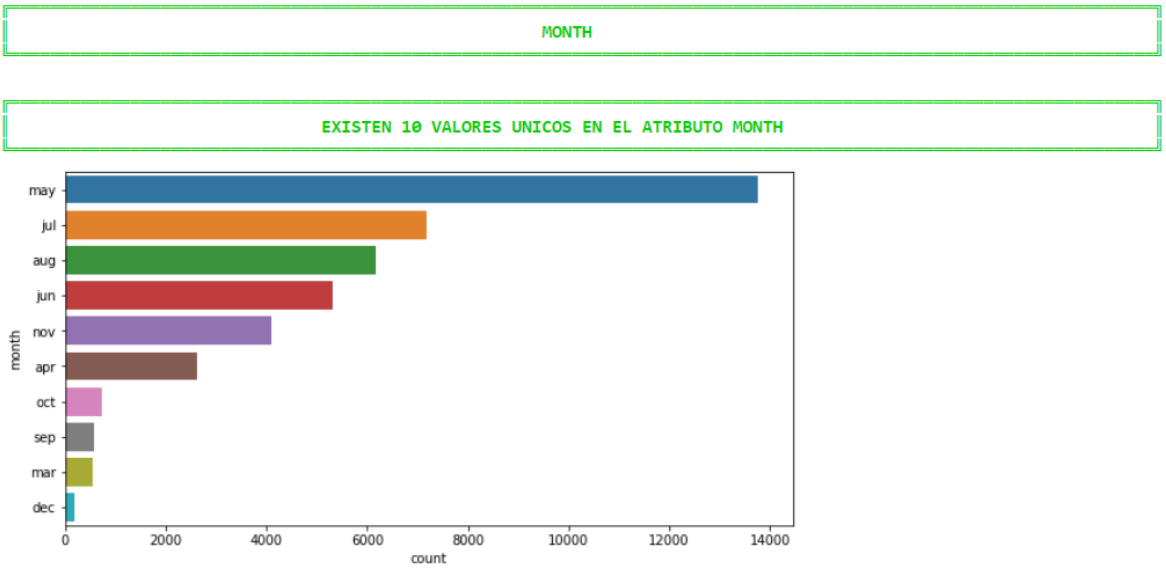


Ilustración 14. Análisis univariado variable Month (Mes)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 14 podemos observar que en la variable Month (Mes último contacto) existen 10 (diez) meses de los cuales el que más sobresale en cantidades sobre los demás es el mes de mayo. Un punto muy importante en esta gráfica es que no se tomaron datos para los meses de enero y febrero.

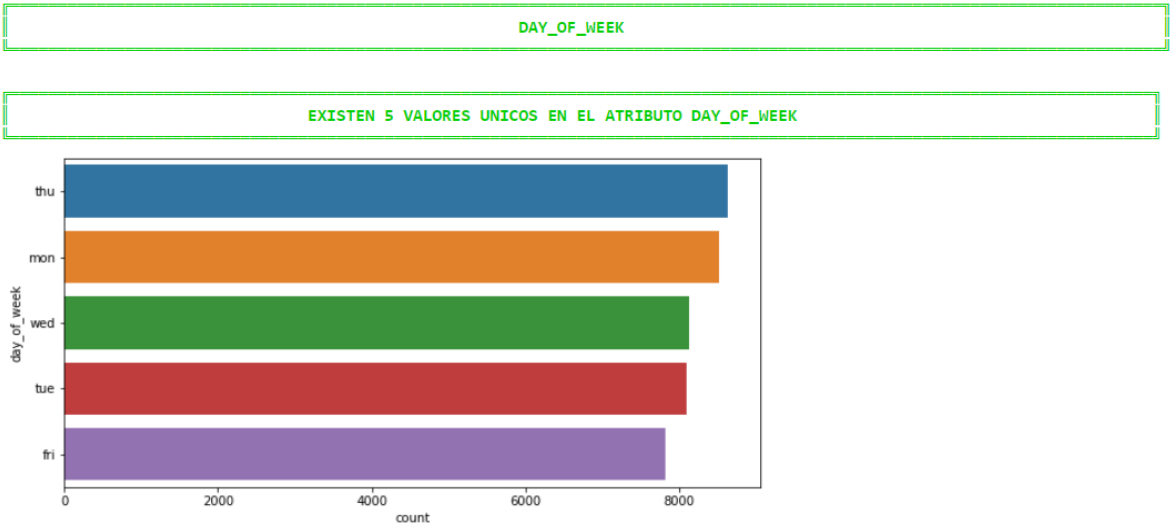


Ilustración 15. Análisis univariado variable Day of Week (Dia de la semana)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 14 podemos observar que en la variable Day of Week (Dia de la semana último contacto) existen 5 (cinco) días que están distribuidos equitativamente. Un punto muy importante en esta gráfica es que no se tomaron datos para los sábados y domingos.

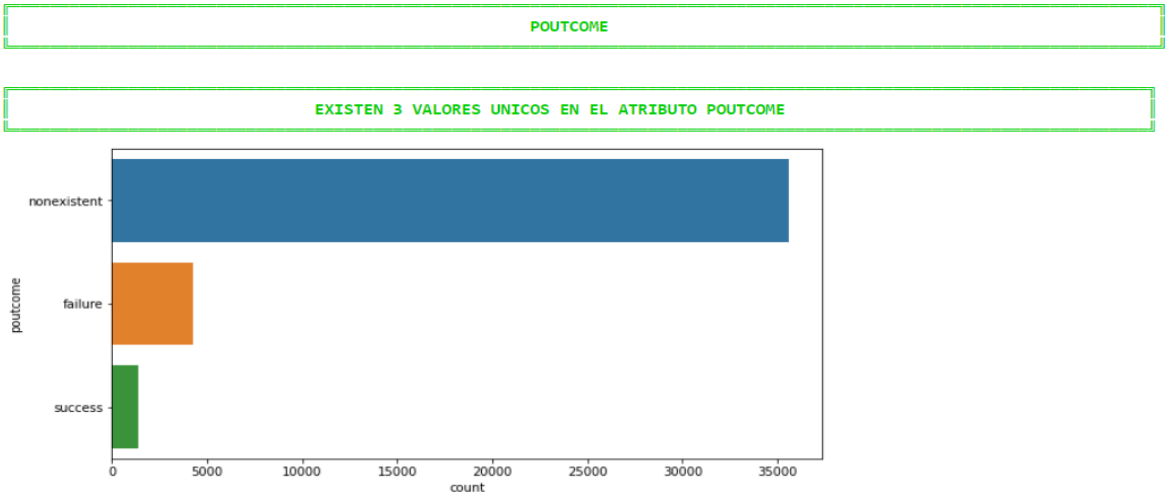


Ilustración 16. Análisis univariado variable Poutcome (Resultado Campaña)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 16 podemos observar que en la variable Poutcome (Resultado de la Campaña) existen 3 (tres) posibles valores de los cuales el que más sobresale en cantidades es el nonexistent (No se tuvo un resultado de la campaña anterior). Se debe tener en cuenta que el valor nonexistent se presenta debido a que el cliente no fue contactado previamente en las campañas ofertadas por la entidad financiera mediante el marketing directo.

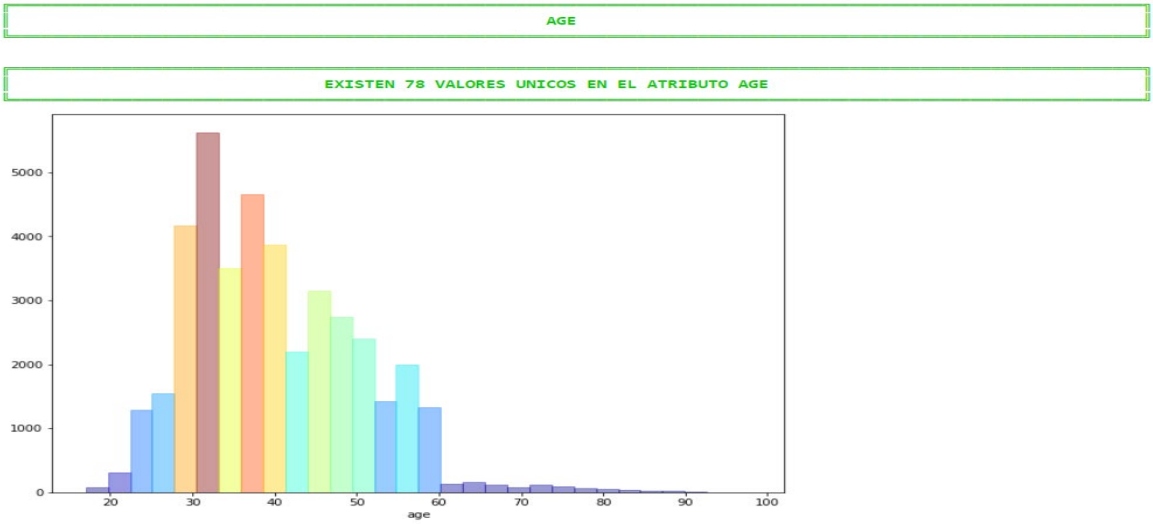


Ilustración 17. Análisis univariado variable Age (Edad)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 17 podemos observar que en la variable Age (Edad) existen 78 (setenta y ocho) edades de los cuales el rango de edad que más sobresale en cantidades se encuentra entre los 30 a 40 años. Así mismo la gráfica representa que hay un porcentaje mínimo hacia las personas mayores a los 60 años.

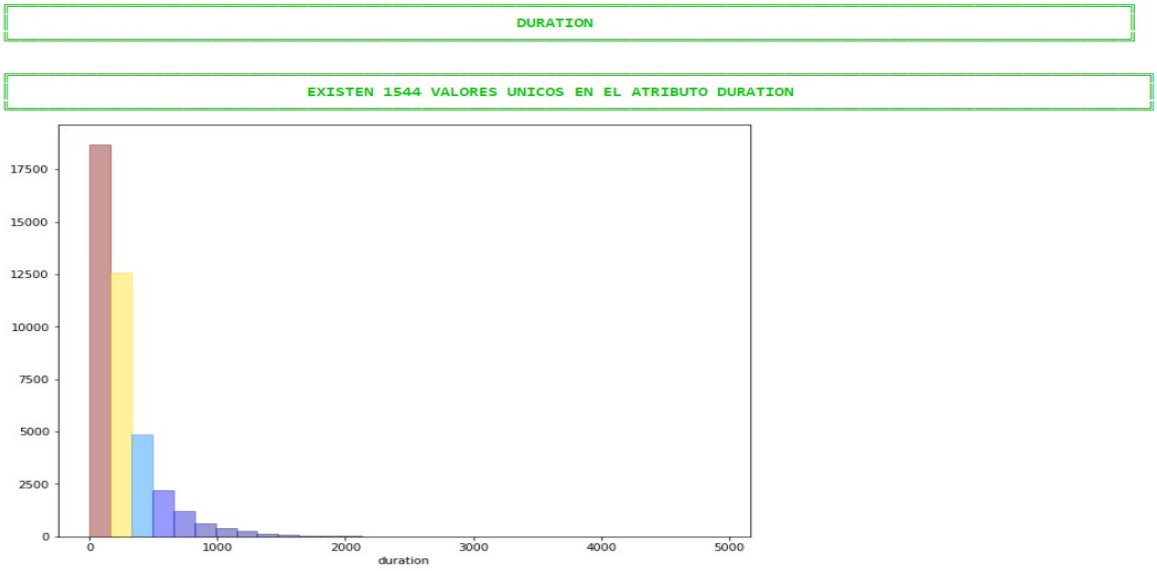


Ilustración 18. Análisis univariado variable Duration (Duración llamada)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 18 podemos observar que en la variable Duration (Duración de la llamada) existen 1.544 (mil quinientos cuarenta y cuatro) posibles valores (segundos) de los cuales el rango de tiempo en segundos que más sobresale en cantidades se encuentra entre 0 a 400 segundos (6 minutos). Así mismo la gráfica representa que hay un porcentaje mínimo de datos después de los 1.000 segundos (16 minutos) los cuales se deberán revisar en la próxima fase de CRISP-DM.

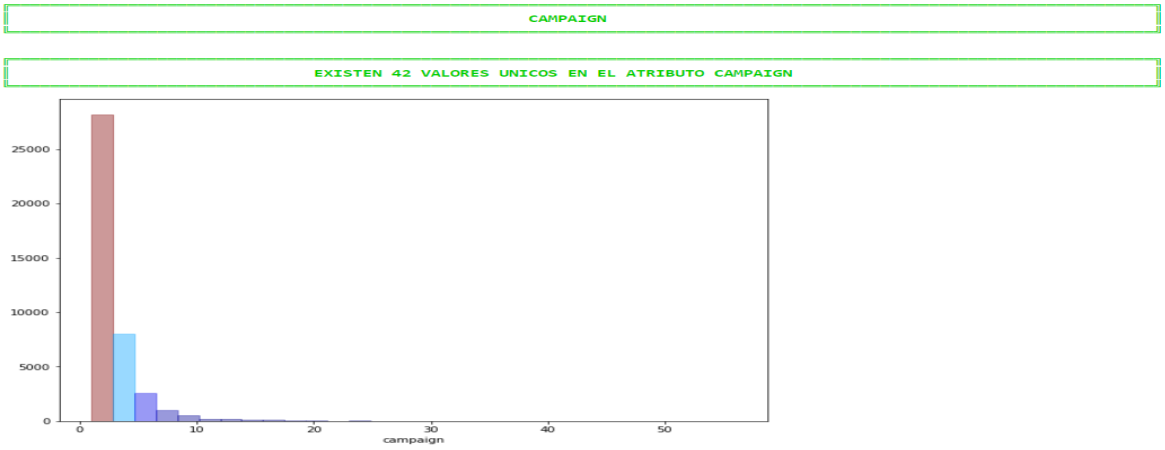


Ilustración 19. Análisis univariado variable Campaign (No. Contactos campaña vigente)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 19 podemos observar que en la variable Campaign (No. Contactos Campaña Vigente) existen 42 (cuarenta y dos) posibles valores (cantidad de contactos realizados durante la campaña vigente) de los cuales el rango de llamadas que más sobresale en cantidades se encuentra entre 1 a 4 llamadas realizadas al cliente. Así mismo la gráfica representa que hay un porcentaje mínimo de datos después de las 10 llamadas los cuales se deberán revisar en la próxima fase de CRISP-DM.

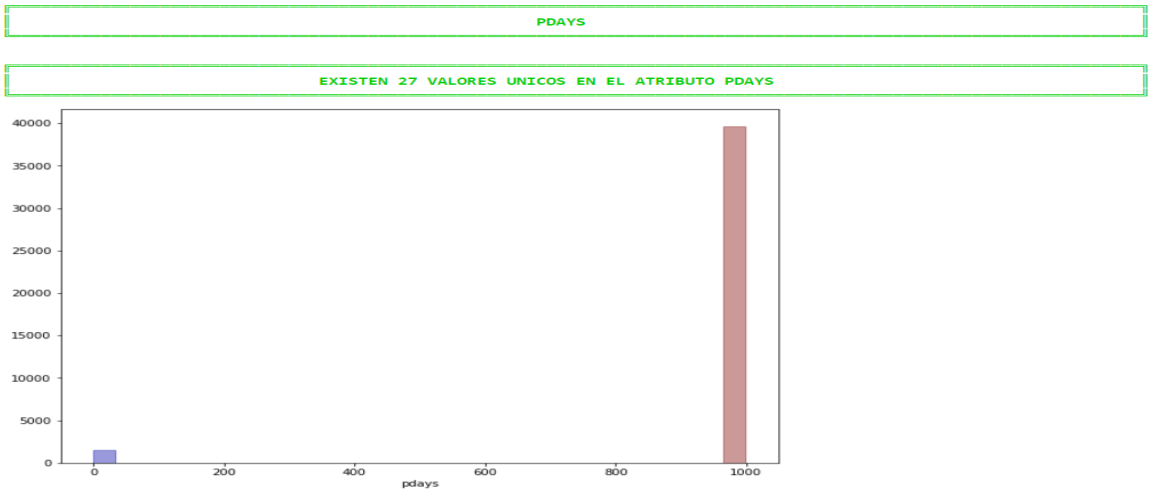


Ilustración 20. Análisis univariado variable Pdays (Diferencia días campaña anterior vs campaña actual)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 19 podemos observar que en la variable Pdays (Diferencia días campaña anterior vs campaña actual) existen 27 (veintisiete) posibles valores (días) de los cuales el rango de días que más sobresale en cantidades se encuentra entre 980 a 1000 días. Se debe tener muy en cuenta que en el atributo Pdays el número 999 representa que el cliente no fue contactado previamente, por lo cual es importante darle un tratamiento especial a este valor en la próxima fase de CRISP-DM.

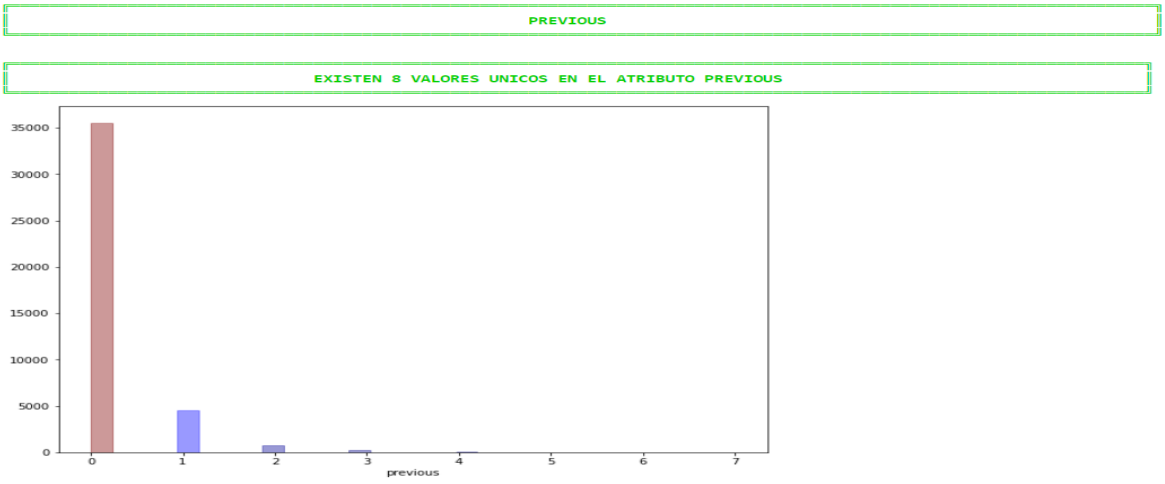


Ilustración 21. Análisis univariado variable Previous (No. contactos campaña anterior)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 21 podemos observar que en la variable Previous (No. Contactos Campaña Anterior) existen 8 (ocho) posibles valores (cantidad de contactos realizados durante la campaña anterior) de los cuales el rango de llamadas que más sobresale en cantidades se encuentra entre 0 a 1 llamada realizada al cliente.

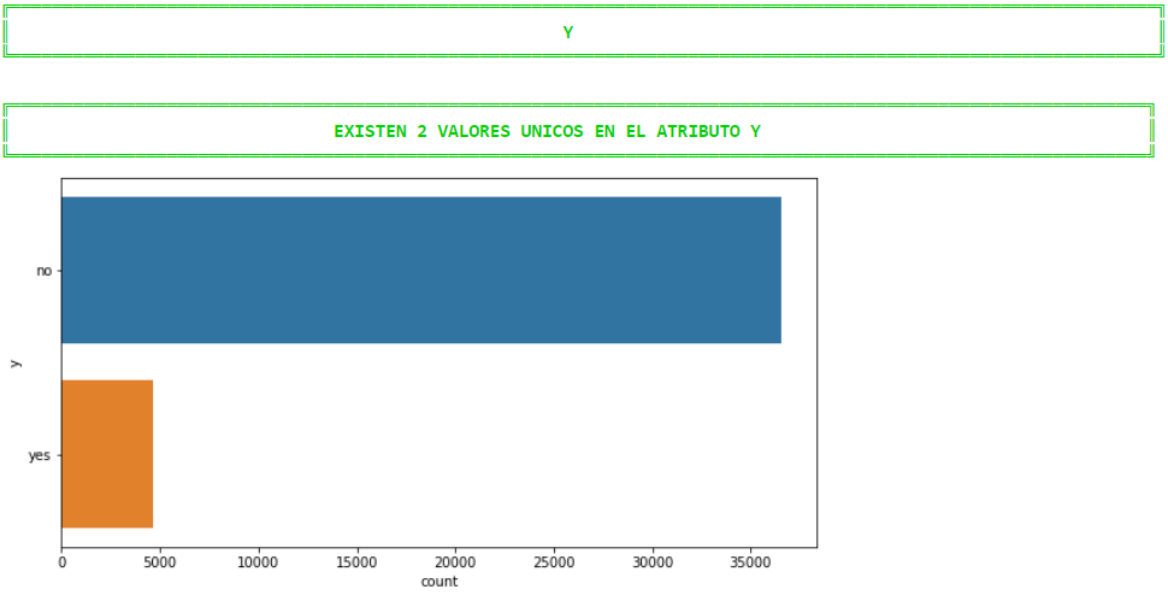


Ilustración 22. Análisis univariado variable Y (Variable Objetivo)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 22 podemos observar que en la variable Y (Objetivo) existen 2 (dos) posibles valores de los cuales el que más sobresale en cantidades es el No (Cliente que no adquirió el producto financiero). Esto quiere decir que nuestro conjunto de datos está muy desbalanceado y se deberán aplicar algunos métodos de balanceo en posteriores etapas para que el modelo pueda realizar buenas predicciones.

7.1.3.2. ANÁLISIS COMPUESTO POR LA VARIABLE OBJETIVO

Procederemos a graficar cada una de las variables del conjunto de datos segmentados por la variable objetivo para ir detectando el comportamiento de los datos que contiene cada una de estas.

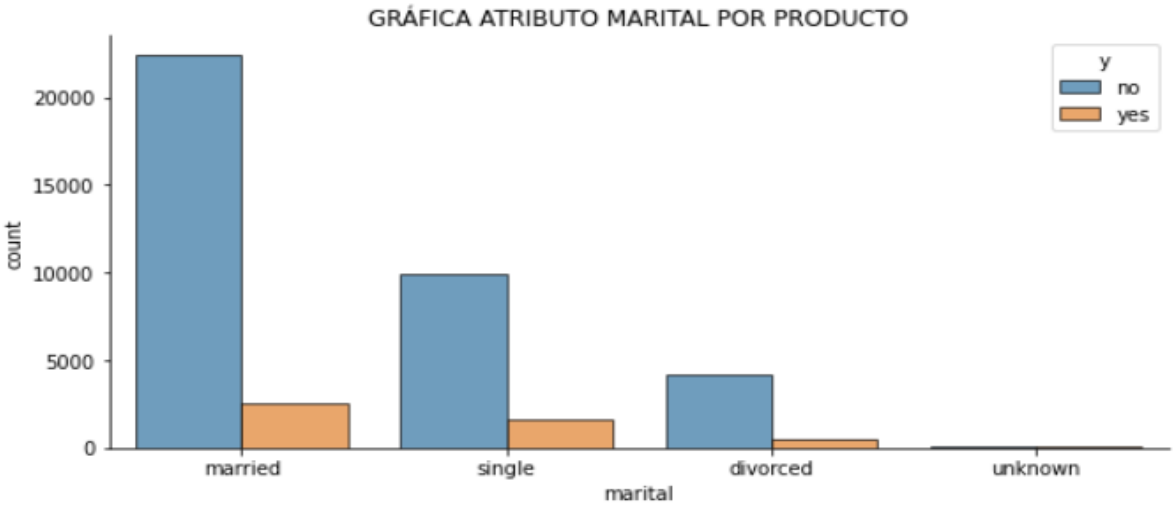


Ilustración 23. Análisis compuesto variable Marital (Estado Civil)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 23 podemos ir observando los diferentes tipos de estado civil que más adquieren el producto financiero ofertado por la entidad bancaria teniendo en cuenta la proporción de información para cada uno de estos. En esta gráfica podemos concluir que los clientes que más adquieren el instrumento financiero se encuentran casados.

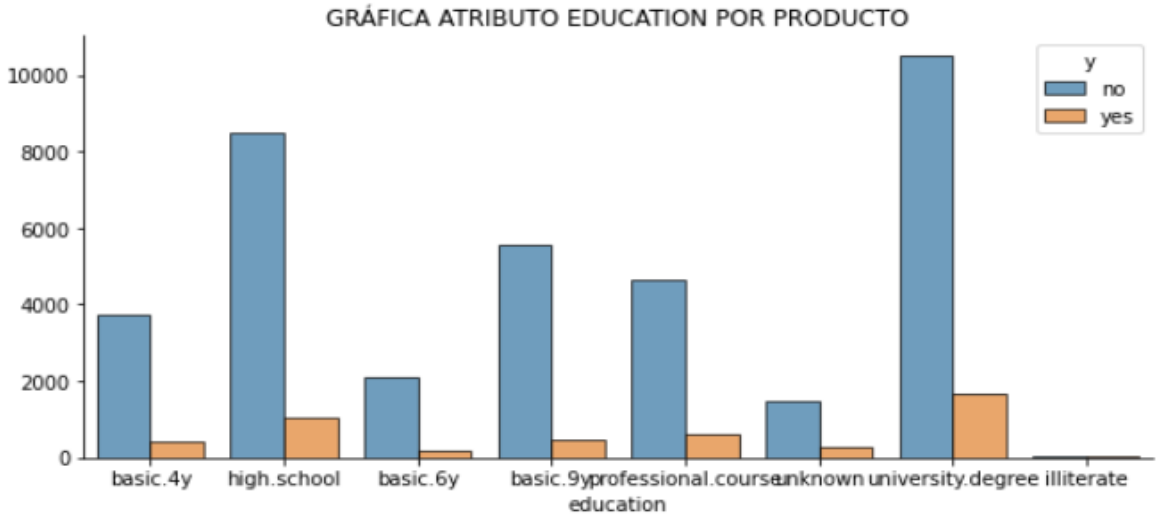


Ilustración 24. Análisis compuesto variable Education (Educación)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 24 podemos ir observando los diferentes niveles de educación que más adquieren el producto financiero ofertado por la entidad bancaria teniendo en cuenta la proporción de información para cada uno de estos. En esta gráfica podemos concluir que los clientes que más adquieren el instrumento financiero son los que tienen un título universitario y finalizado la secundaria.

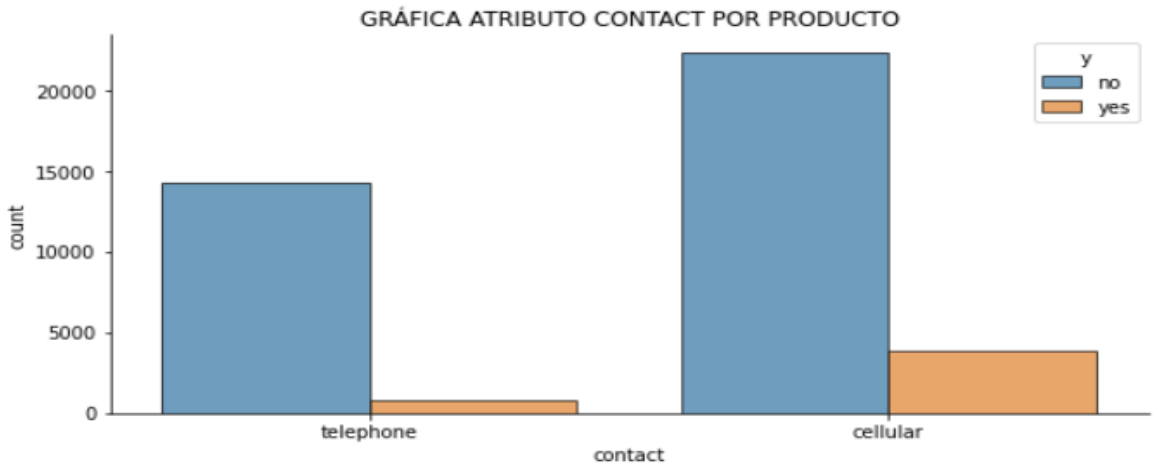


Ilustración 25. Análisis compuesto variable Contact (Contacto)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 25 podemos ir observando los diferentes tipos de comunicación por los cuales más adquieren el producto financiero ofertado por la entidad bancaria teniendo en cuenta la proporción de información para cada uno de estos. En esta gráfica podemos concluir sin duda alguna que los clientes que más adquieren el instrumento financiero son los que reciben una comunicación directa por celular.

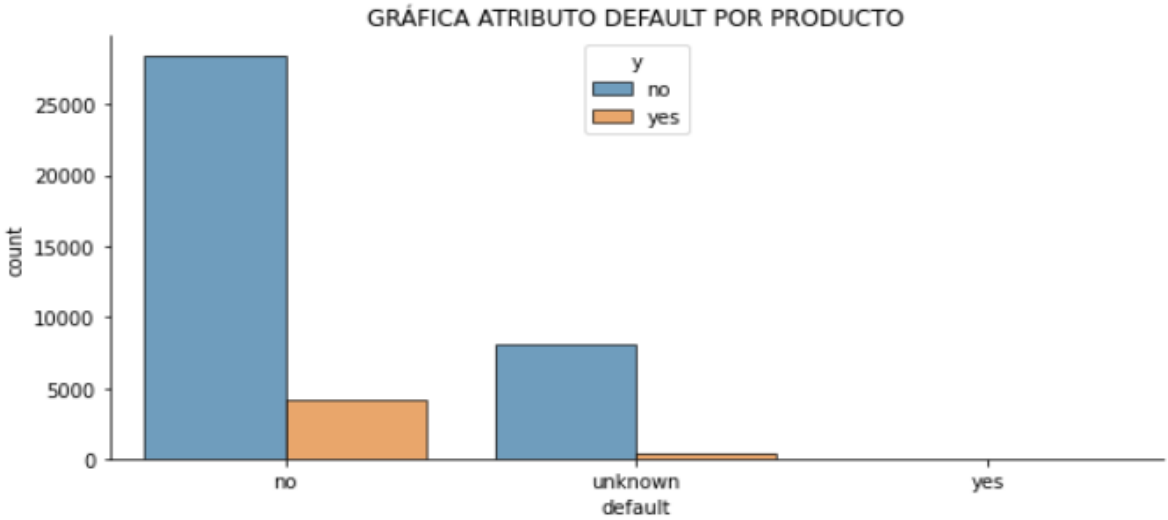


Ilustración 26. Análisis compuesto variable Default (Crédito en mora)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 26 podemos ir observando los clientes que tienen o no un crédito en mora con la entidad financiera. En esta gráfica podemos concluir sin duda alguna que los clientes que más adquieren el instrumento financiero son los que no tienen o no han tenido un crédito en mora con la entidad financiera.

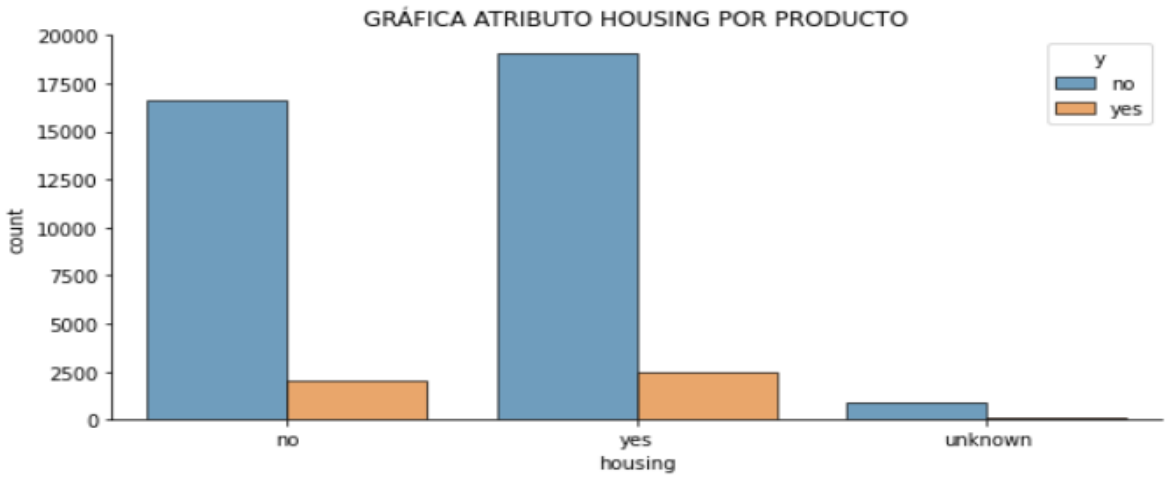


Ilustración 27. Análisis compuesto variable Housing (Crédito de vivienda)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 27 podemos ir observando los clientes que tienen o no un crédito de vivienda con la entidad financiera. En esta gráfica podemos concluir que tanto los clientes que han tenido y no han tenido un crédito de vivienda son los que adquieren el instrumento financiero ofertado en igual proporción.

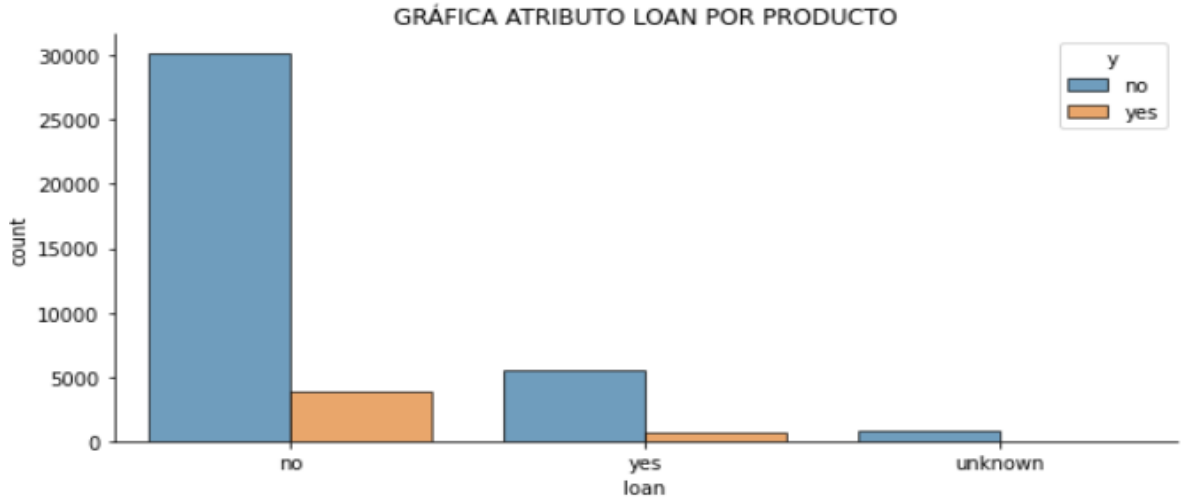


Ilustración 28. Análisis compuesto variable Loan (Préstamo personal)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 28 podemos ir observando los clientes que tienen o no un préstamo personal con la entidad financiera. En esta gráfica podemos concluir que los clientes que no han tenido un préstamo personal con la entidad bancaria son los que más adquieren el instrumento financiero ofertado.

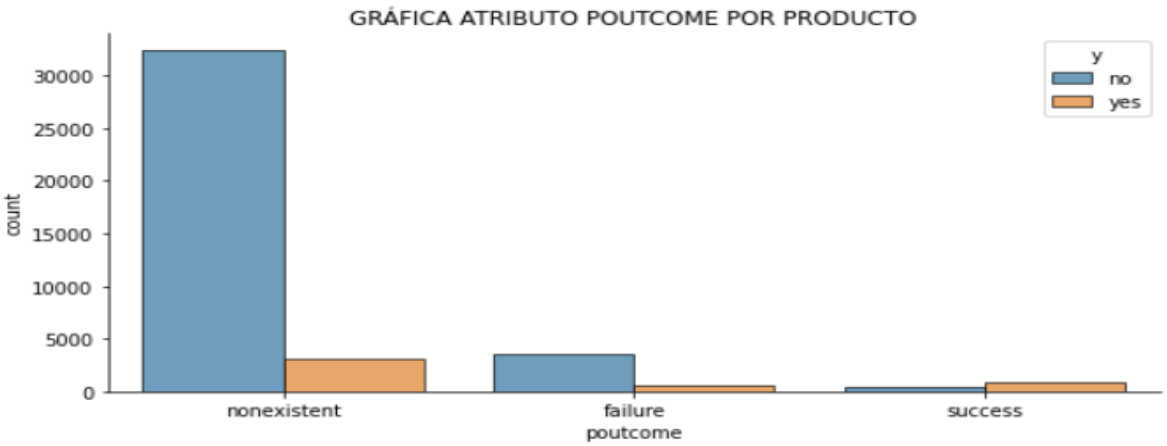


Ilustración 29. Análisis compuesto variable Poutcome (Resultado Campaña)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 29 podemos ir observando los diferentes resultados de la campaña anterior que más adquieren el producto financiero ofertado por la entidad bancaria teniendo en cuenta la proporción de información para cada uno de estos. Se debe tener en cuenta que el valor nonexistent se presenta debido a que el cliente no fue contactado previamente en las campañas ofertadas por la entidad financiera mediante el marketing directo, esto quiere decir que el cliente obtuvo el producto financiero de manera diferente a una llamada telefónica.

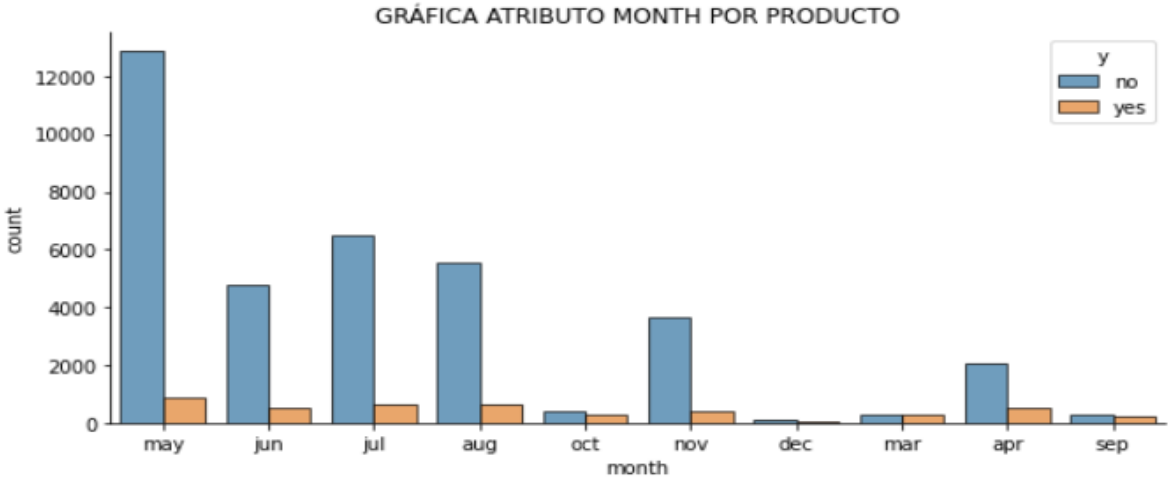


Ilustración 30. Análisis compuesto variable Month (Mes)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 30 podemos ir observando los diferentes meses donde más adquieren el producto financiero ofertado por la entidad bancaria teniendo en cuenta la proporción de información para cada uno de estos. En esta gráfica podemos concluir que los clientes que más adquieren el instrumento financiero se dan entre los meses de mayo a agosto.

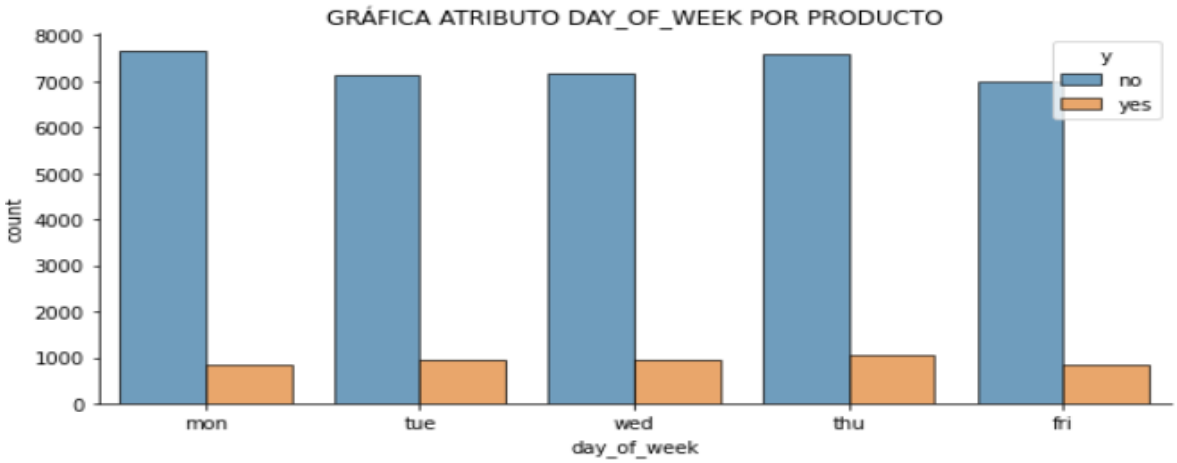


Ilustración 31. Análisis compuesto variable Day of Week (Día de la semana)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 31 podemos ir observando los diferentes días de la semana donde más adquieren el producto financiero ofertado por la entidad bancaria teniendo en cuenta la proporción de información para cada uno de estos. En esta gráfica podemos concluir que los clientes que más adquieren el instrumento financiero se dan en igual proporción de lunes a viernes.

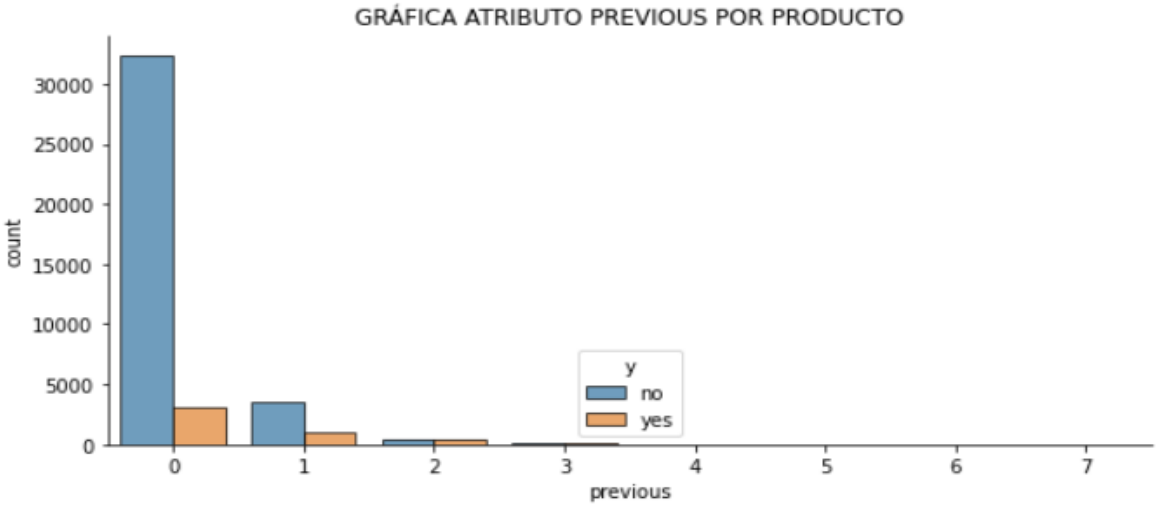


Ilustración 32. Análisis compuesto variable Previous (No. Contactos campaña anterior)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 32 podemos ir observando la cantidad de llamadas que se realizaron en la campaña anterior a un cliente para ofertar el producto financiero. En esta gráfica podemos concluir que los clientes que más adquieren el instrumento financiero son a los cuales no se contactó (esto quiere decir, que el instrumento financiero lo adquirieron por otro medio diferente al marketing directo) y así mismo a los que se les contactó por una única vez.

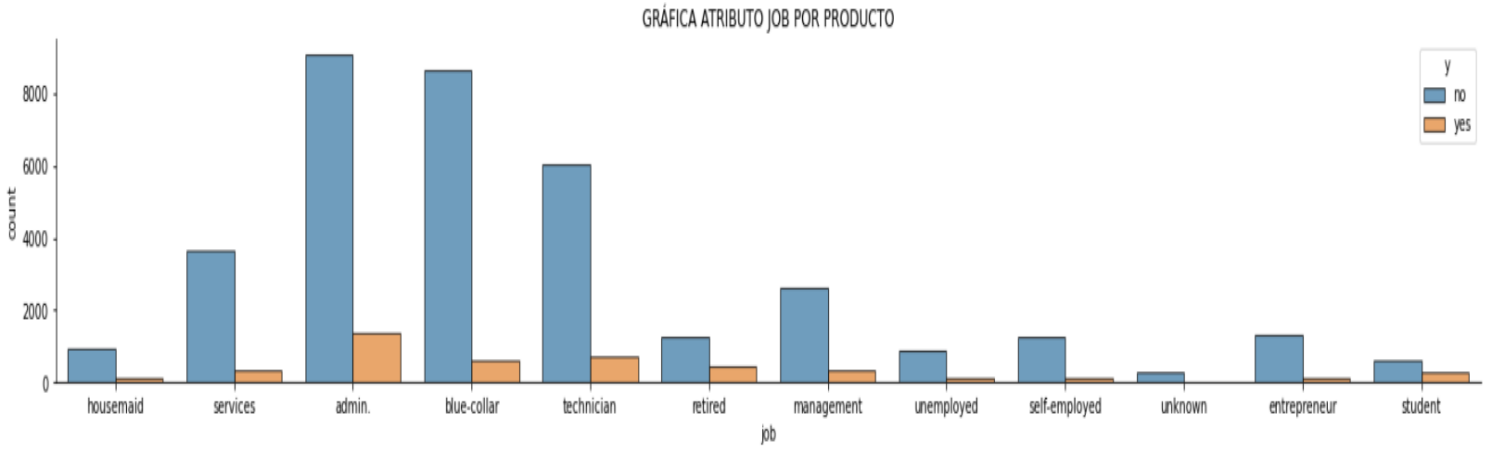


Ilustración 33. Análisis compuesto variable Job (Trabajo)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 33 podemos ir observando los diferentes tipos de trabajo que más adquieren el producto financiero ofertado por la entidad bancaria teniendo en cuenta la proporción de información para cada uno de estos. En esta gráfica podemos concluir que los clientes que más adquieren el instrumento financiero son los que tienen un trabajo como administrativo, los obreros y los técnicos.

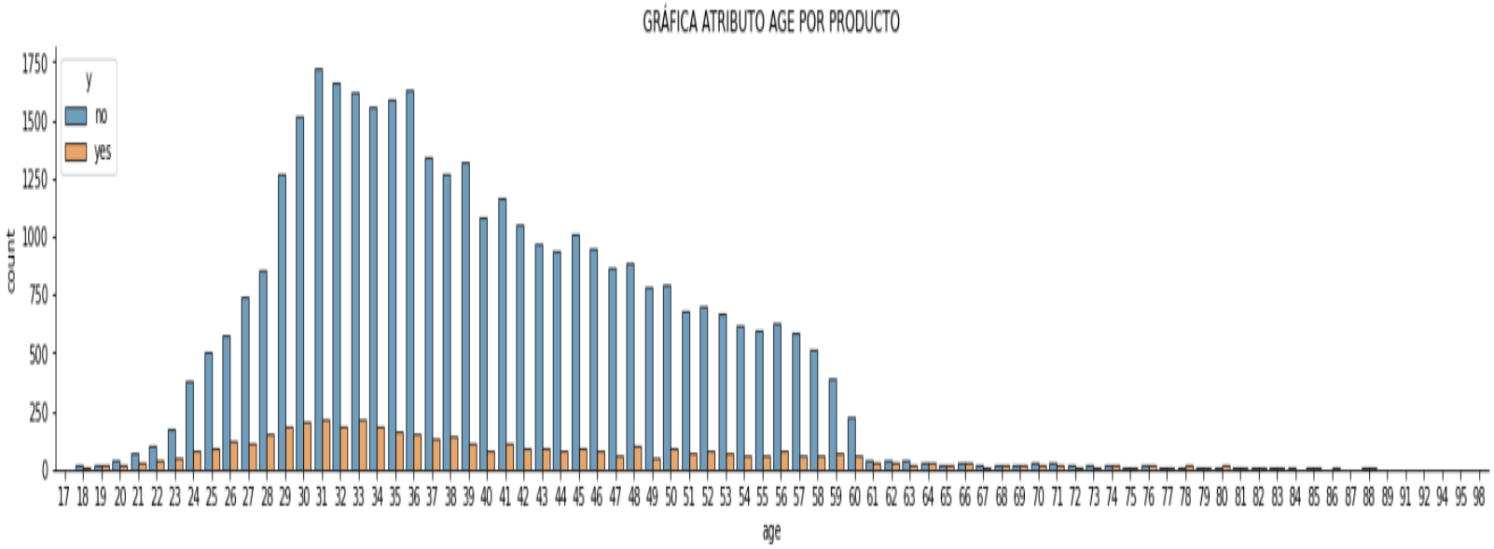


Ilustración 34. Análisis compuesto variable Age (Edad)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 34 podemos ir observando las diferentes edades que más adquieren el producto financiero ofertado por la entidad bancaria teniendo en cuenta la proporción de información para cada uno de estos. En esta gráfica podemos concluir que los clientes que más adquieren el instrumento financiero se encuentran entre las edades de 26 a 41 años.

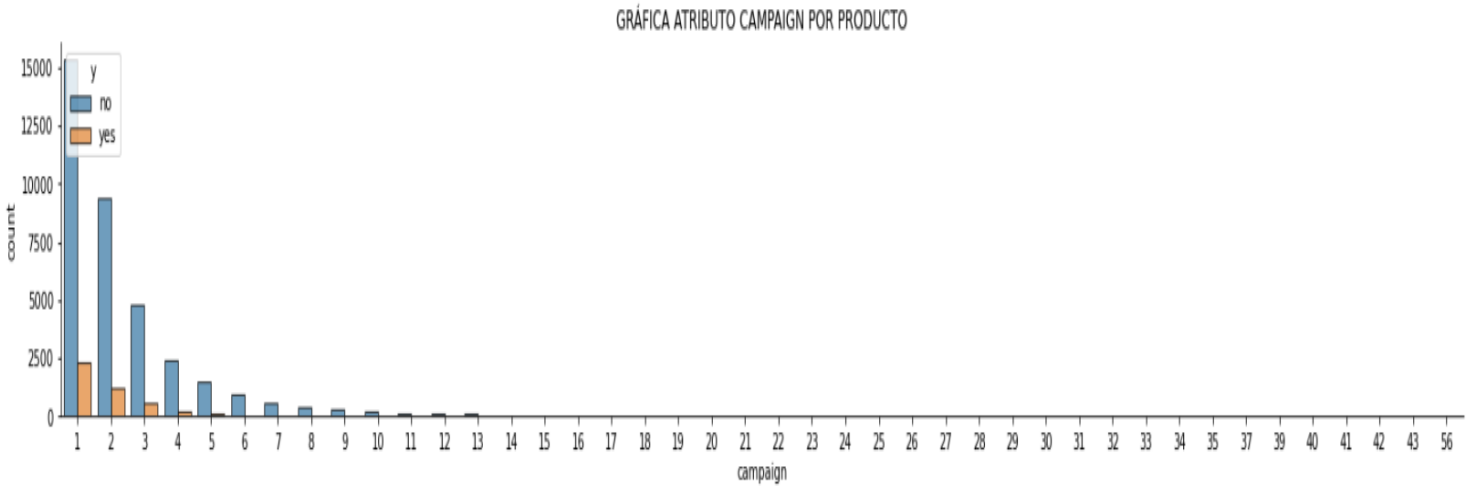


Ilustración 35. Análisis compuesto variable Campaign (Contactos Campaña vigente)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 35 podemos ir observando la cantidad de llamadas que se realizaron en la campaña vigente a un cliente para ofertar el producto financiero. En esta gráfica podemos concluir que los clientes que más adquieren el instrumento financiero son a los cuales se les contacta por una, dos o tres ocasiones en la misma campaña vigente.

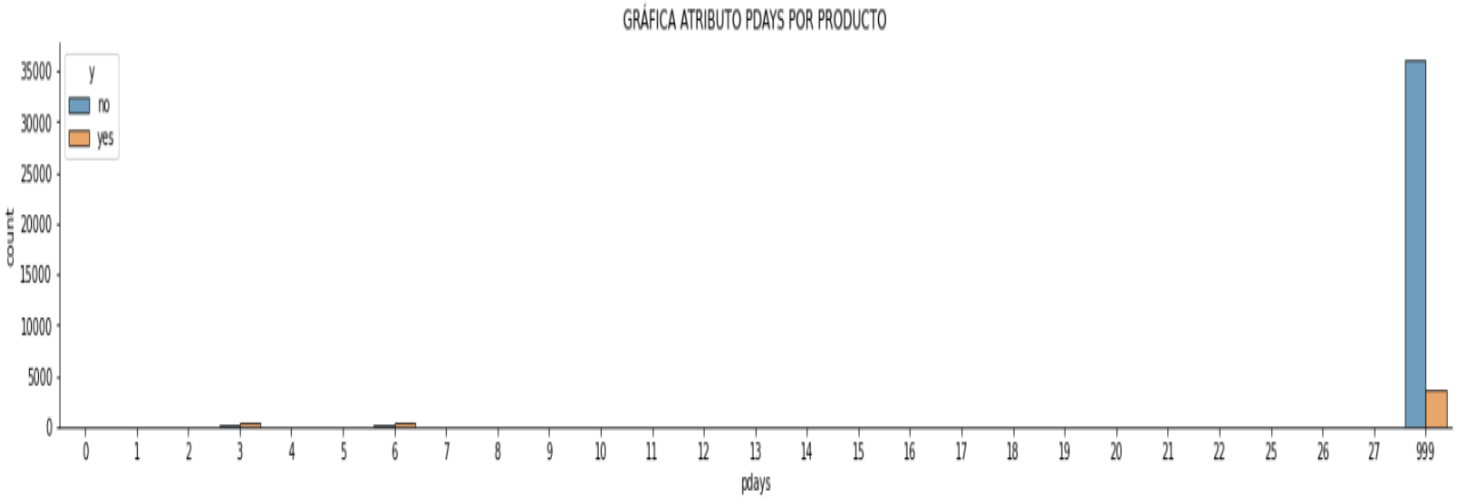


Ilustración 36. Análisis compuesto variable Pdays (Diferencia días campaña anterior vs campaña actual)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 36 podemos ir observando la diferencia de días de la campaña anterior vs campaña actual. Se debe tener muy en cuenta que en el atributo Pdays el número 999 representa que el cliente no fue contactado previamente, por lo cual es importante darle un tratamiento especial a este valor en la próxima fase de CRISP-DM.

7.1.3.3. ANÁLISIS DISTRIBUCIÓN VARIABLES NUMÉRICAS

Procederemos a graficar cada una de las variables numéricas del conjunto de datos para constatar si estas tienen una distribución normal de sus valores.

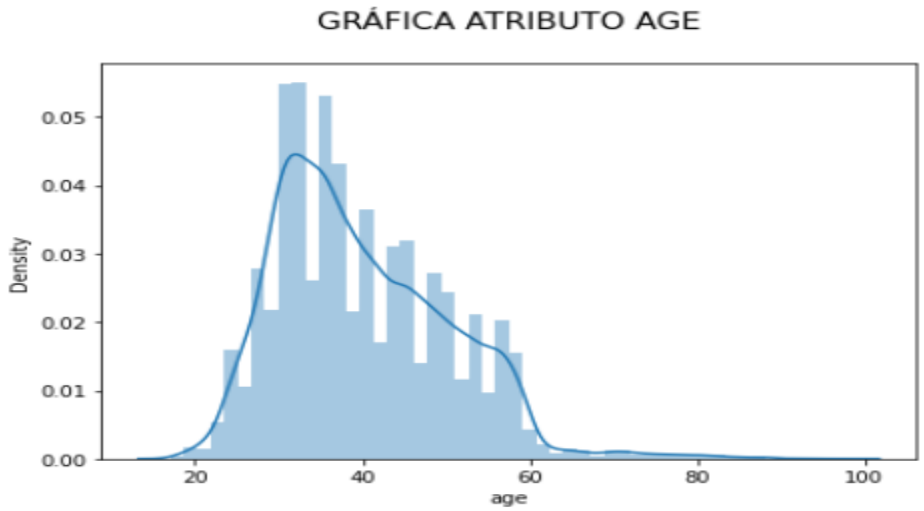


Ilustración 37. Análisis distribución variable Age (Edad)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 37 se puede ver el comportamiento de la variable age (edad) de acuerdo con sus datos contenidos, donde se observa que alcanza a existir una distribución normal de dicha gráfica aunque esta tienda a tener una cola demasiada sesgada hacia el lado derecho.

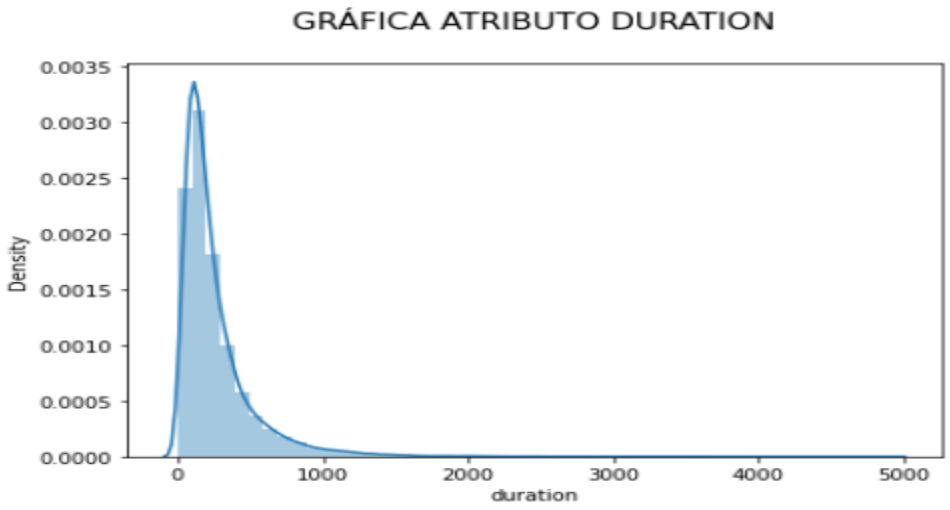


Ilustración 38. Análisis distribución variable Duration (Duración)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 38 se puede ver el comportamiento de la variable duration (duración de una llamada con el cliente) de acuerdo con sus datos contenidos, donde se observa claramente que existe una distribución normal de dicha gráfica aunque esta tienda a tener una cola demasiada sesgada hacia el lado derecho.

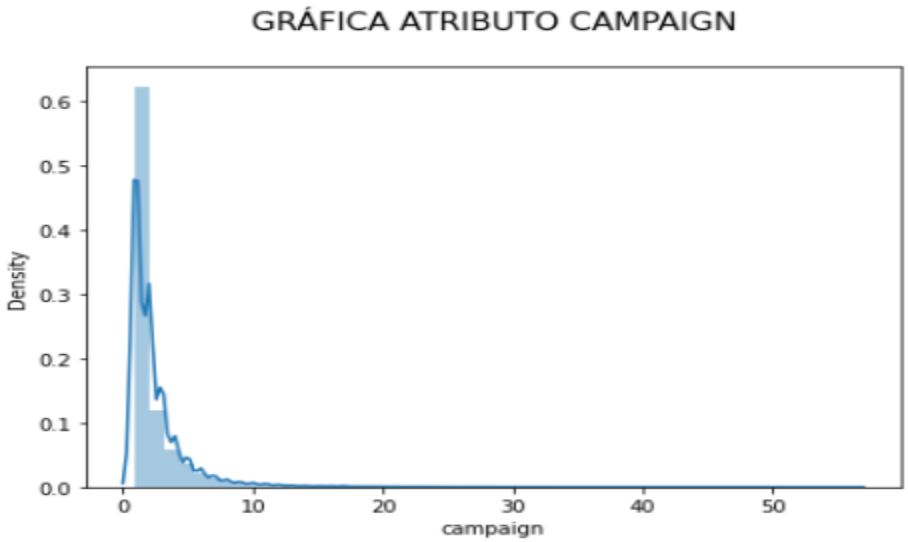


Ilustración 39. Análisis distribución variable Campaign (No. contactos campaña vigente)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 39 se puede ver el comportamiento de la variable Campaign (No. contactos campaña vigente) de acuerdo con sus datos contenidos, donde se observa que no alcanza a existir una distribución normal sino más bien atípica. De igual forma esta tiende a tener una cola demasiada sesgada hacia el lado derecho de los datos.

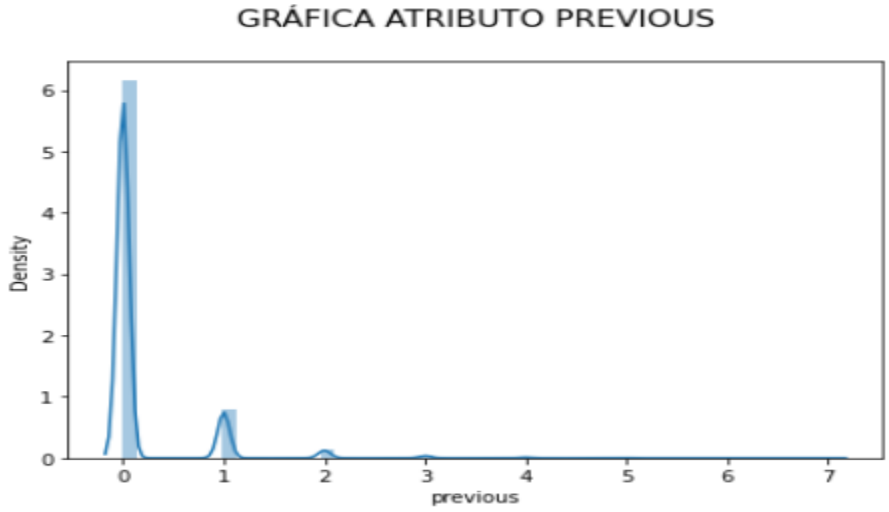


Ilustración 40. Análisis distribución variable Previous (No. contactos campaña anterior)
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 40 se puede ver el comportamiento de la variable Previous (No. contactos campaña anterior) de acuerdo con sus datos contenidos, donde se observa que no existe una distribución normal sino más bien atípica.

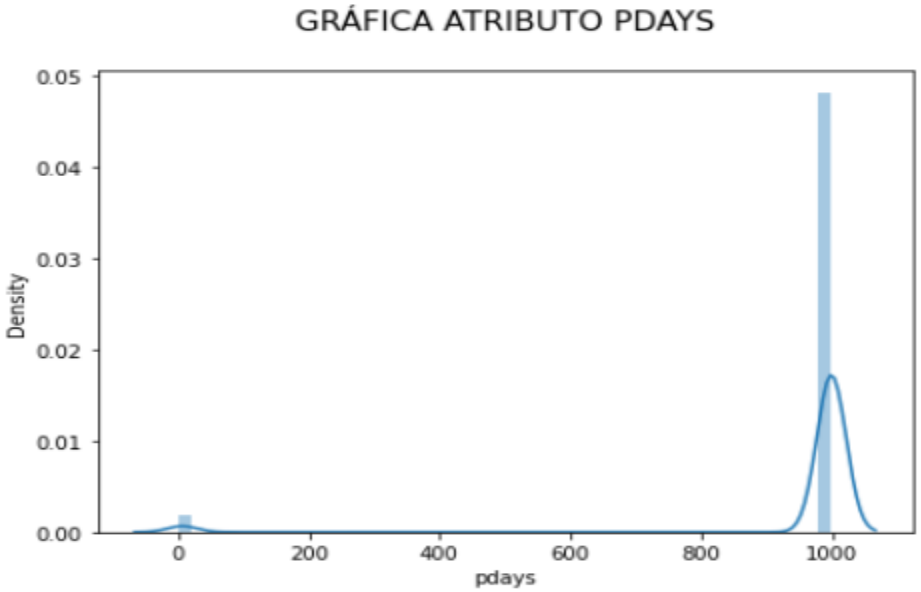


Ilustración 41. Análisis distribución variable Pdays (Diferencia días campaña anterior vs campaña actual) Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 41 se puede ver el comportamiento de la variable Pdays (Diferencia días campaña anterior vs campaña actual) de acuerdo con sus datos contenidos, donde se observa que alcanza a existir una distribución normal al lado derecho de la gráfica aunque esta tienda a tener una cola demasiada sesgada hacia el lado izquierdo.

Se debe tener muy en cuenta que en el atributo Pdays el número 999 representa que el cliente no fue contactado previamente, por lo cual es importante darle un tratamiento especial a este valor en la próxima fase de CRISP-DM.

7.1.3.4. ANÁLISIS DATOS ATÍPICOS VARIABLES NUMÉRICAS

Procederemos a graficar cada una de las variables numéricas del conjunto de datos para detectar posibles datos atípicos en cada una de estas.

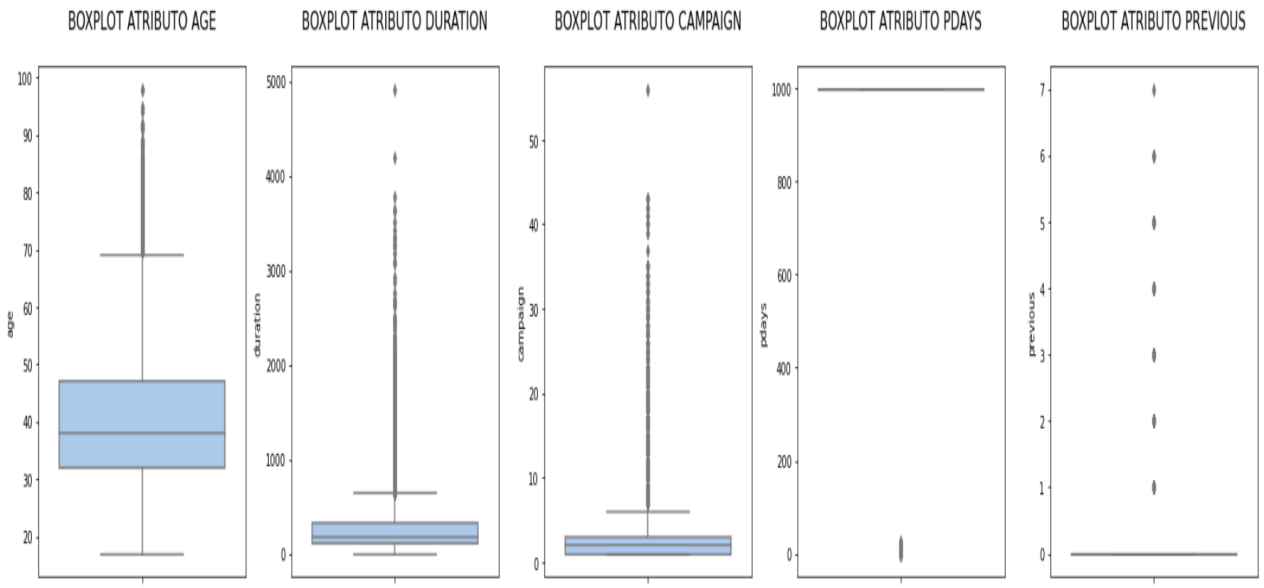


Ilustración 42. Análisis datos atípicos variables numéricas
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 42 podemos observar que para las variables age (edad), duration (duración) y campaign (número de llamadas en la campaña vigente) existe mayor concentración de posibles valores inusuales encima del tercer cuartil y máximo valor.

Para la variable previous (número de llamadas en la campaña anterior) la gráfica aplicada no es la más aconsejable por la reducida cantidad de valores únicos que esta contiene, adicional la gran mayoría de los datos están concentrados en un único valor.

Por otra parte, para la variable pdays (diferencia días campaña anterior vs campaña actual) es muy importante darle un tratamiento especial a este valor en la próxima fase de CRISP-DM, para tener una visión más real de los datos que contiene esta variable.

7.1.3.5. ANÁLISIS DATOS ATÍPICOS POR LA VARIABLE OBJETIVO

Procederemos a graficar cada una de las variables numéricas del conjunto de datos segmentados por la variable objetivo para ir detectando posibles datos atípicos en cada una de estas.

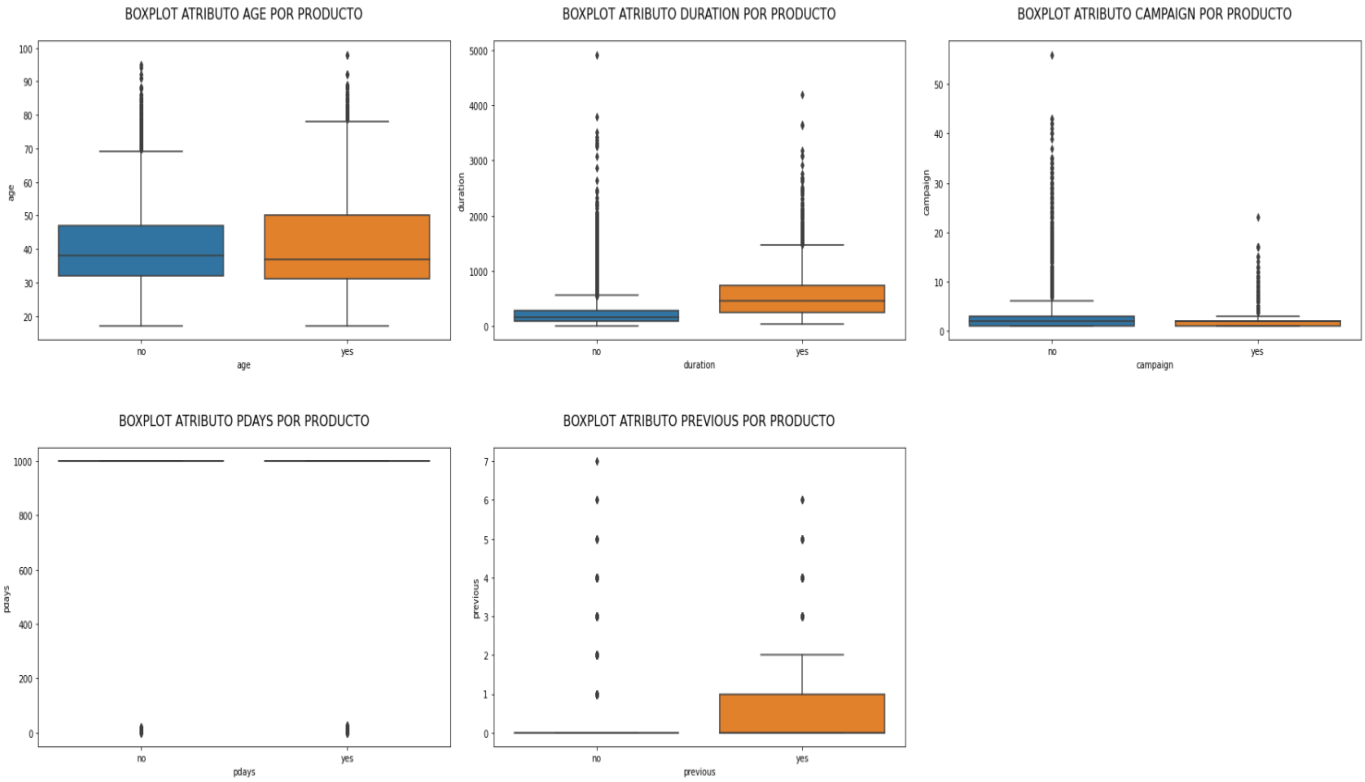


Ilustración 43. Análisis datos atípicos variables numéricas segmentada por la variable objetivo
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 43 al igual que la ilustración anterior podemos observar que para las variables age (edad), duration (duración) y campaign (número de llamadas en la campaña vigente) existe mayor concentración de posibles valores inusuales encima del tercer cuartil y máximo valor.

Para el caso de la variable previous (número de llamadas en la campaña anterior) esta gráfica refleja una mejor visión de los valores atípicos que existen allí. Al igual que en las primeras tres variables existe mayor concentración de posibles valores inusuales encima del tercer cuartil y máximo valor.

7.1.3.6. ANÁLISIS CORRELACIÓN VARIABLES NUMÉRICAS

Por último, pero no menos importante, graficamos una matriz de correlación que nos permita identificar posibles correlaciones entre las variables presentes en este conjunto de datos.

	age	duration	campaign	pdays	previous
age	1.000000	-0.000866	0.004594	-0.034369	0.024365
duration	-0.000866	1.000000	-0.071699	-0.047577	0.020640
campaign	0.004594	-0.071699	1.000000	0.052584	-0.079141
pdays	-0.034369	-0.047577	0.052584	1.000000	-0.587514
previous	0.024365	0.020640	-0.079141	-0.587514	1.000000

Ilustración 44. Tabla de correlación variables numéricas
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

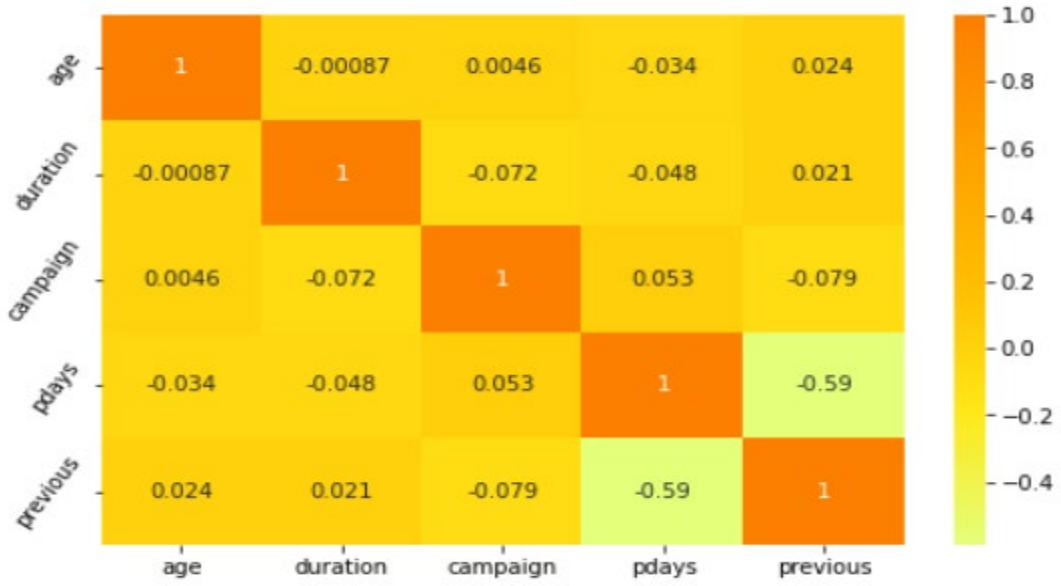


Ilustración 45. Matriz de correlación variables numéricas
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 44 y 45 uno de los puntos más notables en la matriz de correlación tiene que ver con las variables pdays y previous las cuales presentan correlaciones moderadas que se deben tener muy en cuenta en próximas etapas de la metodología CRISP-DM. Cabe resaltar que a la variable pdays se le deberá aplicar un tratamiento especial a los datos que contiene valores 999.

7.1.4. CALIDAD DE LOS DATOS

Luego de realizar los análisis exploratorios e identificar anomalías en algunas de las variables (datos desconocidos y erróneos, distribuciones anormales, datos atípicos, no

agrupamientos de valores contenidas en algunas variables, variables innecesarias, tipos de datos que deben ser categorizados, etc), se procederá a aplicar el preprocesamiento de los datos sobre el conjunto de datos en la próxima etapa de la metodología CRISP-DM (Preparación de los datos).

7.2. PREPARACIÓN DE LOS DATOS

7.2.1. SELECCIÓN DE LOS DATOS

En el conjunto de datos se pudo observar que en la fase anterior (exploratoria) hay una serie de casos que se deben evaluar a nivel de atributos y registros con el objetivo de contar únicamente con los datos necesarios que se le suministrarán al modelo final (en la próxima fase de CRISP-DM).

Como punto de partida para la selección de los datos se iniciará imputando algunos registros desconocidos que están presentes en algunas variables del conjunto de datos, así mismo se imputaran aquellos registros que en gran medida tienden a ser datos atípicos, se eliminarán atributos que no son representativos o no generan gran impacto sobre la variable predictora y aquellos atributos que tengan una correlación alta sobre las variables independientes o con la variable objetivo, se implementará la codificación de etiquetas sobre las variables categóricas y se renombrarán los atributos para que sean más entendibles.

Por último, pero no menos importante se realizará nuevamente un análisis exploratorio sobre el conjunto de datos cuando se haya realizado el preprocesamiento de los datos (en esta fase), con el objetivo de visualizar y observar la diferencia de los datos en limpio con respecto a la fase anterior y así mismo identificar con qué datos y variables se trabajará en la próxima fase de CRISP-DM.

7.2.2. LIMPIEZA DE LOS DATOS

7.2.2.1. IMPUTACIÓN A NIVEL DE REGISTROS

Se imputarán los registros que contengan el valor **unknown** (desconocidos) en cada una de las variables del conjunto de datos donde esté presente y que no conlleve a

eliminar un volumen o porcentaje alto de registros, con el objetivo de tratar de mantener en lo posible la mayor cantidad de registros en este conjunto de datos.

En la tabla 7 se mostrará el resumen de imputación de datos que se realizó sobre algunas variables del conjunto de datos.

Tabla 7. Registros imputados atributos categóricos

ATRIBUTO	VALOR A ELIMINAR	CANTIDAD ELIMINADA
Job	unknown	330
Marital	unknown	71
Education	unknown	1.596
	illiterate	18
Housing	unknown	946

Debido a que la variable **default** (crédito en mora) contiene un porcentaje alto de datos (8.597) desconocidos (unknown), se distribuirán de acuerdo con la información analizada en el conjunto de datos.

En este caso los registros que contengan el valor **unknown** (desconocido) y adicional hayan adquirido el producto financiero (en la variable objetivo corresponde a ***y = yes***) se actualizarán con el valor **yes** en la variable **default** (crédito en mora). La cantidad de registros en este caso está alrededor de 398 registros.

Por otra parte los registros que contengan el valor **unknown** (desconocido) y adicional no hayan adquirido el producto financiero (en la variable objetivo corresponde a ***y = no***) se actualizarán con el valor **no** en la variable **default** (crédito en mora). La cantidad de registros en este caso está alrededor de 7.352 registros.

Todo esto se realizó con el objetivo de no tener presente el valor **unknown** (desconocido) en cada una de las variables del conjunto de datos.

En la variable **Poutcome** (Resultado de la campaña anterior) existen valores con el nombre de **nonexistent** (Representa que no existe información sobre algún resultado de la campaña anterior (satisfactorio o fallido)) y debido a que contiene un porcentaje alto de datos (33.050) **nonexistent**, se distribuirá este valor de acuerdo con la información analizada en el conjunto de datos.

En este caso los registros que contengan el valor **nonexistent** y adicional hayan adquirido el producto financiero (en la variable objetivo corresponde a ***y = yes***) se actualizarán con el valor **success** en la variable **Poutcome** (Resultado de la campaña anterior). La cantidad de registros en este caso está alrededor de 2.919 registros.

Por otra parte los registros que contengan el valor **nonexistent** y adicional no hayan adquirido el producto financiero (en la variable objetivo corresponde a ***y = no***) se actualizarán con el valor **failure** en la variable **Poutcome** (Resultado de la campaña anterior). La cantidad de registros en este caso está alrededor de 30.131 registros.

También fue muy importante comprender que cuando un estado de la campaña fue **success** (exitoso) el cliente si o si tuvo que haber adquirido el producto ofrecido, caso contrario, cuando está presente un resultado **failure** (fallido) el cliente definitivamente no tuvo que haber adquirido el producto ofrecido. Con base a esta información se decidió corregir ese estado en el atributo **Poutcome** (Resultado campaña anterior).

Todo esto se realizó con el objetivo de no tener presente el valor **nonexistent** en esta variable del conjunto de datos.

Dando alcance a la exploración de los datos realizados en la fase anterior donde se identificaron posibles registros atípicos en la ilustración 42 y la ilustración 43 del presente documento, en la siguiente tabla 8 se mostrará el resumen de imputación de datos que se realizó sobre las variables **Duration** (Duración de la llamada en segundos) y **Campaign** (Número de llamadas realizadas en la campaña vigente).

Tabla 8. Registros imputados atributos numéricos continuos

ATRIBUTO	VALOR A ELIMINAR	CANTIDAD ELIMINADA
Duration	Duración mayor o igual a 1.000 segundos	890
Campaign	Número de llamadas mayor o igual a 10	984

7.2.2.2. ELIMINACIÓN A NIVEL DE ATRIBUTOS

Con base a los análisis exploratorios se eliminarán los atributos de contexto social y económico ya que no aportan significancia a nivel individual de cada registro, ni mucho menos a la variable objetivo. Así mismo se eliminarán los atributos **Pdays** (Diferencia días campaña anterior vs campaña actual) y **Previous** (Cantidad de llamadas realizadas en la campaña anterior).

La razón de eliminar el atributo Pdays se debe a que gran porcentaje de sus valores representan el valor numérico 999, lo que significa que estas personas son nuevas y no tienen contacto previo, por lo cual será irrelevante para el modelo.

Por otra parte, se tomó la decisión de eliminar el atributo Previous, ya que gran porcentaje de sus valores representan el valor numérico 0, lo que significa que no se ha contactado a este cliente en campañas anteriores y también será información irrelevante para el modelo.

Así mismo se eliminará el atributo contact (medio de contacto) ya que la comunicación con el cliente se dará únicamente por llamada telefónica, por lo cual será un atributo irrelevante para el modelo.

En la tabla 9 se mostrará el resumen de eliminación de atributos que se realizó sobre el conjunto de datos.

Tabla 9. Atributos eliminados

No.	ATRIBUTOS ELIMINADOS
1	Emp.Var.Rate
2	Cons.Price.Idx
3	Cons.Conf.Idx
4	Euribor3m
5	Nr.Employed
6	Pdays
7	Previous
8	Contact

7.2.2.3. CATEGORIZACIÓN Y TRANSFORMACIÓN DE ATRIBUTOS

Es importante realizar una agrupación o categorización de los datos contenidos en algunas variables del conjunto de datos, ya que estas cuentan con información o datos redundantes que serán más viables manipular en agrupaciones.

Así mismo se decidió realizar una transformación sobre los datos de algunas variables para que sean representativos cuando se le suministren al modelo de machine learning.

En la tabla 10 se mostrará el resumen de agrupación y transformación que se llevó a cabo sobre algunas variables del conjunto de datos.

Tabla 10. Categorización y Transformación de atributos

ATRIBUTO	AGRUPAMIENTO
Age	Personas menores a 30 años se agruparon en la edad de 20 años
	Personas mayores a 31 años y menores de 40 años se agruparon en la edad de 30 años
	Personas mayores a 41 años y menores de 50 años se agruparon en la edad de 40 años
	Personas mayores a 51 años y menores de 60 años se agruparon en la edad de 50 años
	Personas mayores a 61 años se agruparon en la edad de 60 años
Education	Personas que tengan un nivel de educación basic.4y, basic.6y y basic.9y se agruparon en el nuevo nivel Primary.
Month	Se realizó una transformación de la variable Month para convertirla en número del mes. Ejemplo: Enero = 1, Febrero = 2, etc.
Day Of Week	Se realizó una transformación de la variable Day Of Week para convertirla en número del día de la semana. Ejemplo: Lunes = 1, Martes = 2, etc.
Duration	Se realizó una transformación de la variable Duration para convertirla de segundos a minutos.

7.2.2.4. RENOMBRAMIENTO Y ESTANDARIZACIÓN DE ATRIBUTOS

Uno de los puntos fundamentales en el preprocesamiento de los datos es estandarizar el nombre de los atributos (Todo en mayúscula o todo en minúscula, un mismo tipo de fuente, etc.). Para este conjunto de datos se decidió renombrar y estandarizar todos los atributos, ya que estos en muchos casos no representan un nombre claro sobre la información que se tiene almacenada allí.

En la tabla 11 se mostrará el renombramiento y estandarización de los atributos contenidos en este conjunto de datos.

Tabla 11. Renombramiento y estandarización de atributos

ATRIBUTO ANTERIOR	ATRIBUTO NUEVO
age	Age
job	Work_Type
marital	Marital_Status
education	Level_Education
default	Delinquent_Credit
housing	Housing_Credit
loan	Personal_Credit
month	Contact_Month
day_of_week	Contact_Day_Week
duration	Call_Duration
campaign	Number_Calls
poutcome	Previous_Campaign_Result
y	Acquired_Product

7.2.2.5. CODIFICACIÓN ATRIBUTOS CATEGÓRICOS

Para que nuestro modelo de machine learning pueda entrenarse y aprender de los datos es importante que estos se encuentren transformados a valores numéricos, por ende es importante aplicarles una técnica llamada codificación de etiquetas a aquellas variables que aún no son numéricas, es decir sobre las variables categóricas. En esta técnica, a cada dato se le asigna un número entero único. Para este conjunto de datos se le aplicó la técnica de codificación **Label Encoder** sobre las variables categóricas.

En la tabla 12 se mostrará el valor numérico entero asignado a cada una de las variables categóricas a las cuales se les aplicó la técnica de codificación de etiquetas **Label Encoder.**

Tabla 12. Codificación atributos categóricos

ATRIBUTO	VALOR ASIGNADO
Work_Type	0: admin. 1: blue-collar 2: entrepreneur 3: housemaid 4: management

	5: retired 6: self-employed 7: services 8: student 9: technician 10: unemployed
Marital_Status	0: divorced 1: married 2: single
Level_Education	0: primary 1: high.school 2: professional.course 3: university.degree
Delinquent_Credit	0: no 1: yes
Housing_Credit	0: no 1: yes
Personal_Credit	0: no 1: yes
Previous_Campaign_Result	0: failure 1: success
Acquired_Product	0: no 1: yes

7.2.2.6. ANÁLISIS UNIVARIADO Y COMPUESTO DESPUÉS DEL PREPROCESAMIENTO

Se procederá a graficar nuevamente cada una de las variables del conjunto de datos para visualizar y observar la diferencia de los datos en limpio con respecto a la fase anterior.

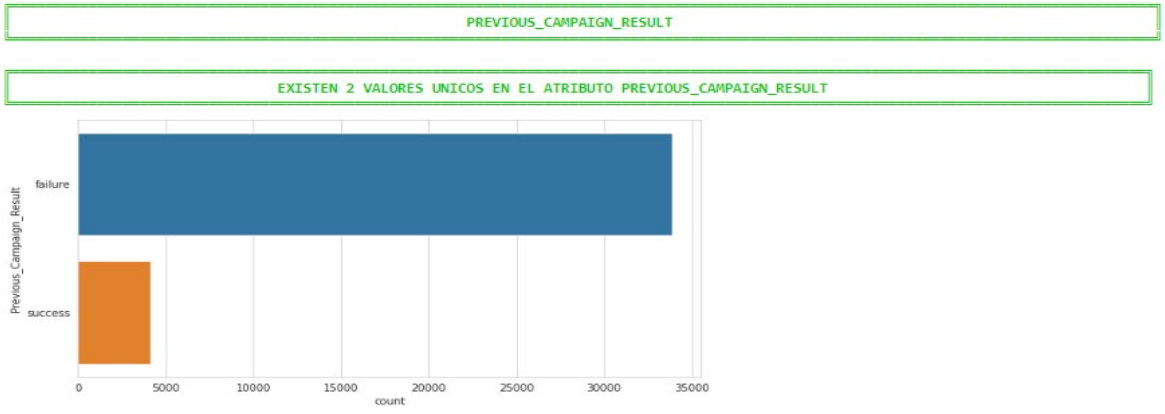


Ilustración 46. Análisis univariado variable Previous Campaign Result
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

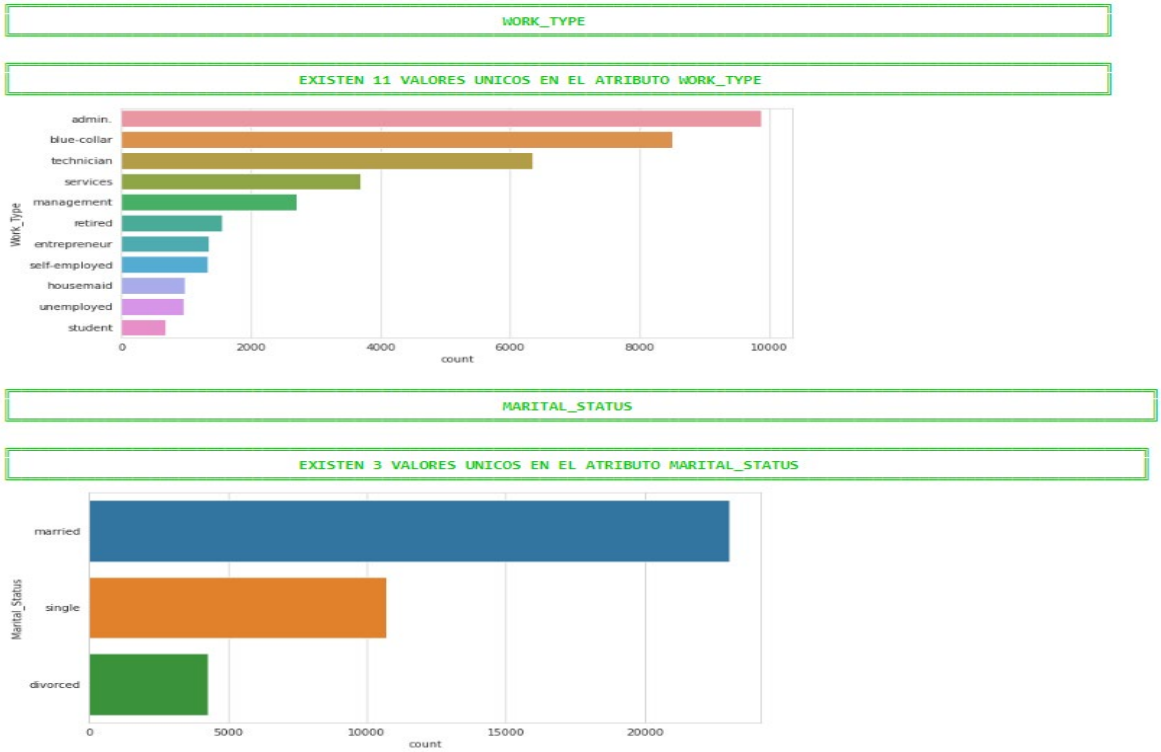


Ilustración 47. Análisis univariado variables Work Type y Marital Status
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

Ilustración 48. Análisis univariado variables Level Education y Delinquent Credit
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

Ilustración 49. Análisis univariado variables Housing Credit y Personal Credit
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

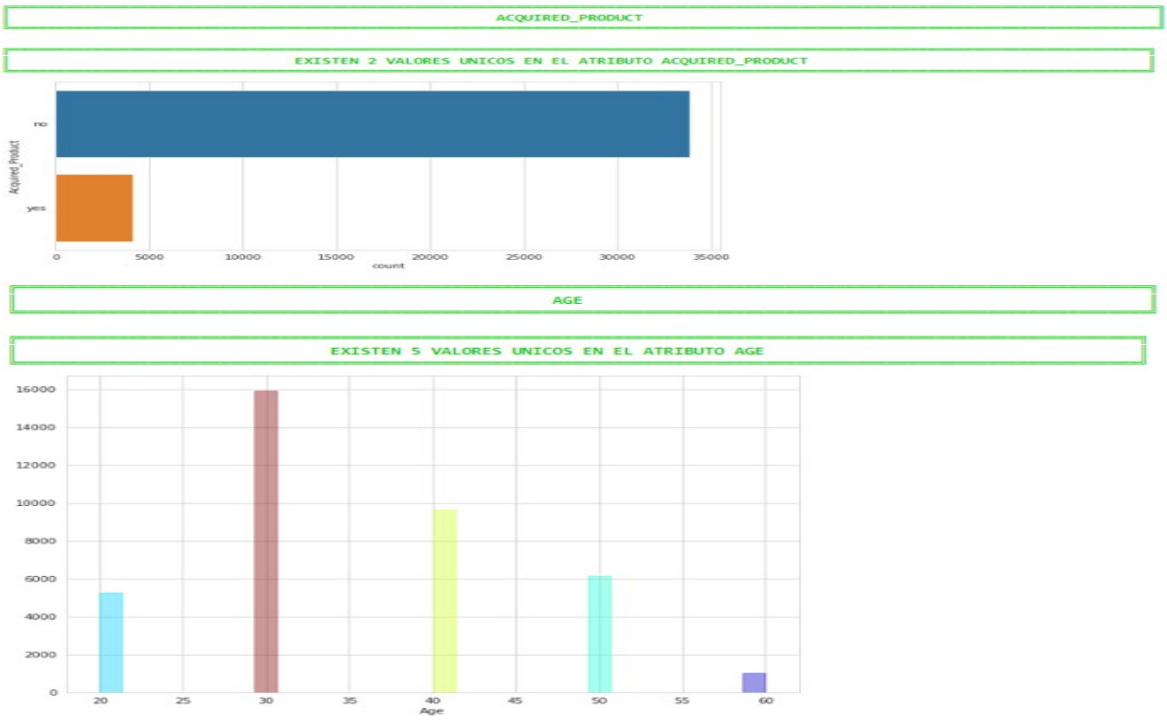


Ilustración 46. Análisis univariado variables Acquired Product y Age
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

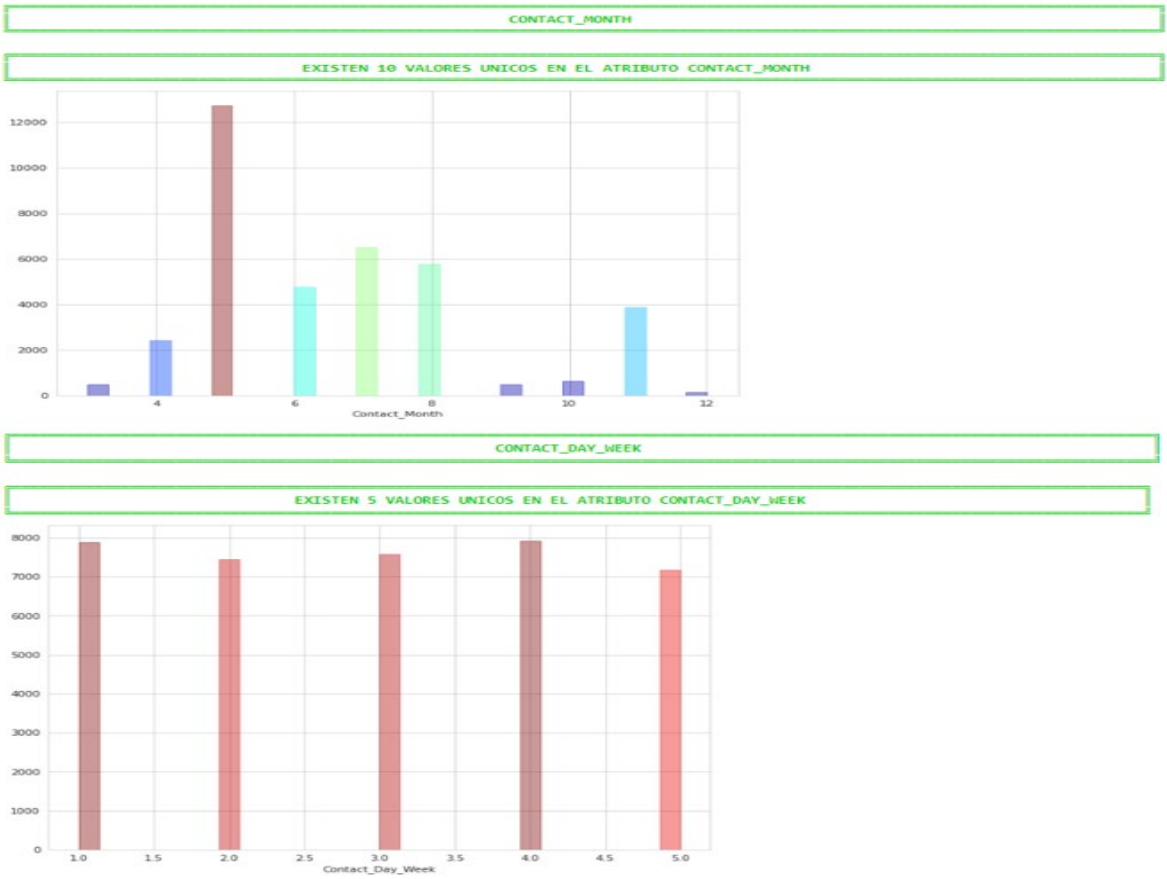


Ilustración 47. Análisis univariado variables Contact Month y Contact Day Week
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

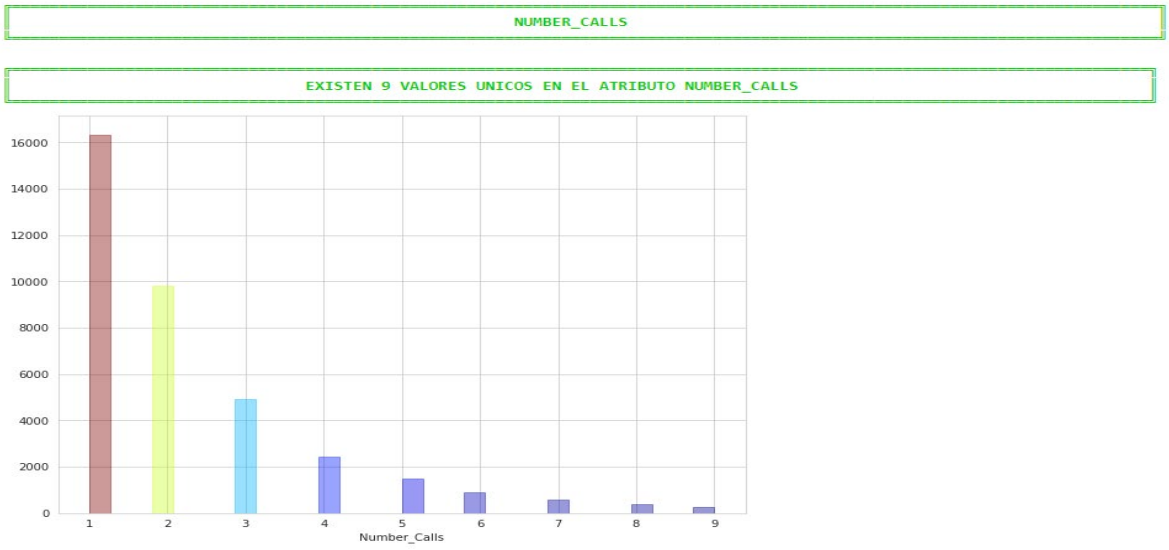


Ilustración 48. Análisis univariado variable Number Calls
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

Ilustración 49. Análisis compuesto variables conjunto de datos después de preprocesamiento
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

De acuerdo con las gráficas plasmadas nuevamente después del preprocesamiento de los datos se puede concluir lo siguiente:

- Ya no existen valores desconocidos (unknown) en los atributos.
- Ya no se presentan valores nonexistent en el atributo Previous_Campaign_Result.
- Ya están agrupados y/o categorizados los atributos a los cuales se les implementó esta actividad. Así mismo ya se ve la transformación en los datos a los cuales se les aplicó alguna técnica descrita en el punto 7.2.2.3 (Categorización y Transformación de atributos).
- Ya están renombrados y estandarizados los atributos.
- Ya no están presentes los atributos que se eliminaron de este conjunto de datos.

7.2.2.7. ANÁLISIS DATOS ATÍPICOS VARIABLES NUMÉRICAS

Se procederá a graficar nuevamente cada una de las variables numéricas del conjunto de datos donde se detectaron posibles datos atípicos en cada una de estas, para visualizar y observar la diferencia de los datos en limpio con respecto a la fase anterior.

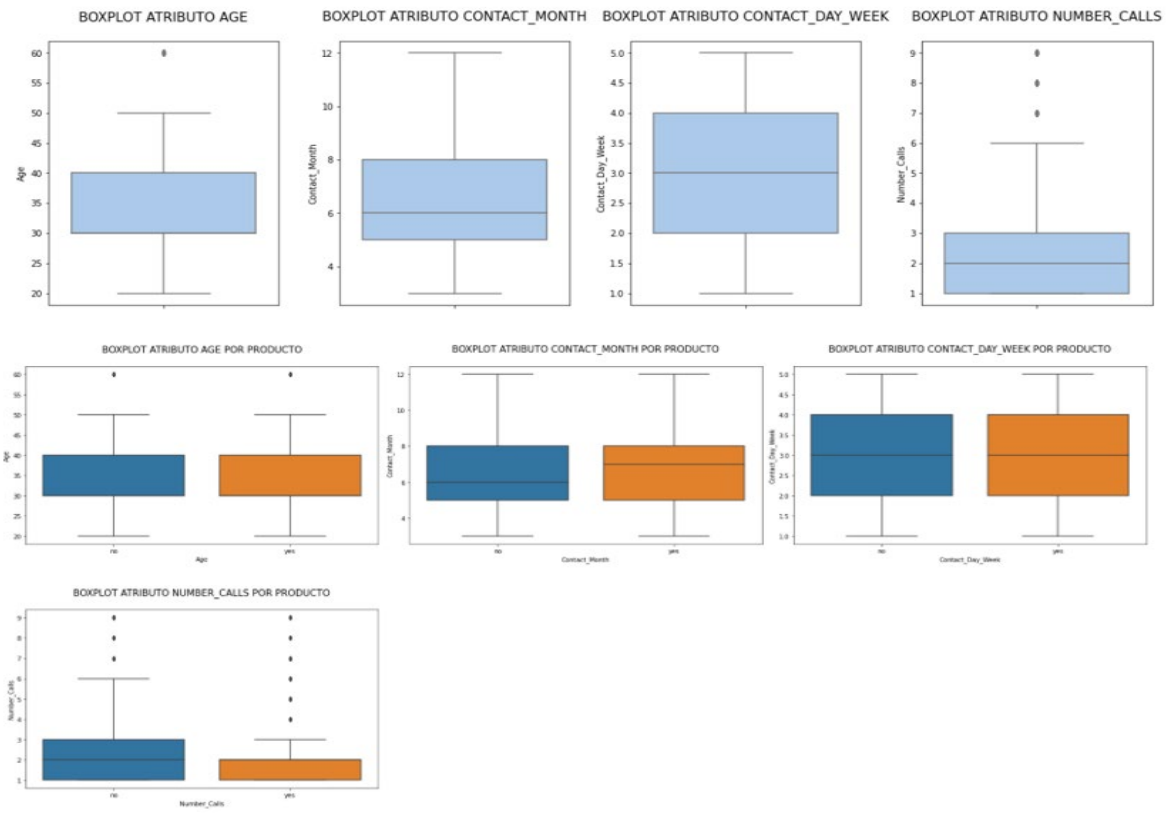


Ilustración 50. Análisis datos atípicos variables numéricas después de preprocesamiento
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En este caso podemos identificar fácilmente que ya no existe un gran porcentaje de datos atípicos y esto gracias al preprocesamiento y/o limpieza que se implementó sobre los datos contenidos en el conjunto de datos.

7.2.2.8. ANÁLISIS CORRELACIÓN VARIABLES CONJUNTO DE DATOS

Se graficará nuevamente la matriz de correlación que nos permita identificar posibles correlaciones entre las variables presentes en este conjunto de datos después de haberles realizado el preprocesamiento de los datos.

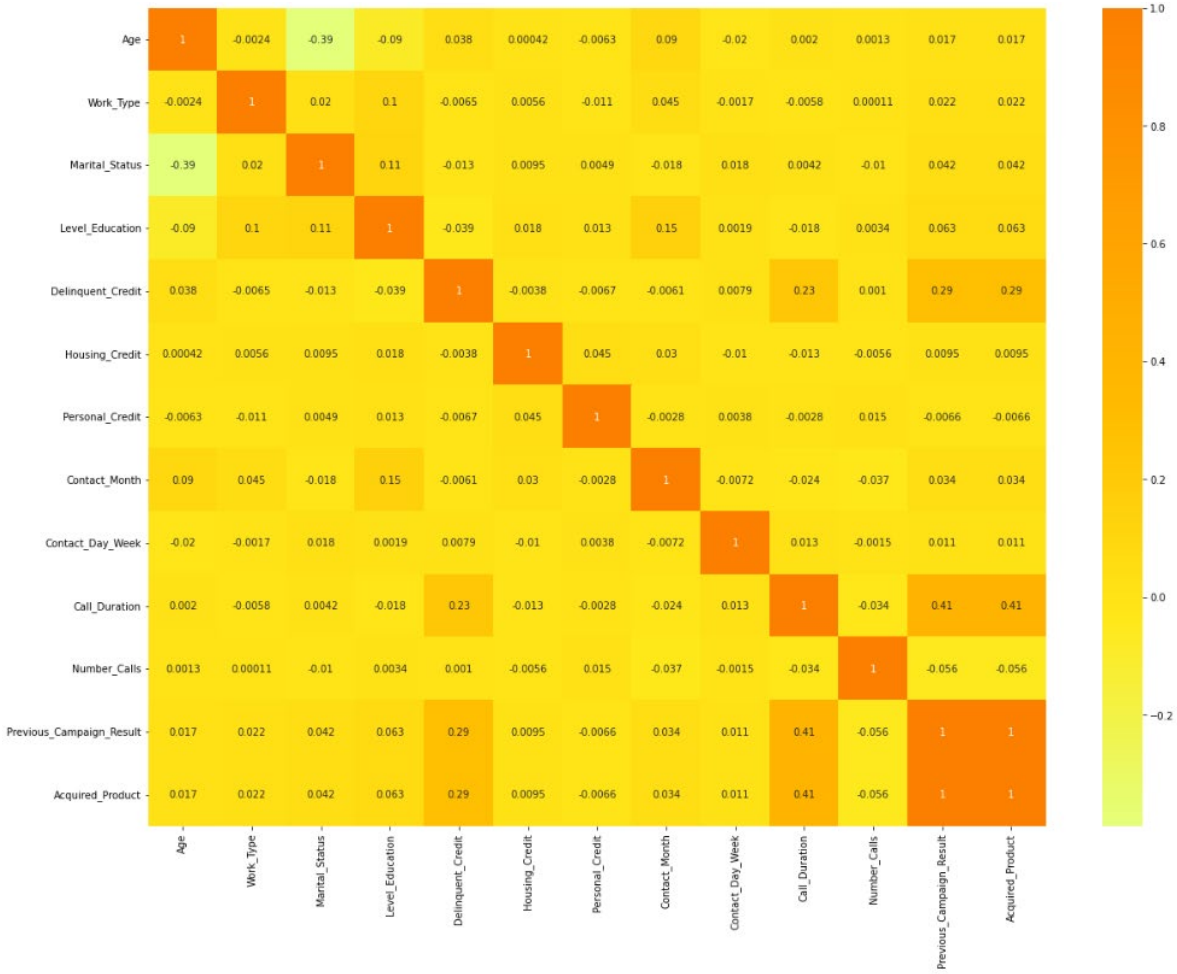


Ilustración 51. Matriz de correlación variables conjunto de datos después de preprocesamiento
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 51 se está presentando una correlación extremadamente fuerte (1) entre la variable **Previous_Campaign_Result** con respecto a la variable predictora **Acquired_Product**. Por ende es necesario eliminar esta variable independiente del conjunto de datos.

Esta correlación nos quiere dar a entender que si la variable **Previous_Campaign_Result** (Resultado de la campaña anterior) es exitosa (success) el cliente si o si tomara el producto ofrecido, mientras que si esta variable presenta un resultado fallido (failure) el cliente definitivamente no tomara el producto ofrecido.

7.2.3. INTEGRACIÓN DE LOS DATOS

En esta fase de CRISP-DM concluimos con el conjunto de datos integrado de toda la limpieza y preprocesamiento realizada sobre los datos expuestos en este documento.

	Age	Work_Type	Marital_Status	Level_Education	Delinquent_Credit	Housing_Credit	Personal_Credit	Contact_Month	Contact_Day_Week	Call_Duration	Number_Calls	Acquired_Product
0	50	3	1	0	0	0	0	5	1	4.35	1	0
1	50	7	1	1	0	0	0	5	1	2.48	1	0
2	30	7	1	1	0	1	0	5	1	3.77	1	0
3	40	0	1	0	0	0	0	5	1	2.52	1	0
4	50	7	1	1	0	0	1	5	1	5.12	1	0
5	40	7	1	0	0	0	0	5	1	3.30	1	0
6	50	0	1	2	0	0	0	5	1	2.32	1	0
8	20	9	2	2	0	1	0	5	1	6.33	1	0
9	20	7	2	1	0	1	0	5	1	0.83	1	0
11	20	7	2	1	0	1	0	5	1	3.70	1	0

Ilustración 52. Conjunto de datos final preprocesado
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

7.3. MODELAMIENTO

7.3.1. SELECCIÓN TÉCNICA DE MODELADO

En esta fase se seleccionarán diferentes técnicas de modelado que permitan dar cumplimiento con los objetivos propuestos del proyecto.

Con el preprocesamiento de los datos ya realizados en fases anteriores se procederá a realizar la división de los datos de entrenamiento y de prueba, se implementará el escalado de atributos, se dará un tratamiento especial a los datos desequilibrados mediante la técnica smote, se efectuarán métricas de evaluación, matrices de confusión, curvas de ROC, se evaluarán las variables más sobresalientes sobre cada uno de los modelos y por último se implementará los hiperparametros sobre cada uno de estos modelos.

7.3.2. DISEÑO Y CONSTRUCCIÓN DE MODELOS

Para el desarrollo de este proyecto se llevarán a cabo los siguientes algoritmos de clasificación de aprendizaje supervisado:

- Logistic Regression
- Decision Tree
- Random Forest
- XGBoost
- Support Vector Machines
- Naïve Bayes

7.3.2.1. DIVISIÓN CONJUNTO DE DATOS

Se iniciará clasificando las variables predictoras de la variable objetivo o predicha para que queden en variables totalmente independientes.

Luego se dividirán los registros del conjunto de datos en una proporción del 70% para entrenamiento y un 30% para pruebas.

En la tabla 13 se representará la cantidad de registros divididos para entrenamiento y prueba:

Tabla 13. Datos entrenamiento y prueba

No. Total Muestras Conjunto de Datos	No. Muestras Entrenamiento Conjunto de Datos	No. Muestras Prueba Conjunto de Datos
36.353	25.447	10.906

7.3.2.2. ESCALADO DE ATRIBUTOS Y MANEJO DATOS DESEQUILIBRADOS

Este conjunto de datos está bastante desequilibrado, lo que refleja que los resultados pueden no representar la precisión real y estarían sesgados. Para superar este desequilibrio, se utilizó la técnica de sobremuestreo (smote) la cual nos permite equilibrar en igual proporción y/o volumen de datos la clase minorista con respecto a la clase mayoritaria que se encuentra sobre la variable objetivo.

Los datos transformados y desequilibrados se normalizan aún más utilizando la técnica Standard Scaler ya que gracias a esta podemos eliminar el sesgo potencial que el modelo puede tener hacia características con magnitudes más altas.

En la tabla 14 se representará la cantidad de registros después de aplicar las técnicas de sobremuestreo (smote) y Standard Scaler:

Tabla 14. Técnica smote y standardscaler

	Clientes Adquirieron Producto Financiero	Clientes No Adquirieron Producto Financiero
No. Muestras Antes de Aplicar Técnica Sobremuestreo	2.592	22.855
No. Muestras Después de Aplicar Técnica Sobremuestreo	22.855	22.855
No. Muestras Final Entrenamiento	45.710	

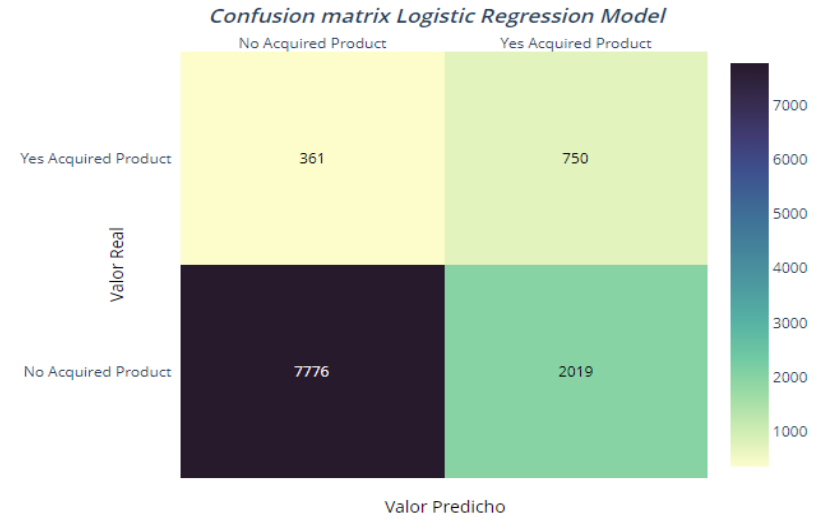
7.3.2.3. MATRICES DE CONFUSIÓN

Esta es una de las técnicas o herramientas de mayor relevancia e importancia en esta fase, ya que nos va a permitir evaluar cada uno de los modelos con base a un conteo de aciertos y errores de la clase predicha en los algoritmos de clasificación.

Uno de los objetivos fundamentales para este proyecto es poder detectar cuál de los modelos será el que mayor aciertos contenga al predecir si un cliente aceptará el producto financiero, por lo cual nos fijamos en el indicador de desempeño de valor F (f1-score), ya que va a permitir combinar las medidas de precision y recall en un sólo valor. Esto es práctico para este proyecto porque hace más fácil el poder comparar el rendimiento combinado de la precisión y la exhaustividad entre varias soluciones.

7.3.2.3.1. LOGISTIC REGRESSION

Se analizará la matriz de confusión del modelo de regresión logística, identificando los indicadores de desempeño más ajustable hacia la clase 1 (Cliente adquiere producto financiero) y evaluando las diferentes combinaciones entre el valor predicho con el valor real.



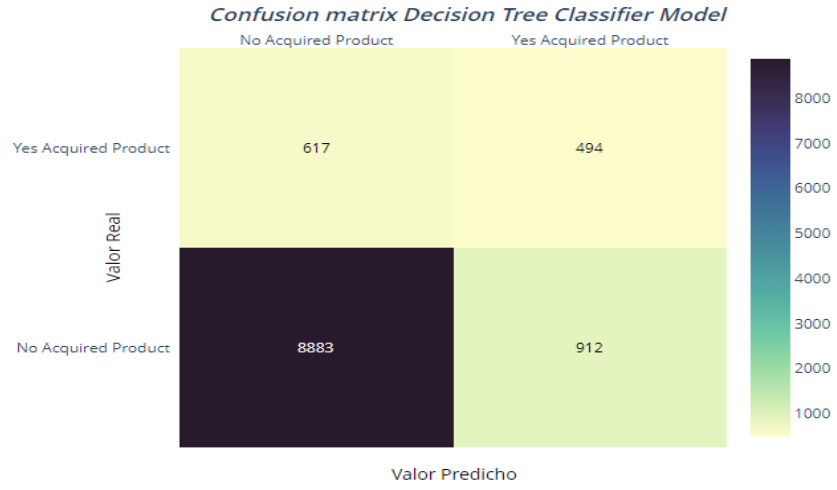
	precision	recall	f1-score	support
0	0.96	0.79	0.87	9795
1	0.27	0.68	0.39	1111
accuracy			0.78	10906
macro avg	0.61	0.73	0.63	10906
weighted avg	0.89	0.78	0.82	10906

Ilustración 53. Matriz de confusión y métricas de evaluación modelo Logistic Regression
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

De acuerdo con los resultados obtenidos en la ilustración 53 se puede indicar que es un algoritmo sobresaliente hacia la clase de interés que queremos abarcar en este proyecto al obtener una puntuación media en las métricas de evaluación reflejadas. Así mismo se puede observar en la matriz de confusión que el algoritmo tuvo una desviación de error en la predicción en 2.380 registros (2.019 Falsos-Positivos y 361 Falsos-Negativos) de 10.906, lo que lo hace un algoritmo poco adaptable para trabajar con este conjunto de datos.

7.3.2.3.2. DECISION TREE CLASSIFIER

Se analizará la matriz de confusión del modelo de árboles de decisión, identificando los indicadores de desempeño más ajustable hacia la clase 1 (Cliente adquiere producto financiero) y evaluando las diferentes combinaciones entre el valor predicho con el valor real.



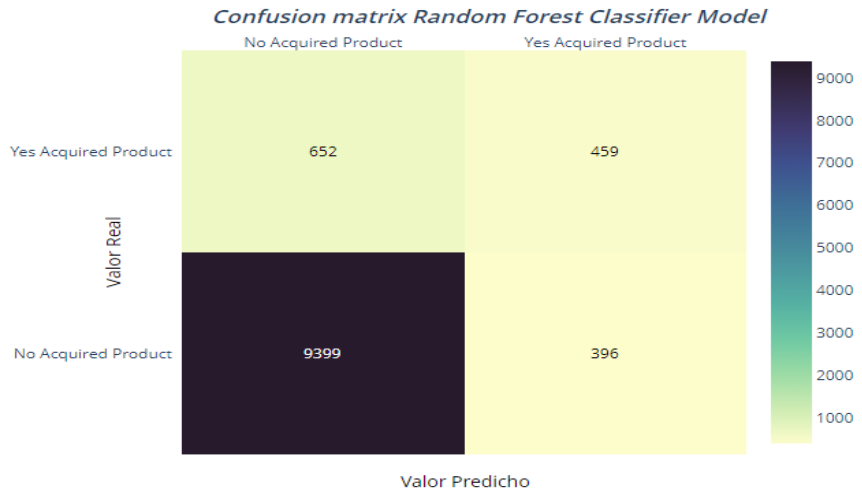
	precision	recall	f1-score	support
0	0.94	0.91	0.92	9795
1	0.35	0.44	0.39	1111
accuracy			0.86	10906
macro avg	0.64	0.68	0.66	10906
weighted avg	0.88	0.86	0.87	10906

Ilustración 54. Matriz de confusión y métricas de evaluación modelo Decision Tree
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

De acuerdo con los resultados obtenidos en la ilustración 54 se puede indicar que es un algoritmo poco eficiente hacia la clase de interés que queremos abarcar en este proyecto al obtener una puntuación muy baja en las métricas de evaluación reflejadas. Así mismo se puede observar en la matriz de confusión que el algoritmo tuvo una desviación de error en la predicción en 1.529 registros (912 Falsos-Positivos y 617 Falsos-Negativos) de 10.906, lo que lo hace un algoritmo poco adaptable para trabajar con este conjunto de datos.

7.3.2.3.3. RANDOM FOREST CLASSIFIER

Se analizará la matriz de confusión del modelo de bosque aleatorio, identificando los indicadores de desempeño más ajustable hacia la clase 1 (Cliente adquiere producto financiero) y evaluando las diferentes combinaciones entre el valor predicho con el valor real.



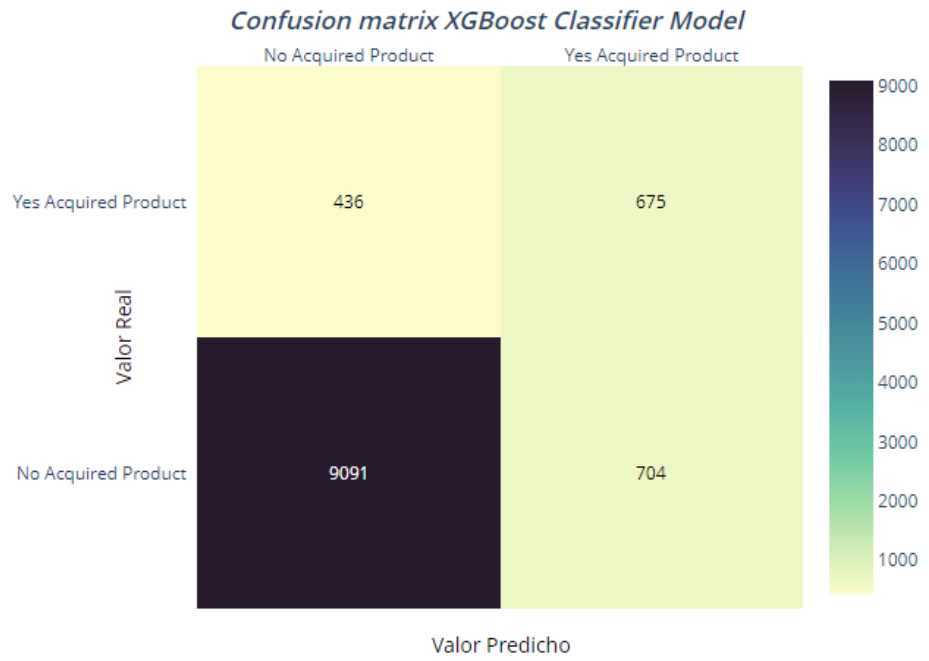
	precision	recall	f1-score	support
0	0.94	0.96	0.95	9795
1	0.54	0.41	0.47	1111
accuracy			0.90	10906
macro avg	0.74	0.69	0.71	10906
weighted avg	0.89	0.90	0.90	10906

Ilustración 55. Matriz de confusión y métricas de evaluación modelo Random Forest
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

De acuerdo con los resultados obtenidos en la ilustración 55 se puede indicar que es un algoritmo muy bueno hacia la clase de interés que queremos abarcar en este proyecto al obtener una puntuación considerablemente buena en las métricas de evaluación reflejadas. Así mismo se puede observar en la matriz de confusión que el algoritmo tuvo una desviación de error en la predicción en 1.048 registros (396 Falsos-Positivos y 652 Falsos-Negativos) de 10.906, lo que lo hace un algoritmo muy adaptable para trabajar con este conjunto de datos.

7.3.2.3.4. XGBOOST CLASSIFIER

Se analizará la matriz de confusión del modelo de xgboost, identificando los indicadores de desempeño más ajustable hacia la clase 1 (Cliente adquiere producto financiero) y evaluando las diferentes combinaciones entre el valor predicho con el valor real.



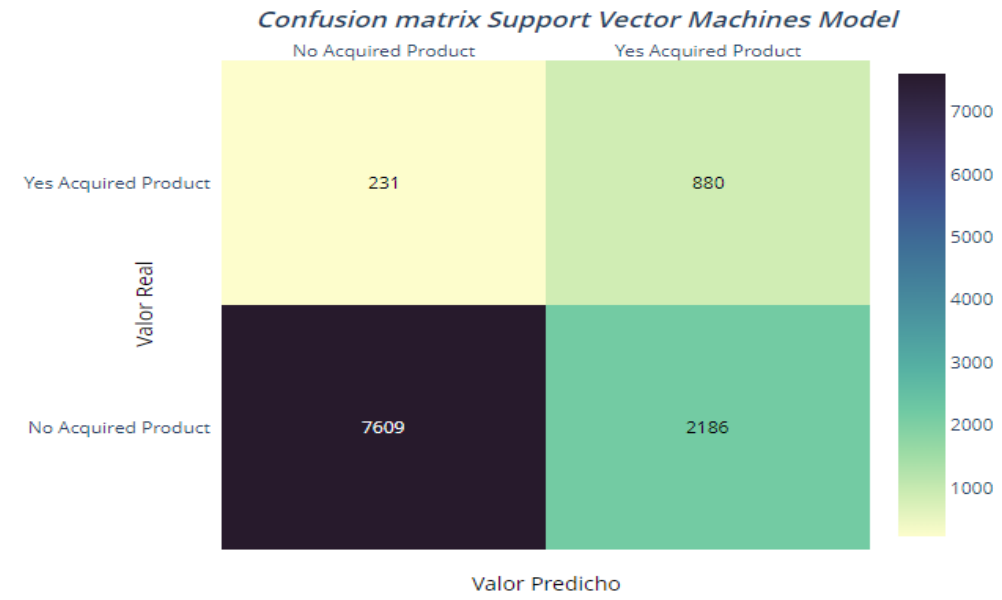
	precision	recall	f1-score	support
0	0.95	0.93	0.94	9795
1	0.49	0.61	0.54	1111
accuracy			0.90	10906
macro avg	0.72	0.77	0.74	10906
weighted avg	0.91	0.90	0.90	10906

Ilustración 56. Matriz de confusión y métricas de evaluación modelo XGBoost
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

De acuerdo con los resultados obtenidos en la ilustración 56 se puede indicar que es un algoritmo bueno hacia la clase de interés que queremos abarcar en este proyecto al obtener una puntuación considerablemente buena en las métricas de evaluación reflejadas. Así mismo se puede observar en la matriz de confusión que el algoritmo tuvo una desviación de error en la predicción en 1.140 registros (704 Falsos-Positivos y 436 Falsos-Negativos) de 10.906, lo que lo hace un algoritmo muy adaptable para trabajar con este conjunto de datos.

7.3.2.3.5. SUPPORT VECTOR MACHINES

Se analizará la matriz de confusión del modelo de máquinas de vectores de soporte, identificando los indicadores de desempeño más ajustable hacia la clase 1 (Cliente adquiere producto financiero) y evaluando las diferentes combinaciones entre el valor predicho con el valor real.



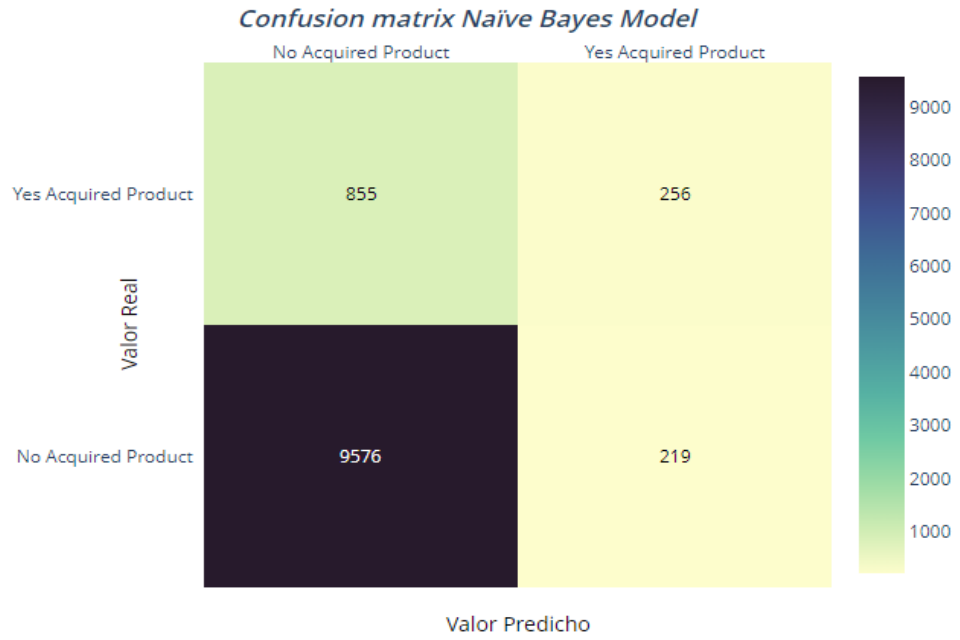
	precision	recall	f1-score	support
0	0.97	0.78	0.86	9795
1	0.29	0.79	0.42	1111
accuracy			0.78	10906
macro avg	0.63	0.78	0.64	10906
weighted avg	0.90	0.78	0.82	10906

Ilustración 57. Matriz de confusión y métricas de evaluación modelo Support Vector Machines
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

De acuerdo con los resultados obtenidos en la ilustración 57 se puede indicar que es un algoritmo muy bueno hacia la clase de interés que queremos abarcar en este proyecto al obtener una puntuación considerablemente buena en las métricas de evaluación reflejadas. Así mismo se puede observar en la matriz de confusión que el algoritmo tuvo una desviación de error en la predicción en 2.417 registros (2.186 Falsos-Positivos y 231 Falsos-Negativos) de 10.906, lo que lo hace un algoritmo muy adaptable para trabajar con este conjunto de datos.

7.3.2.3.6. NAÏVE BAYES

Se analizará la matriz de confusión del modelo bayesiano, identificando los indicadores de desempeño más ajustable hacia la clase 1 (Cliente adquiere producto financiero) y evaluando las diferentes combinaciones entre el valor predicho con el valor real.



	precision	recall	f1-score	support
0	0.92	0.98	0.95	9795
1	0.54	0.23	0.32	1111
accuracy			0.90	10906
macro avg	0.73	0.60	0.63	10906
weighted avg	0.88	0.90	0.88	10906

Ilustración 58. Matriz de confusión y métricas de evaluación modelo Naïve Bayes
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

De acuerdo con los resultados obtenidos en la ilustración 58 se puede indicar que es un algoritmo poco eficiente hacia la clase de interés que queremos abarcar en este proyecto al obtener una puntuación muy baja en las métricas de evaluación reflejadas. Así mismo se puede observar en la matriz de confusión que el algoritmo tuvo una desviación de error en la predicción en 1.074 registros (219 Falsos-Positivos y 855 Falsos-Negativos) de 10.906, lo que lo hace un algoritmo poco adaptable para trabajar con este conjunto de datos.

7.3.2.4. CURVA DE ROC Y ÁREA BAJO LA CURVA AUC

La curva de ROC y área bajo la curva AUC es otra de las técnicas o herramientas muy relevantes e importantes en esta fase. Al igual que la matriz de confusión la curva de ROC es una representación gráfica de los falsos positivos y los verdaderos positivos que va a ir mostrando los modelos posiblemente más óptimos para ir evaluando cuál será el mejor modelo con los resultados más eficientes o eficaces que probablemente sea adapte o ajuste mejor al conjunto de datos de este proyecto.

7.3.2.4.1. LOGISTIC REGRESSION

Se analizará la curva de ROC y área bajo la curva AUC del modelo de regresión logística, para identificar el equilibrio que haya entre la sensibilidad (Verdaderos Positivos) y la especificidad (Falsos Positivos) de este modelo:

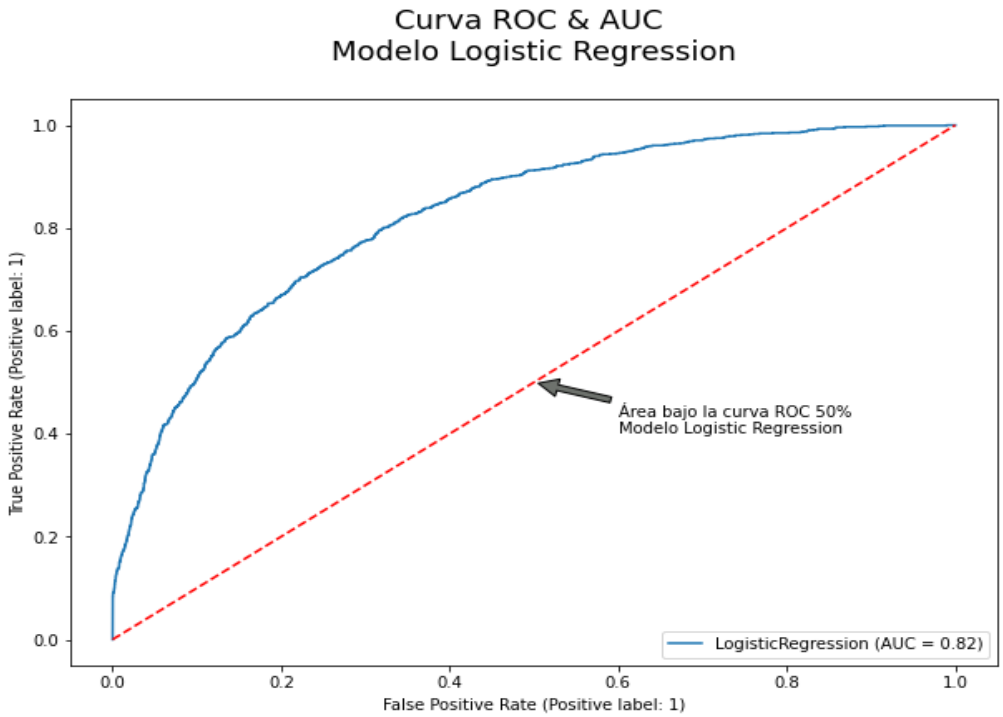


Ilustración 59. Curva ROC y AUC modelo Logistic Regression
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

La ilustración 59 representa que hay un equilibrio medio entre la sensibilidad y la especificidad de este modelo, ya que su curva (en color azul) no tiende a estar tan cercana de la esquina superior izquierda, indicando una medida de rendimiento por mejorar ya que el modelo en este caso dará como resultado una proporción de observaciones que no serán muy bien predichas cuando se le proporcione nuevos conjuntos de datos a predecir.

7.3.2.4.2. DECISION TREE CLASSIFIER

Se analizará la curva de ROC y área bajo la curva AUC del modelo de árboles de decisión, para identificar el equilibrio que haya entre la sensibilidad (Verdaderos Positivos) y la especificidad (Falsos Positivos) de este modelo:

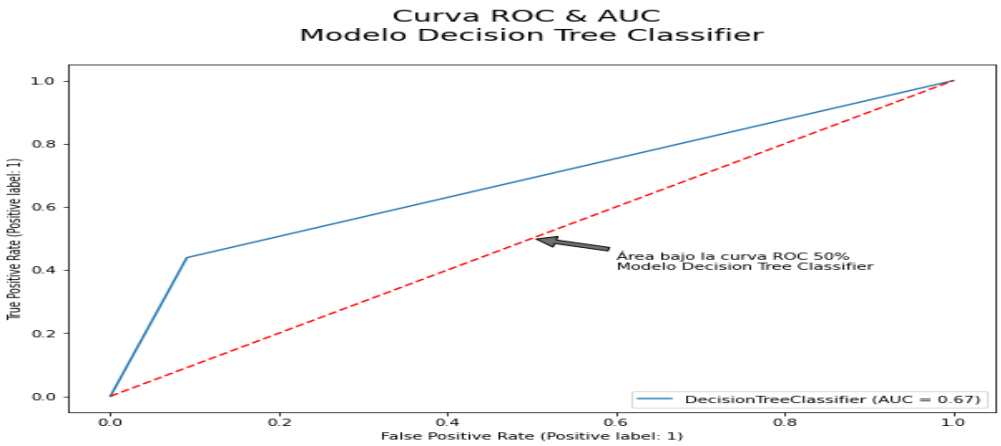


Ilustración 60. Curva ROC y AUC modelo Decision Tree
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

La ilustración 60 representa que no existe un equilibrio entre la sensibilidad y la especificidad de este modelo, ya que su curva (en color azul) está demasiado lejana de la esquina superior izquierda, indicando un rendimiento muy bajo ya que el modelo en este caso dará como resultado una proporción de observaciones muy alta que no serán bien predichas cuando se le proporcione nuevos conjuntos de datos a predecir.

7.3.2.4.3. RANDOM FOREST CLASSIFIER

Se analizará la curva de ROC y área bajo la curva AUC del modelo de bosque aleatorio, para identificar el equilibrio que haya entre la sensibilidad (Verdaderos Positivos) y la especificidad (Falsos Positivos) de este modelo:

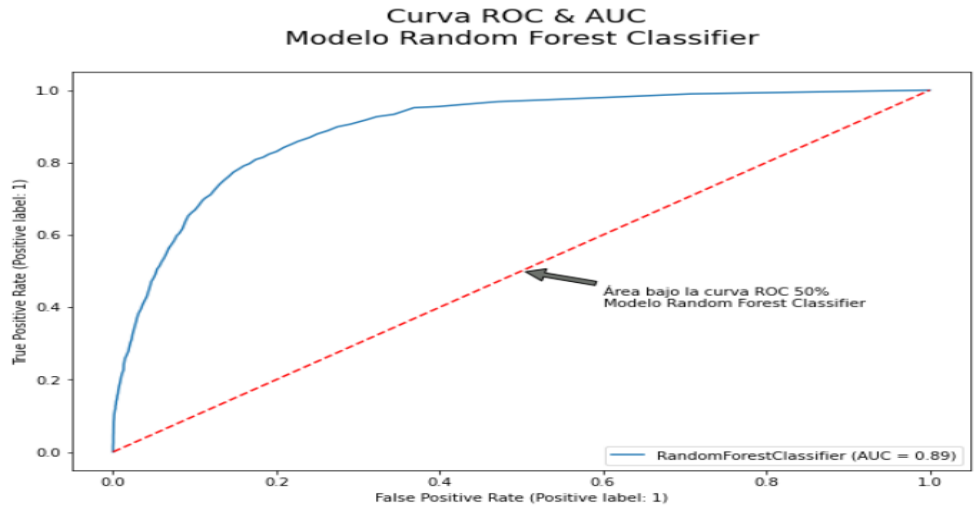


Ilustración 61. Curva ROC y AUC modelo Random Forest
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

La ilustración 61 representa que hay un equilibrio alto entre la sensibilidad y la especificidad de este modelo, ya que su curva (en color azul) tiende a estar bastante cercana de la esquina superior izquierda, indicando un rendimiento muy favorable, lo que dará como resultado una proporción de observaciones que serán muy bien predichas cuando se le proporcione nuevos conjuntos de datos al modelo.

7.3.2.4.4. XGBOOST CLASSIFIER

Se analizará la curva de ROC y área bajo la curva AUC del modelo xgboost, para identificar el equilibrio que haya entre la sensibilidad (Verdaderos Positivos) y la especificidad (Falsos Positivos) de este modelo:

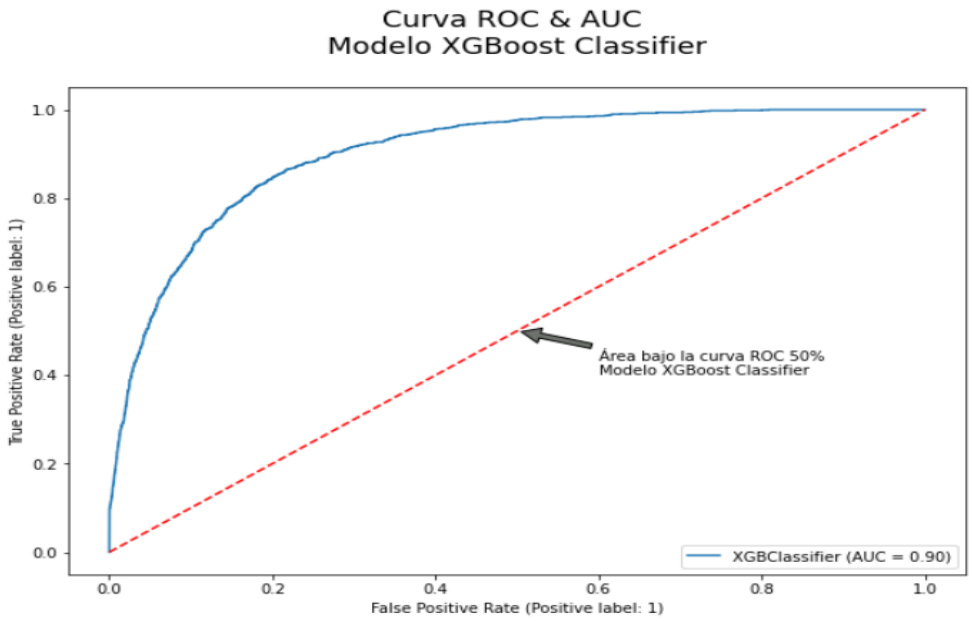


Ilustración 62. Curva ROC y AUC modelo XGBoost
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

La ilustración 62 representa que hay un equilibrio alto entre la sensibilidad y la especificidad de este modelo, ya que su curva (en color azul) tiende a estar bastante cercana de la esquina superior izquierda, indicando un rendimiento muy favorable, lo que dará como resultado una proporción de observaciones que serán muy bien predichas cuando se le proporcione nuevos conjuntos de datos al modelo.

7.3.2.4.5. SUPPORT VECTOR MACHINES

Se analizará la curva de ROC y área bajo la curva AUC del modelo de máquinas de vectores de soporte, para identificar el equilibrio que haya entre la sensibilidad (Verdaderos Positivos) y la especificidad (Falsos Positivos) de este modelo:

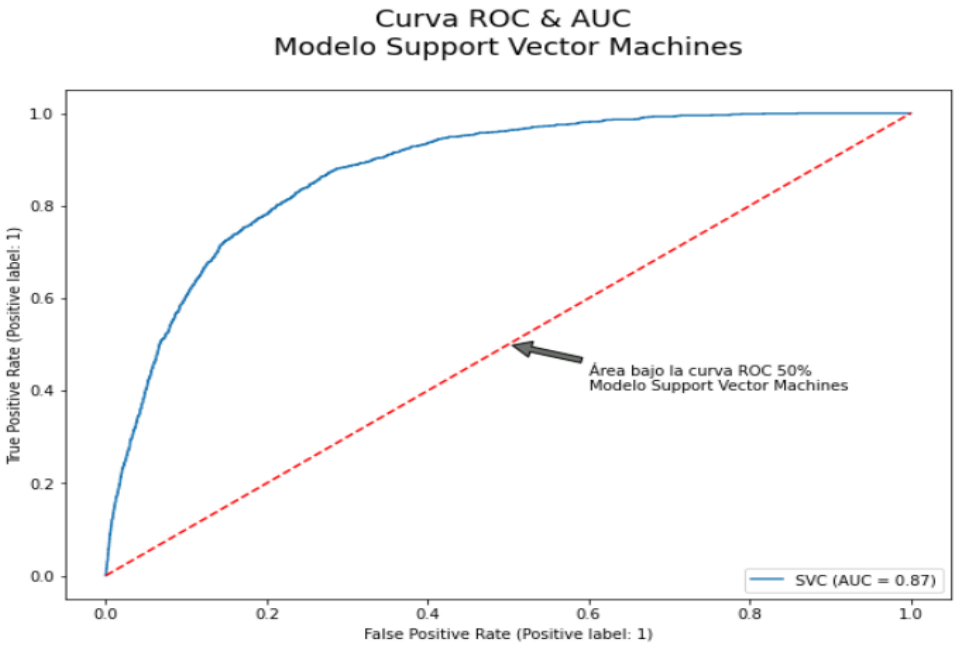


Ilustración 63. Curva ROC y AUC modelo Support Vector Machines
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

La ilustración 63 representa que hay un equilibrio considerable entre la sensibilidad y la especificidad de este modelo, ya que su curva (en color azul) tiende a estar un poco cercana de la esquina superior izquierda, indicando un rendimiento favorable, lo que dará como resultado una proporción de observaciones que serán bien predichas cuando se le proporcione nuevos conjuntos de datos al modelo.

7.3.2.4.6. NAÏVE BAYES

Se analizará la curva de ROC y área bajo la curva AUC del modelo bayesiano, para identificar el equilibrio que haya entre la sensibilidad (Verdaderos Positivos) y la especificidad (Falsos Positivos) de este modelo:

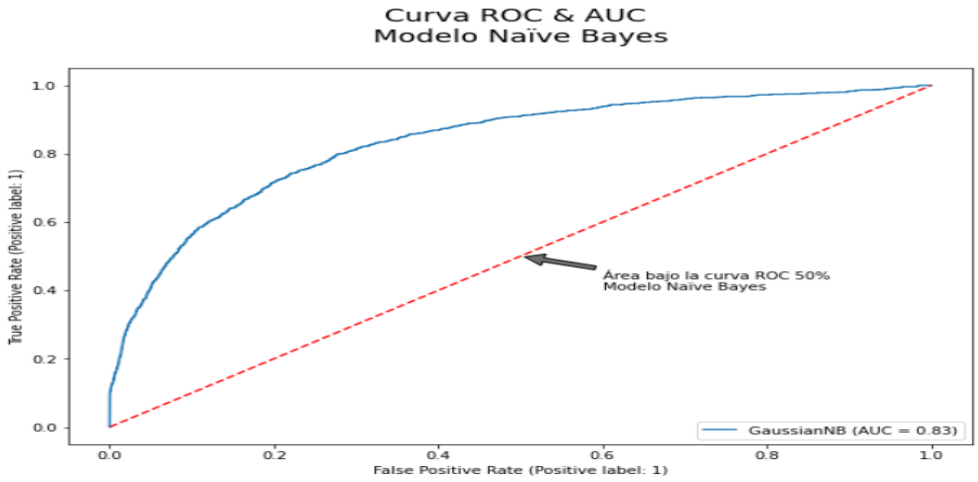


Ilustración 64. Curva ROC y AUC modelo Naïve Bayes
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

La ilustración 64 representa que hay un equilibrio medio entre la sensibilidad y la especificidad de este modelo, ya que su curva (en color azul) no tiende a estar tan cercana de la esquina superior izquierda, indicando una medida de rendimiento por mejorar ya que el modelo en este caso dará como resultado una proporción de observaciones que no serán muy bien predichas cuando se le proporcione nuevos conjuntos de datos a predecir.

7.3.2.5. CURVA ROC Y AUC COMPARACIÓN MODELOS

Se representará la curva de ROC y área bajo la curva AUC del consolidado de modelos descritos anteriormente.

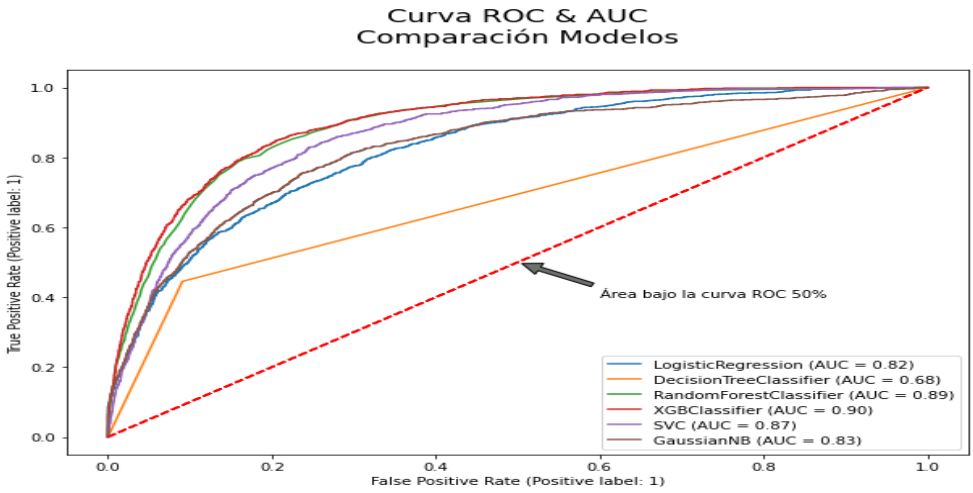


Ilustración 65. Curva ROC y AUC comparación modelos
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

Como conclusión de la ilustración 65 podemos identificar que los modelos con mejor rendimiento sin duda alguna son el bosque aleatorio y el xgboost, ya que estos modelos representan el mejor balance y equilibrio entre la sensibilidad y la especificidad de los datos y observaciones a predecir.

7.3.3. EVALUACIÓN DE MODELOS

En este punto se analizará los indicadores o métricas de desempeño de cada uno de los modelos que se han venido trabajando hasta el momento. Para este caso se evaluarán algunas métricas para ir finalizando con la toma de decisiones del modelo a implementar en la próxima fase.

7.3.3.1. INDICADORES DE DESEMPEÑO

- **Precisión:** Esta métrica permitirá medir el volumen o porcentaje de clientes que estarán interesados en adquirir el producto financiero.
- **Recall:** Esta métrica permitirá identificar el volumen o porcentaje de clientes que el modelo no fue capaz de predecir positivamente, es decir está dejando por fuera a clientes que estarían interesados en adquirir el producto financiero.
- **F1:** Esta métrica combina los dos indicadores anteriores (Precision y Recall) en un solo valor. Es demasiado importante ya que tiene en cuenta el porcentaje y volumen de los clientes que estarán interesados en adquirir el producto financiero y que fueron predichos correctamente como los clientes que estarían interesados en adquirir el producto financiero pero fueron predichos erróneamente.
- **Accuracy:** Esta métrica permitirá medir el porcentaje de aciertos que tuvo el modelo con la predicción de los datos.
- **Specificity:** Esta métrica permitirá medir el volumen o porcentaje de clientes que el modelo predijo estarían interesados en adquirir el producto financiero cuando esto era irreal.

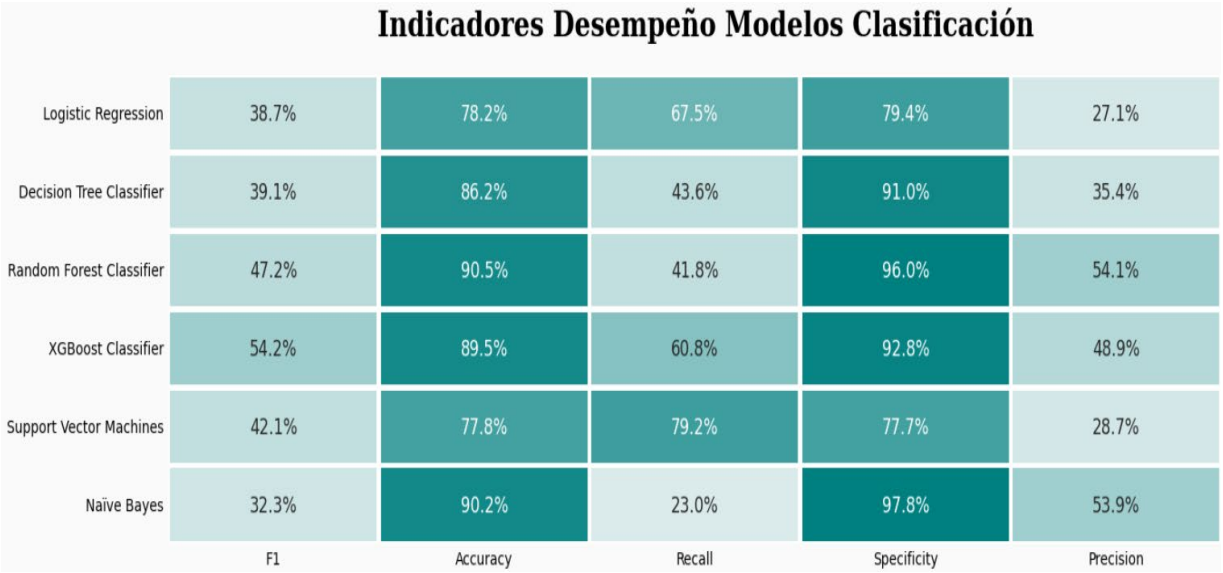


Ilustración 66. Indicadores de desempeño de los modelos
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

Para el desarrollo del presente trabajo el objetivo primordial es identificar y segmentar el grupo de clientes que estarán interesados en adquirir el producto financiero como también identificar qué clientes estaban interesados y no se tuvieron en cuenta. Las métricas más favorables para este proyecto son la precisión y el recall (sensibilidad o exhaustividad), por lo cual se tomará como indicador primordial el F1 ya que contiene los dos indicadores (precisión y recall) en un solo valor.

De acuerdo con el indicador F1 claramente se puede identificar que los mejores modelos para trabajar con este conjunto de datos asociados al proyecto son el bosque aleatorio y el xgboost, como se ha reiterado en análisis previos de esta fase de la metodología CRISP-DM.

7.3.3.2. ATRIBUTOS MÁS REPRESENTATIVOS EN LOS MODELOS

Es importante conocer en cada uno de los modelos cuáles de sus atributos son los que mayor impacto generan para predecir el resultado final o predicho. Aquí se representará cada una de esas variables asociadas a los modelos que mayor poder de predicción tenga sobre la variable objetivo.

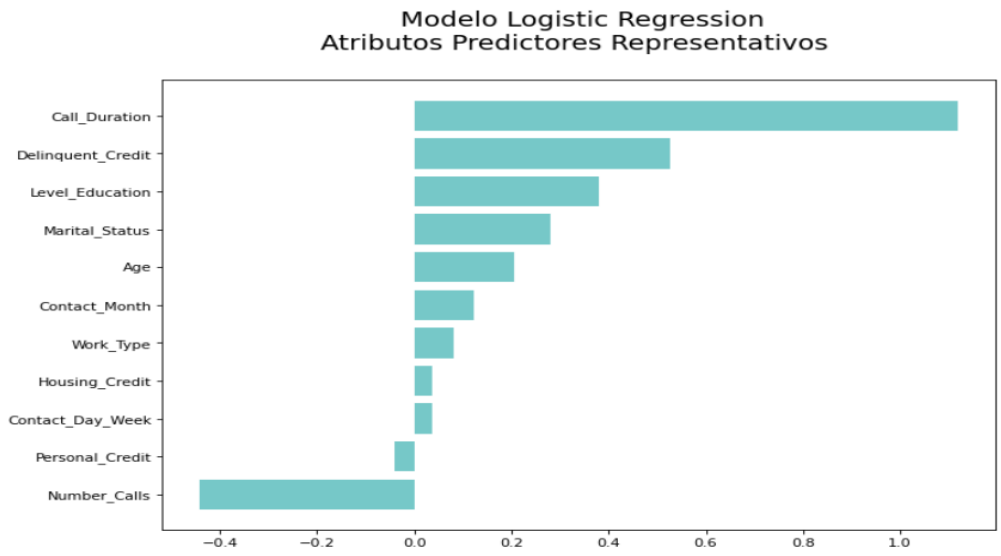


Ilustración 67. Atributos representativos en modelo Logistic Regression
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 67 se puede observar que las variables más representativas o de mayor impacto hacia la variable objetivo en el modelo de regresión logística están dadas en la duración de la llamada, si tiene o no un crédito en mora y su nivel de educación.

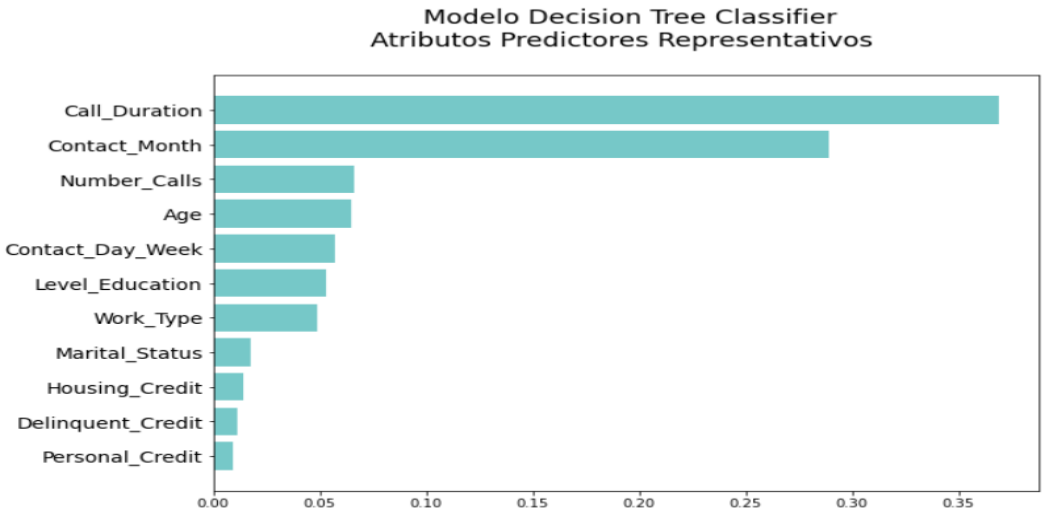


Ilustración 68. Atributos representativos en modelo Decision Tree
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 68 se puede observar que las variables más representativas o de mayor impacto hacia la variable objetivo en el modelo de árbol de decisión están dadas en la duración de la llamada, el mes en que se contactó al cliente y el número de llamadas.

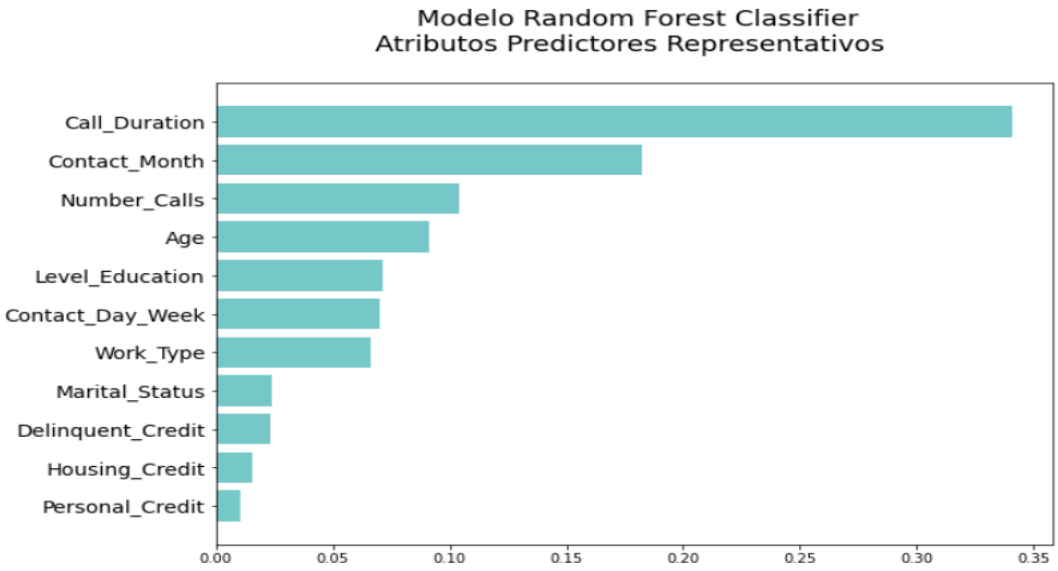


Ilustración 69. Atributos representativos en modelo Random Forest
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 69 se puede observar que las variables más representativas o de mayor impacto hacia la variable objetivo en el modelo de bosque aleatorio están dadas en la duración de la llamada, el mes en que se contactó al cliente y el número de llamadas realizadas.

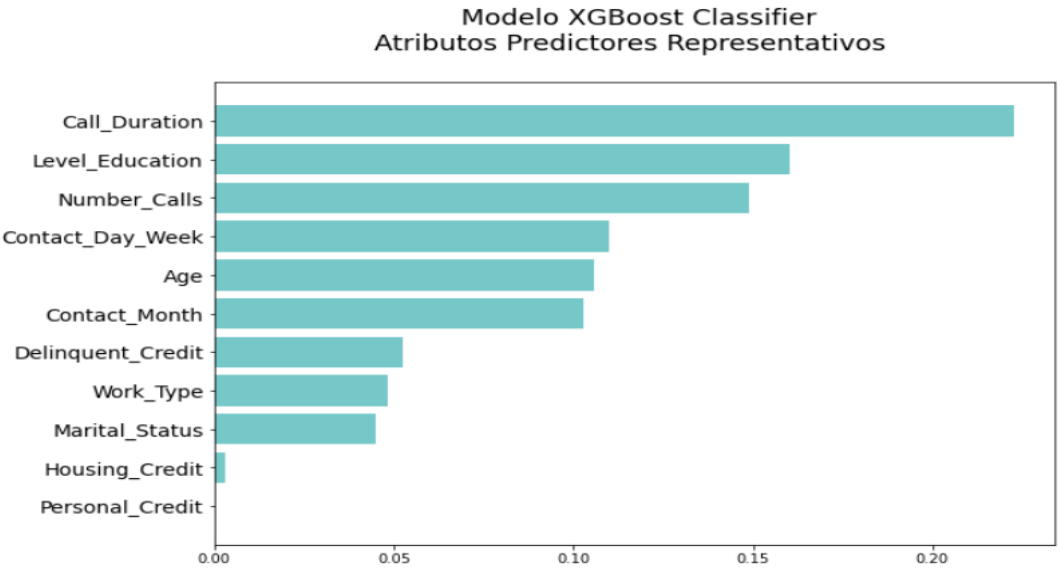


Ilustración 70. Atributos representativos en modelo XGBoost
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 70 se puede observar que las variables más representativas o de mayor impacto hacia la variable objetivo en el modelo xgboost están dadas en la duración de la llamada, el nivel de educación del cliente y el número de llamadas realizadas.

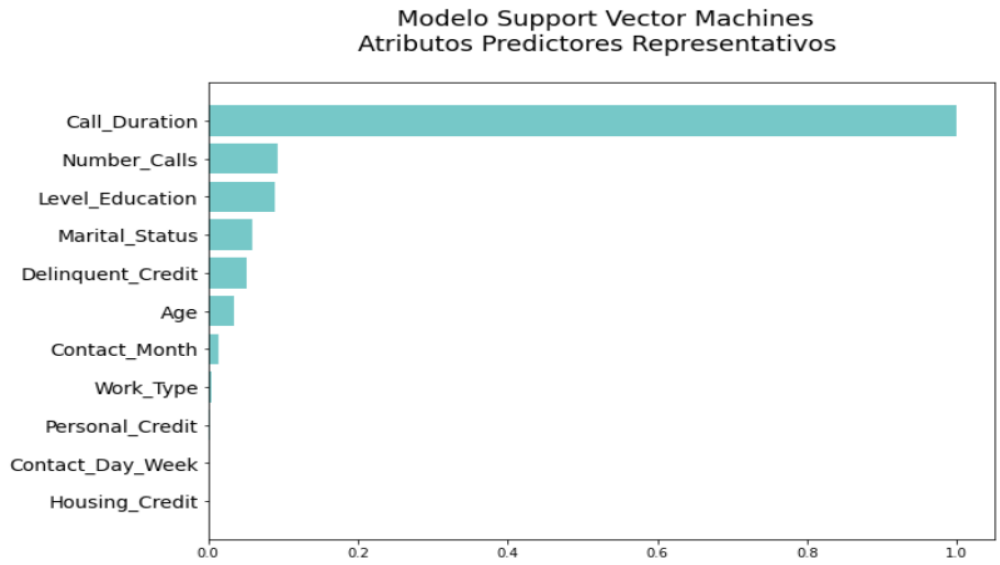


Ilustración 71. Atributos representativos en modelo Support Vector Machines
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 71 se puede observar que las variables más representativas o de mayor impacto hacia la variable objetivo en el modelo de máquinas de vectores de soporte están dadas en la duración de la llamada y el número de llamadas realizadas y el nivel de educación del cliente.

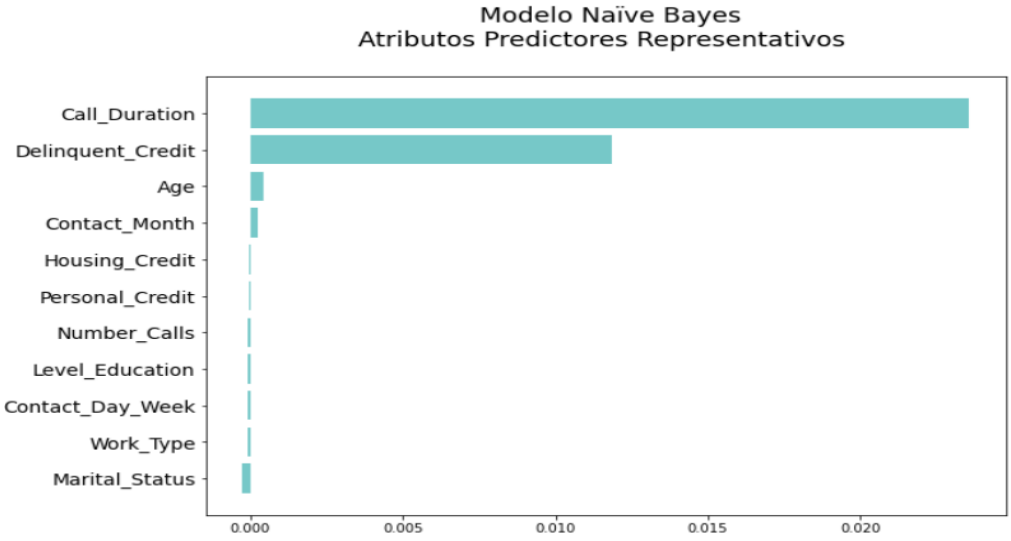


Ilustración 72. Atributos representativos en modelo Naïve Bayes
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 72 se puede observar que las variables más representativas o de mayor impacto hacia la variable objetivo en el modelo bayesiano están dadas en la duración de la llamada, si tiene o no un crédito en mora y la edad del cliente.

7.3.3.3. SELECCIÓN DE HIPERPARAMETROS

Una de las características de vital importancia en los modelos de machine learning es definir correctamente cuáles serán los mejores valores o parámetros de los hiperparametros. Esto ayudará indudablemente a mejorar y ajustar el estimador del modelo del conjunto de datos de entrenamiento.

Para este caso se usará el paquete Grid Search CV, el cual permitirá recorrer los hiperparametros de cada uno de los modelos para obtener los mejores valores que se emplearán en el modelo final.

En la tabla 15 y 16 se representarán los valores establecidos para los hiperparametros contenidos en cada uno de los modelos a evaluar.

Tabla 15. Selección hiperparametros modelo Random Forest

PARÁMETRO	VALOR DESIGNADO	DESCRIPCIÓN
n_estimators	150	Número de árboles en el bosque.
criterion	Gini	Función para medir la calidad de una división.
max_depth	None	Profundidad máxima del árbol. Con el valor None se garantiza que los nodos se expandan hasta que todas las hojas sean puras.
min_samples_split	2	Número mínimo de muestras necesario para dividir un nodo interno.
min_samples_leaf	1	Número mínimo de muestras requeridas para estar en un nodo hoja.
max_features	10	Cantidad máxima de atributos que se consideran para partir un nodo.
max_leaf_nodes	None	Cultiva árboles de la mejor manera. Con el valor None se genera un número ilimitado de nodos en una hoja.
random_state	0	Controla la aleatoriedad del arranque de las muestras utilizadas al construir los árboles.

Tabla 16. Selección hiperparametros modelo XGBoost

PARÁMETRO	VALOR DESIGNADO	DESCRIPCIÓN
n_estimators	150	Número de etapas de impulso a realizar.
learning_rate	0.3	La tasa de aprendizaje que reduce la contribución de cada árbol.
max_depth	5	Profundidad máxima de los estimadores
subsample	0.5	Tomará muestras aleatorias de la mitad de los datos de entrenamiento antes de cultivar árboles para evitar el sobreajuste.
sampling_method	uniform	Método que se va a utilizar para muestrear las instancias de entrenamiento. Con el valor uniform cada instancia de entrenamiento tiene la misma probabilidad de ser seleccionada.
colsample_bytree	1	Es la relación de submuestra de columnas al construir cada árbol.

scale_pos_weight	1	Controlar el equilibrio de pesos positivos y negativos, útil para clases desequilibradas.
max_leaves	5	Número máximo de nodos que se agregarán.

En este caso se evaluaron únicamente los dos modelos que mejor comportamiento han tenido en el transcurso de esta fase y que mejor se ajustaron a los objetivos contenidos en el desarrollo de este proyecto.

De acuerdo con los resultados obtenidos una vez establecidos los mejores hiperparametros para cada uno de los modelos, se concluye que el mejor modelo para implementar sobre el conjunto de datos de este proyecto será el Random Forest Classifier.

El modelo Random Forest obtuvo un 93.58% de puntuación con los mejores hiperparametros establecidos en comparación con el 90.85% de puntuación que obtuvo el modelo XGBoost con los mejores hiperparametros establecidos.

7.4. EVALUACIÓN

A partir de los análisis comparativos entre todos los modelos evaluados en la fase anterior y de acuerdo con los resultados obtenidos en los indicadores de desempeño de cada uno de los algoritmos, se decidió utilizar el algoritmo de clasificación Random Forest como el modelo final.

7.4.1. EVALUACIÓN DE RESULTADOS

En este apartado se entrenará nuevamente el modelo final (Random Forest Classifier) con los mejores hiperparametros establecidos en la fase anterior.

Posterior a ello, sobre el modelo final se aplicará la estrategia ***k-fold cross validation*** también conocida como validación cruzada la cual permitirá:

- Entrenar y probar el modelo final en diferentes o múltiples bloques en el conjunto de datos.
- Identificar si el modelo final se encuentra bajo un problema de sobreajuste (overfitting).

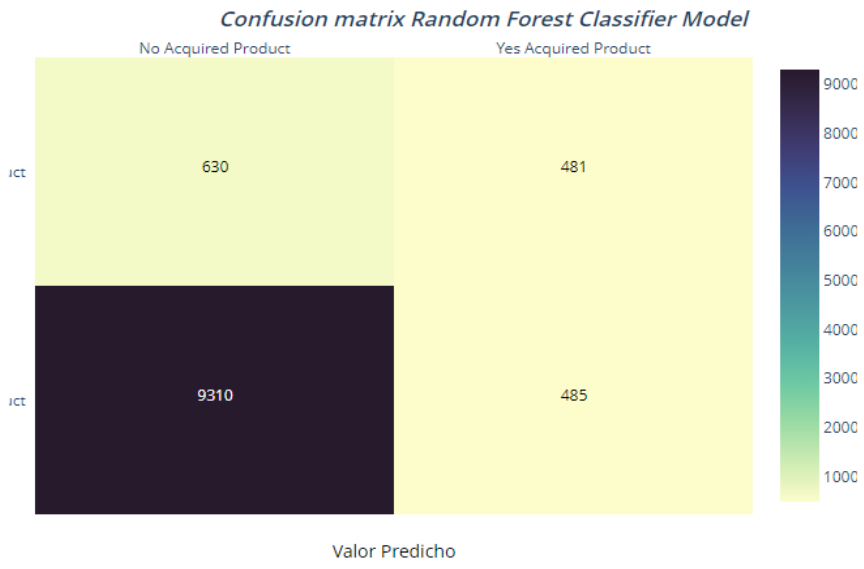
- Validar que el modelo final pueda generalizarse en cada uno de los bloques o pliegues que fueron entrenados y probados con la estrategia de validación cruzada.

Los parámetros a utilizar en esta estrategia (validación cruzada) estarán dados por 10 bloques o divisiones de entrenamiento y prueba y se utilizará como métrica de puntuación para cada uno de los bloques o divisiones el indicador f1.

Como resultado final sobre cada uno de los bloques entrenados y probados bajo esta estrategia de validación cruzada, se tuvo un promedio de puntuación f1 del 93.58% y una desviación estándar del 8.56% entre cada uno de los bloques o pliegues entrenados y probados.

7.4.1.1. MATRIZ DE CONFUSIÓN MODELO FINAL

Se analizará nuevamente la matriz de confusión del modelo final (Random Forest), para identificar posibles mejoras en las predicciones realizadas sobre la clase que más nos interesa en este proyecto (Cliente adquiere producto financiero).



	precision	recall	f1-score	support
0	0.94	0.95	0.94	9795
1	0.50	0.43	0.46	1111
accuracy			0.90	10906
macro avg	0.72	0.69	0.70	10906
weighted avg	0.89	0.90	0.89	10906

Ilustración 73. Matriz de confusión y métricas de evaluación modelo Random Forest
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

De acuerdo con los resultados obtenidos en la ilustración 73 se puede indicar que hubo una mejora hacia la clase de interés que queremos abarcar en este proyecto al obtener una puntuación un poco más alta en comparación a la fase anterior en las métricas de evaluación reflejadas. Aquí se puede observar que en la matriz de confusión el algoritmo tuvo una desviación de error en la predicción en 1.115 registros (485 Falsos-Positivos y 630 Falsos-Negativos) de 10.906.

7.4.1.2. CURVA DE ROC Y AUC MODELO FINAL

Se analizará nuevamente la curva de ROC y área bajo la curva AUC del modelo final (Random Forest), para identificar que el equilibrio entre la sensibilidad (Verdaderos Positivos) y la especificidad (Falsos Positivos) de este modelo se haya mantenido.

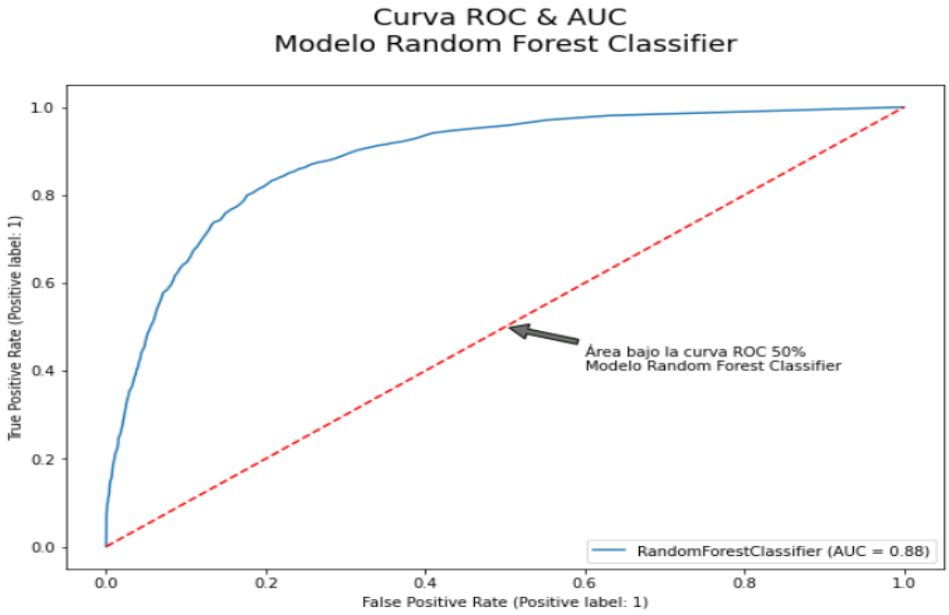


Ilustración 74. Curva ROC y AUC modelo Random Forest
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

La ilustración 74 representa que a pesar de que hubo un punto porcentual más bajo en el área bajo la curva AUC, esta mantiene un equilibrio alto entre la sensibilidad y la especificidad de este modelo, ya que su curva (en color azul) tiende a estar bastante cercana de la esquina superior izquierda, indicando un rendimiento muy favorable para los datos a predecir con el modelo final.

7.4.2. REVISIÓN DE PROCESOS

Como primer paso se ejecutará un proceso de validación de la predicción del modelo final (Random Forest) con respecto a los datos de prueba.

Posterior a esto, se crearán dos escenarios de prueba (unitarios), donde se pueda validar la predicción del modelo final con datos nuevos y desconocidos para este.

En la tabla 17 y 18 se representarán los valores establecidos para cada uno de los escenarios propuestos.

Tabla 17. Escenario de predicción No.1

ATRIBUTO	VALOR DESIGNADO	DESCRIPCIÓN
Age	25	Se estableció una edad de 25 años.
Work_Type	0	Se estableció el tipo de trabajo administrativo.
Marital_Status	2	Se estableció el estado civil soltero(a).
Level_Education	3	Se estableció el nivel de educación título universitario.
Delinquent_Credit	0	Se estableció sin crédito en mora.
Housing_Credit	0	Se estableció sin crédito de vivienda.
Personal_Credit	0	Se estableció sin crédito personal.
Contact_Month	5	Se estableció el mes de mayo de contacto.
Contact_Day_Week	5	Se estableció el día viernes de contacto.
Call_Duration	3	Se establecieron 3 minutos de duración en la llamada.
Number_Calls	2	Se establecieron 2 llamadas realizadas al cliente.

Tabla 18. Escenario de predicción No.2

ATRIBUTO	VALOR DESIGNADO	DESCRIPCIÓN
Age	55	Se estableció una edad de 55 años.
Work_Type	10	Se estableció el tipo de trabajo sin empleo.
Marital_Status	0	Se estableció el estado civil divorciado(a).
Level_Education	1	Se estableció el nivel de educación secundaria.
Delinquent_Credit	1	Se estableció con crédito en mora.
Housing_Credit	1	Se estableció con crédito de vivienda.
Personal_Credit	1	Se estableció con crédito personal.
Contact_Month	12	Se estableció el mes de diciembre de contacto.
Contact_Day_Week	2	Se estableció el día martes de contacto.
Call_Duration	1	Se estableció 1 minuto de duración en la llamada.
Number_Calls	3	Se establecieron 3 llamadas realizadas al cliente.

Como resultado de los 2 (dos) escenarios propuestos podemos concluir lo siguiente:

- En el escenario No.1 el resultado de la salida fue 1 (alta probabilidad que el cliente SI adquiera el producto financiero ofrecido).
- En el escenario No. 2 el resultado de la salida fue 0 (alta probabilidad que el cliente NO adquiera el producto financiero ofrecido).

7.4.3. GENERACIÓN DE REPORTE PROCESO FINAL

Como parte final de esta fase de la metodología CRISP-DM se intentará predecir dos escenarios de carga masiva de datos, con el objetivo de revisar las predicciones que realizará el modelo final (Random Forest) sobre nuevos datos ingeridos y así observar el comportamiento que tendrá el modelo final sobre este conjunto de datos a predecir.

Para el primer escenario de predicción de carga masiva de datos, se utilizará un conjunto de datos de prueba etiquetado con 4.119 registros, que nos brinda el repositorio UCI Machine Learning (UCI) para evaluar y comparar las predicciones finales realizadas por el modelo.

Age_x	Work_Type_x	Marital_Status_x	Level_Education_x	Delinquent_Credit_x	Housing_Credit_x	Personal_Credit_x	Contact_Month_x	Contact_Day_Week_x	Call_Duration_x	Number_Calls_x	Acquired_Product	Will_Acquired_Product
30	1	1	0	0	0	0	11	1	18.18	2	1	1
30	0	0	3	0	0	0	4	1	6.0	1	1	1
20	9	2	2	0	1	0	8	4	1.05	2	1	1
50	1	1	0	0	1	0	5	1	4.88	3	1	1
50	1	0	1	0	1	1	7	1	10.48	2	1	1
50	1	1	0	0	1	0	8	1	14.1	1	1	1
20	0	2	3	0	1	0	3	1	3.58	1	1	1
30	0	1	1	0	1	0	6	3	3.97	1	1	1
40	1	1	1	0	1	1	11	1	7.48	2	1	1
50	0	1	3	0	1	1	8	5	3.4	1	1	1
40	0	1	1	0	0	0	11	1	10.85	2	1	1
30	0	1	1	0	0	0	10	4	2.37	1	1	1
40	0	2	3	0	1	0	6	1	4.05	4	1	1
40	0	1	3	0	0	0	7	1	22.67	3	1	1
30	0	1	3	0	1	0	11	2	1.07	1	1	1
20	0	2	1	0	1	1	3	3	3.73	1	1	1
50	0	2	3	0	0	0	5	1	5.62	1	1	1
40	0	1	3	0	1	1	8	3	4.8	1	1	1
30	0	1	3	0	1	0	9	4	4.53	2	1	1
40	1	0	0	1	0	0	7	1	17.75	3	1	1
30	0	1	3	0	0	0	6	4	2.02	4	1	1
20	0	2	3	0	0	0	8	1	11.93	2	1	1
50	0	0	1	1	0	1	7	1	10.5	3	1	1

Ilustración 75. Predicción carga masiva de datos prueba UCI
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 75 podemos identificar la columna ya predicha *Acquired_Product* (proveniente del conjunto de datos de prueba del repositorio UCI) y así mismo identificar la columna *Will_Acquired_Product* (datos predichos por el modelo).

De los 4.119 registros nuevos que se le brindaron al modelo final, el 69% (283 datos) de estos fueron predichos correctamente (clientes aceptarían el producto financiero ofrecido y lo aceptaron) y el 31% (125 datos) restante fueron predichos con que no aceptarían el producto financiero ofertado pero en este caso sí aceptaron el producto financiero ofertado.

Por otro lado, para las predicciones de los clientes que no aceptarían el producto financiero y realmente no lo aceptaron se tuvo un porcentaje correcto del 32% (1.049 datos) y el 68% (2.250 datos) restante fueron predichos con que aceptarían el producto financiero ofertado pero en este caso no aceptaron el producto financiero ofertado.

Para el segundo escenario de predicción de carga masiva de datos, se utilizará un conjunto de datos propio (creado sin etiquetado) con 10.000 registros, para evaluar las predicciones finales realizadas por el modelo.

Age	Work_Type	Marital_Status	Level_Education	Delinquent_Credit	Housing_Credit	Personal_Credit	Contact_Month	Contact_Day_Week	Call_Duration	Number_Calls	Will_Acquired_Product
20	8	2	0	1	1	1	2	4	2	3	1
52	9	1	2	1	1	0	2	1	3	5	0
57	10	2	3	1	0	1	4	3	4	1	1
32	10	1	2	0	0	1	10	4	2	5	0
35	7	1	0	0	1	0	3	4	4	5	1
20	0	1	0	1	0	0	10	1	3	2	1
44	2	2	0	1	0	1	4	2	3	4	1
36	7	1	0	1	0	0	12	2	4	3	1
54	6	0	3	1	1	0	2	4	4	4	0
26	6	1	1	0	1	1	9	2	1	1	0
52	0	0	2	0	0	1	7	1	3	5	1
23	3	1	3	1	0	0	5	2	5	3	0
23	7	1	2	0	0	0	4	3	4	5	0
31	0	1	1	0	1	1	4	1	2	5	1
35	4	0	0	0	0	1	3	2	4	4	1
23	3	0	3	0	1	1	4	3	6	2	1
47	7	1	3	1	0	0	8	4	2	2	0
21	3	1	1	0	0	1	3	5	1	5	0
53	3	0	3	1	0	0	5	1	1	4	1

Ilustración 76. Predicción carga masiva datos nuevos
Fuente: Elaboración propia generada desde Python a partir de la información recuperada del repositorio UCI Machine Learning

En la ilustración 76 podemos identificar en la columna *Will_Acquired_Product* algunos escenarios donde el modelo Random Forest predijo a que clientes probablemente se les pueda ofertar el producto financiero para que estos lo adquieran. Así mismo se realizó las predicciones sobre los clientes que probablemente no aceptaran el producto financiero ofertado.

De los 10.000 registros nuevos que se le brindaron al modelo final, el 65% (6.513) de estos fueron predichos con que los clientes aceptarían el producto financiero ofrecido y el 35% (3.486) restante fueron predichos con que no aceptarían el producto financiero ofertado.

7.5. DESPLIEGUE

En esta última fase de la metodología CRISP-DM es importante poner en marcha el modelo predictivo construido en etapas anteriores, que haga parte de los procesos productivos de la organización.

7.5.1. PLANIFICACIÓN DESPLIEGUE DEL MODELO

Como primer paso se definirá la creación de un plan de despliegue que permita implementar el modelo de machine learning de manera adecuada y correcta.

Aquí se emplearon algunas técnicas, herramientas y entornos que permitieran establecer la conexión directa entre el modelo final construido y la interfaz gráfica que se utilizará para predecir el resultado final de este proyecto en un entorno web y público.

En la ilustración 77 se muestra la arquitectura a alto nivel empleada para el despliegue del modelo final.

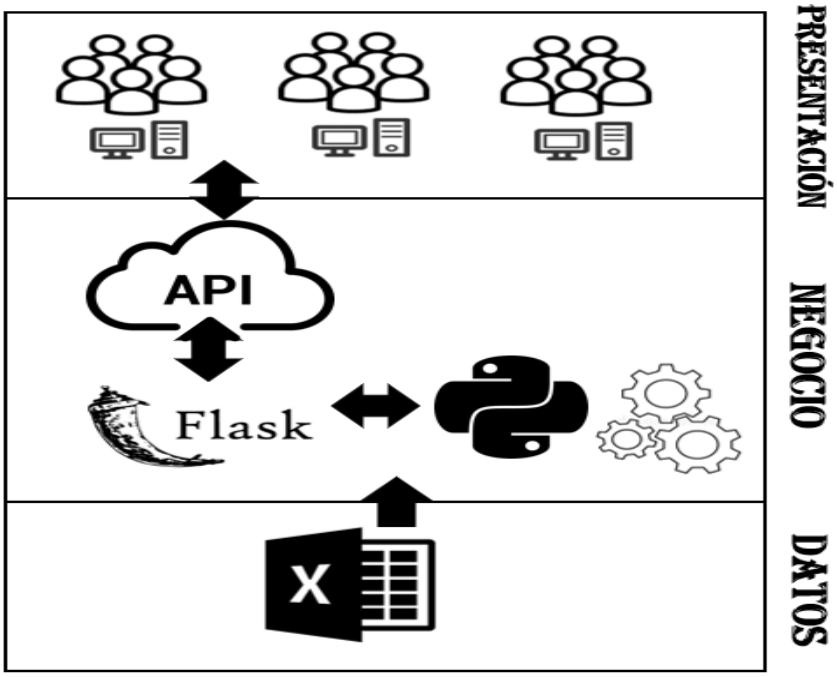


Ilustración 77. Arquitectura alto nivel despliegue (Fuente: Elaboración propia, 2022)

En la tabla 19 se hará una breve descripción de cada una de las capas que fueron utilizadas para implementar el despliegue del modelo de machine learning en el entorno web.

Tabla 19. Capas de arquitectura

Capa	Descripción	Herramienta a Consumir
Datos	Aquí se encuentra el conjunto de datos que contiene toda la información referente a las campañas de marketing directo con los datos con los que fue entrenado al modelo.	Excel
Negocio	Aquí se encuentra todo el proceso y la lógica desarrollada para la construcción del modelo final. Este desarrollo es expuesto a un servidor en la nube a través de una API para brindar el servicio web a los usuarios que realizaran las predicciones planteadas en este proyecto.	Python Flask Google Cloud
Presentación	Es la aplicación web que consumirá los usuarios finales.	Navegadores

7.5.2. PLANIFICACIÓN SEGUIMIENTO Y MANTENIMIENTO

Como segundo paso se deberá planificar el monitoreo y mantenimiento (si aplica) al modelo de machine learning implementado, que permita observar cómo se está comportando el modelo en un ambiente productivo ante nuevos datos que se le están ingiriendo frecuentemente y así poder tomar acciones correctivas que tal vez ayuden a mejorar posibles afectaciones de predicción que esté causando el modelo.

7.5.3. ELABORACIÓN INFORME FINAL

Como tercer paso se deberá realizar la elaboración de un informe final, que recolecte información fundamental acerca del comportamiento del modelo puesto en producción. Todo esto con el objetivo de garantizar una correcta documentación del modelo expuesto en el ambiente productivo para tomar posteriores decisiones evidenciadas en el comportamiento del modelo.

7.5.4. CREACIÓN DOCUMENTO TÉCNICO DEL MODELO

Como cuarto y último paso, se deberá realizar un instructivo o documento técnico del modelo que permita tener una guía para direccionar posibles causales nuevas que puedan determinar si alguna persona o cliente pueda adquirir un producto financiero ofertado por la entidad bancaria.

En el siguiente enlace estará disponible la aplicación web desplegada para dar cumplimiento a los objetivos propuestos en este proyecto:

https://apicliprofinbanc.uc.r.appspot.com

Ilustración 78. Aplicación web modelo predictivo desplegado (Fuente: Elaboración propia, 2022)

En la ilustración 78 se puede observar la aplicación web expuesta públicamente que permitirá predecir si un cliente aceptará o no un producto financiero de acuerdo con una serie de características (variables de entrada).

En este caso se realizaron algunas pruebas, ingresando diferentes tipos de valores sobre las cajas de texto y menús desplegables de la aplicación y efectivamente se pudo observar que el modelo final (Random Forest Classifier) está arrojando la predicción en la parte inferior de la pantalla subrayado en color verde lima.

8. CRONOGRAMA DE TRABAJO

Para este proyecto se creó un plan de trabajo basado en las etapas de la metodología CRISP-DM y se plantea las siguientes actividades que se llevarán a cabo con una duración de 16 semanas, dando inicio en el mes de agosto de 2022 y finalizando en el mes de noviembre aproximadamente.

CRONOGRAMA PROYECTO 2022																				
No	FASES DEL PROYECTO	ACTIVIDAD	MES	AGOSTO				SEPTIEMBRE					OCTUBRE					NOVIEMBRE		
			INICIO-FIN (SEMANA)	8-12	15-19	22-26	29-2	5-9	12-16	19-23	26-30	3-7	10-14	17-21	24-28	31-4	7-11	14-18	21-25	
1	FASE 1 ENTENDIMIENTO DEL NEGOCIO	Entender y describir el negocio																		
2		Evaluar situación actual y establecer los objetivos del modelo																		
3		Elaborar un plan de proyecto																		
1	FASE 2 COMPRENSIÓN DE LOS DATOS	Recopilación de datos iniciales																		
2		Descripción de los datos																		
3		Exploración de datos																		
4		Calidad de los datos																		
1	FASE 3 PREPARACIÓN DE LOS DATOS	Selección de los datos																		
2		Limpieza de los datos																		
3		Integración de los datos																		
1	FASE 4 MODELAMIENTO	Selección técnica de modelado																		
2		Diseño de la evaluación																		
3		Construcción de modelos																		
4		Evaluación de modelos																		
1	FASE 5 EVALUACIÓN	Evaluación de resultados																		
2		Revisión de proceso																		
3		Generación de reporte final																		
1	FASE 6 DESPLIEGUE	Planificación de despliegue del modelo																		
2		Planificación de seguimiento y mantenimiento																		
3		Elaboración informe final																		
4		Creación documento técnico del modelo																		

Ilustración 79. Cronograma del proyecto 2022 (Fuente: Elaboración propia, 2022)

9. PRESUPUESTO

El presupuesto proyectado para el desarrollo de este proyecto en el año 2022 es el siguiente:

PRESUPESTO PROYECTO 2022									
Descripción	Detalle	Roles	Cantidad	Horas X Semana	Semanas Participación	Tipo Unidad	Valor	SubTotal
Recursos Humanos	Horas de levantamiento de información, desarrollo, pruebas, documentación, implementación y honorarios docente	Ingeniero de Proyectos	1	15	16	Hora	$ 7.000	$ 1.680.000
		Ingeniero de Desarrollo	1	20	12		$ 10.000	$ 2.400.000
		Ingeniero QA	1	20	6		$ 6.000	$ 720.000
		Ingeniero Documentador	1	15	7		$ 4.000	$ 420.000
		Honorarios Docente	1	4	10		$ 12.500	$ 500.000
Recursos Bibliográficos	Costos de conexión a internet		4		16	Mes	$ 80.000	$ 1.280.000
Recursos Técnicos	Computador Escritorio		4		16	Indefinido	$ 1.800.000	$ 7.200.000
	Azure Machine Learning Studio	Herramientas con suscripciones gratuitas	2		16	Indefinido	$ -	$ -
	Python				16			
Otros Recursos	Papelería					Indefinido	$ 200.000	$ 200.000
	Transporte						$ 300.000	$ 300.000
	Electricidad						$ 400.000	$ 400.000
	Improvistos						$ 400.000	$ 400.000
						Valor Total	$ 3.219.500	$ 15.500.000

Ilustración 80. Presupuesto del proyecto 2022 (Fuente: Elaboración propia, 2022)

10. CONCLUSIONES

Este trabajo fue muy importante y fundamental para mí como estudiante, ya que mediante este, pude comprender conceptos interesantes en referencia a los diferentes algoritmos que se agrupan dentro del marco de aprendizaje automático supervisado.

Con este proyecto logre adquirir un conocimiento fundamental de la importancia de los modelos de machine learning para automatizar los procesos operativos y su evolución hacia fuentes importantes de información.

Esto sirve como base de apoyo a los niveles medio y alto gerenciales para la toma de decisiones. De la mano con la tecnología, la administración de una empresa lleva consigo el manejo óptimo de toda la información referente a las campañas de marketing directo.

Antes de la implementación del modelo de aprendizaje supervisado, abarque todo el ciclo de vida que conlleva una metodología como CRISP-DM, iniciando principalmente por la exploración de los datos mediante análisis descriptivos, haciendo uso de diferentes tipos de gráficas univariados y compuestos para ver todo el comportamiento de los datos contenidos en cada una de las variables.

Posterior a ello, abarque la fase de preprocesamiento realizando una limpieza sobre los datos de acuerdo con los análisis realizados en la fase anterior, eliminando algunas variables que eran irrelevantes para el modelo, categorizando atributos e implementando técnicas de codificación para hacer uso correcto de las variables categóricas.

Luego construí algunos algoritmos de clasificación de aprendizaje automático supervisado aplicándoles algunas métricas e indicadores que evaluaran el rendimiento de cada uno de los modelos, con el objetivo de seleccionar el que mejor se ajustara al conjunto de datos trabajado en este proyecto y que permitiera predecir si un cliente aceptarían el producto financiero ofertado por la entidad bancaria.

De todos los modelos evaluados, decidí utilizar el algoritmo Random Forest Classifier como modelo final, ya que tenía métricas de evaluación bastante altas en comparación a los demás algoritmos evaluados. A partir del análisis realizado a este modelo, la duración de la llamada, el mes en que se contacta a un cliente y el número de llamadas que se le realizan a dicho cliente tienen el papel principal para determinar la salida (predicción) del modelo final.

Posterior a la elección del modelo, con el uso de algunas herramientas (Python, Librería Flask, HTML, API Google Cloud Plataform) logre desarrollar y exponer públicamente un aplicativo web que consumiera el modelo final desarrollado (Random Forest Classifier) y permitiera predecir si un cliente aceptaría o no un producto financiero de acuerdo con una serie de características (variables de entrada).

Finalmente, puedo concluir que los modelos de machine learning son muy útiles como una herramienta para obtener ventajas competitivas mediante su implantación y uso, apoyando el máximo nivel de la organización.

11. TRABAJOS FUTUROS

Con base al desarrollo de este proyecto, es posible seguir mejorando algunos puntos importantes y fundamentales que conlleva una metodología en la ciencia de datos, aplicando nuevos modelos que no se abordaron en el desarrollo de este trabajo para afianzar aún más las métricas o indicadores de desempeños que estos conllevan.

Se puede contemplar un conjunto de datos más balanceado con variables externas que puedan ser importantes para un tema de marketing directo y que sirva como base para mejorar el desempeño de los modelos de machine learning. Así mismo se desea, que si en dado caso este trabajo se tome como base o insumo a futuro, se pueda ir evaluando la capacidad predictiva del modelo implementado aquí e ir haciendo ajustes a este en caso de ser necesario.

En la fase de despliegue se puede mejorar la aplicación web expuesta o crear una nueva que permita realizar predicciones en lotes, es decir consumir varios registros a la vez y que esta sea capaz de generar una interfaz o un archivo con los resultados de las predicciones hechas por el modelo desarrollado. Es importante realizar un seguimiento continuo y un mantenimiento al modelo cada determinado tiempo para nuevas campañas de marketing directo, así mismo la elaboración de un informe final y la creación de un documento técnico sobre el modelo desplegado en el ambiente productivo.

12. REFERENCIAS BIBLIOGRÁFICAS

[1] Sendinblue, "¿Qué es el marketing directo? Ventajas, canales y ejemplos - Sendinblue," *España*, Aug. 27, 2021. https://es.sendinblue.com/blog/marketing-directo/ (accessed Mar. 31, 2022).

[2] A. Burkov, "The Hundred Page Machine Learning," *Computer (Long. Beach. Calif).*, vol. 2005, no. April, pp. 40–48, 1997, Accessed: Mar. 31, 2022. [Online]. Available: https://books.google.com/books/about/Machine_Learning.html?hl=es&id=EoYBngEACAAJ

[3] IBM, "What is Machine Learning? | IBM," Jul. 15, 2020. https://www.ibm.com/cloud/learn/machine-learning (accessed Mar. 31, 2022).

[4] A. Moreno *et al.*, "Aprendizaje automático," p. 342, 1994, Accessed: May 21, 2022. [Online]. Available: https://upcommons.upc.edu/handle/2099.3/36157

[5] Q. Bi, K. E. Goodman, J. Kaminsky, and J. Lessler, "What is Machine Learning? A Primer for the Epidemiologist," *Am. J. Epidemiol.*, vol. 188, no. 12, pp. 2222–2239, Dec. 2019, doi: 10.1093/AJE/KWZ189.

[6] I. Muhammad and Z. Yan, "SUPERVISED MACHINE LEARNING APPROACHES: A SURVEY," p. 947, 2015, doi: 10.21917/ijsc.2015.0133.

[7] V. Nasteski, "An overview of the supervised machine learning methods," p. 4, 2017, doi: 10.20544/HORIZONS.B.04.1.17.P05.

[8] J. R. Quinlan, "Induction of Decision Trees," *Mach. Learn.*, vol. 1, pp. 81–106, 1986, Accessed: Mar. 31, 2022. [Online]. Available: https://link.springer.com/content/pdf/10.1007/BF00116251.pdf

[9] L. Breiman, "Random Forests," vol. 45, pp. 5–32, 2001, Accessed: Mar. 31, 2022. [Online]. Available: https://link.springer.com/content/pdf/10.1023/A:1010933404324.pdf

[10] M. P. LaValley, "Logistic Regression," *Circulation*, vol. 117, no. 18, pp. 2395–2399, May 2008, doi: 10.1161/CIRCULATIONAHA.106.682658.

[11] DATA SCIENCE, "XGBoost," 2019. https://datascience.eu/es/programacion/xgboost-4/ (accessed Oct. 14, 2022).

[12] Gonzalez Lidgi, "Aprendizaje Supervisado: Support Vector Machine," Mar. 23, 2018. https://aprendeia.com/aprendizaje-supervisado-support-vector-machine/ (accessed Oct. 14, 2022).

[13] K. M. Leung, "Naive Bayesian Classifier," Nov. 2007.

[14] Google, "Google Colab," 2015. https://research.google.com/colaboratory/faq.html?authuser=3&hl=es (accessed Oct. 14, 2022).

[15] Data Science Team, "Matriz de confusión," 2020. https://datascience.eu/es/aprendizaje-automatico/matriz-de-confusion/ (accessed Oct. 14, 2022).

[16] Data Science Team, "¿Qué es una Matriz de Correlación?," 2020. https://datascience.eu/es/matematica-y-estadistica/que-es-una-matriz-de-correlacion/ (accessed Oct. 14, 2022).

[17] P. F. López de Ullibarri Galparsoro I, "Curvas ROC," Sep. 25, 2001. http://webpersonal.uma.es/~jmpaez/websci/BLOQUEI/DocuI/roc.pdf (accessed Oct. 14, 2022).

[18] Gonzalez Ligdi, "Curvas ROC y Área bajo la curva (AUC)," May 31, 2019. https://aprendeia.com/curvas-roc-y-area-bajo-la-curva-auc-machine-learning/ (accessed Oct. 14, 2022).

[19] scikit-learn, “3.3. Metrics and scoring: quantifying the quality of predictions,” 2007. https://scikit-learn.org/stable/modules/model_evaluation.html#scoring-parameter (accessed Oct. 14, 2022).
[20] scikit-learn, “sklearn.preprocessing.LabelEncoder,” 2007. https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.LabelEncoder.html (accessed Oct. 14, 2022).
[21] scikit-learn, “sklearn.preprocessing.StandardScaler,” 2007. https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html (accessed Oct. 14, 2022).
[22] DATA SCIENCE, “SMOTE,” 2020. https://datascience.eu/es/programacion/smote/ (accessed Oct. 14, 2022).
[23] Analytics Lane, “GridSearchCV,” 2019. https://www.analyticslane.com/2018/07/02/gridsearchcv/ (accessed Oct. 14, 2022).
[24] P. C. & P. R. S. Moro, “UCI Machine Learning Repository: Bank Marketing Data Set,” 2014. http://archive.ics.uci.edu/ml/datasets/Bank+Marketing# (accessed Apr. 03, 2022).
[25] G. E. Economía, V. López Güemes Directora, and E. Rocío Rocha Blanco, “TRABAJO DE FIN DE GRADO BUSINESS INTELLIGENCE PARA LA TOMA DE DECISIONES ESTRÁTEGICAS: UN CASO DE APLICACIÓN DE MINERÍA DE DATOS DENTRO DEL SECTOR BANCARIO,” 2019, Accessed: Apr. 03, 2022. [Online]. Available: https://repositorio.unican.es/xmlui/handle/10902/17546
[26] Cerda Walter, “Aumento de la contactabilidad de campañas de marketing directo en base al diseño y construcción de un data mart de contactos de clientes de Banco Falabella,” 2016. https://repositorio.uchile.cl/handle/2250/138657 (accessed Apr. 03, 2022).
[27] N. Villón Cabrera, “Inteligencia Artificial aplicada al marketing: Impacto del uso de Chatbots Cognitivos en la satisfacción del cliente del sector bancario,” 2020, Accessed: Apr. 03, 2022. [Online]. Available: http://hdl.handle.net/10757/652700
[28] Triviño Maria, “MACHINE LEARNING MODEL FOR EFFECTIVENESS EVALUATION OF A DIGITAL MARKETING STRATEGY,” 2021, Accessed: Apr. 03, 2022. [Online]. Available: https://repository.ucatolica.edu.co/bitstream/10983/26706/1/MODELO DE MACHINE LEARNING PARA LA EVALUACIÓN DE EFECTIVIDAD DE UNA ESTRATEGIA DE MARKETING DIGITAL.pdf
[29] Chávez André, “Modelos de machine learning para identificar factores asociados a la adquisición de un crédito efectivo en una entidad financiera,” 2021. http://cybertesis.unmsm.edu.pe/bitstream/handle/20.500.12672/17342/Chavez_pa.pdf?sequence=1&isAllowed=y (accessed Apr. 03, 2022).
[30] Linares Rosa, “Análisis del marketing relacional del sector banca exclusiva de una determinada entidad bancaria,” 2019, Accessed: Apr. 03, 2022. [Online]. Available: https://repositorio.upn.edu.pe/bitstream/handle/11537/22160/Ramirez Linares%2C Rosa Alejandra Cruz.pdf?sequence=6&isAllowed=y
[31] L. Sing’oei and J. Wang, “Data Mining Framework for Direct Marketing: A Case Study of Bank Marketing,” 2013, Accessed: Apr. 03, 2022. [Online]. Available: www.IJCSI.org
[32] T. Parlar and S. K. Acaravci, “Using Data Mining Techniques for Detecting the Important Features of the Bank Direct Marketing Data,” *Int. J. Econ. Financ. Issues*, vol. 7, no. 2, pp. 692–696, 2017, Accessed: Apr. 03, 2022. [Online]. Available: http:www.econjournals.com
[33] K. Wisaeng, “A Comparison of Different Classification Techniques for Bank Direct Marketing,” vol. 3, Sep. 2013, Accessed: Apr. 03, 2022. [Online]. Available:

https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.412.9849&rep=rep1&type=pdf
[34] S. Abbas, “Deposit subscribe Prediction using Data Mining Techniques based Real Marketing Dataset,” *Int. J. Comput. Appl.*, vol. 110, no. 3, pp. 1–7, 2015, doi: 10.5120/19293-0725.
[35] R. Wirth and J. Hipp, “CRISP-DM: Towards a Standard Process Model for Data Mining,” 2015, Accessed: Mar. 31, 2022. [Online]. Available: http://www.cs.unibo.it/~danilo.montesi/CBD/Beatriz/10.1.1.198.5133.pdf