



UNIVERSIDAD MAYOR DE SAN SIMÓN
FACULTAD DE CIENCIAS Y TECNOLOGÍA



DIRECCIÓN DE POSGRADO

DIPLOMADO ESTADÍSTICA APLICADA A LA TOMA DE DECISIONES

TERCERA VERSIÓN

MODELO PREDICTIVO DE CLASIFICACIÓN DE CHURN BASADO EN PATRONES DE CONTACTO PARA UNA STARTUP BOLIVIANA

**PROYECTO PRESENTADO PARA OBTENER EL GRADO DE LICENCIATURA EN
INGENIERÍA INDUSTRIAL
MODALIDAD DOBLE TITULACIÓN**

POSTULANTE : FLAVIA MAYLIN DAVILA PEREZ

TUTOR : M.SC. ING. DANNY LUIS HUANCA SEVILLA

Cochabamba – Bolivia

2025

MODELO PREDICTIVO DE CLASIFICACIÓN DE CHURN BASADO EN PATRONES DE CONTACTO PARA UNA STARTUP BOLIVIANA

Por

Flavia Maylin Davila Pérez

El presente documento, Trabajo de Grado es presentado a la Dirección de Posgrado de la Facultad de Ciencias y Tecnología en cumplimiento parcial de los requisitos para la obtención del grado académico de Licenciatura (o sólo diplomado) en Ingeniería Industrial, modalidad Doble Titulación, habiendo cursado el Diplomado ‘Estadística Aplicada a la Toma de Decisiones’ propuesta por el Centro de Estadística Aplicada (CESA) en su tercera versión.

ASESOR/TUTOR

M.Sc. Ing. Danny Luis Huanca Sevilla

COMITÉ DE EVALUACIÓN

Ing. M.Sc. Ronald Edgar Patiño Tito. (Presidente)

Ing. M.Sc Guillen Salvador Roxana,. (Coordinador)

Ing. M.Sc Espinoza Orosco José (Tribunal)

Ing. M.Sc. Wilson Orlando Trujillo Aranibar (Tribunal)



DIRECCIÓN DE POSGRADO, FACULTAD DE CIENCIAS Y TECNOLOGIA

Cochabamba, Bolivia

Aclaración

Este documento describe el trabajo realizado como parte del programa de estudios de Diplomado ‘Estadística Aplicada a la Toma de Decisiones’ en el Centro de Estadística Aplicada CESA y la Dirección de Posgrado de la Facultad de Ciencias y Tecnología. Todos los puntos de vista y opiniones expresadas en el mismo son responsabilidad exclusiva del autor y no representan necesariamente las de la institución.

Resumen

Las startups bolivianas desempeñan un papel clave en el desarrollo económico y tecnológico del país, al ofrecer soluciones innovadoras en diversos sectores. Para una startup, reducir el riesgo de 'churn' es muy importante, ya que enfrentan un entorno altamente competitivo y en constante cambio.

En el contexto de la startup objeto de este proyecto, el 'Churn Comercial' se refiere a la pérdida de clientes ocurrida 90 días después de una conversión exitosa. El objetivo es identificar los factores clave en los patrones de contacto de los clientes para generar insights aplicables a otras startups similares en Bolivia, además de ofrecer recomendaciones estratégicas alineadas con la visión de la empresa, para mejorar la retención de clientes y fomentar su crecimiento.

Este proyecto se enfoca en el análisis predictivo de 'Churn Comercial' basado en patrones de contacto para una startup con un modelo de negocio B2B y un producto del tipo SaaS del estado plurinacional de Bolivia. La metodología utilizada en el desarrollo de este proyecto abarca desde la obtención de los datos hasta la implementación y mejora de modelos predictivos de machine learning.

Se extrajeron datos desde la API del CRM donde la startup boliviana aloja sus datos, aplicando el enfoque de procesamiento de datos de la arquitectura Medallion. Se tomaron en cuenta los datos de las actividades y características de los negocios, obtenidos entre 2021 y 2024, con un enfoque en los clientes del territorio boliviano.

Este proyecto fue desarrollado siguiendo los lineamientos y la estructura metodológica de minería de datos CRISP-DM. El cual comprende el entendimiento del negocio, la obtención de los datos y su entendimiento, su procesamiento para su posterior uso en el modelado, evaluación y mejora de modelos predictivos de churn comercial en una startup boliviana.

Los resultados obtenidos en las métricas de evaluación globales de los modelos entrenados demuestran la eficiencia tanto del modelo XGBoost como del Decision Tree al procesar una cantidad limitada de datos con clases binarias desbalanceadas. A pesar de que los modelos Naive Bayes y Random Forest también mostraron buenos resultados, XGBoost sobresale por su alta precisión en la diferenciación de clases, con un AUC-ROC de 95%, una precisión del 92%, un Recall del 96%, y un F1-Score del 83%, lo que es clave en bases de datos desbalanceadas. De manera similar, el modelo de Árbol de Decisión presenta un rendimiento sólido con un AUC-ROC del 90%, una precisión del 93%, un Recall del 90%, y un F1-Score del 83%. Estos resultados destacan la capacidad de XGBoost para predecir de manera efectiva el abandono de clientes, superando a otros modelos en cuanto a discriminación entre clases y la identificación de casos relevantes.

Palabras clave

Startup, churn, predicción, patrones de contacto, machine learning

*A mi luz, mi guía, pilar de valores y maestros en la vida, por estar a mi
lado en cada etapa de mi vida apoyándome y brindándome amor
incondicional, gracias mamá y papá.*

Agradecimientos

A mi persona por darme los ánimos y la motivación cada día de poder seguir adelante y poder lograr mis objetivos personales, por dar lo mejor de mí siempre y por mi resiliencia ante las situaciones más adversas

A mi familia, por brindarme su compañía y motivación a lo largo de esta etapa de mi vida.

A los amigos que fueron parte de este proceso, por motivarme y acompañarme en la realización de este proyecto y aportar con su positivismo a mi vida.

A la startup boliviana por confiarme sus datos y permitirme brindarles este proyecto como un aporte a su crecimiento.

A la prestigiosa Universidad Mayor de San Simón por permitirme formarme a lo largo de estos años y darme los mejores años académicos de mi vida, permitiéndome encontrarme y descubrir mis capacidades

A la Dirección de Posgrado de la Facultad de Ciencias y Tecnología por darme la oportunidad de cursar este posgrado y la oportunidad de aprender de primera mano de expertos en el área ayudándome a mejorar mis skills y darme herramientas con las que podré aportar al desarrollo de mi país.

Tabla de contenidos

1.	Introducción.....	1
1.1.	Antecedentes.....	1
1.2.	Justificación.....	3
1.3.	Planteamiento del problema.....	3
1.4.	Objetivo general.....	5
1.4.1.	Objetivos específicos.....	5
2.	Marco teórico.....	6
2.1.	Inteligencia Artificial (IA).....	6
2.2.	Machine Learning.....	6
2.3.	Modelos predictivos de Machine Learning.....	7
2.3.1.	Modelos predictivos de clasificación.....	7
2.3.2.	EDA (Exploratory Data Analysis).....	8
2.3.3.	Variable destino o etiqueta.....	9
2.4.	Modelo CRISP-DM (Cross Industry Standard Process for Data Mining).....	9
2.5.	StartUps.....	9
2.5.1.	Características de una startup.....	10
2.6.	Software As A Service (SaaS).....	10
2.7.	Ciclo de vida de un cliente.....	11
2.7.1.	Churn.....	11
2.8.	CRM (Customer Relationship Management).....	11
2.9.	Arquitectura Medallion.....	12
2.9.1.	Capas (Bronce, Plata y Oro).....	12
2.10.	Descripción estadística.....	13
2.10.1.	Media.....	13
2.10.2.	Mediana (Me).....	13
2.10.3.	Moda (Mo).....	13

2.10.4.	Mínimo (Min) y Máximo (Max).....	13
2.10.5.	Rango	13
2.10.6.	Varianza.....	13
2.10.7.	Desviación estándar	14
2.10.8.	Correlación.....	14
2.10.9.	Prueba Chi-Cuadrado	14
2.10.10.	Prueba ANOVA.....	14
2.11.	Python en la Ciencia de Datos	15
2.11.1.	Manipulación de datos con Python.....	15
2.11.2.	Aprendizaje Automático con Python.....	15
2.11.3.	Métricas de Evaluación con Python.....	16
2.11.4.	Métodos de ajuste y optimización	18
3.	Marco metodológico	19
3.1.	Área de estudio	19
3.2.	Flujograma metodológico	19
3.3.	Fuentes de información	21
3.3.1.	Fuente de datos primaria: PipeDrive CRM.....	21
3.3.2.	Fuentes de datos secundarias.....	21
3.4.	Entendimiento del negocio.....	21
3.4.1.	Objetivos del negocio	21
3.4.2.	Situación actual de la startup.....	22
3.4.3.	Procesos actuales de la empresa	22
3.4.4.	Recursos y Restricciones	23
3.5.	Extracción de datos.....	23
3.5.1.	Capas de la arquitectura Medallion.....	23
3.5.2.	Entendimiento de los datos	24
3.6.	Análisis y preparación de los datos	27
3.6.1.	Análisis Exploratorio de Datos (EDA)	27
3.6.2.	Preparación de los datos.....	50

3.7.	Entrenamiento de Modelos	51
3.7.1.	Análisis del problema.....	51
3.7.2.	Selección de Algoritmos.....	52
3.7.3.	Etapa Preliminar de Implementación	52
3.8.	Evaluación y selección del modelo	55
3.8.1.	Evaluación preliminar de modelos.....	55
3.8.2.	Selección del modelo	62
4.	Resultados y Discusión	63
4.1.	Entendimiento de las necesidades y objetivos.....	63
4.2.	Extracción y comprensión de los datos	64
4.3.	Análisis y proceso de datos.....	64
4.4.	Entrenamiento de Modelos	64
4.4.1.	Resultados de la Evaluación: Modelo Naive Bayes	65
4.4.2.	Resultados de Evaluación: Modelo Logistic Regression.....	67
4.4.3.	Resultados de la Evaluación: Modelo Decision Tree.....	69
4.4.4.	Resultados de la Evaluación: Modelo Random Forest	71
4.4.5.	Resultados de la Evaluación: Modelo XGBoost.....	73
4.4.6.	Resultados globales del entrenamiento	75
4.5.	Selección del modelo predictivo	79
4.5.1.	Importancia de Variables.....	79
4.6.	Discusión de resultados	80
5.	Conclusiones	83
6.	Recomendaciones	85
	Referencias bibliográficas.....	87
	Anexos.....	91
Anexo 1.	Estructura de trabajo: Arquitectura MEDALLION	91
Anexo 2.	Procesamiento de datos capa Bronce.....	92
Anexo 3.	Procesamiento de datos capa Plata (subcapa de estandarización).....	93
Anexo 4.	Procesamiento de datos capa Plata (subcapa de filtrado)	94

Anexo 5.	Procesamiento de datos capa Plata (subcapa de enriquecimiento)	95
Anexo 6.	Procesamiento de datos capa Oro.....	96
Anexo 7.	Análisis exploratorio de datos.....	97
Anexo 8.	Entrenamiento de Modelos.....	98
Anexo 9.	Data Frame resultante de la extracción de datos	99
Anexo 10.	CD	100

Lista de figuras

Figura 1-1 Comparación entre ‘Churn’ y ‘Churn Comercial’ a lo largo del tiempo.....	4
Figura 1-2 Comparación entre ‘Churn’ y ‘Churn Comercial’ Gestión 2024.....	5
Figura 2-1 Clasificación de algoritmos de Machine Learning.....	6
Figura 2-2 Ciclo de vida de un cliente vs el trabajo de Marketing en cada etapa.....	11
Figura 2-3 Estructura de la Arquitectura Medallion	12
Figura 2-2-4 Fuente: (Databricks, 2024).....	12
Figura 2-5 Grafico de Curva ROC	17
Figura 2-6 Grafico de Curva ROC	17
Figura 3-1 Mapa de distribución de clientes en Bolivia	19
Figura 3-2 Flujograma metodológico	20
Figura 3-3 Diagrama del ciclo de vida de un cliente en los procesos de la startup.....	22
Figura 3-4 Diagrama de Entidad Relación.....	25
Figura 3-5 Histograma de distribución de la variable ‘Total_Actividades_com’	33
Figura 3-6 Histograma de distribución de la variable ‘Total_Llamadas_com’	34
Figura 3-7 Histograma de distribución de la variable ‘Llamadas_Efectivas_com’	34
Figura 3-8 Histograma de distribución de la variable ‘Llamadas_No_Efectivas_com’	35
Figura 3-9 Histograma de distribución de la variable ‘WA_Seguimiento_com’	35
Figura 3-10 Histograma de distribución de la variable ‘Reuniones_Hechas’	36
Figura 3-11 Histograma de distribución de la variable ‘Reuniones_Canceladas’	36
Figura 3-12 Histograma de distribución de la variable ‘Total_Actividades_com’	37
Figura 3-13 Histograma de distribución de la variable ‘Total_Llamadas_exp’	37
Figura 3-14 Histograma de distribución de la variable ‘Llamadas_Efectivas_exp’	38
Figura 3-15 Histograma de distribución de la variable ‘Llamadas_No_Efectivas_exp’	38
Figura 3-16 Histograma de distribución de la variable ‘WA_Seguimiento_com’	39
Figura 3-17 Histograma de distribución de la variable ‘Kickoff_Hechas’	39
Figura 3-18 Histograma de distribución de la variable ‘Kickoff _Canceladas’	40
Figura 3-19 Histograma de distribución de la variable ‘Capacitaciones_Hechas’	40

Figura 3-20 Histograma de distribución de la variable ‘Capacitaciones_Canceladas’.....	41
Figura 3-21 Grafica de barras de ‘Tipo de cliente’	41
Figura 3-22 Grafica de barras de '(C) (EXP) Plazo y Pago'	42
Figura 3-23 Grafica de barras de ‘Tipo Primer Contacto’	42
Figura 3-24 Grafica de barras de ‘Rango de Contacto’.....	43
Figura 3-25 Grafica de barras de ‘R1yR2’	43
Figura 3-26 Grafica de barras de ‘Tipo Primera Capacitación’	44
Figura 3-27 Grafica de barras de ‘Onboarding’	44
Figura 3-28 Grafica de barras de ‘Churn Comercial’	45
Figura 3-29 Boxplots de ‘Llamadas_Efectivas_com’ vs ‘Churn Comercial’	45
Figura 3-30 Boxplots de ‘Total_Actividades_exp’ vs ‘Churn Comercial’	46
Figura 3-31 Boxplots de ‘Llamadas_No_Efectivas_exp’ vs ‘Churn Comercial’	46
Figura 3-32 Boxplots de Capacitaciones ‘Hechas’ y ‘Canceladas’ vs ‘Churn Comercial’	47
Figura 3-33 Matriz de correlación de Pearson entre variables numéricas.....	47
Figura 3-34 Matriz de correlación de Pearson entre variables numéricas seleccionadas.....	53
Figura 4-1 Comportamiento de ‘Churn Comercial’ en comparación a 'Churn' a lo largo del tiempo.....	63
Figura 4-2 Comportamiento de ‘Churn Comercial’ en comparación a 'Churn' Gestión 2024.....	63
Figura 4-3 Boxplots de total de actividades entre embudos vs Churn Comercial	64
Figura 4-4 Comparación de Precisión entre modelos antes y después de ajustes	65
Figura 4-7 Matriz de Confusión Modelo Naive Bayes	66
Figura 4-8 Curvas de aprendizaje Modelo Naive Bayes	67
Figura 4-9 Matriz de Confusión Modelo Logistic Regression	68
Figura 4-10 Curvas de aprendizaje Modelo Logistic Regression.....	69
Figura 4-11 Matriz de Confusión Modelo Decision Tree.....	70
Figura 4-12 Curvas de aprendizaje Modelo Decision Tree.....	71
Figura 4-13 Matriz de Confusión Modelo Random Forest.....	72
Figura 4-14 Curvas de aprendizaje Modelo Random Forest	73
Figura 4-15 Matriz de Confusión Modelo XGBoost.....	74
Figura 4-16 Curvas de aprendizaje Modelo XGBoost.....	75

Figura 4-5 Precisión Global entre Modelos.....	76
Figura 4-6 Comparación de AUC-ROC entre modelos.....	76
Figura 4-17 Comparación de métricas de evaluación entre modelos.....	81
Figura 4-18 Comparación de porcentaje de churn entre proyectos	81
Figura 4-19 Gráfica de barras de estados de cuentas.....	82

Lista de tablas

Tabla 3-1 Descripción de los datos.....	25
Tabla 3-2 Descripción de los campos de la tabla de la subcapa plata de enriquecimiento.	27
Tabla 3-3 Detalle de las variables cualitativas	27
Tabla 3-4 Detalle de las variables cuantitativas.....	28
Tabla 3-5 Detalle de las variables cuantitativas del embudo comercial.....	29
Tabla 3-6 Detalle de las variables cuantitativas del embudo experiencia.....	31
Tabla 3-7 Detalle de las variables cualitativas.....	33
Tabla 3-8 Prueba Chi-Cuadrado para variables cualitativas	48
Tabla 3-9 Prueba ANOVA para variables cuantitativas	49
Tabla 4-1 Métricas de Evaluación Modelo Naive Bayes	66
Tabla 4-2 Métricas de Evaluación Modelo Logistic Regression.....	68
Tabla 4-3 Metricas de Evaluación Modelo Decision Tree	70
Tabla 4-4 Métricas de Evaluación Modelo Random Forest	72
Tabla 4-5 Métricas de Evaluación Modelo XGBoost.....	74
Tabla 4-6 Métricas de evaluación de todos los modelos ‘macro avg’.....	77
Tabla 4-7 Métricas de evaluación de todos los modelos ‘weighted avg’.....	78
Tabla 4-8 Comparación de métricas de evaluación entre proyectos	80

1. Introducción

En el entorno empresarial actual, las startups enfrentan el desafío constante de mantener una alta tasa de retención de clientes. Este reto es especialmente relevante en mercados dinámicos, donde los rápidos avances tecnológicos y la alta rotación de clientes exigen estrategias ágiles y efectivas. En este contexto, el fenómeno conocido como ‘churn’, que se refiere a la pérdida de clientes tras una conversión exitosa, representa un obstáculo importante para el crecimiento acelerado, una de las principales características que definen a una startup. Dentro del contexto de la startup sobre la cual se desarrolló el proyecto nos referimos a ‘Churn Comercial’ a la pérdida de clientes, solo si sucede dentro de los 90 días posteriores a una conversión exitosa.

Comprender las causas del ‘Churn Comercial’ y anticipar su ocurrencia es fundamental para diseñar estrategias que no solo mejoren las tasas de retención, sino que también eleven la calidad de la experiencia del cliente. Este proyecto tiene como objetivo desarrollar un modelo predictivo de ‘Churn Comercial’ basado en técnicas de machine learning, siguiendo los lineamientos de la metodología CRISP-DM. El modelo buscará identificar patrones y variables clave que permitan prevenir el abandono de clientes.

El desarrollo del modelo se enfoca en detectar patrones relacionados con la contactabilidad en el embudo comercial de los clientes, con el propósito de diseñar estrategias específicas que reduzcan el ‘Churn Comercial’ durante la etapa de experiencia. La automatización de este proceso permitirá abordar el problema de forma eficiente, mejorando no solo las tasas de retención, sino también la capacidad de tomar decisiones basadas en datos.

A través de esta solución, se busca posicionar a la startup como una organización innovadora y centrada en el cliente, fortaleciendo la relación con sus consumidores y garantizando un crecimiento sostenible en el tiempo.

1.1. Antecedentes

Vivimos en la era de la revolución del conocimiento, cuando el poder de una nación no se determina por el número de soldados en su ejército, sino por el conocimiento que posee. (Negnevitsky, 2005). Los avances tecnológicos cada vez requieren de personas altamente calificadas para poder desarrollar para poder mejorar la calidad de vida de las personas, sin embargo, vemos en la actualidad que este trabajo se está derivando a las ‘Maquinas Inteligentes’ que a lo largo de los años han logrado adquirir el conocimiento y experiencia de personas altamente calificadas y resolver todo tipo de tareas, desde tareas sencillas hasta tareas bastante específicas y complejas.

En 1956, un taller de verano en Dartmouth College reunió a diez investigadores interesados en el estudio de la inteligencia artificial, y nació una nueva ciencia: la inteligencia artificial. Desde principios de la década

de 1950, la tecnología de IA ha pasado de la curiosidad de unos pocos investigadores a una herramienta valiosa para apoyar la toma de decisiones humanas. (Negnevitsky, 2005)

La evolución de la inteligencia artificial desde sus inicios en los años 50's pasando por distintas etapas a lo largo de su historia, desde la era de las grandes ideas y grandes expectativas en la década de los años 60's hasta la frustración y falta de financiamiento allá por los años 70's, seguido del resurgimiento del campo de las redes neuronales artificiales por la década de los 80's, mientras que el verdadero avance ocurrió en 1986, cuando Rumelhart y McClelland reinventaron el algoritmo de retro propagación, lo que popularizó el uso de redes neuronales multilayer. Durante estos años, también se introdujeron nuevas técnicas como el aprendizaje por refuerzo y redes feedforward con funciones de base radial, lo que permitió un significativo progreso en el campo de la inteligencia artificial. (Negnevitsky, 2005)

El aprendizaje automático, mejor conocido como 'Machine Learning', se categoriza como un subcampo de la Inteligencia Artificial. Básicamente el aprendizaje automático implica la construcción de modelos matemáticos para ayudar a comprender los datos (VanderPlas, 2017).

El análisis predictivo emplea modelos estadísticos y algoritmos para analizar datos históricos, con el objetivo de predecir eventos futuros. (IBM, Análisis predictivos, 2024). Actualmente el análisis predictivo ha ganado gran relevancia en múltiples sectores desde el ámbito empresarial y la medicina, hasta el marketing y el turismo entre otras disciplinas, el análisis predictivo se ha convertido en una herramienta esencial para anticipar tendencias, identificar patrones y optimizar la toma de decisiones.

Bolivia, conocida por su cultura conservadora y tradicional, enfrenta desafíos al incorporar tecnologías como la inteligencia artificial en los procesos empresariales debido a una tradición marcada por la desinformación y resistencia al cambio. Sin embargo, en este contexto, el uso innovador del análisis predictivo puede ser un factor clave para anticiparse a las necesidades de los clientes y optimizar servicios, logrando así una ventaja competitiva dentro del mercado. Para ello, es fundamental la comprensión de cómo esta herramienta puede fortalecer su posición en el mercado y adaptarse a un entorno en constante cambio.

El análisis predictivo emerge como una herramienta clave para que las startups y empresas IT bolivianas optimicen sus operaciones y tomen decisiones estratégicas basadas en datos. Su implementación no solo permite anticipar las necesidades de los clientes, sino también mejorar la eficiencia de los servicios y consolidar una ventaja competitiva en el mercado. Para comprender mejor el impacto de esta innovación, se ha llevado a cabo una revisión de diversas fuentes. En este sentido, un estudio realizado por (Urrelo, 2024) emplea modelos de aprendizaje supervisado para identificar las condiciones que influyen en la decisión de abandono por parte de los clientes en una empresa de alojamiento web. Este enfoque proporciona una base metodológica relevante que puede ser aplicada en el sector tecnológico boliviano, permitiendo a las startups desarrollar estrategias más efectivas para mejorar la retención de usuarios y optimizar su crecimiento.

En este sentido, las startups, reconocidas por su alto potencial y velocidad de crecimiento en relación con la inversión inicial (REPSOL, 2023), representan una gran oportunidad para implementar tecnologías

innovadoras dentro de sus procesos de crecimiento. Desde sus primeros pasos, una startup del tipo SaaS (Software as a Service) construye una estructura organizativa flexible y escalable, enfocada en un crecimiento ágil y escalonado, lo que facilita la adopción y aprovechamiento de estas tecnologías para optimizar sus operaciones y decisiones. De esta manera, se presentan como un ejemplo clave para demostrar cómo el uso de nuevas tecnologías en el contexto boliviano no solo puede ser beneficioso, sino también fundamental para impulsar la competitividad y el desarrollo en un mercado global cada vez más digitalizado.

1.2. Justificación

En una startup, la optimización de recursos es esencial para alcanzar un crecimiento sostenible. El objetivo principal es generar resultados significativos sin desperdiciar recursos como ser tiempo, esfuerzo, talento humano o recursos logísticos. Para ello, es crucial direccionar los esfuerzos hacia los clientes con mayor probabilidad de permanencia a largo plazo, evitando aquellos que tendrían altas probabilidades de terminar en ‘churn’. Una estrategia eficaz para lograr esto es el desarrollo de un modelo predictivo basado en patrones de comportamiento a lo largo del ciclo de vida del cliente.

Este modelo de análisis predictivo permitirá a la startup gestionar mejor sus recursos para mejorar la eficiencia en la retención de clientes y mantener una baja tasa de ‘Churn Comercial’, al tiempo que optimiza las estrategias de interacción y contacto.

Mediante el uso de datos históricos, la empresa podrá conocer mejor a sus clientes, identificar sus necesidades y diseñar estrategias personalizadas para mejorar su experiencia con el servicio o producto. Esto no solo incrementará la retención, sino que también posicionará a la empresa de manera competitiva, permitiéndole anticiparse a las expectativas del mercado y asegurar un crecimiento más sólido y sostenible.

1.3. Planteamiento del problema

Se tiene el conocimiento de que dentro del contexto de la startup se distingue entre ‘churn’ y ‘Churn Comercial’. La diferencia radica en que el ‘Churn Comercial’ se refiere al abandono del cliente dentro de los primeros 90 días tras una conversión exitosa. Por otro lado, el ‘churn’ puede ocurrir en cualquier momento posterior a este período inicial de los 90 días.

En la actualidad, el área de experiencia de la startup enfrenta un desafío en la asignación estratégica de recursos para la interacción y contacto con los clientes. La incertidumbre se genera debido a la dificultad de identificar cuáles clientes tienen más probabilidades de hacer ‘Churn Comercial’ y cuáles, por el contrario, permanecerán. Este problema se puede ver reflejado en la gestión de los intentos de contacto durante la etapa comercial, donde las acciones tomadas no siempre se alinean con el comportamiento real de los clientes. Para abordar este reto, es necesario analizar patrones de comportamiento en los la interacción, lo que permitirá orientar de manera más efectiva los recursos y reducir de manera significativa la tasa de ‘Churn Comercial’.

En la Figura 1-1, se puede observar una tendencia creciente en el ‘Churn Comercial’ a lo largo del tiempo, especialmente en los últimos cuatro años. Este patrón sugiere que, a medida que pasa el tiempo, aumenta la cantidad de clientes que abandonan la empresa durante la etapa de onboarding, correspondiente a la etapa de experiencia. Este comportamiento podría estar relacionado con la gestión ineficiente de los recursos destinados al contacto y la interacción con los clientes. La visualización resalta la necesidad urgente de optimizar los esfuerzos de interacción, dirigiéndolos estratégicamente a aquellos clientes que presentan un mayor riesgo de ‘churn’, con el fin de reducir esta tendencia negativa.

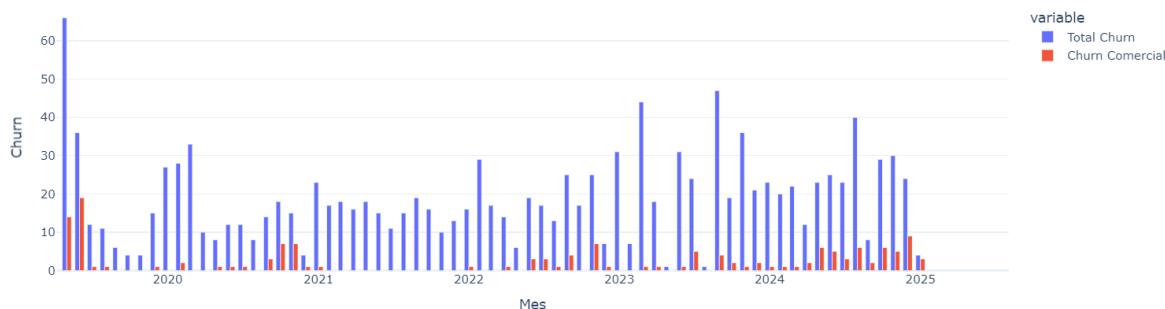


Figura 1-1 Comparación entre ‘Churn’ y ‘Churn Comercial’ a lo largo del tiempo.

Fuente: Elaboración propia, (enero, 2025)

En el primer año de operación de la startup, se registró un índice extraordinariamente alto de ‘churn’, destacando 66 clientes que dejaron el servicio en mayo de 2019. De estos, el 21.21% (14 clientes) correspondieron a ‘Churn Comercial’. Este fenómeno puede atribuirse al hecho de que fue el primer año en que la empresa comercializó su producto SaaS (Sistema de Gestión de Inventarios). Sin embargo, se observa una marcada tendencia a la disminución del ‘churn’ desde mayo hasta noviembre del 2019.

En contraste, durante la gestión 2021, la empresa no presentó casos de ‘Churn Comercial’ y mantuvo un promedio de 15 clientes mensuales que hicieron ‘churn’, lo cual representa un índice relativamente bajo. La ausencia de ‘Churn Comercial’ podría deberse a diversas mejoras implementadas dentro del contexto de la startup, como optimizaciones en los procesos comerciales o mayor adaptación del producto a las necesidades del cliente durante todo ese año.

Dado que las startups se caracterizan por cambios constantes en busca de un crecimiento rápido y escalonado, resulta más relevante analizar el comportamiento del ‘Churn Comercial’ en la última gestión. En la Figura 1-2, se presenta una comparación entre el ‘Churn Comercial’ y el ‘churn’ registrado en los últimos 12 meses, lo que permitirá identificar tendencias actuales y áreas de mejora clave.

El comportamiento del ‘Churn Comercial’ a lo largo de la gestión 2024 muestra una clara tendencia ascendente representando una alta tasa de crecimiento del 20.09% con un promedio mensual a lo largo de la gestión de 3.9 clientes perdidos mes a mes. Este promedio de abandono es bastante alto lo que representa un reto para la startup en el objetivo de crecimiento escalonado y rápido por lo tanto representa un problema para la empresa.

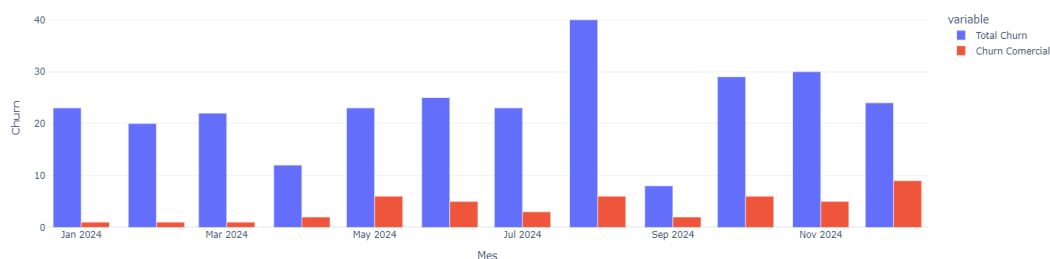


Figura 1-2 Comparación entre 'Churn' y 'Churn Comercial' Gestión 2024

Fuente: Elaboración propia, (enero, 2025)

El desafío radica en identificar a los clientes con mayor riesgo de abandonar su suscripción dentro de los primeros 90 días tras una conversión exitosa en la etapa comercial. Esto permitirá desarrollar estrategias basadas en estos resultados, optimizando los esfuerzos del área de experiencia para priorizar los recursos hacia aquellos clientes con menor probabilidad de abandono.

1.4. Objetivo general

Desarrollar un modelo predictivo de clasificación de 'churn' basado en patrones de contacto con los clientes, utilizando técnicas de machine learning, para apoyar a la toma de decisiones basadas en datos en una startup boliviana.

1.4.1. Objetivos específicos

- Entender las necesidades y objetivos del negocio para mejorar la predicción del churn.
- Extraer datos de los clientes y sus actividades para comprender los patrones de contacto y sus características.
- Analizar y preparar los datos extraídos para su análisis y posterior modelado.
- Entrenar modelos de predicción clasificatorio utilizando técnicas de machine learning.
- Evaluar y seleccionar un modelo predictivo de clasificación efectivo en términos de retención de clientes.

2. Marco teórico

En este capítulo se presentan de manera clara los conceptos, metodologías y elementos clave que utilizaremos en el desarrollo del proyecto. Estos conceptos no solo son la base teórica de nuestro enfoque, sino que también nos guiarán en el análisis y en la toma de decisiones a lo largo del proceso.

2.1. Inteligencia Artificial (IA)

Según el Grupo de Expertos de Alto Nivel sobre Inteligencia Artificial (AI HLEG, por sus siglas en inglés) los sistemas de Inteligencia Artificial (IA) son sistemas de software diseñados por humanos que actúan en la dimensión física o digital percibiendo su entorno mediante la adquisición de datos, interpretándolos, estructurándolos o no, razonando sobre ellos y sobre los conocimientos o procesando la información derivada de estos datos y decidiendo la(s) mejor(es) acción(es) a tomar para alcanzar el objetivo dado' (HLEG, 2019).

2.2. Machine Learning

Machine Learning (ML) o aprendizaje automático es un sub campo de la Inteligencia Artificial que estudia y trabaja sobre algoritmos y modelos estadísticos que los sistemas de computación usan para realizar tareas específicas sin la necesidad de estar explícitamente programados. La principal ventaja de usar algoritmos de ML es que, una vez aprendida la tarea que se le pide, el algoritmo puede realizar el trabajo de manera automática. ML viene siendo usado para distintos propósitos, como ser minería de datos, procesamiento de imágenes, análisis predictivo, análisis semánticos, procesamiento del lenguaje natural, recuperación de información, entre otros (Mahesh, 2020; Shinde, 2018).

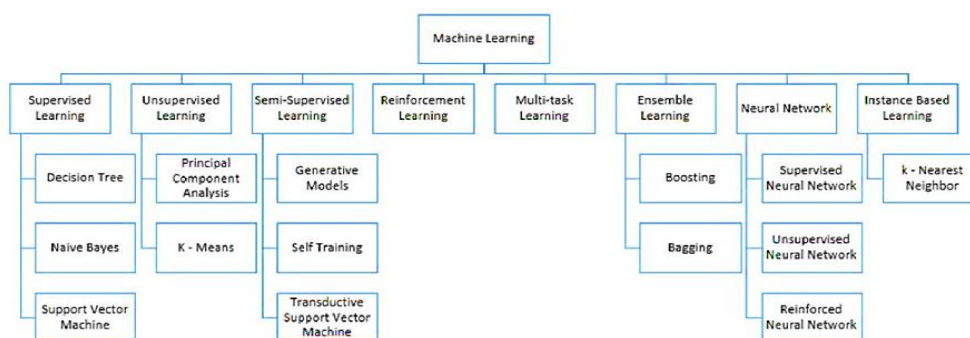


Figura 2-1 Clasificación de algoritmos de Machine Learning

Fuente: (Mahesh, 2020)

Los algoritmos de ML son, en esencia, 'código blando' o 'soft code', y tienen como ideal el emular la forma en que la mente humana procesa (aprende) a partir de señales sensoriales (input data) para cumplir un objetivo o tarea. Para eso, dichos algoritmos tienen que adaptar automáticamente su arquitectura a

través de la repetición y/o experiencia. Ese proceso de adaptación es llamado entrenamiento, donde se proporciona al algoritmo muestras de datos de entrada junto con los resultados esperados. A partir de eso, el algoritmo de ML se configura de tal forma para que no devuelva el resultado esperado solo con los datos de entrada proporcionados, sino que pueda generalizar para producir el resultado esperado a partir de datos nuevos (El Naqa, 2015).

De manera general, el aprendizaje automático puede ser categorizado en dos tipos (VanderPlas, 2017):

- a) **Aprendizaje supervisado:** Hace referencia al modelaje de la relación entre las características medidas de los datos y alguna etiqueta asociada a los mismos. Se subdivide en tareas de clasificación y tareas de regresión.
- b) **Aprendizaje no supervisado:** Consiste en la modelación de características de un conjunto de datos sin referencia a ninguna etiqueta. Los modelos pertenecientes a este tipo de ML son los agrupación y reducción dimensional.

Machine Learning usa distintos tipos de algoritmos para resolver problemas (Fig. 2-3), entre los frecuentemente utilizados se pueden encontrar: clasificadores lineales, regresiones logísticas, Naive Bayes (NB), redes Bayesianas, Árboles de decisiones, Random Forest, Support Vector Machine (SVM), clusterings (K-means, k-nearest neighbour) y redes neuronales artificiales (Shinde, 2018).

2.3. Modelos predictivos de Machine Learning

Los modelos predictivos son técnicas estadísticas y de Machine Learning que a partir de análisis de datos existentes permiten predecir eventos o comportamientos futuros. Son utilizados en diversos campos, principalmente para mejorar la toma de decisiones y planificar estrategias efectivas. Por ejemplo, en Marketing es frecuentemente utilizado en segmentación de clientes y predicciones de abandono (PredikData, s.f.).

De manera general, existen dos tipos de modelos predictivos:

- **Modelos de clasificación:** Son aquellos que permiten predecir la pertenencia a una clase o categoría. En análisis predictivo, estos modelos suelen ser usados para hacer predicciones de tipo binario, y el modelo predice cuál de las dos opciones es la más probable que suceda (Keyrus, s.f.)
- **Modelos de regresión:** Son modelos que permiten predecir valores continuos. Suelen ser utilizados para predecir el rendimiento de algo, ya sea un producto, un proceso o un individuo (Bismart, s.f.)

2.3.1. Modelos predictivos de clasificación

Entre los modelos clasificatorios mas conocidos podemos identificar los siguientes.

2.3.1.1. Modelos clasificatorios de regresión

Los análisis predictivos clasificatorios de regresión relacionan variables entre sí. Pueden ser de tipo logísticas o tipo lineal (Cristianini, 2000)., entre los modelos clasificatorios más utilizados en esta categoría tenemos:

a) Naive Bayes

Es un clasificador probabilístico basado en el teorema de Bayes, que asume independencia condicional entre las características. (Zhang, 2004)

b) Logistic Regression

Es un modelo estadístico utilizado para clasificación binaria, que estima la probabilidad de una clase mediante una función sigmoide. A pesar de su simplicidad, es eficaz en la separación lineal de datos. (Cox, 1958)

2.3.1.2. Modelos predictivos de árboles de decisión

Son modelos de clasificación supervisada que tratan de encontrar la variable que permita dividir un dataset en grupos lógicos que son más diferentes entre sí. Entre los modelos clasificatorios más utilizados en esta categoría tenemos:

a) Decision Tree

Es un modelo basado en una estructura de árbol, donde cada nodo representa una decisión basada en una característica del conjunto de datos. Se utiliza por su interpretabilidad y facilidad de implementación. (Quinlan, 1986)

b) Random Forest

Es un conjunto de árboles de decisión entrenados con diferentes subconjuntos de datos. Mejora la precisión y reduce el sobreajuste combinando múltiples predicciones. (Breiman, 2001)

c) XGBoost

Es una implementación optimizada de gradient boosting, que combina múltiples árboles de decisión y utiliza técnicas de regularización para mejorar el rendimiento en tareas de clasificación y regresión. (Chen, 2016)

2.3.2. EDA (Exploratory Data Analysis)

El Análisis Exploratorio de Datos (EDA, por sus siglas en inglés) es un proceso y etapa esencial en el análisis de datos donde se examina un determinado dataset en orden de descubrir patrones, anomalías, probar hipótesis y supuestos usando métricas estadísticas. El objetivo de la EDA es el de entender de mejor manera la naturaleza y características del dataset bajo estudio, y de extraer información preliminar antes de realizar un modelamiento de manera formal o previo a la formulación de hipótesis. Entre los componentes clave de un EDA se encuentran el resumen de datos, análisis estadístico y la visualización de datos (Milo, 2020; Mukhiya, 2020).

Según el National Institute of Standards and Technology-NIST (NIST Sematech) (Yu, 2010), los objetivos principales de un EDA son: maximizar la comprensión de los datos, identificar estructuras subyacentes, detectar valores atípicos, probar supuestos, desarrollar modelos parsimoniosos.

2.3.3. Variable destino o etiqueta

Las Variables Destino son esenciales en el ámbito del aprendizaje supervisado, dado que representan los atributos que se desean predecir. Durante el proceso de entrenamiento, el modelo adquiere la capacidad de relacionar dichas variables con las características de entrada a partir de datos que han sido etiquetados. Una vez completado el entrenamiento, el modelo puede aplicar este conocimiento para realizar predicciones precisas sobre nuevas instancias que no han sido etiquetadas. (Pineda Pertuz, 2022).

2.4. Modelo CRISP-DM (Cross Industry Standard Process for Data Mining)

CRISP-DM es un modelo de proceso independiente para el desarrollo de proyectos de minería de datos, siendo una de las más utilizadas en este campo (Espinosa-Zúñiga, 2020)

De acuerdo al resumen expuesto por Schröer (Schröer, 2021), este modelo consta de seis etapas:

1. **Comprensión del problema o negocio:** Etapa en la que se entiende y delimita la problemática que pretende solucionar, así como la identificación de los requisitos, supuestos, restricciones y beneficios del proyecto.
2. **Comprensión de datos:** En esta etapa se obtienen los datos que serán utilizados. Se determina el tipo, formato, volumetría y significado de cada dato, así como en la aplicación de estadísticas básicas que nos permitan conocer las propiedades de los datos.
3. **Preparación de datos:** Etapa que consume más tiempo. Se seleccionan los datos que se transformarán con el fin de utilizarlos en la etapa de modelado. Las tareas que se realizan en este punto son: limpieza de datos, creación de indicadores y transformación de datos.
4. **Modelado:** Consiste en la selección de la técnica de modelado. La elección depende del buen entendimiento de las dos primeras etapas.
5. **Evaluación del modelo:** En esta etapa se determina la calidad del modelo con base en el análisis de ciertas métricas estadísticas del mismo, comparando los resultados con resultados previos, o bien, analizando los resultados con apoyo de expertos en el dominio del problema.
6. **Implementación del modelo:** Esta etapa explota, mediante acciones concretas, el conocimiento adquirido mediante el modelo. Aquí también es importante documentar los resultados de manera clara para el usuario final y asegurarse de que todas las etapas de la metodología se documenten debidamente para hacer una revisión del proyecto a fin de obtener lecciones aprendidas durante el proceso. Asimismo, monitorear las acciones para detectar áreas de oportunidad o incluso nuevos problemas.

2.5. StartUps

Una startup es una empresa con poco tiempo de vida que ofrece productos o servicios basados en tecnologías de la información que basa sus estrategias en la incertidumbre pero tiene gran potencial de crecimiento mediante la innovación (Ph.D., 2019).

2.5.1. Características de una startup

Las startups deben seguir ciertas características importantes para poder ser consideradas startups desde sus inicios, algunas de las mas importantes son las siguientes:

1. **Innovación:** siguen un modelo de negocio basado en la innovación y en implementar las ideas en el momento ideal en el mercado para tener éxito.
2. **Altas posibilidades de crecimiento:** siguen una estrategia empresarial que permite el crecimiento rápido y escalonado en el mercado.
3. **Escalabilidad:** tienen una capacidad muy alta de crecimiento y además tienen grandes posibilidades de internacionalizarse, esta característica va muy de la mano con la innovación.
4. **Alto nivel de riesgo:** este es uno de los factores más característicos de las startups, los modelos de negocio de las startups suelen tener altos niveles de riesgo debido a su innovación y que no existe documentación o modelos de negocio previos que demuestren como tener éxito o no.
5. **Experimentación continua:** algo que si es seguro dentro de las startups es la experimentación continua ya que las startups siguen el camino de la innovación se guían por la experimentación. (Ph.D., 2019)

2.6. Software As A Service (SaaS)

El modelo de Software como Servicio (SaaS) se define como un modelo de distribución de software en la nube, en el que las aplicaciones se alojan en servidores de un proveedor y los usuarios acceden a ellas a través de Internet, eliminando la necesidad de instalación y mantenimiento local. Este modelo permite a los usuarios consumir servicios bajo demanda, generalmente mediante suscripciones periódicas, y es gestionado de forma centralizada por el proveedor, quien se encarga de actualizaciones, seguridad y soporte (Gupta, Gupta, & Rai, 2024; Laplante, Zhang, & Voas, 2008)

El SaaS se ha convertido en una herramienta clave para la transformación digital de las empresas, permitiéndoles optimizar recursos y centrarse en actividades de alto valor añadido. Entre sus principales características se pueden destacar:

1. **Acceso remoto:** Los usuarios pueden interactuar con el software desde cualquier dispositivo con conexión a Internet, utilizando un navegador o una interfaz específica.
2. **Modelo de suscripción:** Los costos están basados en el uso, similar al pago por servicios públicos como electricidad, lo que reduce inversiones iniciales en infraestructura (Laplante, Zhang, & Voas, 2008)
3. **Escalabilidad:** Los usuarios pueden ajustar recursos como almacenamiento o funcionalidades según las necesidades del negocio (Gupta, Gupta, & Rai, 2024)
4. **Mantenimiento centralizado:** El proveedor gestiona las actualizaciones y parches, garantizando que los usuarios siempre accedan a la versión más reciente (Gupta, Gupta, & Rai, 2024)

De estas características se derivan una serie de beneficios para las empresas que se deciden por utilizar este tipo de servicios. Por ejemplo, ayuda en la reducción de costos, ya que elimina la necesidad de

adquirir hardwares y/o contratar personal técnico especializado. También permite una mayor flexibilidad operativa, ya que los SaaS se adaptan a las dinámicas cambiantes de las empresas, facilitando la expansión o contracción de servicios según sea necesario y acelerando los procesos de adopción tecnológica (Laplante, Zhang, & Voas, 2008)

2.7. Ciclo de vida de un cliente

El Ciclo de Vida de un Cliente (CVC) se refiere a la progresión que un cliente hace a través de la empresa, desde el primer contacto hasta convertirse en un cliente leal e incluso promotor de la marca (Torres, 2018). La comprensión de este ciclo permite optimizar la experiencia del cliente y afianzar relaciones duraderas. De acuerdo a León de Apio (2017), el CVC está constituido por las siguientes fases:

1. **Adquisición:** Se refiere cuando el cliente adquiere el producto por primera vez.
2. **Conversión:** Sucede cuando el cliente ha probado el producto, le ha gustado y pasa de usar el producto de la competencia a usar el de la empresa.
3. **Crecimiento:** Se da cuando la empresa empieza a experimentar un mayor consumo de su producto, debido principalmente a recomendaciones de los clientes ‘convertidos’.
4. **Retención:** El cliente es fiel a la marca, habituándose a consumir los productos de la empresa.
5. **Reactivación:** Esta etapa ocurre cuando el cliente es atraído por la competencia, ya sea por falta de acciones positivas de la empresa o por lanzamientos y campañas atractivas de otras empresas. (León del Apio, 2017)



Figura 2-2 Ciclo de vida de un cliente vs el trabajo de Marketing en cada etapa

Fuente: Thatzad, 2014 en Loaiza-Torres, 2018

2.7.1. Churn

El ‘Churn’, también conocido como tasa de cancelación o tasa de abandono, hace referencia al porcentaje de clientes que dejan de utilizar los servicios de una empresa durante un determinado tiempo. Es una métrica que permite evaluar la fidelidad de los clientes y la salud general de la empresa. (One.com, s.f.)

2.8. CRM (Customer Relationship Management)

La Gestión de Relaciones con el Cliente, o CRM por sus siglas en inglés, es una estrategia empresarial que tiene como finalidad el construir una cultura de negocio centrada en el cliente. Su objetivo es adquirir,

retener y mejorar la rentabilidad de los clientes mediante una combinación de procesos, tecnologías y personas. Para cumplir dicho objetivo, las empresas que implementan esta filosofía de negocio buscan aumentar la satisfacción y lealtad del cliente mediante servicios personalizados, a través de actividades como la identificación de prospectos y la creación de conocimiento sobre los clientes (Rababah, 2011)

2.9. Arquitectura Medallion

La arquitectura Medallion es un enfoque escalonado de procesamiento de datos que sigue un patrón de diseño en capas, que permite organizar, gestionar y procesar información de forma más eficiente. Este tipo de arquitectura incluye tres capas: bronce, plata y oro, cada uno con funciones y grados de refinamiento distintos, permitiendo que los datos atraviesen por procesos de limpieza y transformación usando data pipelines, mejorando su calidad y adaptabilidad conforme van moviéndose entre las capas. Un patrón de diseño como el de Medallion permite recrear los datos desde la primera capa en cualquier momento, además de facilitar el control de acceso, restringiendo accesos a ciertas capas específicas (Wiselka, 2024).

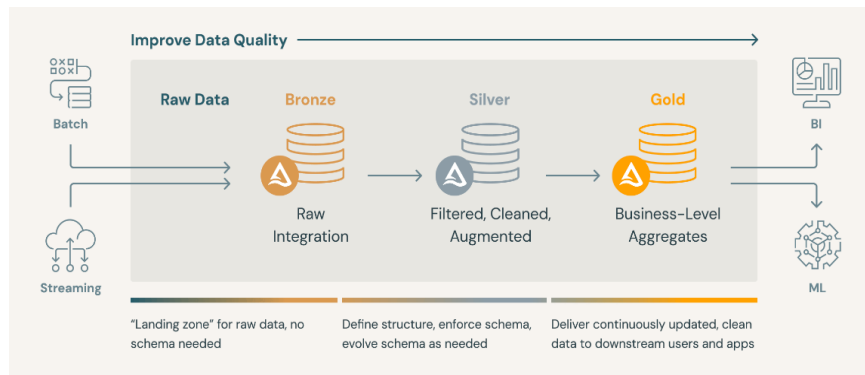


Figura 2-3 Estructura de la Arquitectura Medallion

Figura 2-2-4Fuente: (Databricks, 2024)

2.9.1. Capas (Bronce, Plata y Oro)

Las etapas por las que pasa la data a través del procesamiento de datos en arquitectura Medallion son las siguientes:

1. **Capa Bronce (Bronze Layer):** Contiene datos, y metadatos, en su forma cruda y sin presentar cambios, iguales a los del sistema fuente del cual fueron recuperados.
2. **Capa Plata (Silver Layer):** Aquí se encuentran los datos conformados, limpios y transformados. Para eso, generalmente se realizan trabajos de eliminación de duplicados, manejo de datos corruptos y/o incorrectos que se encuentran en su formato original en la etapa Bronce.
3. **Capa Oro (Gold Layer):** Esta capa contiene datos diseñados para un determinado uso específico, como ser la elaboración de informes y análisis. Los datos usados se pueden seleccionar de múltiples tablas y también de fuentes externas, además de que se pueden realizar diferentes

tipos de agregaciones y transformaciones empresariales específicas. Es en esta capa donde se implementan modelos de datos.

2.10. Descripción estadística

2.10.1. Media

La media, o media aritmética, es la medida de tendencia central más utilizada en el análisis estadístico y representa el promedio de un conjunto de datos. Se calcula a partir de la suma de todos los valores de los datos, dividida entre el número total de datos que componen la muestra observada (Posada Hernández, 2016).

2.10.2. Mediana (Me)

En un conjunto de datos, la mediana representa el lugar central, es decir, es el valor a partir del cual el 50% de las observaciones quedan por debajo de él y el otro 50% por encima. Para calcular la posición de la mediana es necesario ordenar los datos de manera ascendente (Posada Hernández, 2016).

Si bien la media aritmética es la medida de tendencia central más representativa en estadística, para aquellos casos en los que se tienen valores extremos es preferible usar la mediana, ya que no se ve afectada por valores extremos y por tanto no es tan sensible como la media.

2.10.3. Moda (Mo)

La moda es el valor que tiene mayor presencia o frecuencia en un conjunto de datos, puede ser aplicada a las variables cualitativas y cuantitativas discretas o continuas. Para obtener este valor se construyen diagramas de frecuencia y se ubica las características que corresponde a la frecuencia mayor. Un conjunto de datos puede ser unimodal (una moda), bimodal (dos modas) o multimodal (más de dos modas) (Posada Hernández, 2016).

2.10.4. Mínimo (Min) y Máximo (Max)

El valor mínimo de un conjunto de datos hace referencia al valor más bajo. Por el contrario, el máximo estaría representado por el valor más alto. Encontrar estos valores es importante para determinar los límites del rango de distribución de los valores de los datos a utilizar (Posada Hernández, 2016).

2.10.5. Rango

El rango es la medida de dispersión más simple en el análisis de datos. Se lo conoce también como amplitud o recorrido y es la diferencia entre el valor máximo y el valor mínimo de un conjunto de datos. Al estar basada en los valores extremos, no ofrece mucha información sobre la variabilidad de datos, pero puede ser usada como complemento de otras medidas de dispersión (Posada Hernández, 2016).

2.10.6. Varianza

Medida de dispersión que indica que tan alejados están los datos respecto a la media. Está definida como la media de los cuadrados de las diferencias del valor de los datos menos la media aritmética de los mismos. Es empleada junto a la desviación estándar para cuantificar la variabilidad del conjunto en su

totalidad. La varianza poblacional (σ^2) se calcula cuando se tiene la totalidad de los datos de la población, mientras que la varianza de la muestra (s^2) tiene como objetivo convertirse en un estimador de la variación para la población (Posada Hernández, 2016).

2.10.7. Desviación estándar

Considerada la medida de dispersión con mayor representatividad para un conjunto de datos e indica la distribución de los datos alrededor de la media aritmética o promedio. Si este valor es bajo, indica que los datos están agrupados cerca de la media; y por el contrario, cuando la desviación estándar es alta indica que los valores están extendidos sobre un rango más amplio de datos. Se calcula como la raíz cuadrada positiva de la varianza y está denotado por s cuando se estima para la muestra y por σ si es calculado para la población (Posada Hernández, 2016).

2.10.8. Correlación

Medida estadística que indica la relación o dependencia entre dos variables. Es decir, cuando la correlación entre A y B es alta, se sugiere que, si los valores de A aumentan o disminuyen, los valores de B tenderían a sufrir las mismas modificaciones. Empero, cabe destacar que una alta correlación no implica una relación causal directa entre ambas variables (García-Herrero, 2018).

2.10.9. Prueba Chi-Cuadrado

La prueba Chi cuadrado consiste en determinar en una muestra si hay diferencias significativas entre las frecuencias observadas y las especificadas por la ley teórica del modelo con el que se contrasta (frecuencias esperadas). En otras palabras, la prueba de Chi al cuadrado compara entre la tabla observada con una tabla teórica generada a partir del uso de un modelo.

Karl Pearson estableció la siguiente ecuación:

$$\chi^2 = \sum_{i=1}^k \frac{(O_i - E_i)^2}{E_i}$$

Donde:

- O_i = Número de casos observados y clasificados en una determinada celda
- E_i = Número de casos esperados correspondientes a cada celda

Como en cualquier prueba de contraste estadístico, se intenta rechazar la hipótesis nula y aceptar en consecuencia, la hipótesis alternativa. La hipótesis nula se corresponde con la independencia de las variables o que las diferencias entre las frecuencias observadas y esperadas son muy pequeñas, y en consecuencia, el estadístico chi cuadrado también obtendrá un valor muy pequeño (Rigatti, 2017)

2.10.10. Prueba ANOVA

El análisis de la varianza (ANOVA) es una prueba estadística para detectar diferencias en las medias de grupos cuando existe una variable dependiente paramétrica y una o más variables independientes. Puede clasificarse en diferentes tipos, como modelos de efectos fijos unidireccionales y bidireccionales, en

función del número de factores implicados. La técnica asume datos paramétricos, distribución normal, varianzas de grupo similares e independencia de los sujetos, aunque algunos supuestos pueden incumplirse con muestras de gran tamaño. El ANOVA es especialmente útil cuando se comparan tres o más grupos, ya que evita la complejidad y los posibles errores asociados a las pruebas t múltiples (Sawyer, 2009; Larson, 2008)

2.11. Python en la Ciencia de Datos

Python es un lenguaje de programación de código abierto y de tipo interpretado o de script, es decir, que se ejecuta utilizando un programa intermedio llamado intérprete en lugar de compilar el código a lenguaje máquina. Esto hace de Python un lenguaje más flexible y portable. (González Duque, 2011)

2.11.1. Manipulación de datos con Python

Python es vista como una herramienta destacada para el análisis de datos debido a su amplia variedad de bibliotecas, como NumPy, Pandas y Matplotlib, entre otras, y a sus herramientas creadas específicamente para gestionar, procesar, visualizar y modelar datos. Esto convierte a Python en una opción efectiva para convertir datos en información valiosa, facilitando la toma de decisiones a través del análisis de datos en diversas áreas industriales y científicas. (Pineda Pertuz, 2022)

2.11.1.1. Numpy

Abreviatura de Numerical Python, NumPy contiene las estructuras de datos, algoritmos y bibliotecas necesarias para múltiples aplicaciones científicas que involucren datos numéricos, entre los que destacan arreglos multidimensionales (ndarray), funciones para realizar cálculos matriciales, operaciones de álgebra lineal, entre otros. Las matrices NumPy son más eficientes para almacenar y manipular datos que otras estructuras de datos incorporadas en Python. (McKinney, 2022)

2.11.1.2. Pandas

Esta biblioteca combina las ideas de cálculo de arreglos de NumPy con los tipos de capacidades de manipulación de datos que se encuentran en las hojas de cálculo y bases de datos relacionales (como SQL). También permite remodelar, trocear, realizar agregaciones y seleccionar subconjuntos de datos. Pandas es una biblioteca que permite que la manipulación, preparación y limpieza de datos sea más flexible y fluido. (McKinney, 2022)

2.11.1.3. Matplotlib

Matplotlib es la biblioteca de Python más reconocida para generar gráficos y visualizaciones de datos bidimensionales. Su diseño se orientó hacia la creación de gráficos aptos para su publicación. Aunque existen otras bibliotecas de visualización para programadores Python, Matplotlib sigue siendo ampliamente utilizada y se integra de manera adecuada con el resto del ecosistema. (McKinney, 2022)

2.11.2. Aprendizaje Automático con Python

Para desarrollar operaciones de aprendizaje automático con Python, se emplean las siguientes librerías.

2.11.2.1. Scikit-learn

Es una biblioteca de código abierto para aprendizaje automático, que contiene una gran cantidad de algoritmos de clasificación, regresión, agrupamiento, reducción, selección de modelos y preprocesamiento de datos. Estos algoritmos y funciones generalmente están agrupados en submódulos dentro de la biblioteca. (Pineda Pertuz, 2022)

2.11.2.2. XGBoost

XGBoost es una librería avanzada de boosting mediante gradientes distribuidos, diseñada para ser altamente eficiente, adaptable y fácil de usar en diferentes plataformas. Implementa algoritmos de aprendizaje automático bajo el marco de Gradient Boosting, ofreciendo un método paralelo para el boosting de árboles que es conocido por resolver eficazmente muchos problemas en ciencia de datos de manera rápida y precisa. (xgboost, 2022)

2.11.3. Métricas de Evaluación con Python

Para evaluar el rendimiento y efectividad de los modelos se analizan las siguientes métricas en Python

2.11.3.1. Exactitud

Mide el porcentaje de casos que el modelo ha acertado. Se calcula como la proporción entre las predicciones correctas y el número total de predicciones realizadas. Es la métrica por defecto en los algoritmos de Python, utilizando su biblioteca SciKit-Learn, y puede ser evaluada mediante la función `score()`. También está disponible la función `accuracy_score()` del submódulo `metrics` para obtener la exactitud en problemas de clasificación (Pineda Pertuz, 2022)

2.11.3.2. Precisión

La precisión es una métrica que indica la proporción de predicciones correctamente identificadas como positivas con respecto al total de predicciones positivas realizadas, sean estas correctas o incorrectas. En Python, se puede calcular utilizando la librería SciKit-Learn a través de la función `precision_score()` del submódulo `metrics`. Esta métrica es clave para evaluar la efectividad del modelo en la identificación de instancias positivas, ofreciendo información relevante sobre la fiabilidad de sus predicciones afirmativas (Pineda Pertuz, 2022)

2.11.3.3. Sensibilidad o Recall

La sensibilidad, o recall, representa la proporción de casos positivos correctamente identificados en relación con el total de positivos reales. En Python, se puede calcular con la librería SciKit-Learn mediante la función `recall_score()` del submódulo `metrics`. Esta métrica es fundamental para medir la capacidad del modelo en la detección efectiva de instancias positivas, brindando información clave sobre su desempeño en la identificación de casos afirmativos (Pineda Pertuz, 2022)

2.11.3.4. F1-Score

El F1-score es una métrica que combina la precisión y la sensibilidad en un solo valor, lo que resulta útil

cuando se desea evaluar conjuntamente ambas métricas. En Python, se puede calcular utilizando la librería SciKit-Learn a través de la función `f1_score()` del submódulo `metrics`. (Pineda Pertuz, 2022)

2.11.3.5. Curvas ROC

Las Curvas Características Operativas del Receptor (ROC) permiten realizar comparaciones entre distintos clasificadores a partir de los puntajes de sus predicciones, los que se pueden interpretar como probabilidades, variando entre 0 y 1. La curva ROC tiene la estructura mostrada en la Figura 2-4.

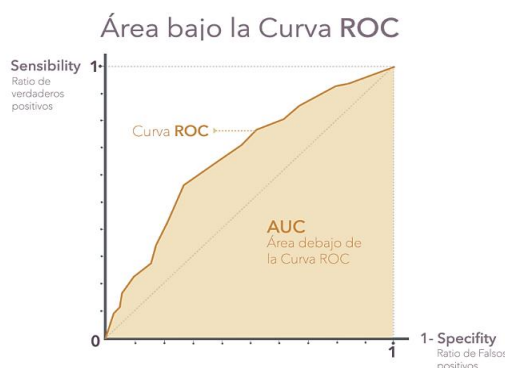


Figura 2-5 Grafico de Curva ROC

Fuente: (Redondo, 2025)

El objetivo es generar modelos cuyo rendimiento se sitúe entre los segmentos $[0, 0] - [0, 1]$ y $[0, 1] - [1, 1]$. Estas curvas pueden generarse en Python utilizando la librería Scikit-Learn con la función `roc_curve()` del submódulo `metrics`. (Pineda Pertuz, 2022)

2.11.3.6. Matriz de confusión

Una matriz de confusión es una herramienta visual que evalúa el rendimiento de un modelo de clasificación, mostrando aciertos y errores entre los valores reales y las predicciones. Usando la librería Scikit-Learn, esta matriz puede generarse usando el método `confusion_matrix()` del submódulo `metrics`. Es muy útil para clasificaciones binarias, donde se asigna un valor 0 a la clase negativa y 1 a la clase positiva. Una matriz de confusión clásica se puede ver en la Figura 2-5. (Pineda Pertuz, 2022)

		Actual	
		0	1
Predicción	0	VN	FN
	1	FP	VP

Figura 2-6 Grafico de Curva ROC

Fuente: (Pineda Pertuz, 2022)

Donde:

- (VP): Verdaderos Positivos. El modelo predice correctamente la salida como positiva.
- (FN): Falsos Negativos. El modelo predice incorrectamente la salida como negativa
- (FP): Falsos Positivos, El modelo predice incorrectamente la salida como positiva
- (VN): Verdaderos Negativos. El modelo predice correctamente la salida como negativa.

Estos elementos permiten que el algoritmo de clasificación tenga una evaluación más completa de su desempeño, analizando tanto los aciertos como los errores en la predicción de cada clase (Pineda Pertuz, 2022)

2.11.4. Métodos de ajuste y optimización

2.11.4.1. Validación Cruzada por k-iteraciones

La validación cruzada con k iteraciones es un método que divide aleatoriamente los datos en k pliegues. En cada iteración, se entrenan el modelo con k-1 pliegues y se prueba con el pliegue restante, repitiendo el proceso k veces para asegurar que cada parte se utilice tanto para entrenar como para probar el modelo.

En Python, este procedimiento se puede implementar con el método StratifiedKFold, el cual es particularmente efectivo para conjuntos de datos desbalanceados, ya que mantiene la proporción de clases en cada porción. (Pineda Pertuz, 2022)

2.11.4.2. SMOTE

El algoritmo SMOTE es un estándar en el manejo de datos desequilibrados por su simplicidad y efectividad. Utiliza un enfoque de sobremuestreo generando ejemplos sintéticos mediante interpolación entre instancias cercanas de la clase minoritaria. Se enfoca en las relaciones entre las características y considera factores como la varianza, la correlación y la distribución de los datos de entrenamiento y prueba. (Fernández, 2018)

2.11.4.3. One-Hot-Encoding

One Hot Encoding es un método para convertir variables categóricas en un formato binario, creando columnas separadas para cada categoría con valores de 0 y 1. Un valor de 1 indica la presencia de la categoría y 0 su ausencia. Este proceso permite que las variables categóricas sean utilizadas eficazmente en modelos de aprendizaje automático, ayudando a capturar relaciones complejas y mejorando el rendimiento de los modelos. Muchos algoritmos requieren entradas numéricas, lo que hace que One Hot Encoding sea esencial. En Python, se puede implementar usando las bibliotecas Pandas o Scikit-learn. (geekforgeeks, s.f.)

3. Marco metodológico

El marco metodológico de este proyecto establece el enfoque que adoptaremos para analizar la tasa de ‘Churn Comercial’ en la startup boliviana. A lo largo del mismo, presentaremos las estrategias que utilizaremos para la obtención y entendimiento de los datos, así como para su procesamiento y diseño de los modelos de Machine Learning. La metodología presentada es fundamental para lograr los objetivos planteados en este proyecto.

3.1. Área de estudio

El área de estudio de este proyecto se centra en el territorio de Bolivia, abarcando todas sus ciudades y regiones. Bolivia se sitúa en el centro de América del Sur, entre los 57°26' y 69°38' de longitud occidental del meridiano de Greenwich y los paralelos 9°38' y 22°53' de latitud sur (INE, 2020)

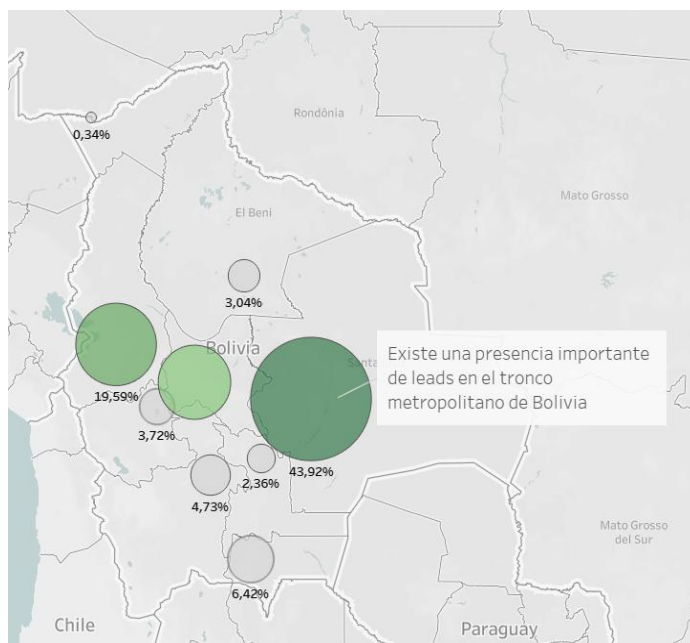


Figura 3-1 Mapa de distribución de clientes en Bolivia

Fuente: Elaboración propia, Noviembre 2024

El objetivo final es poder proporcionar a la startup boliviana una herramienta clave que sea útil para la predicción de ‘Churn Comercial’ que facilite la toma de decisiones basadas en datos.

3.2. Flujograma metodológico

El desarrollo del proyecto sigue los pasos descritos en el flujograma que se muestra en la imagen 3-2. El flujograma sigue la lógica de pasos de la metodología CRISP-DM y de la arquitectura Medallion.

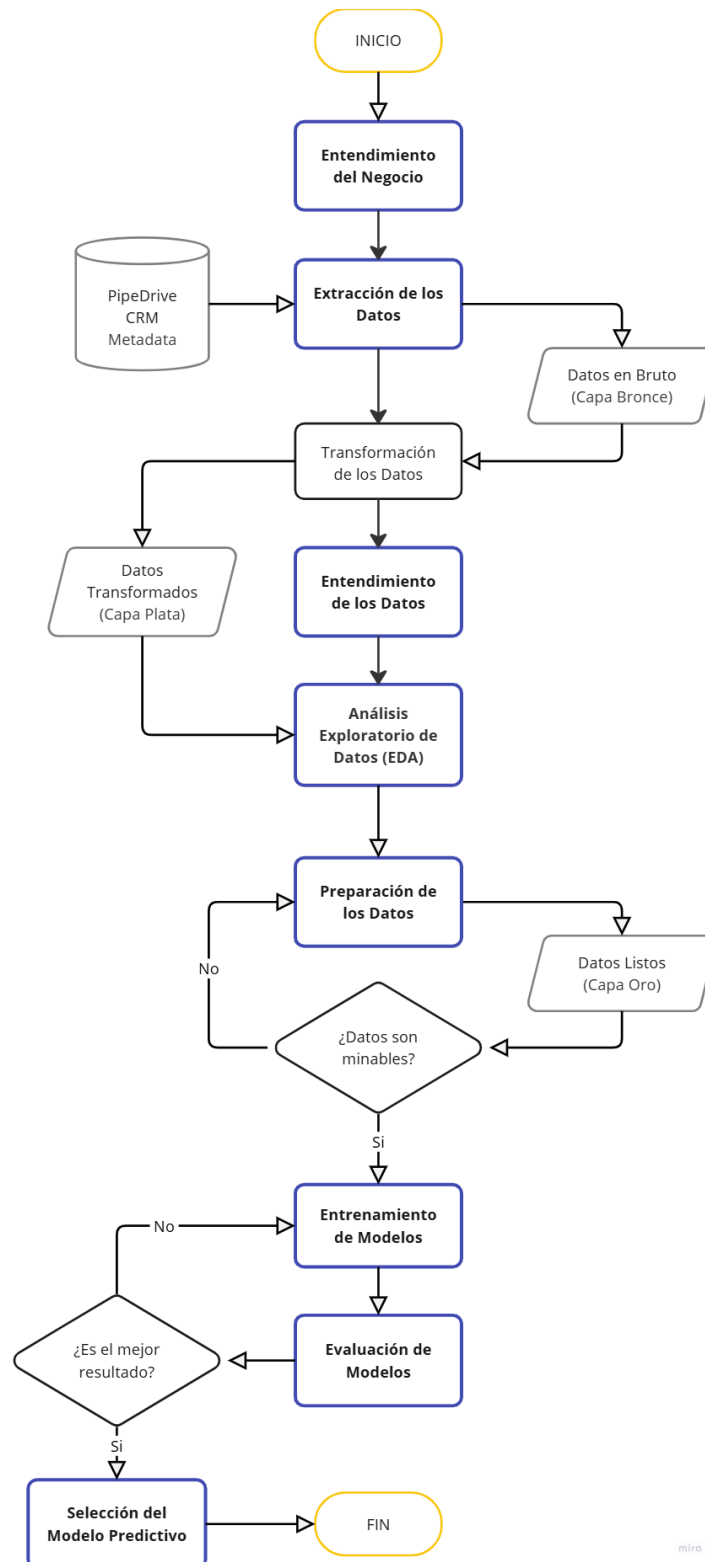


Figura 3-2 Flujograma metodológico

Fuente: Elaboración propia en base a CRISP-DM y Arquitectura Medallion (diciembre, 2024)

3.3. Fuentes de información

3.3.1. Fuente de datos primaria: PipeDrive CRM

Los datos utilizados en este análisis provienen de la base de datos de la startup, específicamente de PipeDrive, el CRM en el que se registran y almacenan todas las interacciones y datos relacionados con sus clientes. La información abarca el período comprendido entre los años 2021 y 2024 y se extrae en formato JSON como metadata a través de la librería ‘requests’ en Python desde la API de PipeDrive (Anexo 1).

Los datos cuentan con un total de cuatro tablas los cuales se dividen en dos tipos de tablas:

- a) **Deals:** Contiene información sobre los negocios creados en PipeDrive, incluyendo datos clave sobre los clientes y sus características.
- b) **Activities:** Registra todas las actividades realizadas con los clientes a lo largo de todo su ciclo de vida, reflejando el seguimiento y la interacción con los clientes.

Esta estructura permite un análisis detallado del comportamiento de los clientes y las dinámicas comerciales dentro de la startup boliviana.

3.3.2. Fuentes de datos secundarias

- **Documentación técnica:** Se utilizarán manuales y guías técnicas.
- **Estudios de casos:** Se analizarán proyectos previos con el mismo enfoque.
- **Literatura académica:** Artículos y documentación sobre análisis de datos se usarán para enriquecer el proyecto con un enfoque académico.

3.4. Entendimiento del negocio

En esta fase, el objetivo principal es comprender a fondo el negocio, incluyendo su misión, visión y objetivos estratégicos. Esto permite establecer las bases que guiarán el desarrollo del proyecto, asegurando su alineación con las necesidades y metas de la empresa. Es fundamental entender que la startup boliviana opera bajo un modelo Business to Business (B2B), dirigido específicamente a pequeñas y medianas empresas del país, además de entender que dentro del contexto de la startup ‘Churn Comercial’ se refiere al abandono del cliente dentro de los primeros 90 días tras una conversión exitosa.

3.4.1. Objetivos del negocio

- Prevenir casos de ‘Churn Comercial’.
- Optimizar recursos en la etapa de experiencia.
- Conocer características de los clientes que hacen ‘Churn Comercial’.

3.4.2. Situación actual de la startup

La startup boliviana, con seis años de trayectoria en el mercado y especializada en su producto SaaS, un sistema de gestión de inventarios, opera con un equipo de menos de cincuenta profesionales. Su estructura organizativa se divide en las siguientes áreas: Marketing, Comercial, Experiencia, Tecnología, Data, People & Happiness, Legal y Contabilidad. A pesar de su corta trayectoria y equipo reducido, esta startup boliviana está sólidamente estructurada y enfocada en un crecimiento rápido y escalable. Su visión a largo plazo incluye la expansión en Latinoamérica, con un enfoque inicial en México y Perú. Su presencia en un nicho innovador la posiciona como pionera en el sector dentro de Bolivia.

La empresa cuenta con una clasificación de clientes en tipos ‘A’, ‘B’ y ‘C’. Esta clasificación fue realizada en un estudio previo a la elaboración de este proyecto, por lo que la inclusión de esta variable es de gran relevancia para la empresa.

Tras un análisis previo sobre la composición de los clientes que hacen ‘Churn Comercial’ por ciudad, no se observa una cifra alarmante que indique que el ‘Churn Comercial’ sea característico de una región en particular. En promedio, el ‘Churn Comercial’ por ciudad es del 11.2%, concentrándose principalmente en las ciudades metropolitanas de Bolivia: Cochabamba, La Paz y Santa Cruz de la Sierra. Esto se debe a la alta concentración de clientes en estas tres ciudades clave de Bolivia. Por lo tanto, la variable ‘Ciudad’ no será relevante en la realización de este proyecto.

3.4.3. Procesos actuales de la empresa

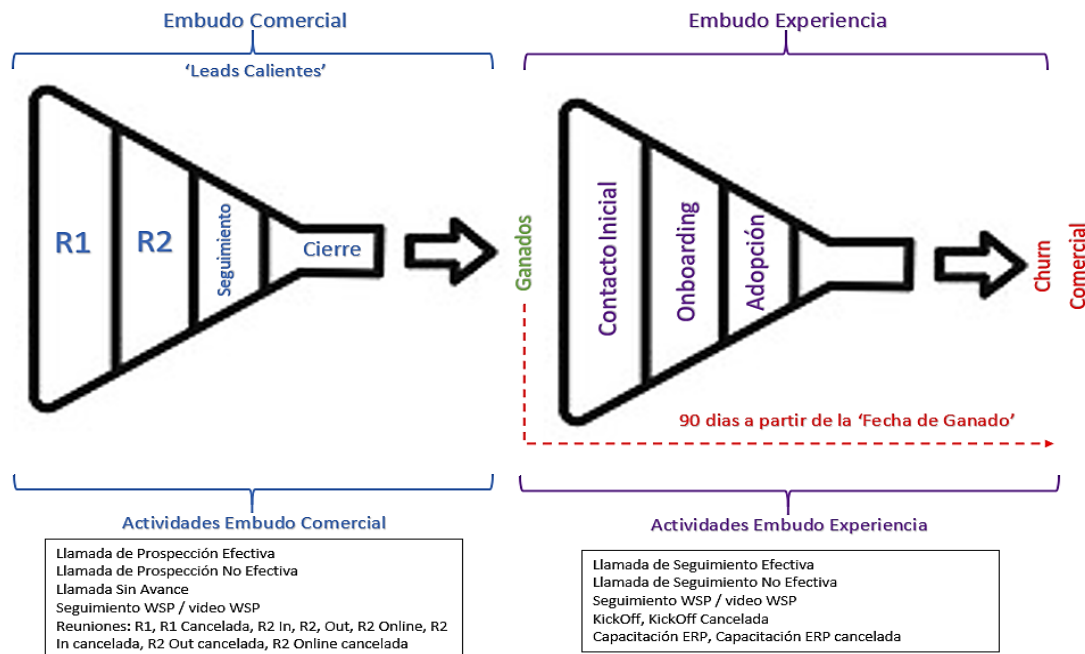


Figura 3-3 Diagrama del ciclo de vida de un cliente en los procesos de la startup

Fuente: Elaboración propia, (enero, 2025)

La empresa actualmente cuenta con embudos que son parte del proceso del ciclo de vida de sus clientes, se realizó un diagrama para poder entender de forma precisa las actividades y eventos que ocurren en cada embudo, es crucial el entendimiento de los procesos que ocurren con los clientes dentro de la empresa ya que eso nos dará una luz para poder realizar los cálculos pertinentes y la selección futura de campos que serán útiles para el modelo predictivo. Se observa a continuación el proceso que será objeto de estudio en la Figura 3-3.

Cabe mencionar un evento importante que puede afectar los resultados del modelo es que, a partir de abril de 2024, se empezaron a considerar las etapas R1 y R2 en lugar de una única etapa denominada 'Reunión'. Cabe destacar que, para el periodo comprendido entre enero de 2021 y abril de 2024, la etapa única 'Reunión' fue asociada retrospectivamente a la etapa R2.

Los clientes una vez ingresan en el proceso comercial deben pasar por distintas etapas, la primera etapa es una reunión R1, para poder llegar a agendar y concretar la Reunión R1 los clientes deben ser contactados mediante llamadas de prospección.

3.4.4. Recursos y Restricciones

- a) **Recursos:** La startup dispone de una base de datos histórica de sus clientes, así como de las herramientas de software gratuitas para la extracción, procesamiento y almacenamiento de datos. Cuenta con un equipo en la empresa dedicado exclusivamente al análisis de datos.
- b) **Restricciones:** Debido a su corta trayectoria en el mercado, la cantidad de información histórica es limitada, ya que aún no cuenta con un volumen significativo de clientes y datos acumulados.

3.5. Extracción de datos

La extracción de datos se llevó a cabo mediante una conexión a la API del CRM PipeDrive, utilizado por la startup para registrar información clave sobre sus clientes. Este proceso permite recopilar datos a lo largo de todo el ciclo de vida del cliente, desde su creación en la base de datos hasta el momento en que es considerado como perdido. La estructura de extracción de datos Medallion con la que se trabajó en este proyecto puede verse en el Anexo 1

Es importante destacar que la base de datos a la que se tiene acceso es de naturaleza privada y confidencial. La obtención de estos datos se realizó bajo un acuerdo previo con la empresa, que incluyó la autorización y los permisos necesarios, así como el cumplimiento de condiciones y términos de seguridad establecidos para garantizar la protección de la información.

3.5.1. Capas de la arquitectura Medallion

Los datos fueron extraídos y organizados siguiendo la estructura Medallion, asegurando un procesamiento eficiente a través de sus capas de bronce, plata y oro.

3.5.1.1. Capa Bronce (Bronce Layer)

Inicialmente se procedió a la creación de filtros en la API del CRM Pipedrive para poder extraer los datos relevantes para el proyecto. Posterior a ello se generó la conexión con la API para la extracción de datos

mediante el uso del lenguaje de programación Python con el uso de la librería ‘requests’ en la consola ‘Visual Studio Code (VS Code)’, esta extracción nos da como resultado una base de datos en bruto en formato JSON (Anexo 2).

3.5.1.2. Capa Plata (Silver Layer)

La capa Plata está compuesta por subcapas que cumplen funciones específicas para procesar y transformar los datos obtenidos en la etapa anterior, los datos se estandarizaron, filtraron y enriquecieron con el uso del lenguaje de programación Python, con ayuda de las librerías ‘Numpy’ y ‘Pandas’, librerías de Python ampliamente utilizadas en ciencia de datos:

1. **Subcapa plata estandarización (Silver standardized sub-layer):** Una vez obtenidos los datos en la capa Bronce, se procedió al mapeo y estandarización de los mismos. En esta subcapa, se asignaron los valores correspondientes a cada elemento extraído como metadata, asegurando que los datos tuvieran una estructura coherente y lista para su posterior análisis (Anexo 3).
2. **Subcapa plata filtrado (Silver filtered sub-layer):** Con los datos estandarizados, se procedió a filtrar las columnas relevantes para el desarrollo del proyecto. Este paso permitió seleccionar únicamente las variables que tienen un valor significativo para el análisis, descartando aquellas que no aportan información útil para el modelo predictivo de ‘Churn Comercial’ (Anexo 4).
3. **Subcapa plata enriquecimiento (Silver enriched sub-layer):** En esta subcapa, se llevó a cabo la integración de las tablas correspondientes mediante la llave principal, unificando la información de manera completa. Además, se realizaron transformaciones de variables, la creación de nuevas variables categóricas y numéricas, y otras operaciones necesarias para enriquecer los datos y prepararlos para un análisis exploratorio de datos (Anexo 5).

3.5.1.3. Capa Oro (Gold Layer)

Finalmente, en la capa Oro se realizaron las agregaciones necesarias y el procesamiento final para obtener una base de datos limpia y estructurada, lista para el entrenamiento del Modelo Predictivo de ‘Churn Comercial’ (Anexo 6).

3.5.2. Entendimiento de los datos

Para obtener un entendimiento completo de los datos, se realizó un análisis exhaustivo utilizando herramientas clave que permitieron explorar y comprender la estructura, patrones y relaciones entre los datos disponibles, además de consultas constantes a las partes interesadas dentro de la startup que pudieran aportar al mejor entendimiento de los datos. Este proceso de análisis inicial sentó las bases para desarrollar modelos predictivos precisos y eficaces, al garantizar que los datos sean interpretados correctamente y utilizados de manera óptima para la toma de decisiones.

Inicialmente, se realizó la búsqueda y selección de los datos necesarios que contienen la información fundamental para el proyecto. Luego, se efectuó un análisis de la organización de cada conjunto de datos, cuyas características se detallan en la Tabla 3-1. Para facilitar la comprensión de las interrelaciones entre las diversas tablas, se incluye un diagrama entidad relación de los datos en la Figura 3-4.

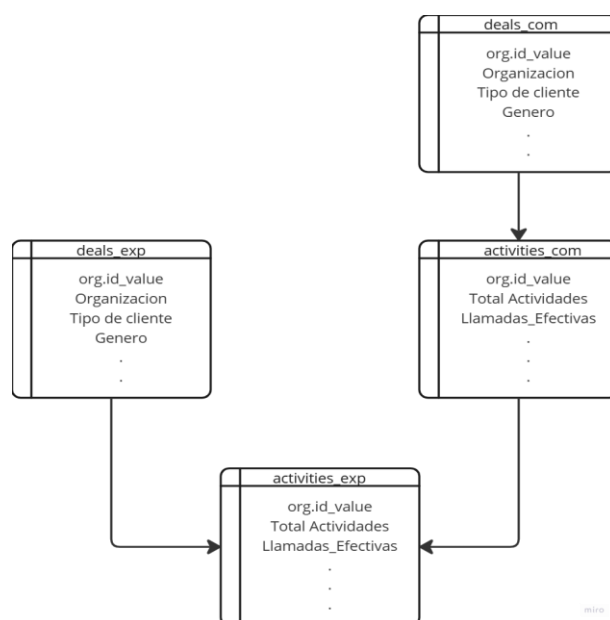


Figura 3-4 Diagrama de Entidad Relación

Fuente: Elaboración propia, (enero, 2025)

Nombre de la tabla	Descripción
'deals_com'	Contiene información sobre características de los clientes 'ganados' correspondientes al embudo comercial
'deals_exp'	Contiene información sobre características de los clientes 'ganados' correspondientes al embudo de experiencia
'activities_com'	Contiene información sobre todas las acciones realizadas con los clientes 'ganados' correspondientes a las etapas dentro del embudo comercial
'activities_exp'	Contiene información sobre todas las acciones realizadas con los clientes 'ganados' correspondientes a las etapas dentro del embudo de experiencia

Tabla 3-1 Descripción de los datos

Fuente: Elaboración propia, (enero, 2025)

Después de tener identificadas y comprendidas las tablas que serían útiles para el desarrollo de este proyecto (extraídas en la Capa Bronce), se procedió a la correspondiente estandarización y filtrado de los campos que serían relevantes para el modelo (datos obtenidos en las subcapas plata: estandarización y filtrado) y finalmente se llevó a cabo la integración de todas las tablas en una resultante, los cálculos de nuevos campos, transformación de campos y finalmente la selección de los campos relevantes para el Análisis Exploratorio de Datos (EDA) correspondiente. A continuación, en la Tabla 3-2, se detallan los campos que fueron finalmente seleccionados como relevantes de la capa plata

Variable	Descripción	Tipo
'org_id.value'	identificador único de cada organización	integer
'Tipo de cliente'	clasificación de clientes tipo ABC	string
'(C) (EXP) Plazo y Pago'	plan de pago del cliente	string
'Total_Actividades_com'	conteo del total de actividades en etapa comercial	integer
'Total_Llamadas_com'	llamadas totales hechos en etapa comercial	integer
'Llamadas_Efectivas_com'	llamadas efectivas hechos en etapa comercial	integer
'Llamadas_No_Efectivas_com'	llamadas no efectivas hechos en etapa comercial	integer
'WA_Seguimiento_com'	seguimientos por WhatsApp en etapa comercial	integer
'Reuniones_Hechas'	reuniones hechas	integer
'Reuniones_Canceladas'	reuniones canceladas	integer
'Tipo Primer Contacto'	tipo de primer contacto	string
'Rango de Contacto'	primer contacto dentro o fuera de rango	string
'R1yR2'	dummie de si tuvo R1 y R2	string
'Total_Actividades_exp'	conteo del total de actividades en etapa experiencia	integer
'Total_Llamadas_exp'	llamadas totales hechas en etapa experiencia	integer
'Llamadas_Efectivas_exp'	llamadas efectivas hechos en etapa experiencia	integer
'Llamadas_No_Efectivas_exp'	llamadas no efectivas hechos en etapa experiencia	integer
'WA_Seguimiento_exp'	seguimientos por WhatsApp en etapa comercial	integer
'Kickoff_Hechas'	Kick Off hechas	integer
'Kickoff_Canceladas'	Kick Off canceladas	integer
'Capacitaciones_Hechas'	capacitaciones ERP hechas	integer
'Capacitaciones_Canceladas'	capacitaciones ERP canceladas	integer
'Tipo Primera Capacitación'	tipo de primera Capacitacion ERP	string
'Onboarding',	estado de Onboarding	string

Variable	Descripción	Tipo
'New Categories'	quiebre estructural desde abril 2024	integer

Tabla 3-2 Descripción de los campos de la tabla de la subcapa plata de enriquecimiento.

Fuente: Elaboración propia, (enero, 2025)

Es importante mencionar que los datos con los que se están trabajando en este proyecto no se tratan de datos poblacionales. Esto se debe a que algunos negocios ya existían antes de enero del 2021 y su historial de interacciones comenzó antes de esa fecha, por lo que no todos los clientes creados en el embudo comercial desde 2021 forman parte de la tabla de experiencia.

3.6. Análisis y preparación de los datos

Se realizó un análisis exhaustivo de los datos y una preparación adecuada para garantizar su efectividad en el entrenamiento de los modelos predictivos.

3.6.1. Análisis Exploratorio de Datos (EDA)

En esta etapa, se realizó un análisis exhaustivo para identificar tendencias, patrones y correlaciones dentro de los datos. Se clasificaron las variables en cualitativas y cuantitativas, lo que permitió una mejor comprensión de la estructura de los datos y facilitó el análisis posterior. Los detalles sobre las variables cualitativas pueden encontrarse en la Tabla 3-3.

Variable	Tipo	Valores
'Tipo de cliente'	nominal	'A', 'B', 'C'
'(C) (EXP) Plazo y Pago'	nominal	'Anual', 'Mensual', 'Otros'
'Tipo Primer Contacto'	nominal	'Efectiva', 'No Efectiva', 'No tuvo', 'Sin Avance'
'Rango de Contacto'	nominal	'Dentro del rango', 'Fuera del rango', 'Sin Llamada de primero contacto'
'R1yR2'	binaria	0: No tuvo R1 y R2, 1: Tuvo R1 y R2
'Tipo Primera Capacitación'	nominal	'Hecha', 'No tuvo'
'Onboarding',	nominal	Finalizado
'New Categories'	binaria	quiebre estructural desde abril 2024
'Churn Comercial'	nominal	quiebre estructural desde abril 2024

Tabla 3-3 Detalle de las variables cualitativas

Fuente: Elaboración propia, (enero, 2025)

Los detalles sobre las variables cuantitativas pueden encontrarse en la Tabla 3-4.

Variable	Tipo
'Total_Actividades_com'	discreto
'Total_Llamadas_com'	discreto
'Llamadas_Efectivas_com'	discreto
'Llamadas_No_Efectivas_com'	discreto
'WA_Seguimiento_com'	discreto
'Reuniones_Hechas'	discreto
'Reuniones_Canceladas'	discreto
'Total_Actividades_exp'	discreto
'Total_Llamadas_exp'	discreto
'Llamadas_Efectivas_exp'	discreto
'Llamadas_No_Efectivas_exp'	discreto
'WA_Seguimiento_exp'	discreto
'Kickoff_Hechas'	discreto
'Kickoff_Canceladas'	discreto
'Capacitaciones_Hechas'	discreto
'Capacitaciones_Canceladas'	discreto

Tabla 3-4 Detalle de las variables cuantitativas

Fuente: Elaboración propia, (enero, 2025)

3.6.1.1. Descripción Estadística

La descripción estadística es el resumen de un conjunto de datos mediante medidas numéricas que representan sus características principales.

Se presentan las variables cuantitativas de acuerdo a los embudos a los que pertenecen, la simbología sigue la siguiente descripción:

- n : Recuento
- \bar{x} : Media

- \tilde{x} : Mediana
- Mín. : Valor mínimo
- Max. : Valor máximo
- Rango : Rango
- IQR : Rango Intercuartílico
- Q1 : Valor del primer cuartil (25%)
- Q2 : Valor del segundo cuartil (50%)
- Q3 : Valor del tercer cuartil (75%)
- σ : Desviación estándar
- σ^2 : Varianza

3.6.1.1.1. Variables del Embudo Comercial

1. Se puede observar las medidas estadísticas de los datos correspondientes al embudo comercial en la Tabla 3-5.

Variable	n	\bar{x}	\tilde{x}	Mín.	Max	Rango	IQR	Q1	Q2	Q3	σ	σ^2
1	373	12.93	10	1	42	41	11	6	10	17	10.20	104.04
2	373	6.40	4	0	21	21	7	2	4	9	5.77	33.30
3	373	4.23	3	0	13	13	4	2	3	6	3.57	12.74
4	373	2.12	1	0	10	10	3	0	1	3	2.85	8.12
5	373	4.96	4	0	15	15	5	2	4	7	4.08	16.65
6	373	1.13	1	0	3	3	1	1	1	2	0.72	0.52
7	373	0.16	0	0	3	3	0	0	0	0	0.51	0.26

Tabla 3-5 Detalle de las variables cuantitativas del embudo comercial

Fuente: Elaboración propia, (enero, 2025)

El total de observaciones para todas las variables de la etapa comercial es de 373, lo que indica que no hay valores faltantes (NaN).

Las variables corresponden a la siguiente numeración:

1. **'Total_Actividades_com'**: La media es de 12.93, lo que sugiere que, en promedio, los clientes completan 12 actividades. La mediana es de 10, lo que significa que la mitad de los clientes realizan 10 actividades o menos. El valor mínimo es 1 y el máximo es 42, indicando que los clientes completan entre 1 y 42 actividades. El rango intercuartílico es de 11, lo que implica que el 50% de los clientes completan entre 6 y 17 actividades. La varianza y la desviación estándar de las actividades son significativas, lo que sugiere una gran variabilidad.

2. **'Total_Llamadas_com'**: La media es de 6.40, lo que sugiere que, en promedio, los clientes reciben 6 llamadas. La mediana es de 4, lo que significa que la mitad de los clientes reciben 4 llamadas o menos. El valor mínimo es 0 y el máximo es 21, lo que indica que algunos clientes no reciben llamadas, mientras que otros pueden recibir hasta 21 llamadas. El rango intercuartílico es de 7, lo que implica que el 50% de los clientes reciben entre 2 y 9 llamadas. Además, la varianza y la desviación estándar son significativas, lo que indica que hay una considerable variabilidad.
3. **'Llamadas_Efectivas_com'**: La media es de 4.23, lo que sugiere que, en promedio, se logran 4 llamadas efectivas. La mediana es de 3, lo que significa que se logran 3 o menos llamadas efectivas con la mitad de los clientes. El valor mínimo es 0 y el máximo es 13, lo que indica que con algunos clientes no se logran llamadas efectivas, mientras que con otros se pueden lograr hasta 13 llamadas efectivas. El rango intercuartílico es de 4, lo que implica que con el 50% de los clientes se logran entre 2 y 6 llamadas efectivas. La varianza y la desviación estándar indican que hay una baja variabilidad.
4. **'Llamadas_No_Efectivas_com'**: La media es de 2.12, lo que sugiere que, en promedio, se tienen 2 llamadas no efectivas. La mediana es de 1, lo que significa que se tiene 1 llamada no efectiva con la mitad de los clientes. El valor mínimo es 0 y el máximo es 10, lo que indica que con algunos clientes no se tienen llamadas no efectivas, mientras que con otros se pueden alcanzar a tener hasta 10 llamadas no efectivas. El rango intercuartílico es de 3, lo que implica que con el 50% de los clientes se tienen entre 0 y 3 llamadas no efectivas. La varianza y la desviación estándar indican que hay una baja variabilidad.
5. **'WA_Seguimiento_com'**: La media es de 4.96, lo que sugiere que, en promedio, se mandan 4 mensajes de seguimiento. La mediana es de 4, lo que significa que se mandan 4 o menos mensajes a la mitad de los clientes. El valor mínimo es 0 y el máximo es 15, lo que indica que a algunos clientes no se le mandan ningún mensaje de seguimiento, mientras que a otros se pueden mandar hasta 15 mensajes. El rango intercuartílico es de 5, lo que implica que al 50% de los clientes se mandan entre 2 y 7 mensajes de seguimiento. La varianza y la desviación estándar indican que hay una baja variabilidad.
6. **'Reuniones_Hechas'**: La media es de 1.13, lo que sugiere que, en promedio, se tiene 1 reunión efectiva. La mediana es de 1, lo que significa que se tiene 1 o ninguna reunión con la mitad de los clientes. El valor mínimo es 0 y el máximo es 3, lo que indica que con algunos clientes no se tiene ninguna reunión, mientras que con otros se pueden tener hasta 3 reuniones, comprendidas entre R1 y R2. El rango intercuartílico es de 1, lo que implica que con el 50% de los clientes se tienen entre 1 y 2 reuniones. La varianza y la desviación estándar indican que hay una baja variabilidad.
7. **'Reuniones_Canceladas'**: La media es de 0.16, lo que sugiere que, en promedio, los clientes no cancelan ninguna reunión. La mediana es de 0, lo que significa que la mitad de los clientes no suelen cancelar reuniones. El valor mínimo es 0 y el máximo es 3, lo que indica que algunos clientes no cancelan reuniones, mientras que otros pueden cancelar hasta 3 reuniones, comprendidas entre R1 y R2. El rango intercuartílico es de 0, lo que implica que el 50% de los clientes no cancela reuniones. La varianza y la desviación estándar indican que hay una baja variabilidad.

3.6.1.1.2. Variables del Embudo Experiencia

Se puede observar las medidas estadísticas de los datos correspondientes al embudo de experiencia en la Tabla 3-6.

Variable	n	\bar{x}	\tilde{x}	Mín.	Max	Rango	IQR	Q1	Q2	Q3	σ	σ^2
1	373	5.62	4	1	17	16	6	2	4	8	4.59	21.07
2	373	1.66	1	0	7	7	3	0	1	3	2.11	4.45
3	373	0.74	0	0	3	3	1	0	0	1	0.97	0.94
4	373	0.88	0	0	5	5	1	0	0	1	1.48	2.19
5	373	2.09	1	0	9	9	3	0	1	3	2.74	7.50
6	373	0.78	0	0	3	3	1	0	0	1	1.01	1.02
7	373	0.26	0	0	2	2	0	0	0	0	0.54	0.29

Tabla 3-6 Detalle de las variables cuantitativas del embudo experiencia

Fuente: Elaboración propia, (enero, 2025)

El total de observaciones para todas las variables de la etapa comercial es de 373, lo que indica que no hay valores faltantes (NaN).

Las variables corresponden a la siguiente numeración:

1. **'Total_Actividades_exp'**: La media es de 5.62, lo que sugiere que, en promedio, los clientes completan 5 actividades. La mediana es de 4, lo que significa que la mitad de los clientes realizan 4 actividades o menos. El valor mínimo es 1 y el máximo es 17, indicando que los clientes completan entre 1 y 17 actividades. El rango intercuartílico es de 6, lo que implica que el 50% de los clientes completan entre 2 y 8 actividades. La varianza y la desviación estándar de las actividades son moderadas, lo que sugiere una gran variabilidad.
2. **'Total_Llamadas_exp'**: La media es de 1.66, lo que sugiere que, en promedio, los clientes reciben 1 llamada. La mediana es de 1, lo que significa que la mitad de los clientes reciben 1 llamadas o ninguna. El valor mínimo es 0 y el máximo es 7, lo que indica que algunos clientes no reciben llamadas, mientras que otros pueden recibir hasta 7 llamadas. El rango intercuartílico es de 3, lo que implica que el 50% de los clientes reciben entre 0 y 3 llamadas. La varianza y la desviación estándar son significativas, lo que indica que hay una considerable variabilidad.
3. **'Llamadas_Efectivas_exp'**: La media es de 0.74, lo que sugiere que, en promedio, se logran 0 llamadas efectivas. La mediana es de 0, lo que significa que no se logran llamadas efectivas con la mitad de los clientes. El valor mínimo es 0 y el máximo es 3, lo que indica que con algunos clientes no se logran llamadas efectivas, mientras que con otros se pueden lograr hasta 3 llamadas efectivas. El rango intercuartílico es de 1, lo que implica que con el 50% de los clientes se logran

entre 0 y 1 llamadas efectivas. La varianza y la desviación estándar indican que hay una baja variabilidad.

4. **'Llamadas_No_Efectivas_exp'**: La media es de 0.88, lo que sugiere que, en promedio, no se tienen llamadas no efectivas. La mediana es de 0, lo que significa que no se tienen llamadas no efectiva con la mitad de los clientes. El valor mínimo es 0 y el máximo es 5, lo que indica que con algunos clientes no se tienen llamadas no efectivas, mientras que con otros se pueden alcanzar a tener hasta 5 llamadas no efectivas. El rango intercuartílico es de 1, lo que implica que con el 50% de los clientes se tienen entre 0 y 1 llamadas no efectivas. La varianza y la desviación estándar indican que hay una variabilidad moderada.
5. **'WA_Seguimiento_exp'**: La media es de 2.09, lo que sugiere que, en promedio, se mandan 2 mensajes de seguimiento. La mediana es de 1, lo que significa que se mandan 1 o ningún mensaje a la mitad de los clientes. El valor mínimo es 0 y el máximo es 9, lo que indica que a algunos clientes no se le mandan ningún mensaje de seguimiento, mientras que a otros se pueden mandar hasta 9 mensajes. El rango intercuartílico es de 3, lo que implica que al 50% de los clientes se mandan entre 0 y 3 mensajes de seguimiento. La varianza y la desviación estándar indican que hay alta variabilidad.
6. **'Capacitaciones_Hechas'**: La media es de 0.78, lo que sugiere que, en promedio, no se tiene una Capacitación ERP efectiva. La mediana es de 0, lo que significa que no se tiene ninguna Capacitación con la mitad de los clientes. El valor mínimo es 0 y el máximo es 3, lo que indica que con algunos clientes no se tiene ninguna capacitación, mientras que con otros se pueden tener hasta 3 capacitaciones. El rango intercuartílico es de 1, lo que implica que con el 50% de los clientes se tienen entre 0 y 1 capacitaciones. La varianza y la desviación estándar indican que hay una baja variabilidad.
7. **'Capacitaciones_Canceladas'**: La media es de 0.26, lo que sugiere que, en promedio, los clientes no cancelan ninguna Capacitación ERP. La mediana es de 0, lo que significa la mitad de los clientes no suelen cancelar capacitaciones. El valor mínimo es 0 y el máximo es 2, lo que indica que algunos clientes no cancelan capacitaciones, mientras que otros pueden cancelar hasta 2 capacitaciones. El rango intercuartílico es de 0, lo que implica que el 50% de los clientes no cancela capacitaciones. La varianza y la desviación estándar indican que hay una baja variabilidad.

3.6.1.1.3. Variables Cualitativas

Se puede observar las medidas estadísticas de los datos correspondientes al embudo de experiencia en la tabla 3-7.

Variable	n	Valores Únicos	Moda	Frecuencia	%
'Tipo de cliente'	373	3	'B'	235	63.00
'(C) (EXP) Plazo y Pago'	373	3	'Anual'	285	76.41
'Tipo Primer Contacto'	373	4	'Efectiva'	222	59.52

Variable	n	Valores Únicos	Moda	Frecuencia	%
'Rango de Contacto'	373	3	'Fuera de Rango'	224	60.05
'R1yR2'	373	2	'0'	310	83.11
'Tipo Primera Capacitación'	373	2	'Hecha'	202	54.16
'Onboarding'	373	2	'No Finalizado'	276	74.00
'Churn Comercial'	373	2	0	338	90.62

Tabla 3-7 Detalle de las variables cualitativas

Fuente: Elaboración propia, (enero, 2025)

La mayoría de los clientes ganados son del tipo B, representando el 63% del total. La gran mayoría de los clientes esta suscrito a un plan de pago Anual, representando el 76.41% del total. El primer contacto con los clientes es generalmente de carácter efectivo, alcanzando un 59.52% del total. Un 60.05% de los clientes son contactados fuera del rango establecido de los 6 minutos. Además, una gran mayoría, el 83.11%, no tiene las reuniones R1 y R2 durante su proceso en el embudo comercial. La primera capacitación para los clientes se realiza en el 54.16% de los casos. También, la mayoría de los clientes no han completado el proceso de onboarding. Finalmente, un 90.62% de los clientes se encuentran activos o han generado un 'churn' después de más de 90 días.

3.6.1.2. Análisis Univariante

Se realizó un análisis por cada variable numérica y categórica. (Anexo 7.)

3.6.1.2.1. Variables del Embudo Comercial

En el histograma de la variable 'Total_Actividades_com' (Figura 3-5), se observa que la mayoría de los clientes completan entre 4 y 10 actividades, con una disminución en la cantidad de clientes a medida que aumentan las actividades, lo que sugiere que hay pocos clientes con seguimiento continuo.

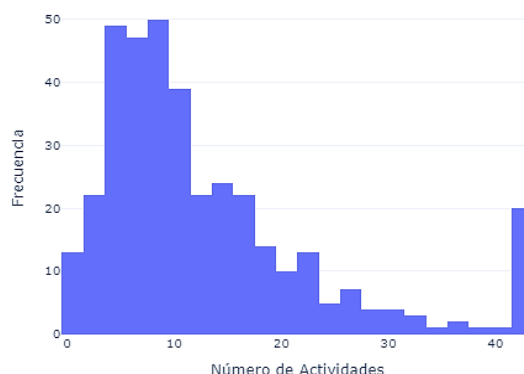


Figura 3-5 Histograma de distribución de la variable 'Total_Actividades_com'

Fuente: Elaboración propia, (enero, 2025)

En el histograma de la variable ‘Total_Llamadas_com’ (Figura 3-6), se observa que la mayoría de los clientes reciben entre 1 y 3 llamadas, con una disminución en la cantidad de clientes a medida que aumentan las llamadas, lo que sugiere que hay pocos clientes con los que se tiene una persecución.

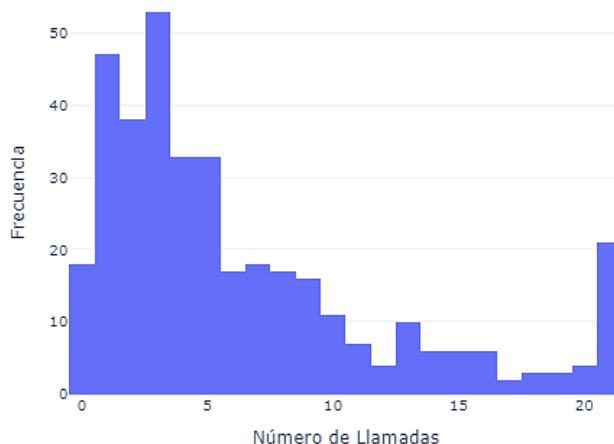


Figura 3-6 Histograma de distribución de la variable ‘Total_Llamadas_com’

Fuente: Elaboración propia, (enero, 2025)

En el histograma de la variable ‘Llamadas_Efectivas_com’ (Figura 3-7), se observa que la mayoría de los clientes reciben entre 1 y 4 llamadas efectivas. A medida que aumenta el número de llamadas, la cantidad de clientes disminuye, lo que sugiere que solo un pequeño grupo de clientes requiere un seguimiento intensivo con altas cantidades de llamadas efectivas en la etapa comercial.



Figura 3-7 Histograma de distribución de la variable ‘Llamadas_Efectivas_com’

Fuente: Elaboración propia, (enero, 2025)

En el histograma de la variable ‘Llamadas_No_Efectivas_com’ (Figura 3-8), se observa que la mayoría de los clientes reciben entre 0 y 1 llamada no efectiva, lo que sugiere que, por lo general, no se realizan llamadas no efectivas en la etapa comercial. A medida que aumenta el número de llamadas no efectivas,

la cantidad de clientes disminuye, indicando que solo con una minoría de clientes se tienen hasta 10 llamadas no efectivas.

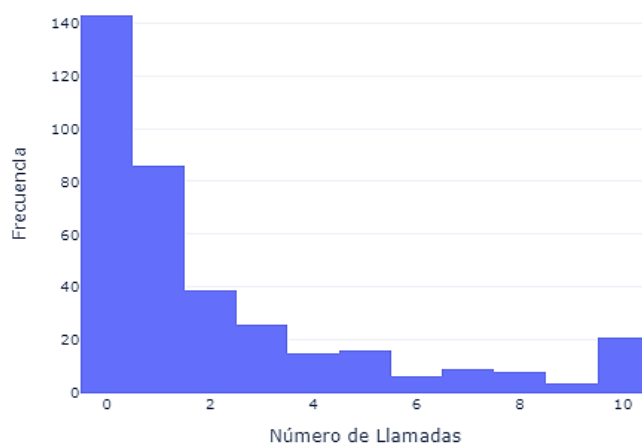


Figura 3-8 Histograma de distribución de la variable 'Llamadas_No_Efectivas_com'

Fuente: Elaboración propia, (enero, 2025)

En el histograma de la variable 'WA_Seguimiento_com' (Figura 3-9), se observa que la mayoría de los clientes reciben entre 2 y 4 mensajes de seguimiento por WhatsApp, lo que sugiere que la mayoría de los clientes tienen un número moderado de seguimientos. A medida que aumenta la cantidad de mensajes, la frecuencia de clientes disminuye, indicando que son pocos los clientes que reciben más de 4 mensajes de seguimiento.

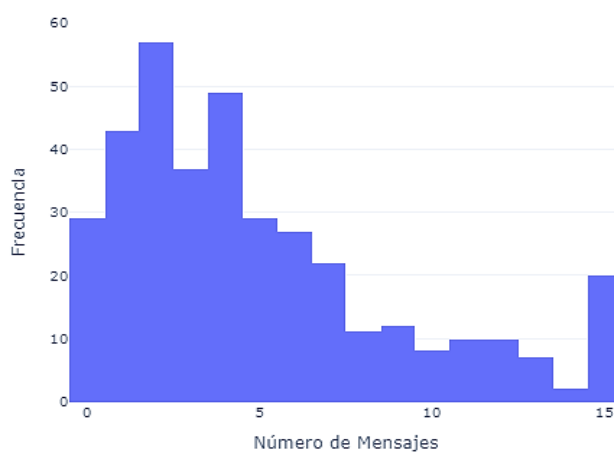


Figura 3-9 Histograma de distribución de la variable 'WA_Seguimiento_com'

Fuente: Elaboración propia, (enero, 2025)

En el histograma de la variable 'Reuniones_Hechas' (Figura 3-10), se observa que la mayoría de los clientes tienen al menos 1 reunión efectiva. Son muy pocos los clientes que no tienen ningún tipo de reunión al igual que los que tienen 2 reuniones, son muy pocos los clientes que tienen más de 2 reuniones.

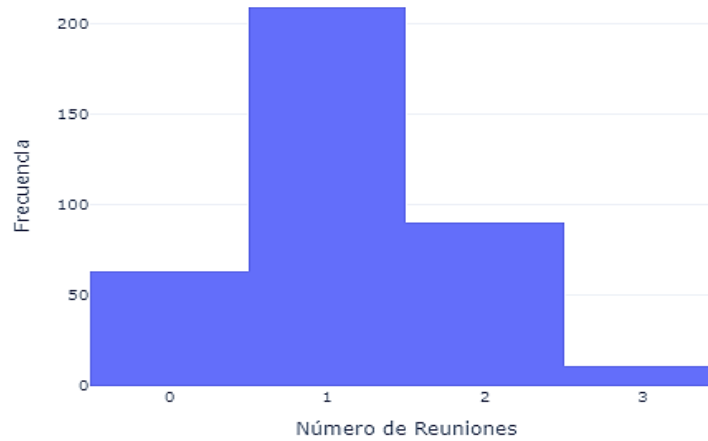


Figura 3-10 Histograma de distribución de la variable 'Reuniones_Hechas'

Fuente: Elaboración propia, (enero, 2025)

En el histograma de la variable 'Reuniones_Canceladas' (Figura 3-11), se observa que la mayoría de los clientes no cancelan reuniones. Son pocos los clientes que cancelan alguna reunión, y en casos excepcionales cancelan hasta 3 reuniones.



Figura 3-11 Histograma de distribución de la variable 'Reuniones_Canceladas'

Fuente: Elaboración propia, (enero, 2025)

3.6.1.2.2. Variables del Embudo Experiencia

En el histograma de la variable 'Total_Actividades_exp' (Figura 3-12), se observa que la mayoría de los clientes completan entre 1 y 5 actividades, con una disminución considerable en la cantidad de clientes a medida que aumentan las actividades, lo que sugiere que hay pocos clientes que completan hasta 17 actividades.

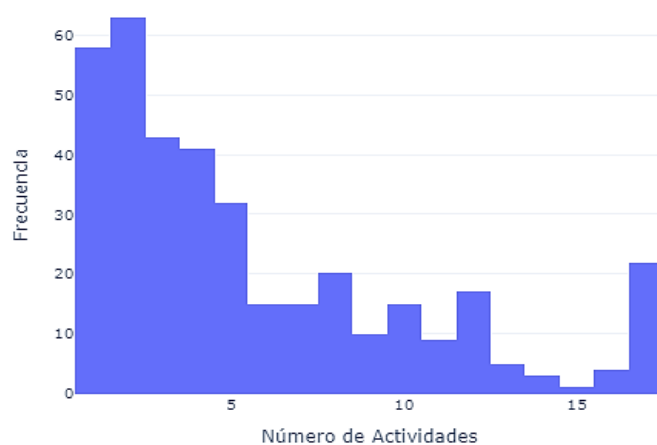


Figura 3-12 Histograma de distribución de la variable 'Total_Actividades_com'

Fuente: Elaboración propia, (enero, 2025)

En el histograma de la variable 'Total_Llamadas_exp' (Figura 3-13), se observa que la mayoría de los clientes reciben entre 0 y 1 llamada, con una disminución notable en la cantidad de clientes a medida que aumenta el número de llamadas, lo que sugiere que en la etapa de experiencia las llamadas no son frecuentes.

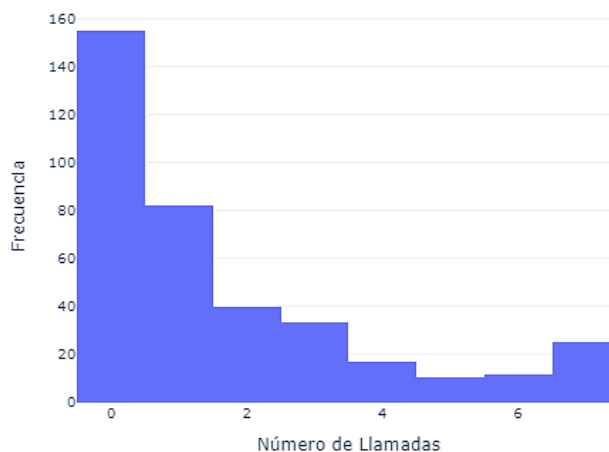


Figura 3-13 Histograma de distribución de la variable 'Total_Llamadas_exp'

Fuente: Elaboración propia, (enero, 2025)

En el histograma de la variable 'Llamadas_Efectivas_exp' (Figura 3-14), se observa que las llamadas efectivas en la etapa de experiencia son poco frecuentes. La cantidad de clientes a los que se les realiza una llamada es reducida, y a partir de la primera llamada, la cantidad de clientes disminuye aún más.

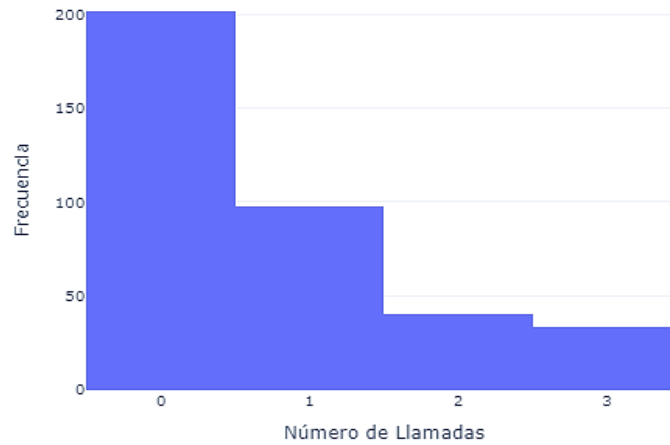


Figura 3-14 Histograma de distribución de la variable 'Llamadas_Efectivas_exp'

Fuente: Elaboración propia, (enero, 2025)

En el histograma de la variable 'Llamadas_No_Efectivas_exp' (Figura 3-15), se observa que la mayoría de los clientes no tienen llamadas no efectivas, lo que sugiere que, en general, las llamadas no efectivas son poco comunes en la etapa de experiencia. A medida que aumenta el número de llamadas no efectivas, la cantidad de clientes disminuye, lo que indica que solo una minoría de clientes tiene hasta 5 llamadas no efectivas.

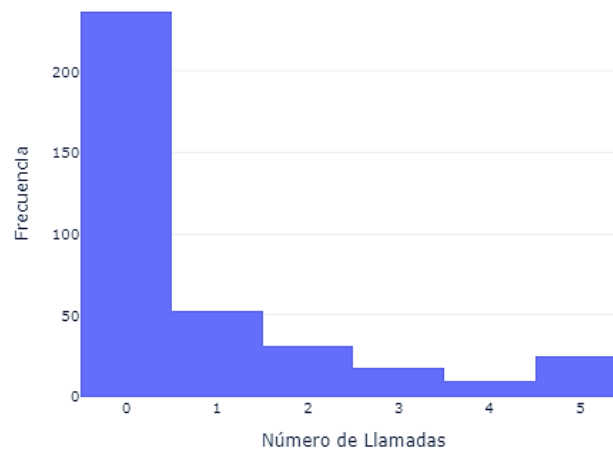


Figura 3-15 Histograma de distribución de la variable 'Llamadas_No_Efectivas_exp'

Fuente: Elaboración propia, (enero, 2025)

En el histograma de la variable 'WA_Seguimiento_exp' (Figura 3-16), se observa que la mayoría de los clientes no reciben mensajes de seguimiento por WhatsApp en la etapa de experiencia, lo que sugiere que no se suele hacer seguimiento por mensajes a los clientes en la etapa de experiencia.

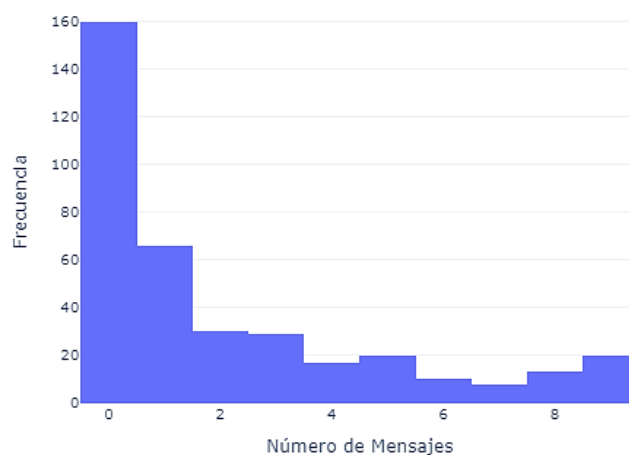


Figura 3-16 Histograma de distribución de la variable 'WA_Seguimiento_com'

Fuente: Elaboración propia, (enero, 2025)

En el histograma de la variable 'Kickoff_Hechas' (Figura 3-17), que la cantidad de clientes que han tenido al menos un Kickoff es igual a la cantidad de clientes que no lo han tenido. En otras palabras, hay una distribución equilibrada entre aquellos que realizaron un Kickoff y aquellos que no, esto puede deberse a un error en el llenado de datos de acuerdo a lo conversado con los encargados del área de experiencia.

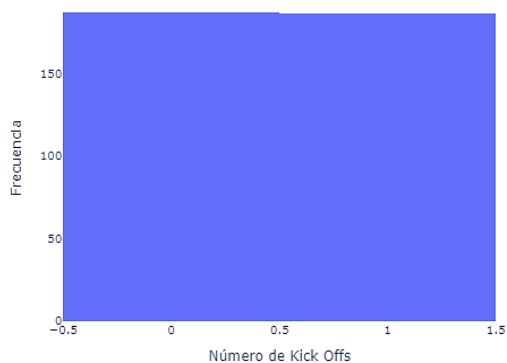


Figura 3-17 Histograma de distribución de la variable 'Kickoff_Hechas'

Fuente: Elaboración propia, (enero, 2025)

En el histograma de la variable 'Kickoff_Canceladas' (Figura 3-18), se observa que la mayoría de los clientes no cancelan la Kickoff, lo que no tiene consistencia con la distribución de la variable 'Kickoff_Hechas'. Son muy pocos los clientes que cancelan la Kickoff. esto también puede deberse a un error en el llenado de datos.

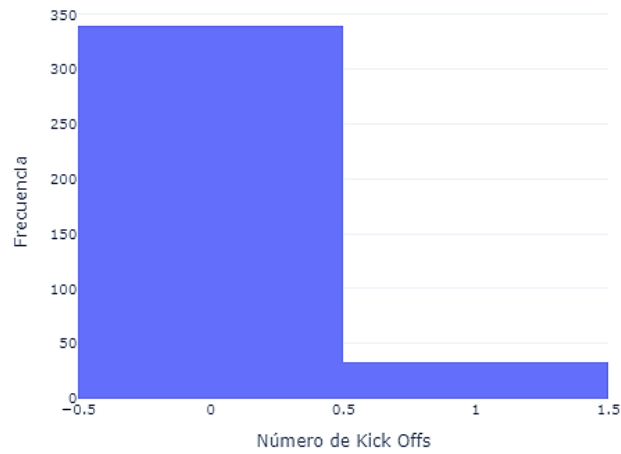


Figura 3-18 Histograma de distribución de la variable 'Kickoff _Canceladas'

Fuente: Elaboración propia, (enero, 2025)

En el histograma de la variable 'Capacitaciones_Hechas' (Figura 3-19), se observa que la mayoría de los clientes no llegan a tener capacitaciones ERP, se ve que son muy poco frecuentes los clientes que cancelan hasta 3 reuniones.

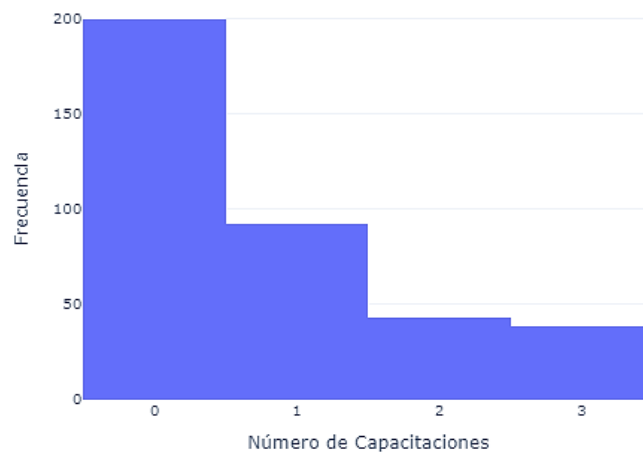


Figura 3-19 Histograma de distribución de la variable 'Capacitaciones_Hechas'

Fuente: Elaboración propia, (enero, 2025)

En el histograma de la variable 'Capacitaciones_Canceladas' (Figura 3-19), se observa que la mayoría de los clientes no cancelan las capacitaciones ERP, se ve que son muy poco frecuentes los clientes que cancelan hasta 3 reuniones.

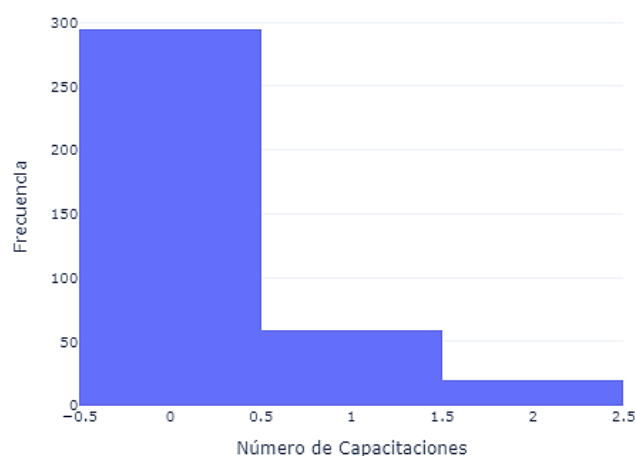


Figura 3-20 Histograma de distribución de la variable 'Capacitaciones_Canceladas'

Fuente: Elaboración propia, (enero, 2025)

3.6.1.2.3. Variables Cualitativas

El tipo de clientes se clasifica en 3 categorías: 'A', 'B' y 'C'. Como se muestra en la Figura 3-21, esta categorización fue realizada en un estudio previo dentro de la startup, el cual agrupa diversas características de los clientes. Se observa que la mayoría de los clientes (63.02%) corresponden a la categoría B.

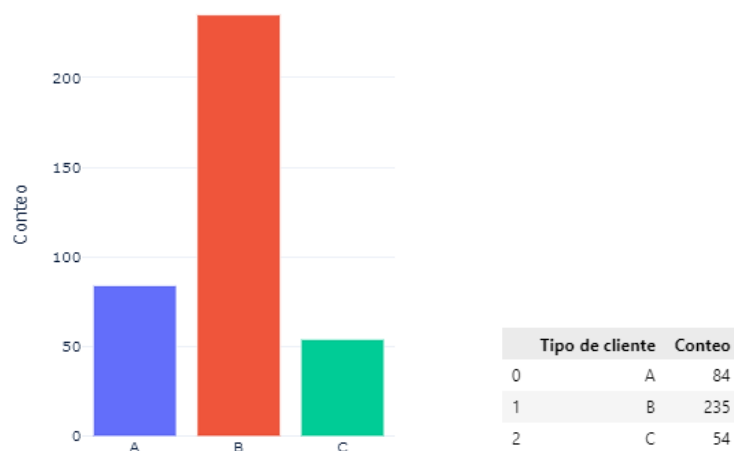


Figura 3-21 Gráfica de barras de 'Tipo de cliente'

Fuente: Elaboración propia, (enero, 2025)

Los clientes se suscriben con un plan de pago que puede ser 'Bianual', 'Anual', 'Semestral', 'Trimestral', 'Mensual' u 'Otros'. Como se muestra en la Figura 3-22, Se observa que la mayoría de los clientes (76.41%) están suscritos un plan de pago 'Anual'.

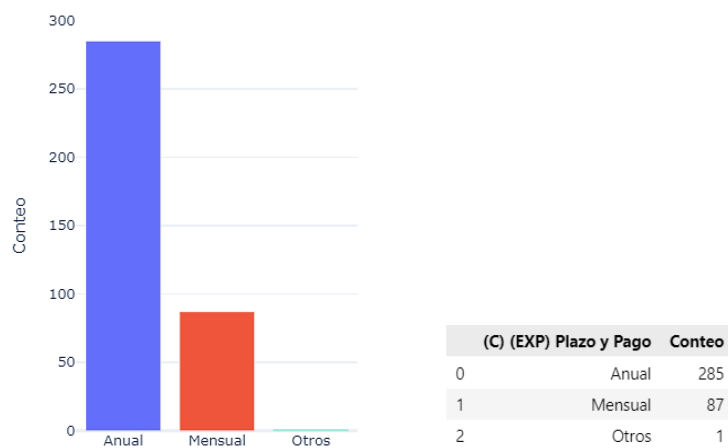


Figura 3-22 Grafica de barras de '(C) (EXP) Plazo y Pago'

Fuente: Elaboración propia, (enero, 2025)

El primer contacto se clasifica en: 'Efectivo', 'No Efectivo', 'Sin Avance' o 'No tuvo'. Como se observa en la Figura 3-23, la mayoría de los clientes tienen un primer contacto 'Efectivo' (59.52%) de los clientes de la empresa.

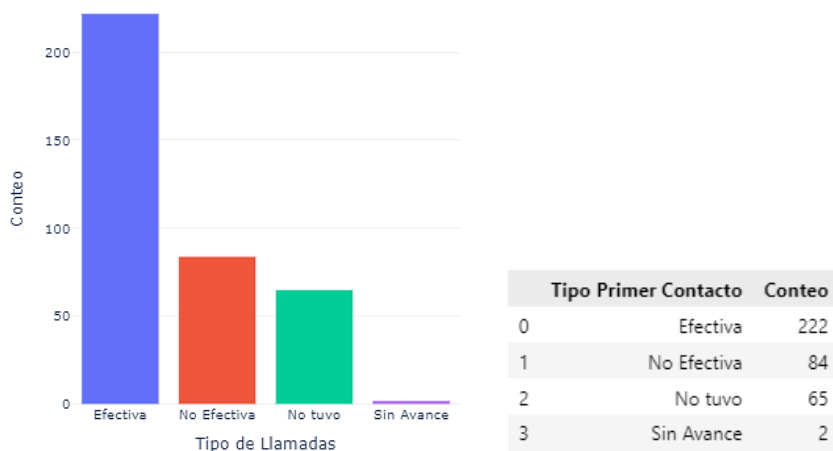


Figura 3-23 Grafica de barras de 'Tipo Primer Contacto'

Fuente: Elaboración propia, (enero, 2025)

El tiempo transcurrido desde la creación del negocio en el CRM hasta el primer contacto en la etapa comercial puede clasificarse en Dentro de rango, Fuera de rango o Sin llamada de primer contacto. Como se observa en la Figura 3-24, en la mayoría de los casos (60.05% de los clientes), este primer contacto ocurre fuera del rango de los 6 minutos establecidos como métrica previa en la startup.

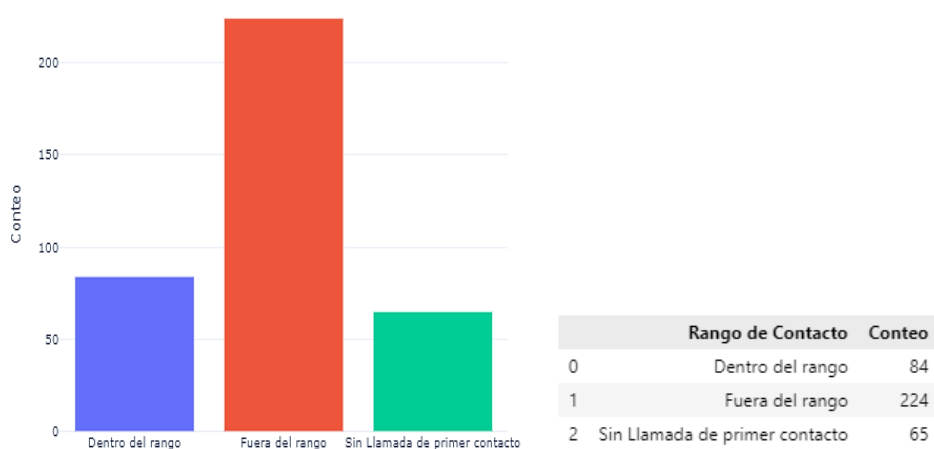


Figura 3-24 Grafica de barras de 'Rango de Contacto'

Fuente: Elaboración propia, (enero, 2025)

El hecho de haber tenido R1 y R2 en la etapa comercial se clasifica en 0 (No) y 1 (Sí). Como se muestra en la Figura 3-25, la mayoría de los clientes (83.10%) no completan ambas reuniones, lo que indica que son pocos los casos en los que se cumplen todas las etapas de la fase comercial.

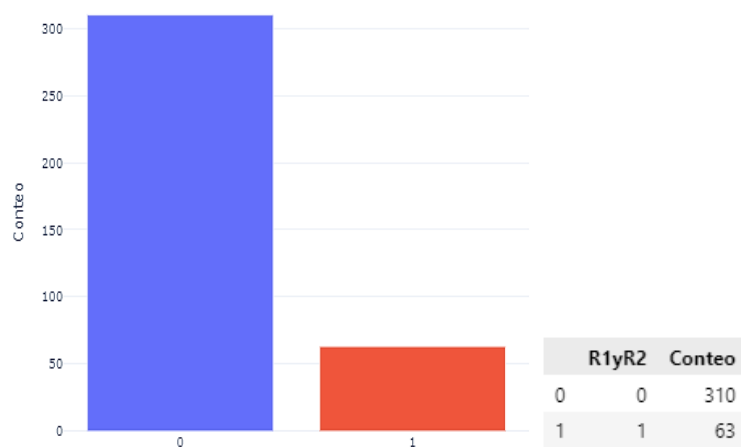


Figura 3-25 Grafica de barras de 'R1yR2'

Fuente: Elaboración propia, (enero, 2025)

El tipo de primera capacitación se clasifica en 'Hecha', 'Cancelada' y 'No tuvo'. Como se muestra en la Figura 3-26, la mayoría de los clientes (54.15%) tiene su primera capacitación marcada como hecha. La distribución es casi equitativa entre quienes la realizaron y quienes no, y no se registran casos de capacitaciones canceladas como tipo de primera capacitación, directamente no llegaron a tener una primera capacitación.

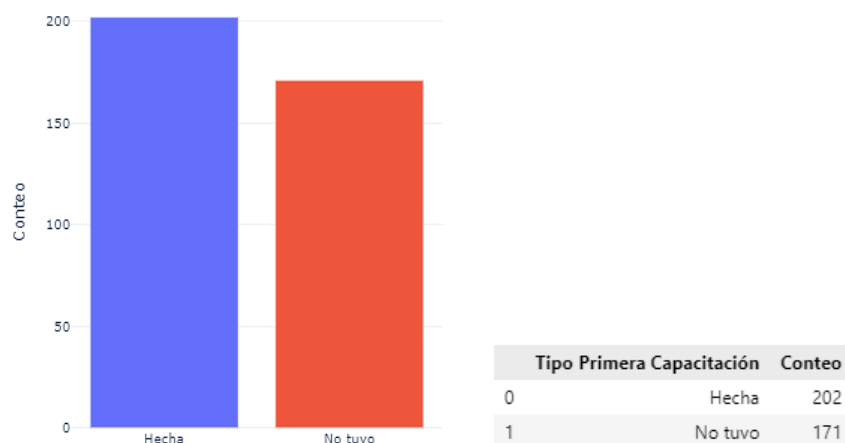


Figura 3-26 Grafica de barras de 'Tipo Primera Capacitación'

Fuente: Elaboración propia, (enero, 2025)

El estado de Onboarding se clasifica en 'Finalizado' y 'No Finalizado'. Como se muestra en la Figura 3-27, la mayoría de los clientes (74%) no llega a completar el proceso de Onboarding.

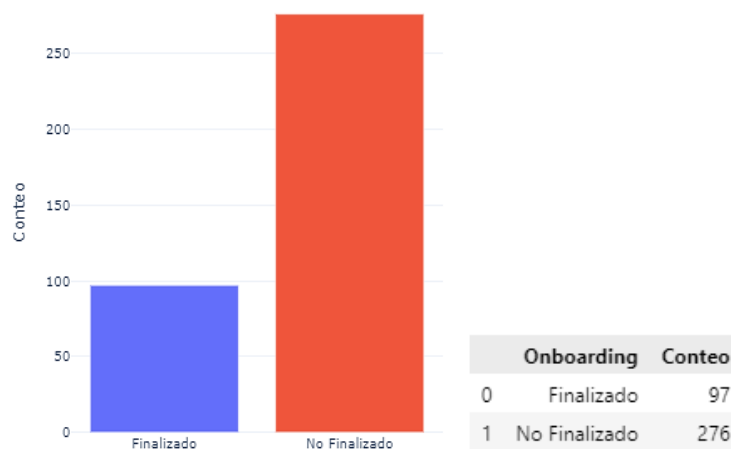


Figura 3-27 Grafica de barras de 'Onboarding'

Fuente: Elaboración propia, (enero, 2025)

El estado de 'Churn Comercial' se clasifica en 0 (No) y 1 (Sí). Como se muestra en la Figura 3-28, solo el 9.38% de los clientes (35 casos) presentan 'Churn Comercial', mientras que el resto permanece suscrito por más de 90 días. Esto indica un desbalance en los datos, con un marcado sesgo hacia la categoría de No Churn Comercial. Como se puede observar en el grafico nuestros datos se encuentran desbalanceados teniendo como clase minoritaria a la clase 1: 'Churn Comercial'.

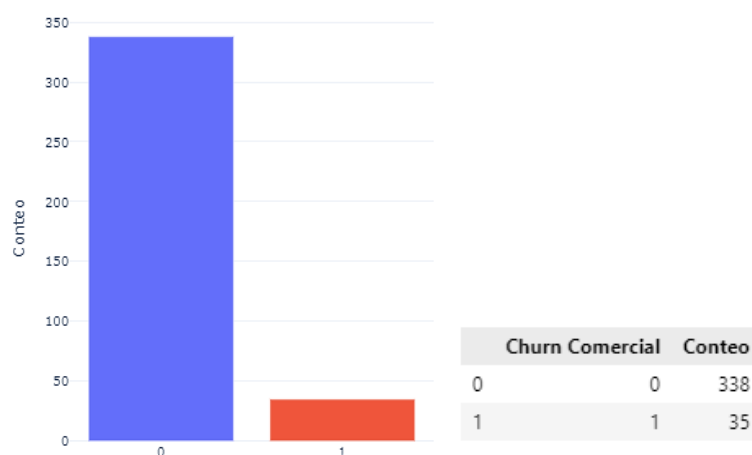


Figura 3-28 Grafica de barras de 'Churn Comercial'

Fuente: Elaboración propia, (enero, 2025)

3.6.1.3. Análisis Bivariado

Se realizó el cruce de datos de las variables cuantitativas por etapas: comercial y experiencia con la variable de interés 'Churn Comercial', donde se encontraron diferencias significantes en las siguientes variables.

Se puede evidenciar una diferencia significativa en los rangos intercuartílicos entre los grupos para la variable 'Llamadas Efectivas' del embudo comercial (Figura 3-29). Además, se observa una diferencia significativa entre los promedios, con un valor de 4.30 para el grupo 'No Churn' y 3.49 para el grupo 'Churn Comercial'.

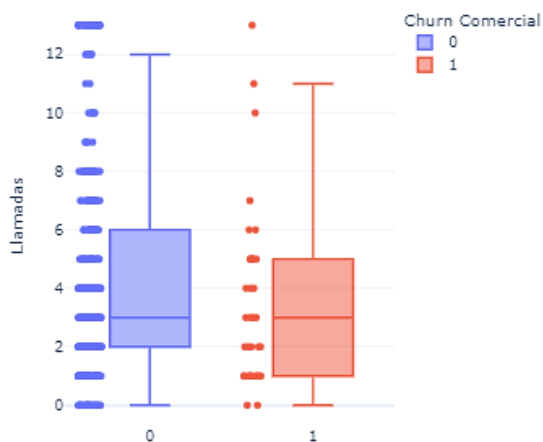


Figura 3-29 Boxplots de 'Llamadas_Efectivas_com' vs 'Churn Comercial'

Fuente: Elaboración propia, (enero, 2025)

Se puede evidenciar una diferencia significativa en los rangos intercuartílicos entre grupos para la variable 'Total_Actividades_exp' del embudo experiencia (Figura 3-30) además de una diferencia resaltante en los cuartiles Q2 entre grupos. Existe una diferencia significativa entre promedios de 5.55 para el grupo 'No Churn' y 6.29 para el grupo 'Churn Comercial'.

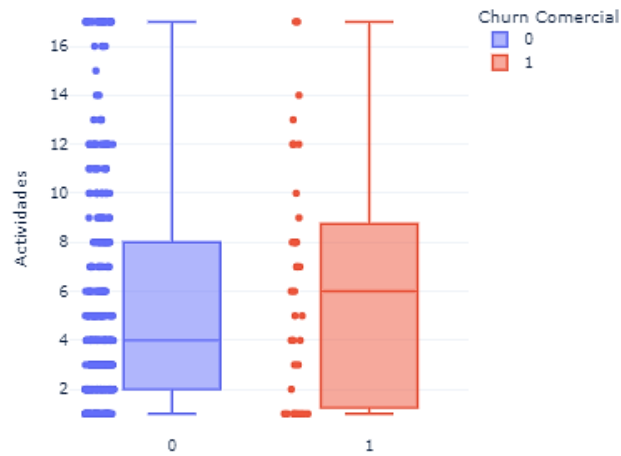


Figura 3-30 Boxplots de 'Total_Actividades_exp' vs 'Churn Comercial'

Fuente: Elaboración propia, (enero, 2025)

Se puede evidenciar una diferencia importante en los rangos intercuartílicos entre grupos para la variable 'Llamadas_No_Efectivas_exp' del embudo experiencia (Figura 3-31) además de una diferencia resaltante en los cuartiles Q2 entre grupos. Existe una diferencia significativa entre promedios de 0.78 para el grupo 'No Churn' y 1.91 para el grupo 'Churn Comercial'.

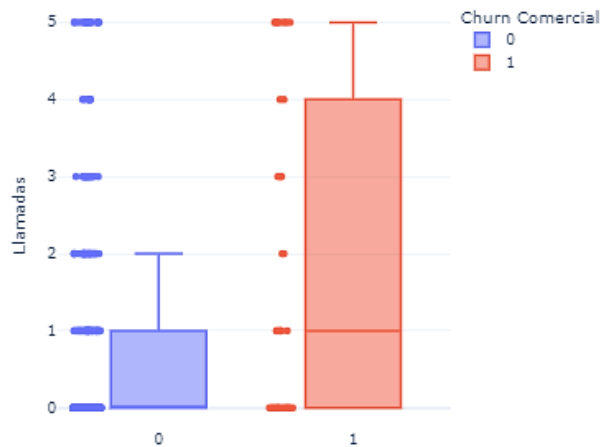


Figura 3-31 Boxplots de 'Llamadas_No_Efectivas_exp' vs 'Churn Comercial'

Fuente: Elaboración propia, (enero, 2025)

Se puede evidenciar una diferencia importante en los rangos intercuartílicos entre los grupos para las variables 'Capacitaciones_Hechas' y 'Capacitaciones_Canceladas' en el embudo de experiencia ((a) y (b) de la Figura 3-32), además de una diferencia destacada en los cuartiles Q2 entre grupos. Esto sugiere que una capacitación cancelada podría aumentar la probabilidad de Churn Comercial, mientras que una capacitación hecha favorecería la retención del cliente.

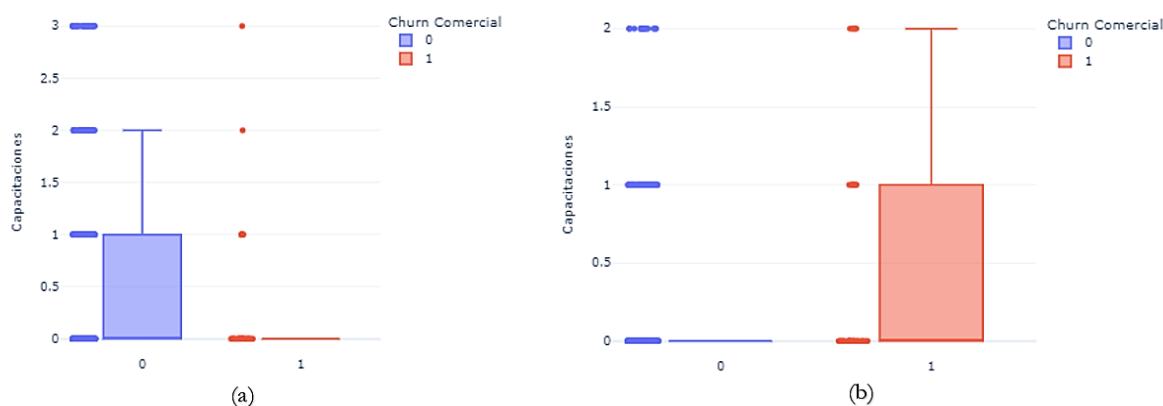


Figura 3-32 Boxplots de Capacitaciones ‘Hechas’ y ‘Canceladas’ vs ‘Churn Comercial’

Fuente: Elaboración propia, (enero, 2025)

3.6.1.4. Correlación entre variables

El análisis de correlación ‘Pearson’ revela una fuerte relación positiva entre las variables de la etapa comercial y aquellas de la etapa de experiencia, como se muestra en la Figura 3-33. Los coeficientes superiores a 0.80 indican una asociación positiva muy fuerte, lo que sugiere una alta correlación lineal y, por ende, una notable redundancia en los datos. Esta correlación podría generar problemas de multicolinealidad en el modelo, ya que las variables derivan de cálculos entre distintos campos.

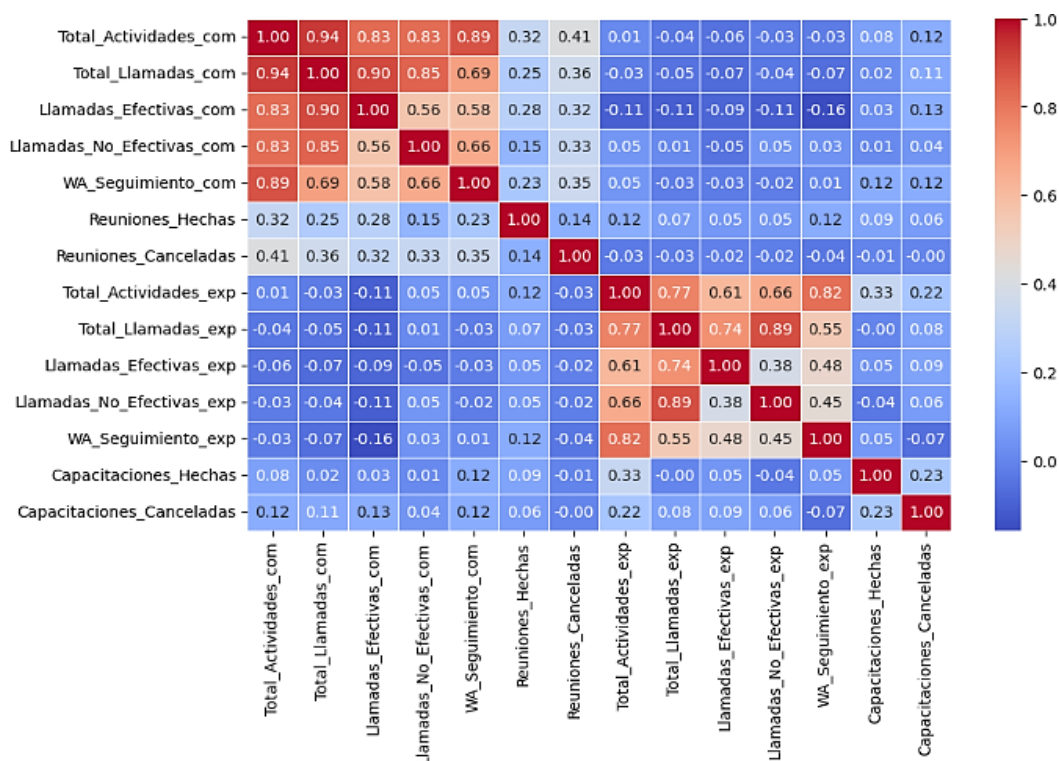


Figura 3-33 Matriz de correlación de Pearson entre variables numéricas.

Fuente: Elaboración propia, (enero, 2025)

3.6.1.5. Prueba Chi-cuadrado para Variables Cualitativas

Se emplea la prueba de Chi-cuadrado para evaluar la relación entre las variables cualitativas y la variable de interés ‘Churn Comercial’ del conjunto de datos.

- Hipótesis Nula (H0): No existe asociación entre las dos variables.
- Hipótesis Alternativa (H1): Existe una asociación entre las dos variables.

Variable	Estadístico Chi-Cuadrado	<i>p-value</i>	Grados de Libertad
'Tipo de cliente'	0.0031	0.9985	2
'(C) (EXP) Plazo y Pago'	126.9717	2.6819e-28	2
'Tipo Primer Contacto'	3.9511	0.2668	3
'Rango de Contacto'	0.9011	0.6373	2
'R1yR2'	0.0081	0.0081	1
'Tipo Primera Capacitación'	2.5199	0.1124	1
'Onboarding'	0.4205	0.5167	1
'New Categories'	0.4802	0.4883	1

Tabla 3-8 Prueba Chi-Cuadrado para variables cualitativas

Fuente: Elaboración propia, (enero, 2025)

El análisis de chi-cuadrado no revela una asociación significativa entre la variable ‘Tipo de cliente’ y la variable de interés. Con un estadístico de 0.0031 y un p-valor de 0.9985 para 2 grados de libertad, se concluye que las distribuciones de ‘Tipo de cliente’ son similares, sugiriendo que no impacta la variable de interés.

La prueba de ‘Tipo Primer Contacto’ arroja un estadístico de 3.9511 y un p-valor de 0.2668 para 3 grados de libertad, indicando que no hay diferencias significativas en las distribuciones entre las categorías. Para ‘Rango de Contacto’, el estadístico es 0.9011 con un p-valor de 0.6373 para 2 grados de libertad, confirmando la ausencia de una relación significativa.

En cambio, las variables ‘R1yR2’ y ‘(C) (EXP) Plazo y Pago’ muestran una asociación significativa con la variable de interés. ‘R1yR2’ presenta un estadístico de 0.0081 y un p-valor de 0.0081 para 1 grado de libertad, mientras que ‘(C) (EXP) Plazo y Pago’ tiene un estadístico de 126.9717 y un p-valor de 2.6819e-28 para 2 grados de libertad, lo que sugiere su influencia en la variable de interés.

Con respecto a ‘Tipo Primera Capacitación’, el estadístico es 2.5199 con un p-valor de 0.1124 para 1 grado de libertad, sin alcanzar significancia estadística. Las pruebas para ‘Onboarding’ y ‘New Categories’

muestran estadísticos de 0.4205 y 0.4802 con p-valores de 0.5167 y 0.4883, respectivamente, sin diferencias significativas.

En resumen, salvo 'R1yR2' y '(C) (EXP) Plazo y Pago', ninguna variable categórica muestra una asociación significativa con la variable de interés.

3.6.1.6. Prueba ANOVA para variables cuantitativas

Se emplea la prueba ANOVA (Análisis de Varianza) para evaluar si existen diferencias significativas entre las medias de los grupos de la variable de interés 'Churn Comercial' del conjunto de datos.

- Hipótesis Nula (H0): No existe diferencias significativas entre las medias de los grupos.
- Hipótesis Alternativa (H1): Existe diferencias significativas entre las medias de los grupos.

Variable	Estadístico F	<i>p-value</i>
'Total_Actividades_com'	0.963	0.3255
'Total_Llamadas_com'	0.9741	0.3243
'Llamadas_Efectivas_com'	1.6567	0.1989
'Llamadas_No_Efectivas_com'	0.2014	0.6539
'WA_Seguimiento_com'	1.2422	0.2658
'Reuniones_Hechas'	0.7119	0.3994
'Reuniones_Canceladas'	0.2622	0.6089
'Total_Actividades_exp'	0.8213	0.3654
'Total_Llamadas_exp'	16.9395	4.7585e-05
'Llamadas_Efectivas_exp'	4.8926	0.0276
'Llamadas_No_Efectivas_exp'	19.5968	1.2593e-05
'WA_Seguimiento_exp'	0.1172	0.7323
'Kickoff_Hechas'	2.5044	0.1144
'Kickoff_Canceladas'	0.0036	0.9520
'Capacitaciones_Hechas'	8.5273	0.0037
'Capacitaciones_Canceladas'	3.7444	0.0537

Tabla 3-9 Prueba ANOVA para variables cuantitativas

Fuente: Elaboración propia, (enero, 2025)

El análisis ANOVA revela que las variables:

- 'Total_Actividades_com',
- 'Total_Llamadas_com',
- 'Llamadas_Efectivas_com',
- 'Llamadas_No_Efectivas_com',
- 'WA_Seguimiento_com',
- 'Reuniones_Hechas',
- 'Reuniones_Canceladas'
- 'Total_Actividades_exp'

no presentan diferencias significativas entre las medias de grupos.

Por otro lado:

- 'Total_Llamadas_exp',
- 'Llamadas_Efectivas_exp',
- 'Llamadas_No_Efectivas_exp' y
- 'Capacitaciones_Hechas'

presentan diferencias significativas, sugiriendo una relación con la variable de interés.

Finalmente, 'Capacitaciones_Canceladas' muestra un valor cercano a la significancia, mientras que 'Kickoff_Hechas' y 'Kickoff_Canceladas' no presentan resultados significativos.

En resumen, las variables:

- 'Total_Llamadas_exp',
- 'Llamadas_Efectivas_exp',
- 'Llamadas_No_Efectivas_exp',
- 'Capacitaciones_Hechas'
- 'Capacitaciones_Canceladas' (en menor medida,)

presentan diferencias significativas con la variable de interés. El resto de las variables no muestran una relación significativa.

3.6.2. Preparación de los datos

3.6.2.1. Tratamiento de valores nulos

Dado que el conjunto de datos se compone principalmente de conteos de ocurrencias de eventos (Actividades), no se presentan valores nulos. Esto se debe a que, al extraer y transformar los datos en las etapas y subetapas de la capa intermedia (Silver Layer), los valores reflejan el número de actividades realizadas en las etapas comercial y de experiencia. Por lo tanto, en este caso, no se registran valores nulos.

3.6.2.2. Tratamiento de valores atípicos

Se reemplazaron los valores atípicos por valores más cercanos a los percentiles comprendidos entre el 95 y 99, lo que ayudó a mitigar el impacto de los valores extremos en los datos, suavizando su influencia y proporcionando una distribución más equilibrada. Al ajustar los outliers hacia el rango de las observaciones más frecuentes, se logra mayor estabilidad en los análisis y modelos posteriores. Esta técnica evita que los valores extremos distorsionen los resultados, mejorando la robustez del análisis y permitiendo una interpretación más precisa. Además, al conservar la mayor cantidad de datos posibles, se optimiza el entrenamiento del modelo y se mejora su capacidad de generalización, favoreciendo decisiones basadas en datos más sólidos.

3.6.2.3. Preparación de variables cualitativas

Las variables cualitativas, excepto la variable destino, se transformaron mediante la técnica de codificación one-hot (one hot encoding). Esta técnica convierte las categorías de una variable cualitativa en un conjunto de variables binarias (0 o 1), donde cada nueva variable indica la presencia o ausencia de una categoría específica. Esto convierte al conjunto de datos en un formato fácilmente entendible para la máquina y facilita el procesamiento de datos cualitativos en modelos de machine learning, permitiendo que las variables sean interpretadas numéricamente sin perder su información esencial.

3.6.2.4. Selección preliminar de variables relevantes

Se realizó una selección preliminar de variables para entrenar el modelo, basándose en criterios y análisis estadísticos, comenzando con un análisis de correlación, seguido de pruebas como Chi-Cuadrado para variables cualitativas y ANOVA para variables cuantitativas. Este paso es crucial en el proceso de modelado de Machine Learning, ya que optimiza el rendimiento del modelo al mejorar su precisión, reducir el tiempo de entrenamiento y facilitar su interpretación. Al eliminar variables irrelevantes o redundantes, no solo se evita el sobreajuste, sino que también se acelera el proceso de entrenamiento, generando modelos más comprensibles y enfocados en los factores realmente importantes.

Además, la selección de variables ayuda a prevenir la multicolinealidad, un problema identificado en el análisis de correlación lineal de Pearson, que puede afectar negativamente a modelos como la regresión logística o los árboles de decisión. Esta etapa contribuye a crear un modelo más robusto y eficiente, maximizando la calidad de las predicciones y facilitando la toma de decisiones basadas en datos.

3.7. Entrenamiento de Modelos

Se entrenaron varios modelos predictivos de clasificación con el objetivo de evaluar su rendimiento y seleccionar el más efectivo para la predicción de ‘Churn Comercial’.

3.7.1. Análisis del problema

Dada la naturaleza del problema de ‘Churn Comercial’ en la startup, se determina que los modelos predictivos de clasificación son ideales para predecir el abandono dentro de los primeros 90 días (métrica de interés). Estos modelos permiten identificar patrones en los datos históricos y predecir cuándo un

cliente tiene más probabilidad de abandonar, facilitando la implementación de estrategias personalizadas para mejorar la retención de clientes.

3.7.2. Selección de Algoritmos

Se opta por seleccionar una variedad de algoritmos que incluyen modelos de regresión, y modelos de árboles. Entre los elegidos se encuentran Naive Bayes, Decision Tree, Random Forest, Logistic Regression y XGBoost. Esta diversidad de modelos se selecciona para abordar distintos aspectos del problema de ‘Churn Comercial’ (Anexo 8).

La selección de los algoritmos se basa en la cantidad de datos disponibles, lo cual es un factor determinante para elegir aquellos que mejor puedan predecir la variable objetivo, considerando las limitaciones de los datos con los que se cuenta.

3.7.3. Etapa Preliminar de Implementación

En esta etapa del proyecto se presenta la estructura básica que se utiliza para un análisis de modelos de machine learning del proyecto. La estructura se divide en 7 secciones:

Sección 1: Importación de Bibliotecas

```
import pandas as pd
from sklearn.model_selection
import train_test_split
from sklearn.metrics
from xgboost import XGBClassifier
import accuracy_score, classification_report, confusion_matrix
import matplotlib.pyplot as plt
import seaborn as sns
```

Se importan las bibliotecas necesarias para el análisis y modelado de datos, incluyendo pandas para la manipulación de datos, Scikit-Learn para la división del conjunto de datos y la evaluación del modelo, XGBoost como algoritmo de clasificación, y las bibliotecas de visualización matplotlib y seaborn.

Sección 2: Carga de Datos

```
df = pd.read_csv(r'..\data\output_silver\contact_metrics_clean.csv')
```

La tabla resultante de la capa plata se importa desde un archivo CSV y se convierte en un DataFrame, un formato optimizado para su manipulación con pandas. Gracias a la estructura Medallion, los datos han sido tratados y depurados, obteniendo una versión limpia y lista para el entrenamiento de modelos de Machine Learning.

Sección 4: Selección de Variables

Se realizó una selección de variables antes del entrenamiento de los modelos, basándose en un análisis estadístico de su importancia.

a) Análisis de Correlación Lineal

Se conservaron las variables con una correlación fuera del rango de -0.05 a 0.05 respecto a la variable de interés (ver Figura 3-34), descartando aquellas con alta correlación entre sí (rango superior a ± 0.8). Las variables seleccionadas en este análisis fueron las siguientes:

- ‘Llamadas_Efectivas_com’
- ‘WA_Seguimiento_com’
- ‘Llamadas_Efectivas_exp’
- ‘Llamadas_No_Efectivas_exp’
- ‘Capacitaciones_Hechas’
- ‘Capacitaciones_Canceladas’

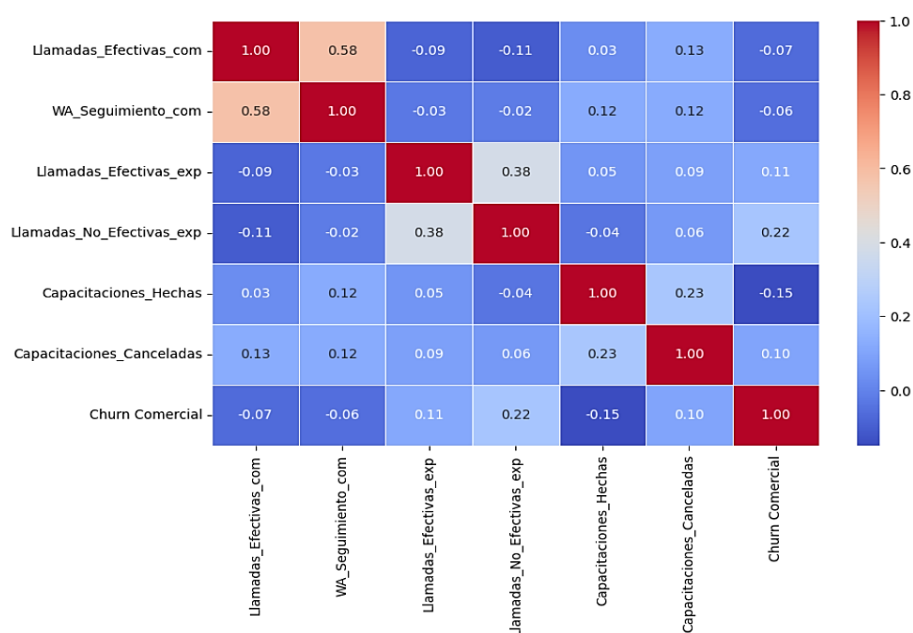


Figura 3-34 Matriz de correlación de Pearson entre variables numéricas seleccionadas.

Fuente: Elaboración propia, (enero, 2025)

b) Prueba de Chi-Cuadrado

Se realizó la prueba Chi-Cuadrado debido a que se están tratando con datos muestrales. Se mantuvieron las variables cualitativas con significancia estadística en la prueba de chi-cuadrado. Las variables seleccionadas en este análisis fueron las siguientes:

- ‘R1yR2’
- ‘(C) (EXP) Plazo y Pago’

c) Prueba ANOVA

Se conservaron las variables cuantitativas con significancia en la prueba ANOVA. Las variables seleccionadas en este análisis fueron las siguientes:

- 'Total_Llamadas_exp'
- 'Llamadas_Efectivas_exp'
- 'Llamadas_No_Efectivas_exp'
- 'Capacitaciones_Hechas' (significancia moderada)

d) Contexto del Negocio

Se incluyó la variable 'Tipo de cliente', considerada esencial para el análisis en el contexto empresarial. La clasificación de los clientes se basó en un conjunto de características definidas previamente por la empresa a partir de estudios previos.

Sección 4: Preprocesamiento

```
features = [
    '(C) (EXP) Plazo y Pago',
    'Tipo de cliente'
]
dummies = pd.get_dummies(df_subset[features])
df_encoded = pd.concat([df_subset.drop(features, axis=1), dummies], axis=1)
```

Se aplicó One-Hot Encoding a las variables cualitativas antes del entrenamiento del modelo, transformándolas en una representación numérica adecuada. Este proceso es esencial para garantizar que los algoritmos de Machine Learning puedan interpretar correctamente la información categórica.

```
# Separar features y target
X = df_encoded.drop(columns=['Churn Comercial'])
y = df_encoded['Churn Comercial']
```

Los datos se dividen en dos conjuntos: 'x', que contiene todas las columnas excepto 'Churn Comercial' (variable objetivo), y 'y', que almacena 'Churn Comercial'. Esta división es fundamental para el entrenamiento y evaluación de los modelos predictivos de clasificación.

Sección 5: Creación de Conjuntos de Entrenamiento y Prueba

```
# Dividir datos en train y test
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.20,
                                                    random_state=42, stratify=y)
```

Los datos se dividen en conjuntos de entrenamiento (X_train, y_train) y prueba (X_test, y_test) utilizando la función train_test_split de Scikit-Learn. Se asigna el 20% de los datos al conjunto de prueba, asegurando la reproducibilidad mediante una semilla aleatoria. El parámetro stratify=y en train_test_split() asegura que las clases se distribuyan proporcionalmente en los conjuntos de entrenamiento y prueba, manteniendo la misma proporción de clases en ambos.

Sección 6: Entrenamiento de Modelos

```
# Crear y entrenar el modelo de Regresión Logística
logistic_regression = LogisticRegression()
logistic_regression.fit(X_train, y_train)
```


En esta sección se importan los clasificadores de sklearn y el clasificador de XGBoost, se crea un modelo y se entrena con los datos de entrenamiento. Se prueban distintos algoritmos, como Decision Tree, Random Forest, Logistic Regression y Naive Bayes, para comparar su rendimiento y determinar el más adecuado.

Sección 7: Predicción

```
# Realizar predicciones en el conjunto de prueba
y_pred = logistic_regression.predict(X_test)
```

Se generan las predicciones sobre el conjunto de prueba ('X_test') utilizando el modelo entrenado y se almacenan los resultados en 'y_pred'.

Sección 8: Evaluación

Esta sección calcula las métricas de evaluación del modelo, como la precisión, el informe de clasificación y la matriz de confusión. Además, se analizan otras métricas de rendimiento, como recall, F1-score, entre otras, para obtener una visión más detallada del desempeño del modelo. A continuación, se presentarán los resultados de estas métricas de evaluación.

3.8. Evaluación y selección del modelo

Se evaluaron las métricas obtenidas tras un entrenamiento preliminar de los modelos, lo que permitió proceder con ajustes y optimizaciones para mejorar su rendimiento.

3.8.1. Evaluación preliminar de modelos

La sección de Evaluación del código representa la estructura estándar para la evaluación de modelos de aprendizaje automático en este proyecto, y se divide en 4 partes clave.

Sección 1: Cálculo de Precisión global

```
# Calcular métricas de evaluación
accuracy = accuracy_score(y_test, y_pred)
```

La precisión del modelo mide su rendimiento general, representando la proporción de predicciones correctas en relación con el total de predicciones realizadas. Para calcularla, utilizamos la función 'accuracy_score' de Python.

Sección 2: Informe de Clasificación

```
# Calcular el informe de clasificación
report = classification_report(y_test, y_pred, target_names=target_names)
```

El Informe de Clasificación, producido a través de la función 'classification_report' en Python, resume las métricas más importantes, tales como precisión, recall, F1-score y soporte, proporcionando un análisis detallado del desempeño del modelo para cada clase. Esto permite una evaluación más exhaustiva del rendimiento del modelo en distintas categorías.

Sección 3: Matriz de Confusión

```
# Calcular y mostrar la matriz de confusión en texto
confusion_df = pd.DataFrame(confusion_matrix(y_test, y_pred,
labels=logistic_regression.classes_,
                                index=logistic_regression.classes_,
                                columns=logistic_regression.classes_))
```

El código crea una matriz de confusión para evaluar el desempeño del modelo de clasificación, comparando las predicciones realizadas (y_pred) con las etiquetas reales del conjunto de prueba (y_test). La matriz se organiza en un DataFrame, lo que permite interpretar fácilmente el rendimiento del modelo, mostrando las clasificaciones correctas e incorrectas por cada clase.

Sección 4: Área bajo la Curva ROC (AUC-ROC)

```
# Calcular el AUC-ROC para cada clase
auc_roc_scores = {}
for i in range(len(logistic_regression.classes_)):
    auc_roc_scores[logistic_regression.classes_[i]] =
        roc_auc_score(y_test == logistic_regression.classes_[i],
                        y_pred == logistic_regression.classes_[i])
```

El código calcula el Área bajo la Curva ROC (AUC-ROC) para cada clase en un modelo de clasificación, utilizando la función 'roc_auc_score' de Scikit-Learn para evaluar la capacidad del modelo para distinguir entre las clases. El resultado es un diccionario llamado 'auc_roc_scores', que contiene las puntuaciones AUC-ROC para cada clase. Este análisis ayuda a comprender cómo el modelo equilibra la tasa de verdaderos positivos y falsos positivos en cada clase, ofreciendo información sobre su sensibilidad y especificidad.

3.8.1.1. Ajuste de Modelos

Los Procesos de Optimización de Modelos tienen como objetivo encontrar la configuración más adecuada de los hiperparámetros para maximizar la precisión de los algoritmos de Machine Learning. Para ello, emplean estrategias como la Grid Search y Random Search, combinadas con validación cruzada (cross-validation), con el fin de evaluar y ajustar continuamente el rendimiento del modelo. Este enfoque busca alcanzar un equilibrio óptimo entre ajuste y generalización, logrando así modelos más precisos y eficientes.

```
# Realizar SMOTE para balancear las clases
smote = SMOTE(sampling_strategy='auto', random_state=42)
X_train_bal, y_train_bal = smote.fit_resample(X_train, y_train)
```

Como se mencionó y observó en el Análisis Exploratorio de Datos univariado para la variable de interés 'Churn Comercial', el conjunto de datos presenta dos obstáculos relevantes para el desarrollo de este proyecto, por un lado, la cantidad limitada de datos (373 observaciones) y, por otro lado, el desbalance de las clases en la variable de interés (ver Figura 3-27). Ambos factores pueden afectar el rendimiento y la precisión de los modelos predictivos, ya que un conjunto de datos pequeño limita la capacidad de generalización del modelo, mientras que el desbalance puede llevar a que el modelo se sesgue hacia la clase mayoritaria, reduciendo su efectividad para predecir la clase minoritaria.

Para abordar el desbalanceo en los datos, se utilizó la técnica SMOTE (Synthetic Minority Over-sampling Technique) previo al entrenamiento de todos los modelos

, que consiste en generar muestras sintéticas de la clase minoritaria para equilibrar la distribución entre las clases. Esta técnica crea nuevas instancias basadas en los puntos cercanos de la clase minoritaria, lo que ayuda a mejorar la capacidad predictiva del modelo, evitando que el algoritmo se sesgue hacia la clase mayoritaria. Al aplicar SMOTE, se optimiza la representación de ambas clases en el conjunto de entrenamiento, lo que mejora la precisión y robustez del modelo, especialmente en escenarios con un desbalance significativo en las clases de salida.

3.8.1.1.1. Ajuste Modelo Naive Bayes

Los parámetros ajustados para mejorar el modelo Naive Bayes fueron los siguientes

```
# Definir el modelo Naive Bayes con parámetros ajustados
naive_bayes = GaussianNB(priors=[0.1, 0.9])
```

Se ajustó el parámetro `priors`, que establece la probabilidad a priori para cada clase, se ajusta este parámetro debido al desbalanceo de las clases.

```
# Definición de hiperparámetros
param_grid = {
    'var_smoothing': [1e-10, 1e-9, 1e-8, 1e-7, 1e-6, 1e-5, 1e-4, 1e-3]}
```

Se definen los hiperparámetros, donde:

- **var_smoothing:** Este parámetro de suavizado ajusta la cantidad pequeña que se añade a las varianzas de las características, con el fin de mejorar la estabilidad numérica del modelo.

3.8.1.1.2. Ajuste Modelo Logistic Regression

Los parámetros ajustados para el modelo de regresión logística fueron:

```
# Definir el modelo de Regresión Logística con parámetros ajustados
logistic_regression = LogisticRegression(max_iter=700,
                                         class_weight = {0: 1, 1: 20},
                                         random_state=42)
```

Se ajusta el parámetro `class_weight` para manejar el desbalance de clases, y también se define `max_iter=500` para establecer un número máximo de iteraciones para la convergencia del modelo.

```
# Definición de hiperparámetros
param_grid = {
    'C': [0.01, 0.1, 1, 10, 100, 1000, 10000],
    'penalty': ['l1', 'l2'],
    'solver': ['liblinear', 'saga']}
```

Se definen los hiperparámetros, donde:

- **C:** Controla la regularización, equilibrando el ajuste del modelo y la penalización de errores.
- **penalty:** Define el tipo de regularización (L1 o L2) para evitar coeficientes grandes.
- **solver:** Especifica el algoritmo para optimizar los coeficientes del modelo.

3.8.1.1.3. Ajuste Modelo Decision Tree Classifier

Los prametros ajustados para el modelo de árbol de decisión fueron los siguientes:

```
# Definir el modelo de DecisionTreeClassifier con parámetros ajustados
decision_tree = DecisionTreeClassifier(
    random_state=42,
    max_depth=5,
    min_samples_split=10,
    min_samples_leaf=5)
```

Se define un árbol de decisión con parámetros iniciales como max_depth=5, min_samples_split=10 y min_samples_leaf=5 para evitar el sobreajuste y asegurar que el árbol no sea demasiado complejo.

```
# Definición de hiperparámetros
param_dist = {
    'criterion': ['gini', 'entropy', 'log_loss'],
    'max_depth': [None] + list(range(5, 20)),
    'min_samples_split': list(range(3, 50, 5)),
    'min_samples_leaf': list(range(1, 20)),
    'min_impurity_decrease': [0.0] + [i / 1000.0 for i in range(1, 51)],
    'class_weight': [None, 'balanced', {0: 1, 1: 10}, {0: 1, 1: 20}],
    'max_features': ['sqrt', 'log2', None],
    'splitter': ['best', 'random']}
```

Se definen los hiperparámetros, donde:

- **criterion:** Función para medir la calidad de una división (puede ser 'gini', 'entropy', o 'log_loss').
- **max_depth:** Profundidad máxima del árbol. Limita el número de niveles que puede tener.
- **min_samples_split:** Mínimo de muestras necesarias para dividir un nodo. Controla la complejidad del árbol.
- **min_samples_leaf:** Mínimo de muestras en cada hoja. Evita hojas con pocas muestras, reduciendo el riesgo de sobreajuste.
- **min_impurity_decrease:** Disminución mínima de la impureza para hacer una división. Sirve para regularizar el árbol.
- **class_weight:** Ajuste de pesos de las clases, útil para manejar clases desbalanceadas (puede ser 'balanced' o un diccionario de pesos).
- **max_features:** Número de características a considerar en cada división (puede ser 'sqrt', 'log2', o None).
- **splitter:** Criterio para dividir los nodos (puede ser 'best' o 'random').

Estos parámetros controlan la complejidad del modelo y cómo se maneja la división de los nodos.

3.8.1.1.4. Ajuste de Modelo Random Forest

Los ajustes realizados al modelo de RandomForest son los siguientes:

```
# Definir el modelo RandomForestClassifier con parámetros ajustados
random_forest = RandomForestClassifier(
    random_state=42,
    max_depth=5,
    min_samples_split=5,
    min_samples_leaf=2,
    class_weight={0: 1, 1: 20})
```

Los parámetros ajustados para el modelo de Random Forest fueron:

- **random_state=42:** Asegura que los resultados sean reproducibles.
- **max_depth=5:** Limita la profundidad de los árboles para evitar el sobreajuste.
- **min_samples_split=5:** Requiere al menos 5 muestras para dividir un nodo, lo que ayuda a generalizar mejor.
- **min_samples_leaf=2:** Exige al menos 2 muestras en cada hoja, evitando divisiones demasiado específicas.
- **class_weight={0: 1, 1: 20}:** Aumenta el peso de la clase minoritaria (clase 1), lo que mejora la predicción de esa clase en un conjunto de datos desbalanceado.

Estos ajustes buscan reducir el sobreajuste y mejorar el rendimiento del modelo en datos desbalanceados.

```
# Definición de hiperparámetros
param_dist = {
    'n_estimators': [50, 100, 200, 300],
    'max_features': ['sqrt', 'log2', None],
    'bootstrap': [True, False],
    'max_depth': [3, 5, 7, None],
    'min_samples_split': [2, 5],
    'min_samples_leaf': [1, 2]}
```

Se definen los hiperparámetros, donde:

- **n_estimators:** Número de árboles en el bosque aleatorio. Un mayor número puede mejorar el rendimiento, pero también aumenta el tiempo de computación.
- **max_features:** Número máximo de características a considerar para dividir un nodo. Puede ser 'sqrt' (raíz cuadrada del número total de características), 'log2' o None (considera todas las características).
- **bootstrap:** Controla si se utiliza muestreo con reemplazo para crear los subconjuntos de datos de cada árbol. Puede ser True (usando reemplazo) o False (sin reemplazo).
- **max_depth:** Profundidad máxima de los árboles. Limita el número de niveles que un árbol puede tener. Un valor mayor permite árboles más profundos y complejos.
- **min_samples_split:** Número mínimo de muestras requeridas para dividir un nodo. Si es un valor más alto, genera árboles más pequeños y menos propensos al sobreajuste.

- **min_samples_leaf**: Número mínimo de muestras requeridas para ser una hoja. Similar al anterior, controla la complejidad y sobreajuste.
- **class_weight**: Permite ajustar los pesos de las clases. En este caso, la clase 1 tiene más peso (20) que la clase 0 (1), lo que es útil para manejar clases desbalanceadas.

Este conjunto de parámetros controla la complejidad y la capacidad del modelo para ajustarse a los datos, especialmente para problemas con clases desbalanceadas, como en este caso.

3.8.1.1.5. Ajuste de Modelo XGBoost

Los ajustes realizados al modelo de XGBoost son los siguientes:

```
# Crear el modelo de XGBoost con parámetros ajustados
xgboost_model = XGBClassifier(
    objective='binary:logistic',
    eval_metric='logloss',
    n_jobs=1)
```

Los parámetros ajustados para el modelo de XGBoost fueron:

- **objective='binary:logistic'**: El modelo se utilizará para una tarea de clasificación binaria.
- **eval_metric='logloss'**: Utiliza la pérdida logarítmica como métrica de evaluación.
- **n_jobs=1**: Especifica que se usará un solo núcleo de procesamiento.

En conjunto, estos parámetros mejoran la precisión y la capacidad del modelo para generalizar, reduciendo el sobreajuste y manejando clases desbalanceadas.

```
# Definición de hiperparámetros
param_dist = {
    'learning_rate': [0.001, 0.005],
    'n_estimators': [100, 200, 300],
    'max_depth': [2],
    'subsample': [0.7, 0.8, 0.9],
    'colsample_bytree': [0.7, 0.8],
    'gamma': [0.1, 0.3, 0.5],
    'reg_alpha': [0.5, 1, 5],
    'reg_lambda': [2, 5, 10],
    'min_child_weight': [7, 10]
}
```

Se ajusta el espacio de hiperparámetros para incluir opciones para:

- **learning_rate**: Controla la tasa de actualización de los coeficientes del modelo.
- **n_estimators**: Define el número de árboles en el modelo de XGBoost.
- **max_depth**: Especifica la profundidad máxima de los árboles para evitar sobreajuste.
- **subsample**: Ajusta la fracción de muestras utilizadas para entrenar cada árbol.
- **colsample_bytree**: Establece la fracción de características utilizadas por árbol.
- **gamma**: Controla la regularización sobre la complejidad del modelo.

- **reg_alpha**: Regula la penalización L1 para los coeficientes.
- **reg_lambda**: Regula la penalización L2 para los coeficientes.
- **min_child_weight**: controla la cantidad mínima de peso.

3.8.1.2. Optimización de modelos

Se utiliza GridSearchCV para realizar una búsqueda exhaustiva de hiperparámetros en los modelos Naive Bayes, Logistic Regression, probando todas las combinaciones definidas en los hiperparametros 'param_grid' durante el proceso de ajuste. La métrica de rendimiento empleada es 'accuracy'. En este contexto, 'model_classifier' hace referencia al modelo específico, y el procedimiento evalúa y selecciona la mejor combinación de hiperparámetros para maximizar la precisión del modelo.

```
# Búsqueda de hiperparámetros con validación cruzada
grid_search = GridSearchCV(naive_bayes, param_grid, scoring='accuracy', cv=cv)
grid_search.fit(X_train_bal, y_train_bal)
```

Se utilizó RandomizedSearchCV para los modelos de Decision Tree, Random Forest y XGBoost, con un espacio de búsqueda definido en 'param_dist'. La métrica de evaluación fue la precisión y se aplicó validación cruzada (cv).

```
# Búsqueda aleatoria de hiperparámetros
random_search = RandomizedSearchCV(
    decision_tree,
    param_distributions=param_dist,
    cv=cv,
    n_iter=30,
    random_state=42)
random_search.fit(X_train_bal, y_train_bal)
```

3.8.1.2.1. Validación Cruzada

La validación cruzada se lleva a cabo con 'StratifiedKFold' para todos los modelos de aprendizaje automático, utilizando 5 pliegues, lo que asegura que la distribución de las clases se mantenga equilibrada en cada pliegue durante el proceso de evaluación.

```
# Validación cruzada
cv = StratifiedKFold(n_splits=5, shuffle=True, random_state=42)
```

3.8.1.2.2. Mejoras en el Rendimiento de Modelos

Se añadió la calibración de los modelos Naive Bayes y Regresión Logística utilizando el método 'sigmoid' de CalibratedClassifierCV para ajustar las probabilidades predichas y mejorar la clasificación en problemas con clases desbalanceadas. Además, se ajustó el umbral de clasificación para la clase 1 a 0.6, buscando mejorar la precisión de la clase minoritaria. Este enfoque proporciona una mayor fiabilidad en las predicciones y es especialmente útil en modelos sensibles al desbalance de clases.

```
# Calibrar el modelo utilizando CalibratedClassifierCV
calibrated_logistic = CalibratedClassifierCV(best_logistic_regression,
method='sigmoid', cv=cv)
calibrated_logistic.fit(X_train_scaled, y_train_bal)
```

```
# Realizar predicciones con el mejor modelo calibrado
y_pred_prob = calibrated_logistic.predict_proba(X_test_scaled)[: , 1]

# Ajustar el umbral para la clase 1
threshold = 0.6
y_pred = (y_pred_prob >= threshold).astype(int)
# Evaluar el modelo con el nuevo umbral
accuracy = accuracy_score(y_test, y_pred)
```

3.8.2. Selección del modelo

Se realizó una comparativa exhaustiva de las métricas de evaluación de los modelos predictivos de clasificación, los cuales fueron entrenados, ajustados y optimizados previamente. Este análisis permitió evaluar el rendimiento de cada modelo, teniendo en cuenta métricas como precisión, Recall, F1-Score y AUC-ROC, para seleccionar el más adecuado. La comparación de las métricas nos permitió identificar el modelo más adecuado para predecir eficientemente el ‘Churn Comercial’, asegurando la selección del modelo con mejor capacidad de discriminación y ajuste a las características del conjunto de datos.

4. Resultados y Discusión

En esta sección se detalla el análisis del estudio, incluyendo los datos clave y los resultados obtenidos a lo largo de todo el proyecto.

4.1. Entendimiento de las necesidades y objetivos

En sus inicios (2019) la startup, registró un alto índice de ‘churn’, con 66 clientes anulando su suscripción, de los cuales el 21.21% (14 leads) correspondieron a ‘churn comercial’ como se puede observar en la Figura 4-1. Sin embargo, desde medio año se observó una disminución significativa. La gestión 2021, la empresa no experimentó ‘Churn Comercial’, manteniendo un promedio de 15 clientes perdidos mensualmente, lo que representó una tasa de abandono relativamente baja.

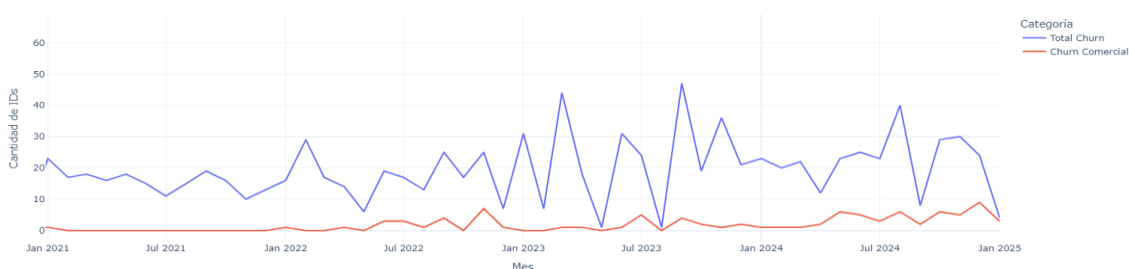


Figura 4-1 Comportamiento de ‘Churn Comercial’ en comparación a ‘Churn’ a lo largo del tiempo.

Fuente: Elaboración propia, (enero, 2025)

En la gestión 2024, el ‘Churn Comercial’ mostró una tendencia ascendente como se puede observar en la Figura 4-2., con un crecimiento del 20.09% y un promedio mensual de 3.9 clientes perdidos.

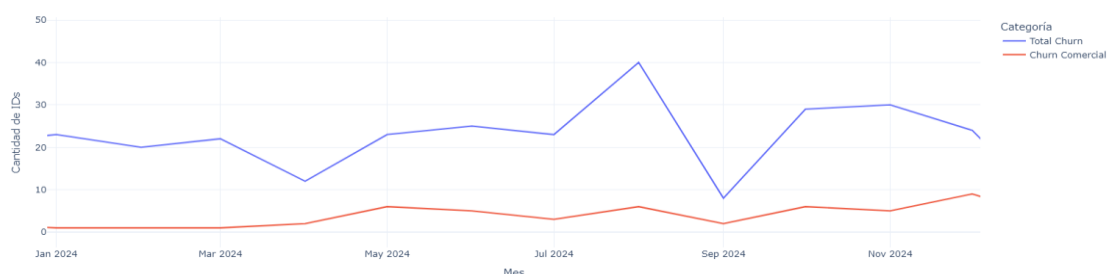


Figura 4-2 Comportamiento de ‘Churn Comercial’ en comparación a ‘Churn’ Gestión 2024

Fuente: Elaboración propia, (enero, 2025)

Este incremento es crucial para la startup, ya que un aumento en el ‘churn comercial’ indica problemas en la retención de clientes durante la etapa de onboarding, lo que afecta el crecimiento rápido y escalado de la empresa. Es vital abordar este desafío identificando a los clientes con mayor riesgo de abandono dentro de los primeros 90 días para optimizar los recursos y reducir el ‘Churn Comercial’

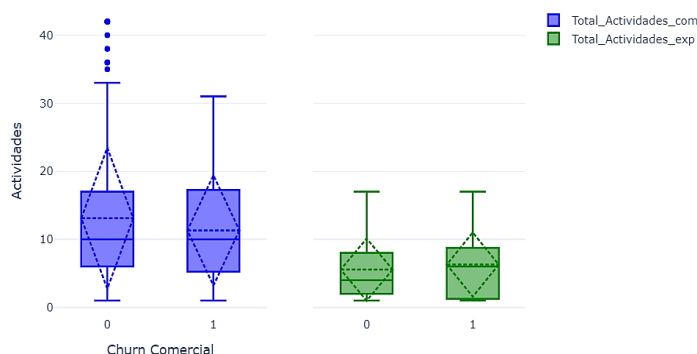


Figura 4-3 Boxplots de total de actividades entre embudos vs Churn Comercial

Fuente: Elaboración propia, (enero, 2025)

Se observa una diferencia significativa en el total de actividades entre los embudos, con una diferencia en las medias de 6 a 10 en el grupo que experimenta 'Churn Comercial'. Esto podría indicar que se dedica más seguimiento y recursos a la etapa comercial, mientras que la etapa de experiencia no recibe la misma atención en términos de interacción con los clientes.

4.2. Extracción y comprensión de los datos

La extracción y análisis de datos se realizaron en VSCode, utilizando librerías como pandas, requests, numpy y matplotlib para la manipulación, visualización y agrupamiento de la información. Esto permitió la construcción de un DataFrame final, listo para su exploración y posterior uso en el entrenamiento de modelos predictivos de clasificación en Machine Learning (Ver Anexo 9). Sin embargo, uno de los principales desafíos fue la comprensión de la API del CRM Pipedrive, debido a su estructura y documentación, lo que dificultó la integración de los datos en las primeras etapas del proceso. Finalmente en esta etapa se obtuvo información relevante sobre clientes y sus interacciones, permitiendo identificar patrones de contacto asociados a la retención o pérdida de clientes

4.3. Análisis y proceso de datos

Se llevó a cabo un proceso de limpieza y transformación de datos para garantizar su calidad y adecuación al modelado. Esto incluyó el manejo de valores nulos mediante imputación estratégica (percentiles 95 al 99), la estandarización de variables para mejorar la estabilidad de los modelos y la creación de nuevas características relevantes basadas en patrones de interacción de los clientes. Además, se identificó un desbalance significativo en la variable objetivo 'Churn Comercial', lo que afectó el rendimiento de los modelos predictivos.

4.4. Entrenamiento de Modelos

En esta fase del proyecto, se completó la preparación de datos y el entrenamiento de modelos para predecir el 'Churn Comercial'. Se limpiaron los datos, seleccionando variables clave mediante análisis estadísticos y transformándolas con One-Hot Encoding. Luego, los datos fueron divididos en conjuntos de entrenamiento y prueba, aplicando SMOTE para balancear clases. Se entrenaron modelos como

regresión logística, Naive Bayes y árboles de decisión, evaluándolos con métricas como precisión y AUC-ROC. Finalmente, se ajustaron los hiperparámetros con Grid Search y Random Search, optimizando el rendimiento para seleccionar el mejor modelo predictivo.

El ajuste de hiperparámetros en el entrenamiento de modelos de machine learning juega un papel clave en su rendimiento predictivo. Los modelos optimizados, que han pasado por procesos de calibración, ajuste de umbral y validación cruzada, han mostrado mejoras significativas en comparación con sus versiones preliminares. En particular, se observa una mejora en la precisión después del ajuste en los modelos Logistic Regression (88% a 92%), Decision Tree (92% a 93.33%) y XGBoost (88.00% a 92.00%). Esta mejora sugiere que la elección cuidadosa de los hiper parámetros puede optimizar un modelo de forma óptima como en el caso de Logistic Regression con una mejora del 5.33% en la precisión. Estos resultados destacan la importancia de ajustar los hiperparámetros y la optimización de modelos en la construcción de modelos para optimizar su rendimiento y confiabilidad (ver Figura 4-4).

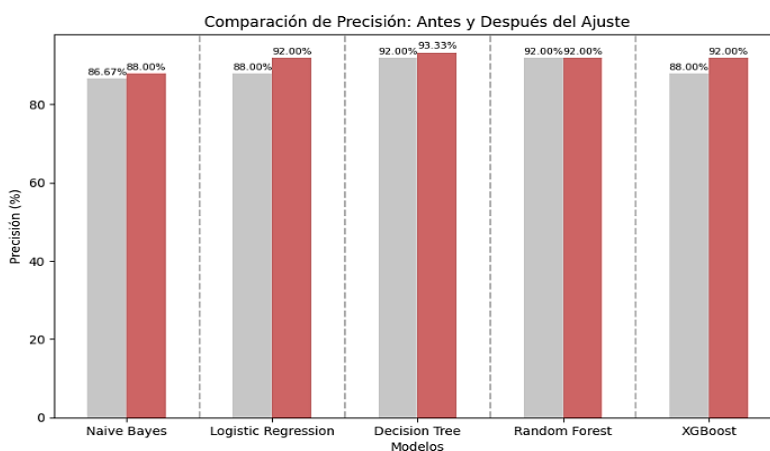


Figura 4-4 Comparación de Precisión entre modelos antes y después de ajustes

Fuente: Elaboración propia, (enero, 2025)

4.4.1. Resultados de la Evaluación: Modelo Naive Bayes

Los resultados de la evaluación del modelo Naive Bayes contempla el análisis e interpretación de las siguientes métricas:

Matriz de Confusión

La matriz de confusión del modelo Naive Byes para la clasificación de ‘Churn Comercial’, presentada en la Figura 4-7, se interpreta de la siguiente manera:

- 60 casos fueron correctamente clasificados como ‘No Churn Comercial’ (0), clientes que no abandonan y fueron identificados de forma correcta.
- 6 casos fueron correctamente clasificados como ‘Churn Comercial’ (1), clientes que abandonan y fueron identificados de forma correcta

- 8 casos fueron falsos positivos, el modelo predijo que sería un caso de ‘Churn Comercial’ pero en realidad los clientes no abandonaron.
- 1 caso fue falso negativo, el modelo predijo que no sería un caso de ‘Churn Comercial’ pero en realidad el cliente si abandonó.

El modelo tiene un buen desempeño identificando clientes que no abandonan (60 aciertos), pero tiene dificultades al predecir correctamente los clientes que sí abandonan, ya que solo detecta 6 casos de ‘Churn Comercial’, mientras que comete 8 falsos positivos y 1 falso negativo. Esto sugiere que el modelo puede estar sesgado hacia la clase mayoritaria (‘No Churn Comercial’), lo que podría afectar su utilidad para predecir ‘Churn Comercial’ con precisión.

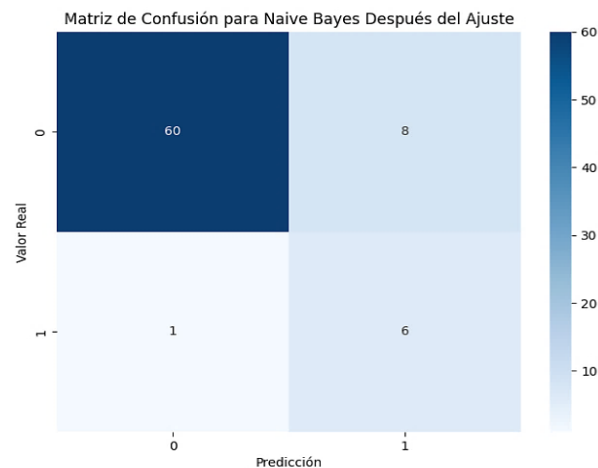


Figura 4-5 Matriz de Confusión Modelo Naive Bayes

Fuente: Elaboración propia, (enero, 2025)

Métricas de Clasificación

El modelo Naive Bayes ajustado alcanza una precisión global del 86.67%, con un buen desempeño en la clase ‘No Churn Comercial’ (0) (100% precisión, 87% Recall, 93% F1-score). Sin embargo, presenta limitaciones en la detección de ‘Churn Comercial’ (1), con una precisión del 44%, un Recall del 100% y F1-Score de 61%, lo que indica dificultades para identificar clientes que abandonan. El AUC-ROC de 0.9338 sugiere que logra diferenciar entre clases efectivamente.

Clase	Precisión	Recall	F1-Score	AUC-ROC
0	0.98	0.88	0.93	0.8697
1	0.43	0.86	0.57	0.8697

Tabla 4-1 Métricas de Evaluación Modelo Naive Bayes

Fuente: Elaboración propia, (enero, 2025)

Curva de Aprendizaje

En la evaluación del modelo preliminar, se observa que la precisión del conjunto de entrenamiento es ligeramente superior a la del conjunto de prueba, lo que indica un posible sobreajuste. A medida que el tamaño del conjunto de entrenamiento aumenta, la brecha entre ambas precisiones se reduce hasta solaparse, aunque la precisión de prueba no muestra una mejora significativa, estabilizándose en torno al 85%. Este comportamiento sugiere que el modelo inicial tiene dificultades para generalizar correctamente y podría estar memorizando patrones específicos en los datos de entrenamiento.

Después de los ajustes, el modelo muestra un comportamiento más estable en la curva de aprendizaje. Se evidencia una mejor convergencia entre la precisión de entrenamiento y prueba, especialmente a medida que se incrementa el tamaño del conjunto de entrenamiento. La precisión de prueba alcanza valores cercanos a los obtenidos en entrenamiento, lo que indica una mejor capacidad de generalización. Además, la reducción de la brecha entre ambas curvas sugiere que el modelo ajustado ha reducido el sobreajuste, logrando un balance más adecuado. (Ver Figura 4-8).

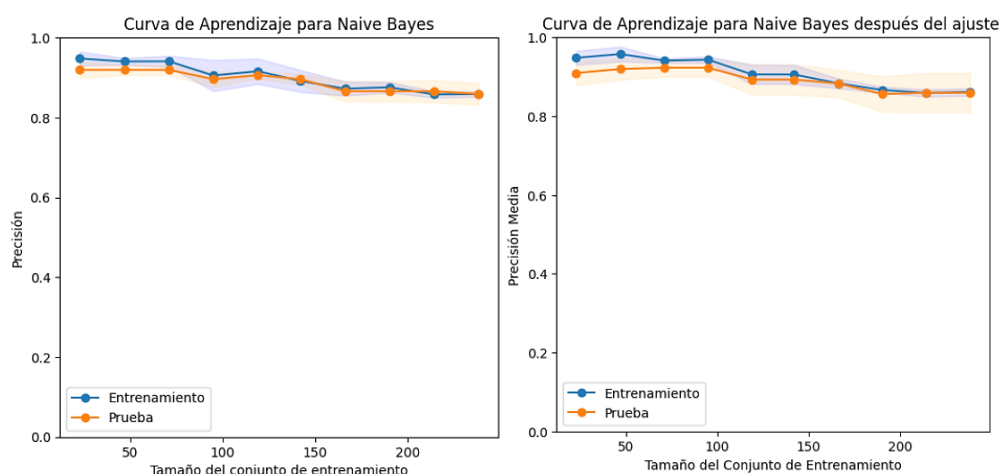


Figura 4-6 Curvas de aprendizaje Modelo Naive Bayes

Fuente: Elaboración propia, (enero, 2025)

4.4.2. Resultados de Evaluación: Modelo Logistic Regression

La matriz de confusión del modelo Logistic Regression para la clasificación de 'Churn Comercial', presentada en la Figura 4-9, se interpreta de la siguiente manera:

- 66 casos fueron correctamente clasificados como 'No Churn Comercial' (0), clientes que no abandonan y fueron identificados de forma correcta.
- 3 casos fueron correctamente clasificados como 'Churn Comercial' (1), clientes que abandonan y fueron identificados de forma correcta
- 2 casos fueron falsos positivos, el modelo predijo que sería un caso de 'Churn Comercial' pero en realidad los clientes no abandonaron.

- 4 casos fueron falsos negativos, el modelo predijo que no serían un caso de ‘Churn comercial’ pero en realidad los clientes si abandonaron.

El modelo tiene un buen desempeño identificando clientes que no abandonan (66 aciertos), pero tiene dificultades al predecir correctamente los clientes que sí abandonan, ya que solo detecta 3 casos de ‘Churn Comercial’, mientras que comete apenas 2 falsos positivos pero 4 falsos negativos. Esto sugiere que el modelo puede estar sesgado hacia la clase mayoritaria (‘No Churn Comercial’), sin embargo, logra identificar correctamente una mayor proporción de casos de ‘Churn Comercial’ en comparación con los falsos positivos, lo que indica un buen equilibrio en la clasificación de la clase minoritaria.

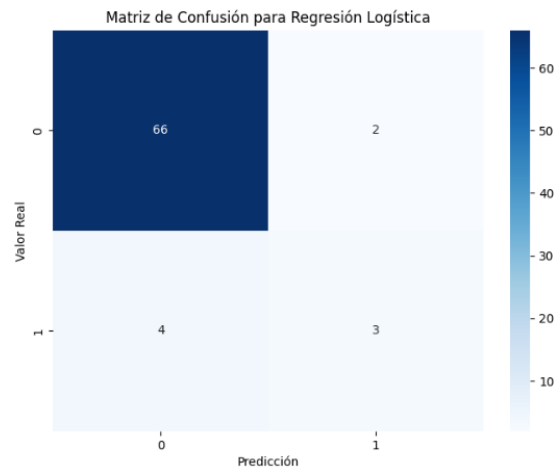


Figura 4-7 Matriz de Confusión Modelo Logistic Regression

Fuente: Elaboración propia, (enero, 2025)

Métricas de Clasificación

El modelo Logistic Regression ajustado alcanza una precisión global del 92%, con un buen desempeño en la clase ‘No Churn Comercial’ (0) (94% precisión, 97% Recall, 96% F1-score). Sin embargo, presenta limitaciones en la detección de ‘Churn Comercial’ (1), con una precisión del 60%, un Recall del 43% y F1-Score de 50%, lo que indica dificultades para identificar clientes que abandonan, pero una mejora significativa en el balanceo de clasificación de clases. El AUC-ROC de 0.6 significa que logra apenas diferenciar entre clases. (Ver Tabla 4-2).

Clase	Precisión	Recall	F1-Score	AUC-ROC
0	0.94	0.97	0.96	0.6996
1	0.60	0.43	0.50	0.6996

Tabla 4-2 Métricas de Evaluación Modelo Logistic Regression

Fuente: Elaboración propia, (enero, 2025)

Curva de Aprendizaje

En el modelo preliminar, la precisión en el conjunto de entrenamiento se mantiene alta, pero la del conjunto de prueba muestra ligeras variaciones sin una tendencia clara de mejora, lo que sugiere un posible sobreajuste. En contraste, en la versión ajustada, aunque la precisión del entrenamiento disminuye gradualmente, la precisión del conjunto de prueba se estabiliza y converge con la de entrenamiento a medida que aumenta el tamaño del conjunto de datos. Este comportamiento indica una mejor capacidad de generalización, reduciendo el riesgo de sobreajuste y logrando un desempeño más consistente en datos no vistos (Ver Figura 4-10). El comportamiento de la precisión sugiere que el modelo podría ajustarse mejor a una mayor cantidad de datos.

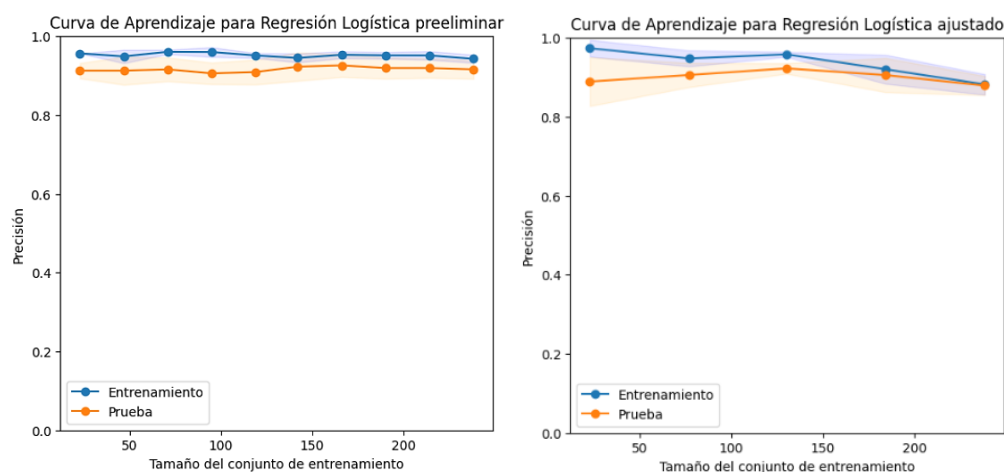


Figura 4-8 Curvas de aprendizaje Modelo Logistic Regression

Fuente: Elaboración propia, (enero, 2025)

4.4.3. Resultados de la Evaluación: Modelo Decision Tree

Los resultados de la evaluación del modelo Decision Tree contempla el análisis e interpretación de las siguientes métricas:

Matriz de Confusión

La matriz de confusión del modelo Decision Tree para la clasificación de 'Churn comercial', presentada en la Figura 4-11, se interpreta de la siguiente manera:

- 63 casos fueron correctamente clasificados como 'No Churn Comercial' (0), clientes que no abandonan y fueron identificados de forma correcta.
- 7 casos fueron correctamente clasificados como 'Churn Comercial' (1), clientes que abandonan y fueron identificados de forma correcta
- 5 casos fueron falsos positivos, el modelo predijo que sería un caso de 'Churn Comercial' pero en realidad los clientes no abandonaron.
- 0 casos fueron falsos negativos, el modelo predijo que no sería un caso de 'Churn Comercial' pero en realidad los clientes si abandonaron.

El modelo tiene un buen desempeño, similar al de Logistic Regression identificando clientes que no abandonan (63 aciertos), pero tiene dificultades al predecir correctamente los clientes que sí abandonan, ya que solo detecta 7 casos de ‘Churn Comercial’, mientras que comete 5 falsos positivos. Esto sugiere que el modelo puede estar sesgado hacia la clase mayoritaria (‘No Churn Comercial’), lo que podría afectar su utilidad para predecir ‘Churn Comercial’ con precisión.

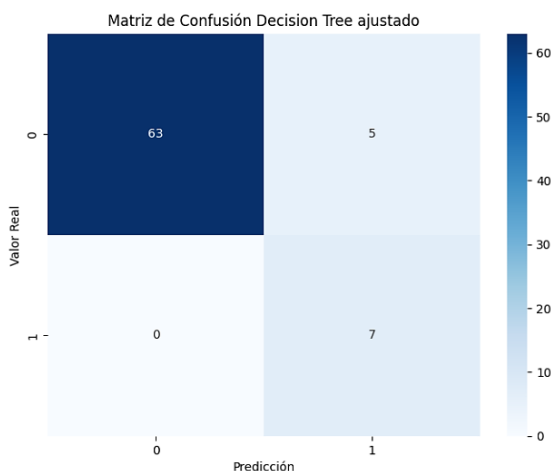


Figura 4-9 Matriz de Confusión Modelo Decision Tree

Fuente: Elaboración propia, (enero, 2025)

Métricas de Clasificación

El modelo Decision Tree ajustado alcanza una precisión global del 93.33%, con un buen desempeño en la clase ‘No Churn Comercial’ (0) (0.98% precisión, 94% Recall, 96% F1-score). Sin embargo, presenta limitaciones en la detección de ‘Churn Comercial’ (1), con una precisión del 60%, un Recall del 86% y F1-Score de 71%, lo que indica dificultades para identificar clientes que abandonan. El AUC-ROC de 0.8992 sugiere que logra diferenciar entre clases efectivamente. En general, el modelo presenta un rendimiento equilibrado en cuanto a las métricas entre las diferentes clases. (Ver Tabla 4-3).

Clase	Precisión	Recall	F1-Score	AUC-ROC
0	0.98	0.94	0.96	0.8992
1	0.60	0.86	0.71	0.8992

Tabla 4-3 Metricas de Evaluación Modelo Decision Tree

Fuente: Elaboración propia, (enero, 2025)

Curva de Aprendizaje

En el modelo preliminar, la precisión en el conjunto de entrenamiento es perfecta, lo que indica que el modelo ha memorizado los datos, pero la precisión en el conjunto de prueba presenta una ligera variabilidad sin mejorar significativamente, lo que sugiere un sobreajuste. Aunque la precisión en el

conjunto de prueba aumenta con el tamaño del conjunto de entrenamiento, las fluctuaciones en su valor indican que el modelo no ha logrado generalizar de manera efectiva. Por otro lado, en la versión ajustada, la precisión en el entrenamiento disminuye gradualmente, pero la precisión en el conjunto de prueba muestra una tendencia de mejora constante y se estabiliza a medida que aumenta el tamaño del conjunto de datos. Este comportamiento sugiere que el modelo ha mejorado su capacidad de generalización, reduciendo el riesgo de sobreajuste y logrando un desempeño más estable y consistente en datos no vistos. (Ver Figura 4-12). El comportamiento de la precisión sugiere que el modelo podría ajustarse mejor a una mayor cantidad de datos.

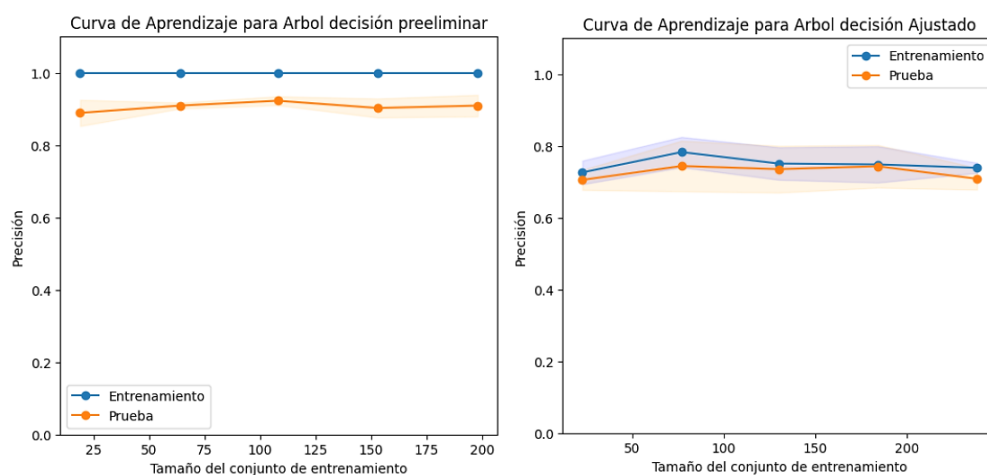


Figura 4-10 Curvas de aprendizaje Modelo Decision Tree

Fuente: Elaboración propia, (enero, 2025)

4.4.4. Resultados de la Evaluación: Modelo Random Forest

Los resultados de la evaluación del modelo Random Forest contempla el análisis e interpretación de las siguientes métricas:

Matriz de Confusión

La matriz de confusión del modelo Random Forest para la clasificación de 'Churn Comercial', presentada en la Figura 4-13, se interpreta de la siguiente manera:

- 63 casos fueron correctamente clasificados como 'No Churn Comercial' (0), clientes que no abandonan y fueron identificados de forma correcta.
- 6 casos fueron correctamente clasificados como 'Churn Comercial' (1), clientes que abandonan y fueron identificados de forma correcta
- 5 casos fueron falsos positivos, el modelo predijo que sería un caso de 'Churn Comercial' pero en realidad los clientes no abandonaron.
- 1 caso fue falso negativo, el modelo predijo que no sería un caso de 'Churn Comercial' pero en realidad el cliente sí abandonó.

El modelo tiene un buen desempeño identificando clientes que no abandonan (63 aciertos), pero tiene dificultades al predecir correctamente los clientes que sí abandonan, ya que solo detecta 6 casos de ‘Churn Comercial’, mientras que comete 5 falsos positivos y 1 falso negativo. Esto sugiere que el modelo puede estar sesgado hacia la clase mayoritaria (‘No Churn Comercial’), lo que podría afectar su utilidad para predecir churn con precisión. Esto indica que el modelo podría estar inclinado hacia la clase mayoritaria (‘No Churn Comercial’), pero a su vez, es capaz de identificar correctamente una mayor proporción de casos de ‘Churn Comercial’ en relación con los falsos positivos, lo que refleja un buen equilibrio en la clasificación de la clase minoritaria.

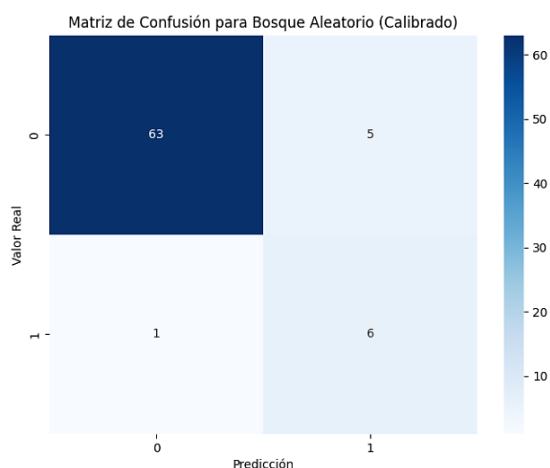


Figura 4-11 Matriz de Confusión Modelo Random Forest

Fuente: Elaboración propia, (enero, 2025)

Métricas de Clasificación

El modelo Random Forest ajustado alcanza una precisión global del 92%, con un buen desempeño en la clase ‘No Churn Comercial’ (0) (98% precisión, 93% Recall, 95% F1-score). Sin embargo, presenta limitaciones en la detección de ‘Churn Comercial’ (1), con una precisión del 55%, un Recall del 86% y F1-Score de 67%, lo que indica dificultades para identificar clientes que abandonan. El AUC-ROC de 0.8918 sugiere que logra diferenciar entre clases efectivamente. (Ver Tabla 4-4).

Clase	Precisión	Recall	F1-Score	AUC-ROC
0	0.98	0.93	0.95	0.8918
1	0.55	0.86	0.67	0.8918

Tabla 4-4 Métricas de Evaluación Modelo Random Forest

Fuente: Elaboración propia, (enero, 2025)

Curva de Aprendizaje

En el modelo preliminar, la precisión en el conjunto de entrenamiento se mantiene constante en 100%, mientras que la precisión en el conjunto de prueba muestra variabilidad sin una tendencia clara, lo que sugiere un posible sobreajuste. A pesar de que la precisión del conjunto de prueba aumenta con el tamaño del conjunto de entrenamiento, la brecha entre ambos conjuntos se mantiene, lo que indica que el modelo podría estar memorizando los datos. En el modelo ajustado, la precisión en el conjunto de entrenamiento muestra una disminución progresiva, lo que refleja un modelo más balanceado. La precisión en el conjunto de prueba mejora en ciertas etapas, aunque presenta fluctuaciones, lo que podría indicar que aún hay margen de mejora en la generalización del modelo. Sin embargo, la convergencia de las precisiones de entrenamiento y prueba en algunos puntos indica una mayor capacidad de generalización, reduciendo el riesgo de sobreajuste en comparación con el modelo preliminar. (Ver Figura 4-14).

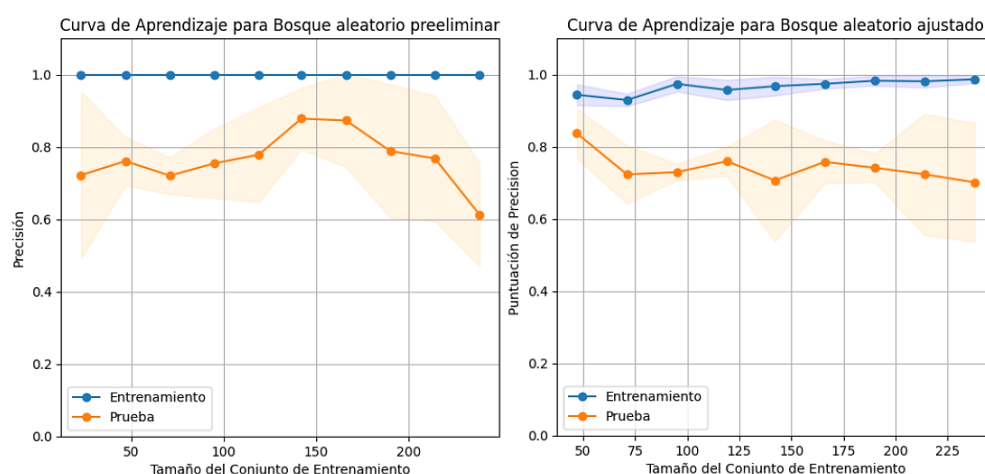


Figura 4-12 Curvas de aprendizaje Modelo Random Forest

Fuente: Elaboración propia, (Enero, 2025)

4.4.5. Resultados de la Evaluación: Modelo XGBoost

Los resultados de la evaluación del modelo XGBoost contempla el análisis e interpretación de las siguientes métricas:

Matriz de Confusión

La matriz de confusión del modelo XGBoost para la clasificación de 'Churn Comercial', presentada en la Figura 4-15, se interpreta de la siguiente manera:

- 62 casos fueron correctamente clasificados como 'No Churn Comercial' (0), clientes que no abandonan y fueron identificados de forma correcta.
- 7 casos fueron correctamente clasificados como 'Churn Comercial' (1), clientes que abandonan y fueron identificados de forma correcta
- 6 casos fueron falsos positivos, el modelo predijo que sería un caso de 'Churn Comercial' pero en realidad los clientes no abandonaron.

- Ningún caso fue falso negativo.}

El modelo tiene un buen desempeño identificando clientes que no abandonan (62 aciertos), pero tiene dificultades al predecir correctamente los clientes que sí abandonan, ya que solo detecta 7 casos de ‘Churn Comercial’, mientras que comete 6 falsos positivos. Esto sugiere que el modelo puede estar sesgado hacia la clase mayoritaria (‘No Churn Comercial’), lo que podría afectar su utilidad para predecir ‘Churn Comercial’ con precisión.

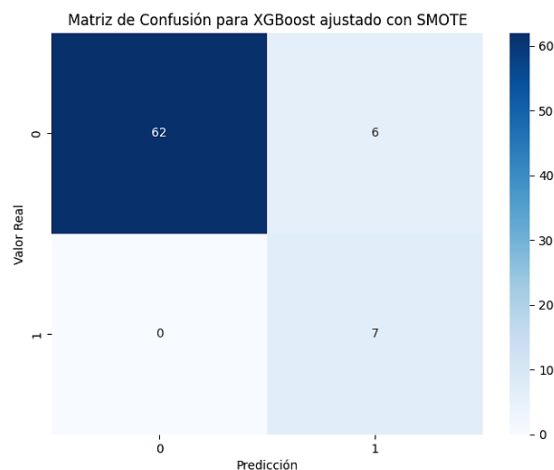


Figura 4-13 Matriz de Confusión Modelo XGBoost

Fuente: Elaboración propia, (Enero, 2025)

Métricas de Clasificación

El modelo Random Forest ajustado alcanza una precisión global del 92%, con un buen desempeño en la clase ‘No Churn Comercial’ (0) (100% precisión, 87% Recall, 93% F1-score). Sin embargo, presenta limitaciones en la detección de ‘Churn Comercial’ (1), con una precisión del 44%, un Recall del 100% y F1-Score de 61%, lo que indica dificultades para identificar clientes que abandonan. El AUC-ROC de 0.9338 sugiere que logra diferenciar entre clases efectivamente. (Ver Tabla 4-4).

Clase	Precisión	Recall	F1-Score	AUC-ROC
0	1.00	0.91	0.95	0.8351
1	0.54	1.00	0.70	0.8351

Tabla 4-5 Métricas de Evaluación Modelo XGBoost

Fuente: Elaboración propia, (enero, 2025)

Curva de Aprendizaje

En el modelo preliminar, la precisión en el conjunto de entrenamiento es consistentemente alta, alcanzando valores cercanos al 100%, lo que indica que el modelo ha memorizado los datos. Sin embargo,

la precisión en el conjunto de prueba muestra ligeras variaciones sin una tendencia clara de mejora, lo que sugiere un posible sobreajuste. Por otro lado, en la versión ajustada, la precisión en el entrenamiento disminuye levemente, mientras que la precisión en el conjunto de prueba comienza con valores más bajos, pero muestra una tendencia de mejora constante y se estabiliza en niveles superiores a medida que aumenta el tamaño del conjunto de datos. Este comportamiento sugiere que el modelo ha mejorado su capacidad de generalización, reduciendo el riesgo de sobreajuste y logrando un desempeño más estable y consistente en datos no vistos. (Ver Figura 4-16).

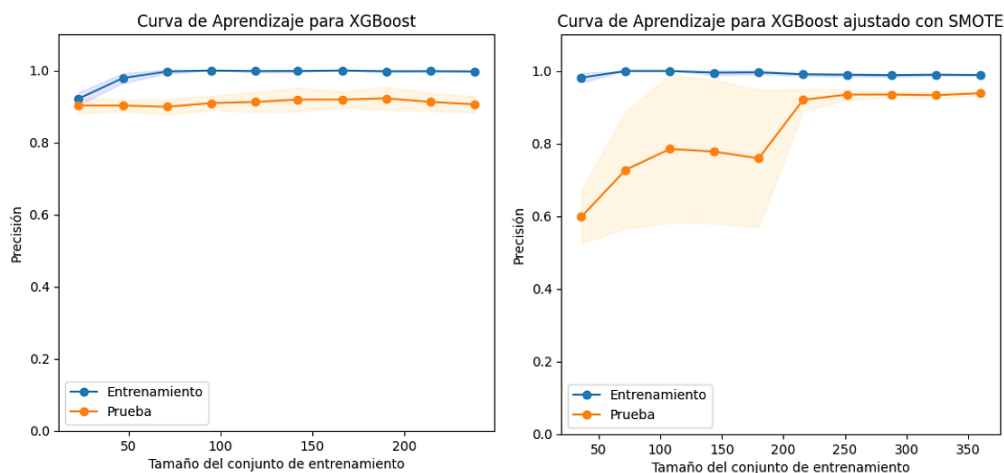


Figura 4-14 Curvas de aprendizaje Modelo XGBoost

Fuente: Elaboración propia, (Enero, 2025)

4.4.6. Resultados globales del entrenamiento

En esta sección se presentan los resultados obtenidos del entrenamiento de los modelos predictivos de clasificación, evaluados según diversas métricas globales de desempeño clave para determinar la capacidad predictiva de cada modelo en la tarea de predicción de 'Churn Comercial'

4.4.6.1. Precisión Global de Modelos

Los valores de Accuracy (precisión global) reflejan la capacidad de cada modelo para clasificar correctamente las muestras en el conjunto de prueba. Los resultados del entrenamiento de los distintos modelos para este proyecto dieron los resultados observados en la Figura 4-5.

- **Decision Tree:** lidera en el mayor índice de precisión entre modelos con un 93.33% de precisión. Esto significa que logra clasificar correctamente el 93% de las muestras de manera efectiva.
- **XGBoost, Random Forest y Logistic Regression:** Demuestran una sólida precisión en la clasificación, alcanzando un 92%. Esto lo convierte en una opción bastante efectiva.
- **Naive Bayes:** alcanzando el 88% de precisión, indica un rendimiento significativamente inferior en comparación al resto, esto significa que el modelo Naive Bayes logra apenas clasificar correctamente el 88% de los casos.

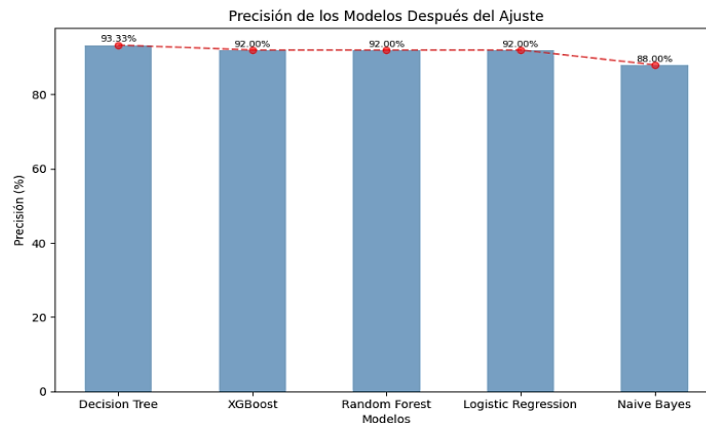


Figura 4-15 Precisión Global entre Modelos

Fuente: Elaboración propia, (enero, 2025)

4.4.6.2. Área bajo la Curva ROC de los modelos (AUC-ROC)

Los valores de AUC-ROC reflejan la capacidad de cada modelo para diferenciar entre las clases positivas y negativas en el conjunto de prueba. La Figura 4-6 muestra los resultados obtenidos tras el entrenamiento y ajuste de los modelos en este proyecto:

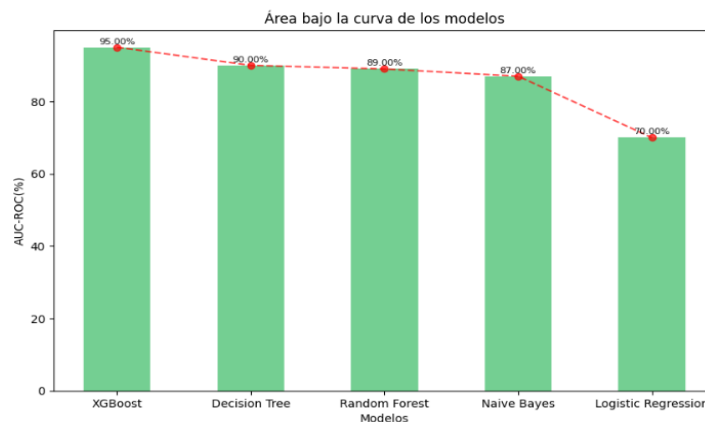


Figura 4-16 Comparación de AUC-ROC entre modelos

Fuente: Elaboración propia, (enero, 2025)

- **XGBoost:** Este modelo tiene el mayor valor de AUC-ROC, alcanzando un 95%. Esto indica una alta capacidad de discriminación entre clases, convirtiéndolo en la opción más robusta.
- **Decision Tree:** Logra un AUC-ROC del 90%, reflejando un desempeño sólido en la clasificación de los casos.
- **Random Forest:** Presenta un AUC-ROC del 89%, mostrando una precisión comparable a la del Decision Tree, con una ligera mejora en estabilidad.

- **Naive Bayes:** Obtiene un 87% de AUC-ROC, lo que indica un rendimiento aceptable, pero inferior a los modelos mencionados anteriormente.
- **Logistic Regression:** Registra el valor más bajo, con un 70% de AUC-ROC, lo que sugiere limitaciones en la capacidad de distinguir entre clases en este conjunto de datos.

4.4.6.3. Métricas 'macro_avg'

Los resultados generales de las métricas de evaluación, considerando el 'macro avg' de todos los modelos empleados en este proyecto, se resumen en la Tabla 4-6. El 'macro avg' calcula las métricas de manera independiente para cada clase, lo que significa que no se ve influenciado por la distribución de las clases.

Modelo	Precisión	Recall	F1-Score	AUC-ROC
Naive Bayes	0.88	0.87	0.75	0.87
Logistic Regression	0.92	0.70	0.73	0.70
Decision Tree	0.93	0.90	0.83	0.90
Random Forest	0.92	0.89	0.81	0.89
XGBoost	0.92	0.96	0.83	0.95

Tabla 4-6 Métricas de evaluación de todos los modelos 'macro avg'

Fuente: Elaboración propia, (enero, 2025)

En la evaluación comparativa de los modelos basada en métricas clave 'macro avg', el modelo XGBoost destaca por su alto Recall de 96%, lo que refleja su capacidad sobresaliente para identificar correctamente los casos de la clase minoritaria. Su AUC-ROC de 95% también indica una excelente capacidad de discriminación entre las clases, a pesar de que su precisión (92%) es similar a la de otros modelos como Logistic Regression (92%) y Random Forest (92%). Su F1-Score de 83% muestra un buen equilibrio entre precisión y Recall.

El modelo Decision Tree, con una precisión de 93%, un Recall de 90% y un F1-Score de 83%, ofrece un rendimiento sólido, pero no alcanza el nivel de discriminación del XGBoost. Por otro lado, el modelo Random Forest, con una precisión de 92%, un Recall de 89%, y un F1-Score de 81%, también muestra un buen rendimiento, pero no logra igualar el Recall del XGBoost.

El modelo Logistic Regression, con una precisión de 92%, presenta un Recall de solo 70%, lo que limita significativamente su capacidad para identificar correctamente la clase minoritaria. Su AUC-ROC de 70% indica una baja capacidad de discriminación entre las clases.

Finalmente, el modelo Naive Bayes, con una precisión de 88%, un Recall de 87% y un F1-Score de 75%, tiene un rendimiento relativamente bueno, pero sus métricas de Recall y F1-Score lo sitúan por debajo de los otros modelos en términos de efectividad para identificar la clase minoritaria.

En conclusión, el modelo XGBoost es el más equilibrado en términos de precisión, Recall y discriminación entre clases, y sería la opción preferida si se prioriza la identificación de la clase minoritaria y la capacidad general del modelo.

4.4.6.4. Métricas 'weighted_avg'

Los resultados generales de las métricas de evaluación, considerando el 'weighted avg' de todos los modelos empleados en este proyecto, se resumen en la Tabla 4-7. El "weighted avg" tomando en cuenta la distribución de clases en el conjunto de datos, lo que significa que se ve influenciado por la distribución de las clases.

Modelo	Precisión	Recall	F1-Score	AUC-ROC
Naive Bayes	0.88	0.88	0.90	0.87
Logistic Regression	0.92	0.92	0.91	0.70
Decision Tree	0.93	0.93	0.94	0.90
Random Forest	0.92	0.92	0.93	0.89
XGBoost	0.92	0.92	0.93	0.95

Tabla 4-7 Métricas de evaluación de todos los modelos 'weighted avg'

Fuente: Elaboración propia, (enero, 2025)

En la evaluación comparativa de los modelos basada en las métricas 'weighted avg', el modelo Decision Tree se destaca por su alto F1-Score de 94%, lo que indica un excelente equilibrio entre precisión y Recall, especialmente considerando que la clase minoritaria recibe un peso significativo. Además, su AUC-ROC de 90% lo coloca casi a la par con el modelo de Random Forest, que presenta un AUC-ROC de 89%. Los modelos XGBoost y Random Forest, con F1-Scores de 93% y 94%, muestran un rendimiento sólido, pero no alcanzan el nivel de discriminación del modelo Decision Tree en cuanto al F1-Score.

Por otro lado, el modelo Naive Bayes tiene una precisión de 88%, pero destaca en Recall con un 88%, lo que indica que es más sensible a la identificación de la clase minoritaria. Sin embargo, su desempeño sigue siendo inferior en comparación con los demás modelos.

La Logistic Regression, con una precisión de 92%, un Recall de 70%, un F1-Score de 73% y un AUC-ROC de 70%, muestra un rendimiento limitado en comparación con los otros modelos, especialmente en su capacidad para detectar correctamente las clases minoritarias y su discriminación entre clases. A pesar de su precisión relativamente alta, su desempeño en términos de Recall y AUC-ROC es considerablemente más bajo, lo que refleja ciertas limitaciones en su capacidad predictiva.

En conclusión, el modelo Decision Tree parece ser el modelo más equilibrado en cuanto a precisión, Recall y F1-Score, destacándose como el mejor modelo bajo la métrica 'weighted avg', seguido de cerca por los modelos XGBoost y Random Forest, que también ofrecen un rendimiento sólido.

4.5. Selección del modelo predictivo

Finalmente podemos concluir que el mejor modelo para las características de los datos con los que se han trabajado en el proyecto es XGBoost destaca como el modelo más equilibrado y robusto, con un excelente rendimiento tanto en 'macro avg' como en 'weighted avg'. En 'macro avg', se sobresale en AUC-ROC (0.95) y Recall (0.96), mostrando una buena capacidad para identificar la clase minoritaria. En 'weighted avg', mantiene un rendimiento sólido con un F1-Score de 0.93 y un AUC-ROC similar al de otros modelos, pero con un mejor manejo de ambas clases. En resumen, XGBoost es el modelo más robusto y adecuado para este caso.

4.5.1. Importancia de Variables

Según los resultados obtenidos en cuanto a importancia de las variables en el modelo seleccionado XGBoost, las variables mas relevantes para el modelo serian:

- **(C) (EXP) Plazo y Pago_Mensual (0.506052):** Esta variable tiene la mayor importancia, lo que indica que es muy relevante para la predicción del modelo.
- **(C) (EXP) Plazo y Pago_Anual (0.418233):** También tiene una importancia significativa, aunque ligeramente inferior a la anterior. Esto sugiere que el "Plazo y Pago Anual" también tiene un fuerte impacto en las predicciones.
- **Capacitaciones_Hechas (0.020052):** Aunque tiene una importancia mucho menor que las variables anteriores, sigue siendo algo relevante para la predicción. La variable está aportando algo de valor al modelo.
- **Capacitaciones_Canceladas (0.010440):** Esta variable también tiene baja importancia, pero sigue siendo considerada en el modelo.
- **WA_Seguimiento_com (0.006954):** Aunque con un valor pequeño, sigue siendo algo importante en las predicciones.
- **Tipo de cliente_C (0.006800):** La categoría "C" del tipo de cliente tiene una pequeña relevancia, indicando que afecta ligeramente el modelo.
- **Llamadas_Efectivas_exp (0.006357) y Llamadas_No_Efectivas_exp (0.006040):** Ambas tienen importancia similar, mostrando que las llamadas efectivas y no efectivas en la experiencia (exp) tienen un impacto moderado en la predicción.
- **R1yR2 (0.005158):** Esta variable también tiene una baja importancia, pero sigue aportando algo al modelo.
- **Llamadas_Efectivas_com (0.004963):** Similar a las variables anteriores, tiene un impacto pequeño.
- **Tipo de cliente_B (0.004590) y Tipo de cliente_A (0.004362):** Aunque representan diferentes categorías del tipo de cliente, su impacto es bajo en comparación con otras variables.
- **(C) (EXP) Plazo y Pago_Otros (0.000000):** Esta variable tiene una importancia nula, lo que significa que no está contribuyendo al modelo en absoluto. Probablemente, podrías considerar eliminarla en futuros entrenamientos.

En conclusión, podemos afirmar que las variables con mayor importancia son las relacionadas con Plazo y Pago (Mensual y Anual). Las variables de las actividades asociadas a Capacitaciones y Llamadas en la etapa de experiencia tienen un impacto mucho menor más no nulo, mientras que los tipos de clientes ‘C’ tienen más relevancia en el modelo que los tipos ‘A’ y ‘B’. La variable ‘(C)(EXP)Plazo y Pago_Otros’ no está aportando al modelo, por lo que se puede prescindir de ella sin afectar el rendimiento.

4.6. Discusión de resultados

En estudios previos sobre la aplicación de aprendizaje automático supervisado para la predicción del abandono de clientes, se evidencian los resultados obtenidos por los modelos propuestos por (Urrelo, 2024). En su investigación, se emplearon distintos algoritmos supervisados para la predicción del churn, coincidiendo en su mayoría con los modelos entrenados y evaluados en este proyecto. Podemos observar la comparativa de las métricas de evaluación obtenidas en el estudio de (Urrelo, 2024) y las métricas de evaluación obtenidas en este proyecto en la Tabla 4-8, donde ‘A’ representa las métricas obtenidas en el presente proyecto y ‘B’ las métricas obtenidas en el proyecto mencionado.

Modelo	Precisión		Recall		F1-Score		AUC-ROC	
	A	B	A	B	A	B	A	B
Naive Bayes	0.88	0.56	0.87	0.56	0.75	0.56	0.87	0.74
Logistic Regression	0.92	0.42	0.70	0.43	0.73	0.41	0.70	0.62
Decision Tree	0.93	0.90	0.90	0.90	0.83	0.90	0.90	0.91
Random Forest	0.92	0.92	0.89	0.87	0.81	0.90	0.89	0.92

Tabla 4-8 Comparación de métricas de evaluación entre proyectos

Fuente: Elaboración propia, (enero, 2025)

A pesar de que el proyecto considerado para la discusión no incluye el entrenamiento del modelo XGBoost, se tomara en cuenta el modelo seleccionado en base a las métricas de evaluación y se comparara con el modelo elegido en este proyecto. Los resultados obtenidos en el proyecto de (Urrelo, 2024) son bastante similares a los resultados obtenidos en este proyecto, como se pueden ver a continuación en la Figura 4-17.

Se encuentran diferencias notables entre resultados, tanto en la cantidad de datos con los que se trabajaron como en las características de los mismos. Mientras que en el proyecto de (Urrelo, 2024) se trabajó con un total de 1,849 datos, para el presente proyecto se utilizaron 373 datos, reflejando una diferencia bastante considerable en cuanto a cantidad de datos entre proyectos, sin embargo, ambos enfrentaron el desafío de contar con un volumen limitado de datos para el entrenamiento de los modelos.

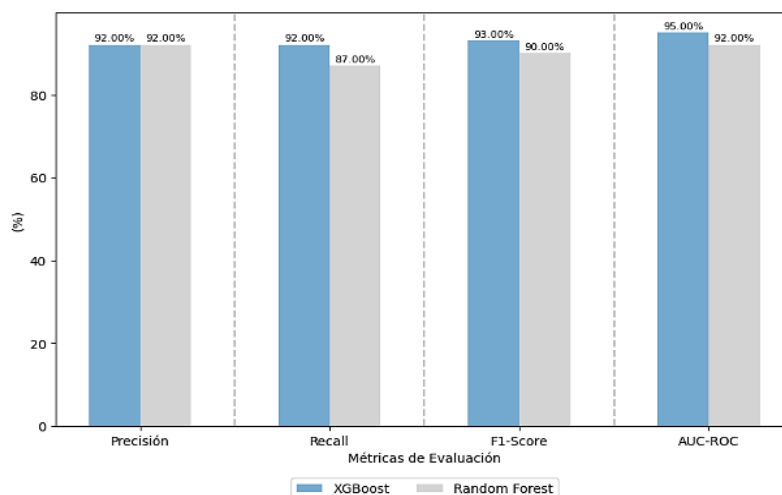


Figura 4-17 Comparación de métricas de evaluación entre modelos

Fuente: Elaboración propia, (enero, 2025)

Además, puede verse un contraste significativo en la característica de los datos, como se puede evidenciar en la Figura 4-18, mientras que en el proyecto de (Urrelo, 2024) se contó con un porcentaje de datos etiquetados como churn del 43% ('A'), en este proyecto solo el 9.38% ('B') de los datos corresponden al churn.

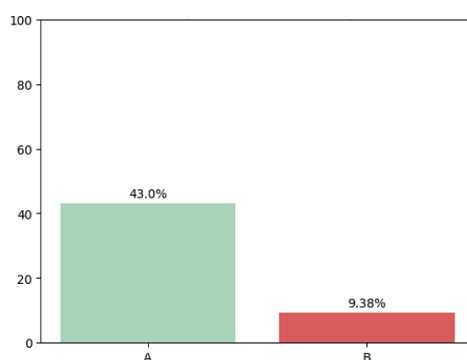


Figura 4-18 Comparación de porcentaje de churn entre proyectos

Fuente: Elaboración propia, (enero, 2025)

Otro aspecto a considerar al momento de evaluar las diferencias entre los resultados de los proyectos es que, el modelo de Urrelo aplica una clasificación multiclase como se puede ver en la Figura 4-19, para predecir la tasa de abandono en una empresa de alojamiento web, mientras que el modelo desarrollado en este proyecto se basa en una clasificación binaria de churn comercial (ver Figura 3-28), centrado exclusivamente en el abandono dentro de los primeros 90 días posteriores a la suscripción en una startup. Este modelo utiliza los patrones de contacto como principal criterio de análisis, a diferencia del enfoque de (Urrelo, 2024), que emplea características de los clientes como variables predictoras. Sin embargo, estas características ya están implícitamente consideradas en este proyecto a través de la clasificación del Tipo de Cliente.'

Es importante resaltar otra diferencia significativa al comparar ambos estudios: la composición de los datos utilizados. Mientras que en el proyecto de (Urrelo, 2024) se trabajó con datos balanceados (tomando en cuenta que 3 clases son consideradas como churn: ‘Suspended’, ‘Terminated’ y ‘Cancelled’), en este proyecto se manejaron datos desbalanceados (ver Figura 3-28), siendo el ‘churn’ la clase minoritaria. Esto implica que el procesamiento previo de los datos antes del entrenamiento fue más exhaustivo, con el fin de reducir el impacto del desbalance y evitar que afectara el rendimiento del modelo.

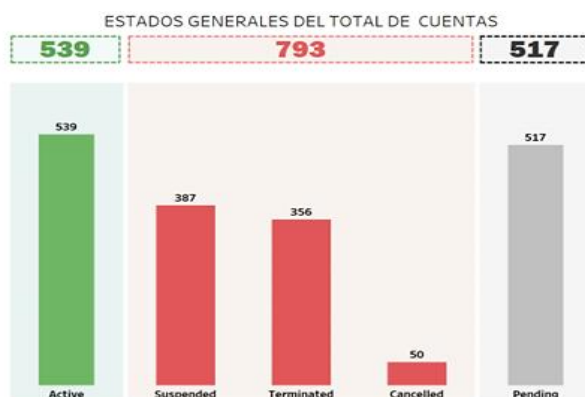


Figura 4-19 Gráfica de barras de estados de cuentas

Fuente: (Urrelo, 2024)

A pesar de que en ambos proyectos se decidieron por modelos diferentes, es relevante destacar las similitudes entre ellos, ya que ambos lograron obtener buenos resultados con el entrenamiento de modelos de Decision Tree. Este modelo podría ser una opción factible por su alta precisión; sin embargo, en este proyecto fue descartado debido a su bajo AUC-ROC por a la naturaleza de los datos utilizados. En consecuencia, podemos concluir que los modelos de Decision Tree son bastante efectivos para manejar datos con características comunes en ambos proyectos, como la cantidad limitada de datos, o para tratar con datos con clases desbalanceadas, como los utilizados en este proyecto, demostrando buenos resultados y un rendimiento sólido.

Según los resultados obtenidos en este estudio, se puede concluir que el objetivo planteado para este proyecto se logró de manera exitosa. El modelo de XGBoost ha demostrado su efectividad en la predicción del churn comercial de clientes, alcanzando una alta precisión del 92% en la variable objetivo binaria. Además, destacó por su capacidad de ajuste y generalización, incluso considerando la naturaleza de los datos. Su principal fortaleza fue el AUC-ROC, que con un 95% obtuvo el mejor desempeño entre los modelos evaluados, característica relevante por tratarse de una clasificación binaria con clases desbalanceadas.

5. Conclusiones

El análisis de datos y el desarrollo del modelo predictivo identificaron factores clave del ‘Churn Comercial’, permitiendo entender las necesidades y objetivos del negocio. No obstante, en el proceso la centralización de la información se vio dificultada por las diferentes perspectivas de las partes interesadas, lo que complicó la alineación de objetivos y requerimientos. A pesar de estos obstáculos, se logró una comprensión efectiva, lo que permitió que el proyecto se desarrollara de forma exitosa.

La extracción y normalización de datos se realizó de manera exitosa, aunque el proceso presentó varios desafíos, ya que consumir datos desde la API del CRM Pipedrive representó algo nuevo para el equipo. Esta tarea resultó compleja, ya que se tuvo que asegurar que los datos se transformaran correctamente en la capa bronce, convirtiéndolos en un formato más comprensible para su posterior análisis y procesamiento. Para lograrlo, se utilizaron herramientas de programación como Python en la consola de Visual Studio Code, junto con librerías especializadas como Pandas, NumPy y Requests, lo que facilitó la integración eficiente y fluida de los datos desde fuentes externas.

Durante el análisis exploratorio de datos se identificaron hallazgos clave sobre los patrones de contacto de los clientes a lo largo de su ciclo de vida. Estos hallazgos no solo reflejan la situación actual de la empresa, sino que también proporcionan valiosos recursos para optimizar la toma de decisiones y detectar oportunidades de mejora. Se analizaron los patrones de contacto de 373 clientes registrados entre 2021 y 2024, de los cuales 169 realizaron ‘churn’. Sin embargo, solo 35 de ellos correspondieron a ‘Churn Comercial’, es decir, abandonaron dentro de los primeros 90 días posteriores a su suscripción.

Según el análisis de datos, en 2024 se observó un aumento en el ‘Churn Comercial’ de clientes hacia fin de año, posiblemente influenciado por factores internos desconocidos y externos, como la crisis económica del país. Bolivia enfrentó crisis económicas agravadas por el alza del dólar, bloqueos e incendios, causando desabastecimiento, 8.8% de inflación y pérdidas millonarias (Aliaga, 2024).

La FEPC reportó 900 cancelaciones de matrículas comerciales en Cochabamba (FEPC, 2024). En Santa Cruz, la Fedemype registró el cierre de al menos 7,000 unidades productivas (eju, 2025) mientras que en El Alto, la Fermype indicó que el 80% de las pequeñas empresas cerraron en 2024 (Chambi, 2024). Al operar bajo un modelo B2B, su servicio de control de inventarios y facturación depende de la estabilidad de sus clientes, por lo que la reducción en la actividad empresarial impactó directamente en su retención. Esto se reflejó en el aumento del ‘churn’ a lo largo de 2024 y, en consecuencia, en el incremento del ‘Churn Comercial’ en la startup boliviana.

El análisis exploratorio de datos reveló que la mayoría de los clientes ganados son de tipo B (63%) y optan por el plan anual, lo cual podría deberse a la reciente introducción de planes mensuales en los últimos meses del 2024. El primer contacto en la etapa comercial suele ser efectivo, y no es usual que los clientes cancelen reuniones o capacitaciones.

No se observa una gran cantidad de clientes que hayan completado las reuniones R1 y R2 en la etapa comercial, ya que la división de estas reuniones se implementó en abril de 2024. Antes de esa fecha, solo se registraba como 'reunión'. Sin embargo, una proporción significativa de clientes ha cumplido con ambas reuniones, lo que sugiere que el cambio en el flujo de trabajo ha sido exitoso en mejorar la conversión de clientes.

Se observó una diferencia significativa en la cantidad de actividades entre los embudos. La mediana del embudo comercial es de 10 actividades, mientras que la del embudo de experiencia es de 4. Esto sugiere un mayor seguimiento en el área comercial, lo que podría afectar negativamente la retención de clientes a largo plazo. Sin embargo, también podría deberse a que en la etapa de experiencia no se requiere cumplir con tantas actividades a comparación de la etapa comercial. Se vio también que un mayor número de llamadas efectivas en la etapa comercial reduce la probabilidad de 'Churn Comercial'. En contraste, las llamadas no efectivas en la etapa de experiencia incrementan el riesgo de 'Churn Comercial'. Además, la cancelación de reuniones R1 o R2 aumenta significativamente el riesgo, mientras que una capacitación efectiva mejora las probabilidades de retención a largo plazo.

La preparación de datos a lo largo de las capas de la arquitectura Medallion, bajo la cual se trabajó en este proyecto, se logró con éxito utilizando librerías como NumPy y Pandas para la manipulación de datos, y plotly.express, matplotlib.pyplot y seaborn para la visualización y el análisis exploratorio. Sin embargo, se tuvo que prescindir de varias variables potencialmente relevantes para el análisis, como los tiempos entre eventos, debido a la falta de datos, ya que más del 50% de los datos resultaron faltantes tras el cálculo. Esto fue consecuencia de la omisión en el llenado de datos durante los procesos de captura.

Se entrenaron con éxito cinco modelos predictivos clasificatorios, destacándose entre ellos los modelos XGBoost y Decision Tree como los modelos con la mayor precisión y el mejor ajuste para la predicción de los datos proporcionados. Durante el proceso, se llevaron a cabo ajustes y optimizaciones, utilizando herramientas como la calibración de datos y la modificación del umbral de aprendizaje, con el objetivo de mejorar los resultados debido a la naturaleza binaria y desbalanceada de los datos, así como para lograr una mayor capacidad de generalización en el entrenamiento de los modelos.

Se evaluaron los modelos entrenados presentaron buenos resultados en métricas globales. XGBoost destacó como el más efectivo, con un AUC-ROC de 95% y Recall de 96%, lo que garantiza una alta capacidad de discriminación y detección de la clase minoritaria. Decision Tree y Random Forest también mostraron buen rendimiento (AUC-ROC de 90% y 89%), pero con menor Recall. En contraste, Logistic Regression tuvo un AUC-ROC de 70% y bajo Recall, limitando su capacidad predictiva. Naive Bayes, aunque competitivo en Recall (87%), obtuvo un menor F1-Score.

Se seleccionó el modelo predictivo de clasificación XGBoost debido a sus excelentes métricas de evaluación. Aunque su precisión global fue del 92%, ligeramente inferior al 93.33% de Decision Tree, su métrica AUC-ROC, alcanzó un destacado 95%. Esta métrica fue clave debido a la naturaleza desbalanceada de los datos, el modelo mostró una mejor capacidad para distinguir entre clases, manteniendo un buen equilibrio entre precisión, F1-Score y Recall, evitando el overfitting, y generalizando mejor a medida que aumenta la cantidad de datos.

6. Recomendaciones

A partir de los hallazgos obtenidos en el desarrollo de este proyecto, se han generado un conjunto de sugerencias y recomendaciones fundamentadas en datos, con el objetivo de mejorar la situación actual de la empresa y proponer la implementación y mejorar la efectividad de los modelos de aprendizaje automático en la predicción del ‘Churn Comercial’.

Para el desarrollo de este proyecto, fue necesario extraer los datos directamente del CRM de la empresa. Sin embargo, durante la exploración y análisis, se identificaron deficiencias en la introducción de datos que afectan la calidad y precisión de los modelos predictivos. Por ello, se recomienda la implementación de formatos estandarizados para el llenado de campos en el CRM. Asimismo, se detectaron actividades con significados ambiguos, lo que complica el entendimiento de los procesos y los datos. También se identificó la presencia de actividades obsoletas en el CRM que, aunque ya no se utilizan, continúan apareciendo en los registros, afectando la calidad de la información disponible. Otro hallazgo crítico fue la omisión recurrente en la introducción de datos en ambos embudos, tanto comercial como de experiencia, lo que limita la precisión de los análisis y modelos predictivos.

Dado que la calidad de los datos influye directamente en el desempeño del modelo, esta observación es de carácter prioritario. Para abordar estas deficiencias, se recomienda la capacitación del personal en el uso adecuado del CRM, garantizando un registro más preciso y así datos más confiables. Esto permitirá evitar procesos retrospectivos de limpieza manual de datos ahorrando tiempo y optimizará la eficiencia de su análisis. Una vez optimizado el llenado de datos, se recomienda incluir más variables en la predicción de ‘Churn Comercial’. La baja calidad de los datos obligó a omitir factores clave, como el tiempo entre eventos, limitando la precisión del modelo. Incorporar estas variables no solo mejoraría la predicción, sino que también aportaría insights valiosos para decisiones estratégicas.

En cuanto a los resultados del análisis, se recomienda a la empresa priorizar la obtención de llamadas efectivas en la etapa comercial, ya que se ha observado que pueden reducir las probabilidades de ‘Churn.Comercial’. En la etapa de experiencia, también es clave garantizar interacciones eficaces, pues las llamadas inefectivas aumentan el riesgo de abandono. Se recomienda revisar el análisis del tiempo mínimo de primer contacto, ya que no se ha encontrado relevancia en su relación con el ‘Churn Comercial’. Además, el porcentaje de clientes contactados en ese tiempo es bajo, lo que sugiere que la estrategia debe ser ajustada.

También se recomienda un seguimiento constante a los clientes en las etapas comercial y de experiencia. Actualmente, el embudo comercial realiza más seguimientos que el embudo de experiencia, lo que podría afectar la retención. Esto es fundamental, ya que se ha identificado una correlación significativa entre los patrones de contacto en la etapa de experiencia y el ‘Churn Comercial’. Por lo tanto, es clave reforzar las el seguimiento constante en esta etapa, especialmente para evitar la cancelación de reuniones, ya que esto incrementa considerablemente las probabilidades de ‘Churn Comercial’. Esto podría lograrse implementando procedimientos estrictos para el cumplimiento de capacitaciones, en los que se gestione

de manera efectiva la cancelación o postergación de reuniones, exigiendo un mayor compromiso por parte de los clientes.

Los resultados de este proyecto han demostrado que los modelos predictivos, como XGBoost, son herramientas clave y efectivas para identificar a los clientes en riesgo de ‘Churn Comercial’. La startup boliviana podría aprovechar estos modelos para priorizar a los clientes que muestren señales tempranas de abandono, ofreciéndoles un trato preferencial para reducir el riesgo de ‘Churn Comercial’. Además, es fundamental monitorear continuamente el desempeño de estos modelos y ajustarlos a medida que se recopilan más datos, asegurando su efectividad y mejores resultados a largo plazo. Se ha observado que el desempeño de los modelos mejora al ajustar parámetros y aplicar técnicas de optimización. Por lo tanto, se recomienda monitorear constantemente el rendimiento del modelo al implementarlo, especialmente al aumentar la cantidad de datos y variables. Aunque los resultados actuales muestran un modelo robusto y eficiente con los datos limitados, es crucial seguir evaluando su efectividad a medida que se amplía la información.

Dado que la mayoría de los clientes adquiridos son del tipo ‘B’ y se ve una preferencia por los planes anuales, se recomienda realizar un análisis más detallado de la segmentación de clientes. Esto permitirá adaptar los planes y servicios de forma personalizada, lo que incrementará la satisfacción y reducirá las probabilidades de ‘Churn Comercial’. A medida que se recopilen más datos, se podrán ajustar los planes de suscripción para satisfacer mejor las necesidades de cada segmento.

Finalmente, en cuanto al contexto social y económico del país, se recomienda prestar especial atención a factores que puedan influir en los clientes, como las crisis económicas, que han afectado negativamente a varios sectores, especialmente al sector empresarial, que constituye una parte clave de los clientes de la startup. Es crucial considerar estos factores para anticipar posibles cambios en el comportamiento de los clientes y ajustar las estrategias de retención de forma proactiva. La empresa debe desarrollar estrategias resilientes que le permitan adaptarse rápidamente a los cambios del entorno, asegurando la satisfacción de sus clientes durante períodos difíciles y fortaleciendo así la lealtad a largo plazo.

Se espera que estas recomendaciones permitan a la empresa optimizar las interacciones con los clientes, mejorar la calidad del servicio y tomar decisiones basadas en datos. Esto contribuirá a reducir el ‘Churn Comercial’, mejorar la retención de clientes y asegurar un crecimiento sostenible a largo plazo para la startup.

Referencias bibliográficas

- Aliaga, J. (2024, 12 18). *France 24*. Retrieved from Bolivia despide un 2024 con una crisis múltiple y avizora un 2025 de alta tensión: <https://www.france24.com/es/am%C3%A9rica-latina/20241218-bolivia-despide-un-2024-con-una-crisis-m%C3%BAltiple-y-avizora-un-2025-de-alta-tensi%C3%B3n>
- Bismart. (s.f.). *Técnicas y tipos de análisis predictivo: clasificación vs. regresión*. Retrieved 01 22, 2025, from <https://blog.bismart.com/tipos-analisis-predictivo-clasificacion-regresion#:~:text=Podemos%20diferenciar%20entre%20dos%20tipos,de%20regresi%C3%B3n%20modelan%20variables%20continuas>
- Breiman, L. (2001). Random forests. . *Machine Learning Journal*.
- Chambi, P. P. (2024, 11 25). *El Alto*. Retrieved from En El Alto 80% de pequeñas empresas cerraron y los que quedan no podrán pagar el aguinaldo 2024: <https://www.elaltono.com.bo/ciudad/20241125/en-el-alto-80-de-pequenas-empresas-cerraron-y-los-que-quedan-no-podran-pagar-el#:~:text=Sobrescribir%20enlaces%20de%20ayuda%20a%20la%20navegaci%C3%B3n,quedan%20no%20podr%C3%A1n%20pagar%20el%20aguinaldo%202024>.
- Chen, T. &. (2016). XGBoost: A scalable tree boosting system. . *KDD Conference on Knowledge Discovery and Data Mining*.
- Cox, D. R. (1958). The regression analysis of binary sequences. . *Journal of the Royal Statistical Society: Series B*.
- Cristianini, N. &.T. (2000). *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press. doi:<https://doi.org/10.1017/CBO9780511801389>
- Databricks. (2024). *Medallion Architecture*. Retrieved from <https://www.databricks.com/glossary/medallionarchitecture>
- eju. (2025, 01 11). *Fedemype Santa Cruz reporta el cierre de al menos 7 mil unidades productivas en 2024 por la crisis económica*. Retrieved from <https://eju.tv/2025/01/fedemype-santa-cruz-reporta-el-cierre-de-al-menos-7-mil-unidades-productivas-en-2024-por-la-crisis-economica/#:~:text=econ%C3%B3mica%20%E2%80%93%20eju.tv-,Fedemype%20Santa%20Cruz%20reporta%20el%20cierre%20de%20al%20menos%207,2024%20>
- El Naqa, I. &. (2015). *Machine Learning in Radiation Oncology: Theory and Applications*. (R. L. I. El Naqa, Ed.) Springer International Publishing. doi:https://doi.org/10.1007/978-3-319-18305-3_1
- Espinosa-Zúñiga, J. J. (2020). Aplicación de metodología CRISP-DM para segmentación geográfica de una base de datos pública. *Ingeniería, investigación y tecnología*, 21(1).

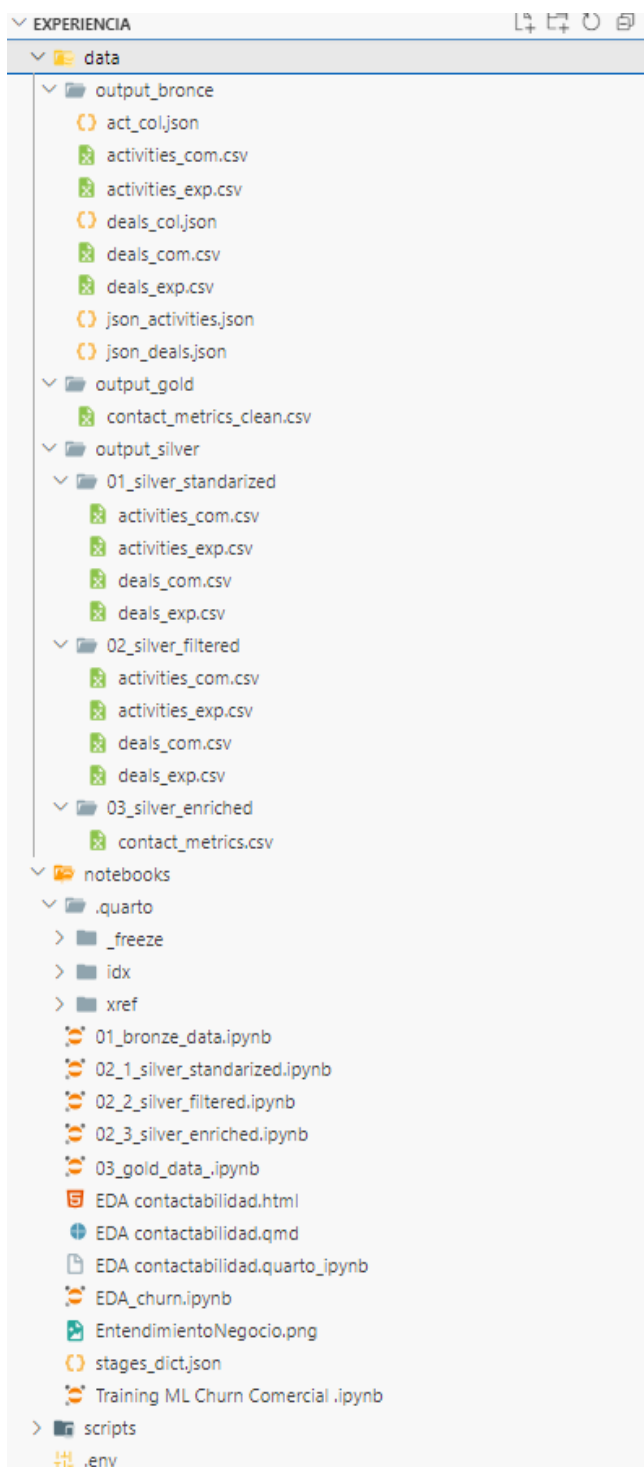
- FEPC. (2024, 12). *Reporte Empresarial*. Retrieved from chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://fepc.bo/archivos/DIRCOM/Gestio n%202024/Reporte%20Empresarial/2024/Diciembre%202024/REPORTE%20EMPRESAR IAL%202024%20.pdf
- Fernández, A. G. (2018). SMOTE for learning from imbalanced data: Progress and challenges, marking the 15-year anniversary. *Journal of Artificial Intelligence Research*, 863-905.
- García-Herrero, J. B. (2018). *Ciencia de datos: Técnicas analíticas y aprendizaje estadístico. Un enfoque práctico*. Altaria Publicaciones Alfaomega.
- geekforgeeks. (s.f.). *One Hot Encoding in Machine Learning*. Retrieved from GeeksforGeeks: <https://www.geeksforgeeks.org/ml-one-hot-encoding/>
- González Duque, R. (2011). *Python para todos*. Retrieved from <http://mundogeek.net/tutorial-python/>
- Greenacre, M. G. (2022). Principal component analysis. *Nature Reviews Methods Primers*, 2(1), 100.
- Gupta, M., Gupta, D., & Rai, P. (2024). Exploring the Impact of Software as a Service (SaaS) on Human Life. *EAI Endorsed Transactions on Internet of Things*, 10.
- HLEG, A. (2019). High-level expert group on artificial intelligence. Ethics guidelines for trustworthy AI. 6.
- IBM. (2024, 05 14). *Análisis predictivos*. Retrieved from https://www.ibm.com/es-es/topics/predictive-analytics?utm_source=chatgpt.com
- INE. (2020). *ASPECTOS GEOGRAFICOS*. Retrieved from <https://www.ine.gob.bo/index.php/bolivia/aspectos-geograficos/#:~:text=Limita%20al%20norte%20y%20este,y%20al%20sudoeste%20con%20Chile.&text=El%20sistema%20nacional%20de%20carreteras,integran%20a%20todos%20los%20departamentos>.
- Keyrus. (s.f.). *Las 11 técnicas más utilizadas en el modelado de análisis predictivo*. Retrieved 01 22, 2025, from <https://keyrus.com/sp/es/insights/las-11-tecnicas-mas-utilizadas-en-el-modelado-de-analisis-predictivos/>
- Laplante, P. A., Zhang, J., & Voas, J. (2008). What's in a Name? Distinguishing between SaaS and SOA. *IT Professional*, 10(3), 50.
- Larson, M. G. (2008). Analysis of variance. *Circulation*, 117(1), 115-121.
- León del Apio, J. (2017). *Online Marketing Analytics: Entender a nuestros clientes analizando su ciclo de vida*. Retrieved from Analítica Web: <https://www.analiticaweb.es/entender-clientesanalizando-ciclo-vida/>
- Mahesh, B. (2020). Machine learning algorithms - a review. *International Journal of Science and Research (IJSR)*, 9(1), 381-386.

- McKinney, W. (2022). *Python for Data Analysis*. O'Reilly Media, Inc.
- Milo, T. &. (2020). *Automating exploratory data analysis via machine learning: An overview, Proceedings of the 2020 ACM SIGMOD international conference on management of data*.
- Mukhiya, S. K. (2020). *Hands-On Exploratory Data Analysis with Python: Perform EDA techniques to understand, summarize, and investigate your data*. Packt Publishing Ltd.
- Negnevitsky, M. (2005). Artificial Intelligence: A Guide to Intelligent Systems. In M. Negnevitsky, *Artificial Intelligence: A Guide to Intelligent Systems* (p. 415). England: PearsonEducationLimited.
- One.com. (s.f.). *One.com*. Retrieved 01 20, 2025, from ¿Qué es el churn y cómo calcularlo?: <https://www.one.com/es/tiendaonline/que-es-churn>
- Ph.D., Y. L.-R. (2019). *STARTUP Y SUS METODOLOGÍAS PARA NO FRACASAR*. Bogotá, Colombia.
- Pineda Pertuz, C. M. (2022). *Aprendizaje automático y profundo en Python*. Ra-Ma S.A. Editorial y Publicaciones.
- Posada Hernández, G. J. (2016). *Elementos básicos de estadística descriptiva para el análisis de datos*. Fondo Editorial Luis Amigó.
- PredikData. (s.f.). *¿Qué son y para qué se usan los modelos predictivos?* Retrieved 01 22, 2025, from <https://predikdata.com/es/que-son-y-para-que-se-usan-los-modelos-predictivos/>
- Quinlan, J. R. (1986). Induction of decision trees. . *Machine Learning*.
- Rababah, K. M. (2011). Customer relationship management (CRM) processes from theory to practice: The pre-implementation plan of CRM system. *International Journal of e-Education, e-Business, e-Management and e-Learning*, 1(1), 22-27.
- Redondo, M. (2025). *mamel*. Retrieved from Modelos Predictivos y la Curva ROC: <https://mamel.es/modelos-predictivos-y-la-curva-roc/>
- REPSOL. (2023, Septiembre 11). *Innovacion, tecnologia y talento*. Retrieved from ¿Que es una startup?: <https://www.repsol.com/es/energia-futuro/personas/que-es-una-startup/index.cshtml#:~:text=%C2%BFQu%C3%A9%20son%20las%20startup%3F,de%20manera%20%C3%A1gil%20y%20r%C3%A1pida>.
- Rigatti, S. J. (2017). Random forest. *Journal of Insurance Medicine*, 47(1), 31-39.
- Sawyer, S. F. (2009). Analysis of variance: The fundamental concepts. *Journal of Manual & Manipulative Therapy*, 17(2), 27E-38E.
- Schröer, C. K. (2021). A systematic literature review on applying CRISP-DM process model. *Procedia Computer Science*, 526-534.
- Shinde, P. P. (2018). *A review of machine learning and deep learning applications, 2018 Fourth international conference on computing communication control and automation (ICCUBEA)*. IEEE.

- Torres, J. S. (2018). Del ciclo de vida del producto al ciclo de vida del cliente. Una aproximación hacia una construcción teórica del ciclo de vida del cliente. *Investigación & Negocios*, 11(18), 110.
- Urrelo, J. O. (2024). *MODELO DE ANÁLISIS PREDICTIVO PARA EL ABANDONO DE CLIENTES EN UNA EMPRESA DE ALOJAMIENTO WEB CON HERRAMIENTAS DE MACHINE LEARNING*. Cochabamba: Dirección de Posgrado de la Facultad de Ciencias y Tecnología.
- VanderPlas, J. (2017). Python Data Science Handbook. In J. VanderPlas, *Python Data Science Handbook* (p. 517). United States of America: Kristen Brown.
- Wiselka, M. (2024). *Development of Modern Data Platform using Medallion Architecture*. Retrieved from <https://urn.fi/URN:NBN:fi:amk-2024111828633>
- xgboost. (2022). *DMLC XGBoost*. Retrieved from XGBoost Documentation: <https://xgboost.readthedocs.io/en/stable/>
- Yu, C. H. (2010). Exploratory data analysis in the context of data mining and resampling. *International Journal of Psychological Research*, 3(1), 9-22.
- Zhang, H. (2004). The Optimality of Naive Bayes. *Proceedings of the Seventeenth International Florida Artificial Intelligence Research Society Conference*. California.

Anexos

Anexo 1. Estructura de trabajo: Arquitectura MEDALLION



Anexo 2. Procesamiento de datos capa Bronze

The screenshot shows a Jupyter Notebook titled '01_bronze_data.ipynb'. The left sidebar displays a file explorer with the following structure:

- EXPERIENCIA
 - .pycache
 - .Rproj.user
 - data
 - output_bronze
 - act.col.json
 - activities.com.csv
 - activities.exp.csv
 - deals.col.json
 - deals.com.csv
 - deals.exp.csv
 - json.activities.json
 - json.deals.json
 - output_gold
 - contact_metrics_clean.csv
 - output_silver
 - 01_silver_standardized
 - activities.com.csv
 - activities.exp.csv
 - deals.com.csv
 - deals.exp.csv
 - 02_silver_filtered
 - activities.com.csv
 - activities.exp.csv
 - deals.com.csv
 - deals.exp.csv
 - 03_silver_enriched
 - contact_metrics.com.csv
 - contact_metrics.exp.csv
 - contact_metrics.csv
 - notebooks
 - quarto
 - freeze
 - idx
 - xref
 - 01_bronze_data.ipynb
 - 02_1_silver_standardized.ipynb
 - 02_2_silver_filtered.ipynb
 - 02_3_silver_enriched.ipynb
 - 03_gold_data.ipynb
 - EDA_contactabilidad.quar...
 - EDA_churn.ipynb
 - EntendimientoNegocio.p...
 - Graficas.ipynb
 - stages.dict.json
 - Training ML Churn Comer...
 - scripts
 - .env
 - service_account.json

The main content area shows the following sections and code:

Credentials

```
load_dotenv()

# credentials
api_token_exp = os.getenv("api_token_exp")
api_token_com = os.getenv("api_token_com")
company_domain = os.getenv("company_domain")
```

GET DATA

GET Metadata

```
# getting metadata - Deals
json_deals_metadata = deals_metadata(
    api_token_exp,
    company_domain
)
df_deals_metadata = pd.json_normalize(json_deals_metadata)
```

```
# getting metadata - Activities
json_act_metadata = activities_metadata(
    api_token_exp,
    company_domain
)
df_act_metadata = pd.json_normalize(json_act_metadata)
```

GET Data from Filters

Activities

```
# getting activities from filter 13666
json_activities_com = get_activities(
    user_id = 0,
    filter_id=13666,
    done=True,
    api_token=api_token_com,
    company_domain= company_domain
)
```

```
# getting activities from filter
json_activities_exp = get_activities(
    user_id = 0,
    filter_id=13671,
    done=True,
    api_token=api_token_exp,
    company_domain= company_domain
)
```

Deals

```
# getting deals from filter All 'Mensual' Deals without
json_deals_com = get_deals(
    filter_id=14375,
    api_token=api_token_com,
    company_domain= company_domain
)
```

https://github.com/datafla/Modelo-Predictivo-Churn-Comercial/blob/main/notebooks/01_bronze_data.ipynb

Anexo 3. Procesamiento de datos capa Plata (subcapa de estandarización)

The screenshot shows a Jupyter Notebook titled '02_1_silver_standardized.ipynb'. The left sidebar displays a file explorer with the following structure:

- EXPERIENCIA
 - pycache
 - .Rproj.user
 - data
 - output_bronce
 - act_col.json
 - activities_com.csv
 - activities_exp.csv
 - deals_col.json
 - deals_com.csv
 - deals_exp.csv
 - json_activities.json
 - json_deals.json
 - output_gold
 - contact_metrics_clean.csv
 - output_silver
 - 01_silver_standardized
 - activities_com.csv
 - activities_exp.csv
 - deals_com.csv
 - deals_exp.csv
 - 02_silver_filtered
 - activities_com.csv
 - activities_exp.csv
 - deals_com.csv
 - deals_exp.csv
 - 03_silver_enriched
 - contact_metrics_com.csv
 - contact_metrics_exp.csv
 - contact_metrics.csv
 - notebooks
 - .quarto
 - freeze
 - idx
 - xref
 - 01_bronce_data.ipynb
 - 02_1_silver_standardized.ipynb (selected)
 - 02_2_silver_filtered.ipynb
 - 02_3_silver_enriched.ipynb
 - 03_gold_data.ipynb
 - EDA_contactabilidad_quar...
 - EDA_churn.ipynb
 - EntendimientoNegocio.p...
 - Graficas.ipynb
 - stages_dict.json
 - Training ML Churn Comer...
 - scripts
 - .env
 - service_account.json

The main area of the notebook shows the following code sections:

Import Bronce Data

```
[3] # Import Bronce Data from csv
# Activities
df_activities_com = pd.read_csv(r'..\data\output_bronce\activities_com.csv', dtype=str)
df_activities_exp = pd.read_csv(r'..\data\output_bronce\activities_exp.csv', dtype=str)

# Deals
df_deals_com = pd.read_csv(r'..\data\output_bronce\deals_com.csv', dtype=str)
df_deals_exp = pd.read_csv(r'..\data\output_bronce\deals_exp.csv', dtype=str)
```

MAP DATA

Restructuring JSON

```
[4] # Import Bronce Data from JSON
with open(r'..\data\output_bronce\json_deals.json', "r") as json_file:
    json_deals_metadata = json.load(json_file)

with open(r'..\data\output_bronce\json_activities.json', "r") as json_file:
    json_act_metadata = json.load(json_file)

with open(r'..\data\output_bronce\act_col.json', "r") as json_file:
    act_col = json.load(json_file)

with open(r'..\data\output_bronce\deals_col.json', "r") as json_file:
    deals_col = json.load(json_file)

with open('stages_dict.json', 'r') as f:
    stages_dict = json.load(f)
```

Mapping Columns

```
[5] # Restructuring metadata - Deals
restructured_deals = restructure_metadata(json_deals_metadata)

[6] # Restructuring metadata - Activities
restructured_activities = restructure_metadata(json_act_metadata)

[7] # mapping columns - Activities
act_map = {field['key']: field['name'] for field in act_col}

[8] # mapping activities
df_activities_com.rename(columns=act_map, inplace=True)
df_activities_exp.rename(columns=act_map, inplace=True)

[9] # mapping columns - Deals
deals_map = {field['key']: field['name'] for field in deals_col}

[10] # mapping activities
df_deals_com.rename(columns=deals_map, inplace=True)
df_deals_exp.rename(columns=deals_map, inplace=True)
```

https://github.com/datafla/Modelo-Predictivo_Churn-Comercial/blob/main/notebooks/02_1_silver_standardized.ipynb

Anexo 4. Procesamiento de datos capa Plata (subcapa de filtrado)

The screenshot displays a Jupyter Notebook titled '02_2_silver_filtered.ipynb'. The left sidebar shows a file explorer with a project structure including 'data', 'notebooks', and 'scripts'. The main area contains three code blocks:

Import Standardized Data

```
# Export Standardized Data to CSV

# Activities
df_activities_com = pd.read_csv(r'..\data\output_silver\01_silver_standardized\activities_com.csv', low_memory=False)
df_activities_exp = pd.read_csv(r'..\data\output_silver\01_silver_standardized\activities_exp.csv', low_memory=False)

# Deals
df_deals_com = pd.read_csv(r'..\data\output_silver\01_silver_standardized\deals_com.csv', low_memory=False)
df_deals_exp = pd.read_csv(r'..\data\output_silver\01_silver_standardized\deals_exp.csv', low_memory=False)
```

[3]

DATA SELECTION

Columns Selection - Deals

```
# filtering deals columns
columns = [
    'org_id.value', # to merge with pipeline
    'Negocio creado el',
    'Fecha de ganado',
    'Fecha de cierre prevista',
    'Origen',
    'Tipo de cliente',
    'Tiempo de contacto (min)',
]

df_deals_com = df_deals_com[columns]
```

[4]

```
# filtering deals columns
columns = [
    'org_id.value',
    'Negocio creado el',
    'Fecha de ganado',
    'Fecha de perdido',
    'Origen',
    'Tipo de cliente',
    'Canal Agrupado',
    'Tiempo de contacto (min)',
    '(C) (EXP) Plazo y Pago',
    '(EXP) Fecha Kickoff',
    '(EXP) Fecha de finalización de onboarding',
]

df_deals_exp = df_deals_exp[columns]
```

[5]

Columns Selection - Activities

```
# filtering activities columns
columns = [
    'Negocio',
    'Organización',
    'org_name',
    'Tipo',
    'Fecha de vencimiento',
    'Hora de vencimiento',
    'Hora en que se marcó como completada'
]

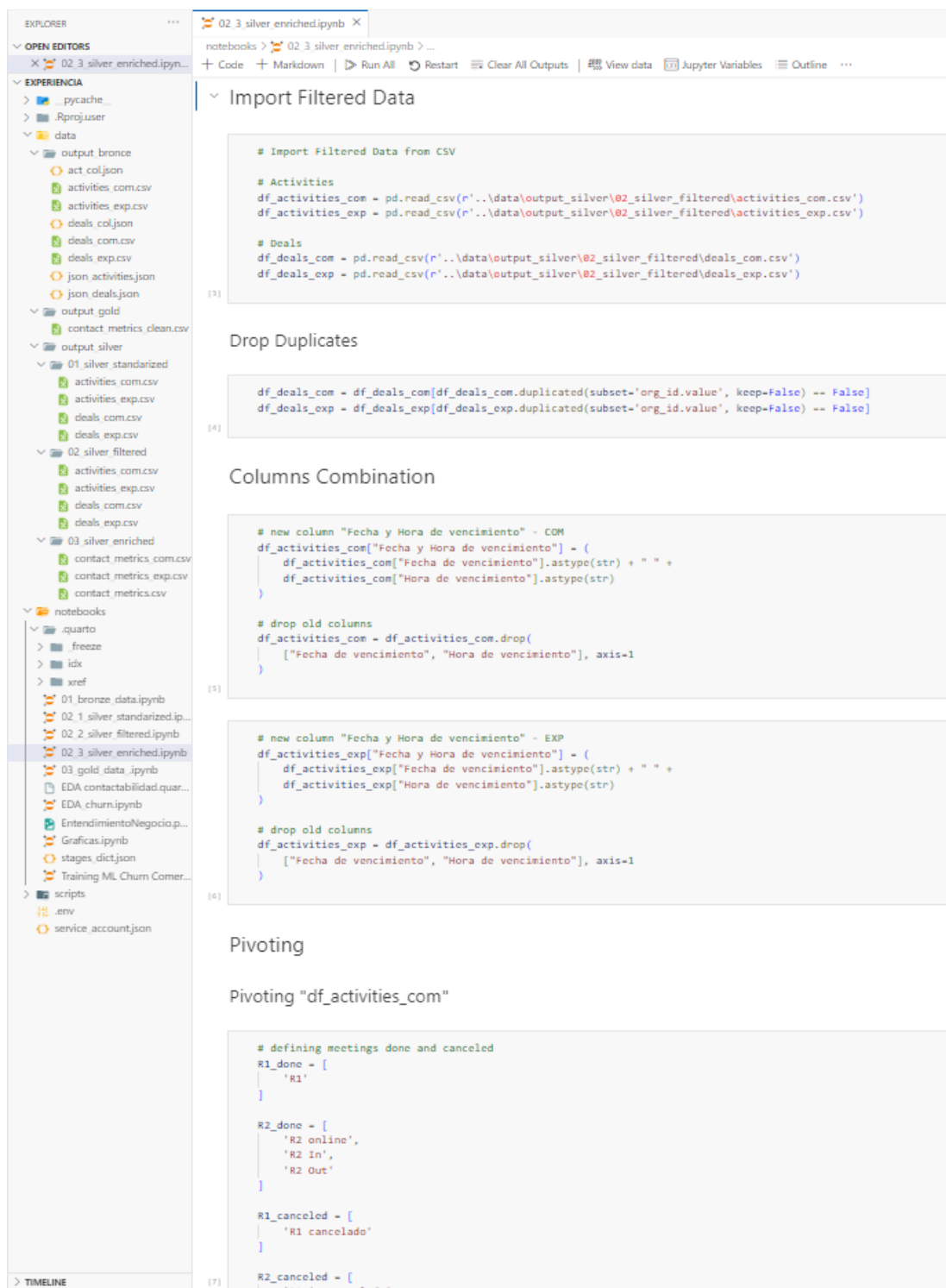
df_activities_com = df_activities_com[columns]
df_activities_exp = df_activities_exp[columns]
df_activities_churn = df_activities_churn[columns]

# review the selected columns
```

[6]

https://github.com/datafla/Modelo-Predictivo-Churn-Comercial/blob/main/notebooks/02_2_silver_filtered.ipynb

Anexo 5. Procesamiento de datos capa Plata (subcapa de enriquecimiento)



EXPLORER

notebooks > 02_3_silver_enriched.ipynb > ...

+ Code + Markdown | ▶ Run All | ⌂ Restart | 🗑 Clear All Outputs | 📊 View data | 📄 Jupyter Variables | 📖 Outline ...

Import Filtered Data

```
# Import Filtered Data from CSV

# Activities
df_activities_com = pd.read_csv(r'..\data\output_silver\02_silver_filtered\activities_com.csv')
df_activities_exp = pd.read_csv(r'..\data\output_silver\02_silver_filtered\activities_exp.csv')

# Deals
df_deals_com = pd.read_csv(r'..\data\output_silver\02_silver_filtered\deals_com.csv')
df_deals_exp = pd.read_csv(r'..\data\output_silver\02_silver_filtered\deals_exp.csv')
```

Drop Duplicates

```
df_deals_com = df_deals_com[df_deals_com.duplicated(subset='org_id.value', keep=False) == False]
df_deals_exp = df_deals_exp[df_deals_exp.duplicated(subset='org_id.value', keep=False) == False]
```

Columns Combination

```
# new column "Fecha y Hora de vencimiento" - COM
df_activities_com["Fecha y Hora de vencimiento"] = (
    df_activities_com["Fecha de vencimiento"].astype(str) + " " +
    df_activities_com["Hora de vencimiento"].astype(str)
)

# drop old columns
df_activities_com = df_activities_com.drop(
    ["Fecha de vencimiento", "Hora de vencimiento"], axis=1
)

# new column "Fecha y Hora de vencimiento" - EXP
df_activities_exp["Fecha y Hora de vencimiento"] = (
    df_activities_exp["Fecha de vencimiento"].astype(str) + " " +
    df_activities_exp["Hora de vencimiento"].astype(str)
)

# drop old columns
df_activities_exp = df_activities_exp.drop(
    ["Fecha de vencimiento", "Hora de vencimiento"], axis=1
)
```

Pivoting

Pivoting "df_activities_com"

```
# defining meetings done and canceled
R1_done = [
    'R1'
]

R2_done = [
    'R2 online',
    'R2 in',
    'R2 Out'
]

R1_canceled = [
    'R1 cancelado'
]

R2_canceled = [
    'R2 cancelado'
]
```

https://github.com/datafla/Modelo-Predictivo-Churn-Comercial/blob/main/notebooks/02_3_silver_enriched.ipynb

Anexo 6. Procesamiento de datos capa Oro

EXPLORER

OPEN EDITORS

03 gold_data.ipynb

EXPERIENCIA

data

output bronze

act.col.json

activities.com.csv

activities.exp.csv

deals.col.json

deals.com.csv

deals.exp.csv

json.activities.json

json.deals.json

output gold

contact_metrics_clean.csv

output silver

01 silver standardized

activities.com.csv

activities.exp.csv

deals.com.csv

deals.exp.csv

02 silver filtered

activities.com.csv

activities.exp.csv

deals.com.csv

deals.exp.csv

03 silver enriched

contact_metrics.com.csv

contact_metrics.exp.csv

contact_metrics.csv

notebooks

quarto

freeze

idx

xref

01 bronze_data.ipynb

02 1 silver standardized.ip...

02 2 silver_filtered.ipynb

02 3 silver_enriched.ipynb

03 gold_data.ipynb

EDA contactabilidad.quar...

EDA churn.ipynb

EntendimientoNegocio.p...

Graficas.ipynb

stages.dict.json

Training ML Churn Comer...

scripts

env

service.account.json

03 gold_data.ipynb

notebooks > 03 gold_data.ipynb > ...

+ Code + Markdown + Run All + Restart + Clear All Outputs + View data + Jupyter Variables + Outline + ...

Import Filtered Data

```
# import CSV to DataFrame
df = pd.read_csv(r'..\data\output_silver\silver_enriched\contact_metrics.csv')
```

```
df['Tipo de cliente'] = df['Tipo de cliente'].str.strip()
```

outliers treatment

Debido a que los campos seleccionados se tratan de conteos de actividades y variables categoricas nuestro dataset no cuenta con valores nulos

```
variable="Total_Actividades_com"
fig = px.box(
    df,
    y=variable,
    points="all",
    title=f'Box Plot de {variable}',
    template="plotly_white"
)
fig.update_layout(width=400, height=500)
fig.show()
```

Box Plot de Total_Actividades_com

```
fig = px.histogram(
    df,
    x=variable,
    title=f'Histograma de {variable}',
    template="plotly_white",
    nbins=30
)
fig.update_layout(width=400, height=500)
fig.show()
```

Histograma de Total_Actividades_com

https://github.com/datafla/Modelo-Predictivo-Churn-Comercial/blob/main/notebooks/03_gold_data.ipynb

Anexo 7. Análisis exploratorio de datos

EXPLORER

OPEN EDITORS

EDA_churn.ipynb

EXPERIENCIA

data

output_bronze

act_col.json

activities_com.csv

activities_exp.csv

dsals_col.json

dsals_com.csv

dsals_exp.csv

json_activities.json

json_dsals.json

output_gold

contact_metrics_clean.csv

output_silver

01_silver_standardized

activities_com.csv

activities_exp.csv

dsals_com.csv

dsals_exp.csv

02_silver_filtered

activities_com.csv

activities_exp.csv

dsals_com.csv

dsals_exp.csv

03_silver_enriched

contact_metrics_com.csv

contact_metrics_exp.csv

contact_metrics.csv

notebooks

quarto

_fraseo

ids

seef

01_bronze_data.ipynb

02_1_silver_standardized.ip...

02_2_silver_filtered.ipynb

02_3_silver_enriched.ipynb

03_gold_data.ipynb

EDA contactabilidad quar...

EDA_churn.ipynb

EntendimientoNegociop...

Gráficas.ipynb

stages_dict.json

Training ML Churn Comer...

scripts

amr

service_account.json

EDA_churn.ipynb

notebooks > EDA_churn.ipynb > ...

+ Code + Markdown | ▶ Run All | Clear All Outputs | Outline ...

Import Data

```
# Import CSV to DataFrame
df = pd.read_csv(r'..\data\output_silver\contact_metrics_clean.csv')
```

```
nus_columns_com = [ ... ]
```

Análisis Univariado

```
df.describe(include='object').T
```

	count	unique	top	freq
Tipo de cliente	373	3	B	235
Tipo Primer Contacto	373	4	Efectiva	222
Rango de Contacto	373	3	Fuera del rango	224
Tipo Primera Capacitación	373	2	Hecha	202
Onboarding	373	2	No Finalizado	276
(C) (EXP) Plazo y Pago	373	3	Annual	285

```
# Agrupamos datos categóricos "Tipo de cliente"
variable = 'Tipo de cliente'
df_cat = df.groupby(variable)['org_id.value'].agg(Cconteo='count').reset_index()
```

```
fig = px.bar(
    df_cat,
    x=variable,
    y='Cconteo',
    color=variable,
    template='plotly_white'
)

fig.update_layout(
    title='<b>Tipo de Cliente</b>',
    width=400,
    height=500,
    xaxis_title='',
    yaxis_title='Cconteo',
    title_x = 0.5,
    showlegend=False
)

fig.show()
```

Tipo de Cliente

```
df_cat
```

	Tipo de cliente	Cconteo
0	A	84
1	B	235
2	C	54

https://github.com/datafla/Modelo-Predictivo_Churn-Comercial/blob/main/notebooks/EDA_churn.ipynb

Anexo 8. Entrenamiento de Modelos

EXPLORER

Training ML Churn Comercial .ipynb

notebooks > Training ML Churn Comercial .ipynb > ...

+ Code + Markdown | ▶ Run All | ⌂ Restart | 🗑 Clear All Outputs | 📊 View data | 📄 Jupyter Variables | 📖 Outline

OPEN EDITORS

Training ML Churn Comercial...

EXPERIENCIA

__pycache__

__projuser

data

output_bronze

act_col.json

activities_com.csv

activities_exp.csv

deals_col.json

deals_com.csv

deals_exp.csv

json_activities.json

json_deals.json

output_gold

contact_metrics_clean.csv

output_silver

01_silver_standardized

activities_com.csv

activities_exp.csv

deals_com.csv

deals_exp.csv

02_silver_filtered

activities_com.csv

activities_exp.csv

deals_com.csv

deals_exp.csv

03_silver_enriched

contact_metrics_com.csv

contact_metrics_exp.csv

contact_metrics.csv

notebooks

quarto

_freeze

idx

srcif

01_bronze_data.ipynb

02_1_silver_standardized.ip...

02_2_silver_filtered.ipynb

02_3_silver_enriched.ipynb

03_gold_data.ipynb

EDA_contactabilidad.quar...

EDA_churn.ipynb

EntendimientoNegocio.ip...

Graticas.ipynb

stages_dict.json

Training ML Churn Comer...

scripts

env

service_account.json

Sección 1: Importacion de Librerias

```

import pandas as pd
import numpy as np
from scipy.stats import chi2_contingency
from scipy.stats import f_oneway
from imblearn.over_sampling import SMOTE
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix, recall_score, f1_score
import matplotlib.pyplot as plt
import seaborn as sns

```

Sección 2: Carga de Datos

```

# Import CSV to DataFrame
df = pd.read_csv(r'..\data\output_gold\contact_metrics_clean.csv')

```

Sección 3: Selección de Variables

```

num_columns_com = [ ...

subset_cols = num_columns_com + num_columns_exp + label ...

```

ANÁLISIS DE CORRELACION LINEAL

```

corr_matrix = df_subset.corr()

plt.figure(figsize=(10, 6))
sns.heatmap(corr_matrix, annot=True, cmap='coolwarm', fmt=".2F", linewidths=0.5)
plt.title("Matriz de Correlación")
plt.show()

```

Matriz de Correlación

	Total_Actividades_com	Total_Llamadas_com	Llamadas_Efectivas_com	Llamadas_No_Efectivas_com	WA_Seguiemiento_com	Reuniones_Hechas	Reuniones_Canceladas	Total_Actividades_exp	Total_Llamadas_exp	Llamadas_Efectivas_exp	Llamadas_No_Efectivas_exp	WA_Seguiemiento_exp	Kickoff_Hechas	Kickoff_Canceladas	Capacitaciones_Hechas	Capacitaciones_Canceladas	Churn Comercial
Total_Actividades_com	1.00	0.94	0.83	0.83	0.89	0.32	0.41	0.01	0.04	0.36	0.33	0.03	0.03	0.05	-0.03	0.08	0.12
Total_Llamadas_com	0.94	1.00	0.90	0.85	0.69	0.25	0.36	-0.02	-0.05	-0.07	-0.04	-0.07	0.11	-0.03	0.02	0.11	-0.05
Llamadas_Efectivas_com	0.83	0.90	1.00	0.56	0.58	0.28	0.32	0.11	0.11	0.35	0.13	0.16	0.11	0.01	0.03	0.13	-0.07
Llamadas_No_Efectivas_com	0.83	0.85	0.56	1.00	0.66	0.15	0.33	0.05	0.01	-0.05	0.05	0.03	0.09	-0.05	0.01	0.04	-0.32
WA_Seguiemiento_com	0.89	0.69	0.58	0.66	1.00	0.23	0.35	0.05	-0.03	-0.02	-0.02	0.01	-0.00	-0.03	0.12	0.12	-0.06
Reuniones_Hechas	0.32	0.25	0.28	0.15	0.23	1.00	0.14	0.12	0.07	0.05	0.05	0.12	0.06	0.02	0.09	0.06	0.04
Reuniones_Canceladas	0.41	0.36	0.32	0.33	0.35	0.14	1.00	-0.03	-0.03	-0.02	-0.02	-0.04	0.06	0.05	-0.01	-0.00	0.03
Total_Actividades_exp	0.01	-0.03	0.11	0.05	0.05	0.12	-0.01	1.00	0.77	0.61	0.66	0.82	-0.19	-0.04	0.33	0.22	0.05
Total_Llamadas_exp	-0.04	-0.05	0.11	0.01	0.03	0.07	-0.03	0.77	1.00	0.74	0.89	0.55	-0.28	-0.09	-0.00	0.08	0.21
Llamadas_Efectivas_exp	-0.04	-0.07	0.09	0.05	0.03	0.05	0.02	0.61	0.74	1.00	0.38	0.48	-0.33	-0.09	0.05	0.09	0.11
Llamadas_No_Efectivas_exp	-0.03	-0.04	0.11	0.05	-0.02	0.05	-0.02	0.66	0.89	0.38	1.00	0.45	-0.10	-0.05	-0.04	0.06	0.22
WA_Seguiemiento_exp	0.03	0.07	0.16	0.03	0.01	0.12	0.04	0.82	0.55	0.48	0.45	1.00	0.31	0.11	0.05	0.07	0.07
Kickoff_Hechas	0.05	0.11	0.11	0.09	-0.00	0.00	-0.19	-0.29	-0.12	-0.18	-0.31	0.31	1.00	0.03	-0.02	0.00	-0.30
Kickoff_Canceladas	-0.03	-0.03	0.01	0.05	0.03	0.07	0.05	0.04	0.09	0.06	0.05	0.11	0.03	1.00	0.02	0.15	0.00
Capacitaciones_Hechas	0.08	0.02	0.03	0.01	0.12	0.09	-0.01	0.33	-0.00	0.05	-0.34	0.05	-0.02	0.02	1.00	0.23	-0.15
Capacitaciones_Canceladas	-0.12	0.11	0.13	0.04	0.12	0.06	-0.00	0.22	0.08	0.09	0.06	-0.07	0.06	0.15	0.23	1.00	0.10
Churn Comercial	-0.05	-0.05	-0.07	-0.32	-0.06	0.04	0.03	0.05	0.21	0.11	0.22	-0.02	-0.08	-0.00	-0.15	0.10	1.00

<https://github.com/datafla/Modelo-Predictivo-Churn-Comercial/blob/main/notebooks/Training%20ML%20Churn%20Comercial%20.ipynb>

98

Anexo 9. Data Frame resultante de la extracción de datos

df

✓ 0.0s Open 'df' in Data Wrangler

	org id.value	Tipo de cliente	Total Actividades com	Total Llamadas com	Llamadas Efectivas com	Llamadas No Efectivas com	WA Seguimiento com	Reuniones Hechas	Reuniones Canceladas	Tipo Primer Contacto	...
0	515.0	B	30	16	7	9	13	1	0	No Efectiva	...
1	2249.0	B	12	8	4	4	3	1	0	Efectiva	...
2	3120.0	B	9	6	5	1	2	1	0	Efectiva	...
3	5181.0	B	10	3	2	1	6	1	0	No Efectiva	...
4	8446.0	A	8	3	3	0	4	1	0	Efectiva	...
...
368	243947.0	B	4	1	1	0	1	2	0	Efectiva	...
369	244105.0	B	9	0	0	0	9	0	0	No tuvo	...
370	244173.0	A	8	2	2	0	4	2	0	Efectiva	...
371	244384.0	A	16	5	4	1	9	2	0	Efectiva	...
372	245287.0	B	5	0	0	0	4	1	0	No tuvo	...

373 rows × 26 columns

◀

Anexo 10.CD

Revise el CD adjunto, haga clic en el siguiente enlace para acceder a toda la documentación del proyecto o escanee el código QR a continuación.

https://github.com/datafla/Modelo-Predictivo_Churn-Comercial