UNIVERSIDAD CATÓLICA SANTO TORIBIO DE MOGROVEJO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN

Implementación de un modelo de minería de datos para predecir la deserción de los clientes en una empresa de telecomunicaciones

TESIS PARA OPTAR EL TÍTULO DE
INGENIERO DE SISTEMAS Y COMPUTACIÓN

AUTOR

MIRKO BRUNO VELA LOPEZ

ASESOR

MARIA YSABEL ARANGURI GARCIA
https://orcid.org/0000-0001-9220-5801

Chiclayo, 2022

Implementación de un modelo de minería de datos para predecir la deserción de los clientes en una empresa de telecomunicaciones

PRESENTADA POR:

MIRKO BRUNO VELA LOPEZ

A la Facultad de Ingeniería de la
Universidad Católica Santo Toribio de Mogrovejo
para optar el título de

INGENIERO DE SISTEMAS Y COMPUTACIÓN

APROBADA POR:

Roger Ernesto Alarcon Garcia
PRESIDENTE

Jessie Leila Bravo Jaico
SECRETARIO

Maria Ysabel Aranguri Garcia
VOCAL

DEDICATORIA

A mis padres y hermanos que siempre apoyan mis decisiones, confían en mi buen desempeño y sobre todo que siempre están en los momentos difíciles de mi vida.

AGRADECIMIENTOS

Agradecer a Dios por guiar mis pasos a lo largo de esta investigación. A mi asesora de tesis la Ing. María Ysabel Arangurí García, por su tutoría a lo largo de este proceso, orientando mis ideas, compartiendo sus conocimientos y sobre todo por su gran labor como docente. A la empresa que me permitió emplear su información y en general, a todo aquel que apoyó a la realización de la presente investigación.

INFORME FINAL DE TESIS

INFORME DE ORIGINALIDAD

13%	13%	2%	%
INDICE DE SIMILITUD	FUENTES DE INTERNET	PUBLICACIONES	TRABAJOS DEL ESTUDIANTE

FUENTES PRIMARIAS

1	hdl.handle.net Fuente de Internet	3%
2	tesis.usat.edu.pe Fuente de Internet	2%
3	www.revistas.espol.edu.ec Fuente de Internet	1%
4	Repositorio.Unap.Edu.Pe Fuente de Internet	<1%
5	www.iartificial.net Fuente de Internet	<1%
6	www.redalyc.org Fuente de Internet	<1%
7	1library.co Fuente de Internet	<1%
8	repositorio.autonoma.edu.co Fuente de Internet	<1%
9	dspace.ucuenca.edu.ec Fuente de Internet	<1%

ÍNDICE

RESUMEN 8

ABSTRACT 9

I. INTRODUCCIÓN 10

II. REVISIÓN DE LA LITERATURA 12

2.1. ANTECEDENTES 12

2.1.1. ANTECEDENTES INTERNACIONALES 12

2.1.2. ANTECEDENTES NACIONALES 15

2.1.3. ANTECEDENTES LOCALES 17

2.2. BASES TEÓRICO CIENTÍFICAS 17

2.2.1. DATO E INFORMACIÓN 17

2.2.1.1. Dato 17

2.2.1.2. Información 17

2.2.1.3. Conocimiento 17

2.2.2. MINERÍA DE DATOS 18

2.2.2.1. Definición 18

2.2.2.2. Tipos de Datos 18

2.2.2.3. Grupos de Métodos 19

2.2.3. MODELOS PREDICTIVOS 19

2.2.3.1. Modelos basados en clasificación 20

2.2.4. MÉTRICAS DE EVALUACIÓN 20

2.2.4.1. Matriz de confusión 20

2.2.4.2. Precision (Precisión) 20

2.2.4.3. Acurracy (Exactitud) 20

2.2.4.4. Recall (Exhaustividad/Sensibilidad) 20

2.2.4.5. AUC (Área bajo la curva) 21

2.2.4.6. F1 Score (Valor F1) 21

III. MATERIALES Y MÉTODOS 21

3.1. TIPO DE INVESTIGACIÓN 21

3.2. MÉTODOS DE INVESTIGACIÓN 21

3.3. TÉCNICAS E INSTRUMENTOS DE RECOLECCIÓN DE DATOS 21

3.4. PROCEDIMIENTOS 22
3.4.1. METODOLOGÍA DE DESARROLLO 22
3.4.2. PRODUCTO ACREDITABLE 23
3.4.3. MANUAL DE USUARIO 24
3.5. MATRIZ DE CONSISTENCIA 25
3.6. CONSIDERACIONES ÉTICAS 27

IV. RESULTADOS Y DISCUSIÓN 28

4.1. EN BASE A LA METODOLOGÍA UTILIZADA 28
4.1.1. FASE #1. COMPRENSIÓN DEL NEGOCIO 28
4.1.2. FASE #2. COMPRENSIÓN DE LOS DATOS 31
4.1.3. FASE #3. PREPARACIÓN DE DATOS 36
4.1.4. FASE #4. MODELAMIENTO 39
4.1.5. FASE #5. EVALUACIÓN 51
4.1.6. FASE #6. DESPLIEGUE 56
4.2. IMPACTOS ESPERADOS 62
4.2.1. IMPACTOS ECONÓMICOS 62
4.2.2. IMPACTOS SOCIALES 62
4.2.3. IMPACTOS EN TECNOLOGÍA 62
4.2.4. IMPACTOS AMBIENTALES 63

V. CONCLUSIONES 67

VI. RECOMENDACIONES 69

VII. ANEXOS 73

ANEXO N° 01. CARTA DE ACEPTACIÓN DE LA ENTIDAD PARA LA EJECUCIÓN DEL PROYECTO 73
ANEXO N° 02. CONSTANCIA DE APROBACIÓN DEL PRODUCTO ACREDITABLE DE LA ENTIDAD DONDE SE EJECUTÓ LA TESIS 74
ANEXO N° 03. GUIA DE ENTREVISTA: COMPRENSIÓN DEL CONTEXTO Y LAS VARIABLES DE ANÁLISIS 75
ANEXO N° 04. MANUAL DE USUARIO 76
ANEXO N° 05. PRUEBAS DE CAJA BLANCA 89
ANEXO N° 06. PRUEBAS DE CAJA NEGRA 94

LISTA DE TABLAS

TABLA I. MÉTODOS DE INVESTIGACIÓN 21
TABLA II. TÉCNICAS E INSTRUMENTOS DE RECOLECCIÓN DE DATOS 22
TABLA III. MATRIZ DE CONSISTENCIA 25
TABLA IV. PLAN DE PROYECTO 30
TABLA V. DICCIONARIO DE DATOS 32
TABLA VI. CARACTERÍSTICAS COMPARATIVAS DE LOS ALGORITMOS EMPLEADOS 40
TABLA VII. MATRIZ DE CONFUSIÓN 42

LISTA DE FIGURAS

FIG. 1. VISIÓN DE MINERÍA DE DATOS 18
FIG. 2. VISUALIZACIÓN DE DATOS EN GOOGLE COLAB 31
FIG. 3. AGRUPACIÓN DE LOS DATOS NUMÉRICOS Y CATEGÓRICOS 33
FIG. 4. ANÁLISIS DESCRIPTIVO DE LAS VARIABLES NUMÉRICAS 34
FIG. 5. IMPUTACIÓN DE VALORES NULOS EN LA VARIABLE FACTURACION 36
FIG. 6. RECODIFICACIÓN DE LA VARIABLE CATEGÓRICA SEGMENTO 37
FIG. 7. CONSOLIDACIÓN DE LA DATA FINAL 38
FIG. 8. EQUILIBRADO DE LA DATA 39
FIG. 9. DETERMINAR LA DATA DE ENTRENAMIENTO Y PRUEBA 39
FIG. 10. ALGORITMO DECISION TREE 43
FIG. 11. ALGORITMO RANDOM FOREST 44
FIG. 12. ALGORITMO BAGGING 45
FIG. 13. ALGORITMO ADABOOST 46
FIG. 14. ALGORITMO XGBOOST 48
FIG. 15. ALGORITMO SVM RECALIBRADO 50
FIG. 16. COMPARACIÓN DE LOS RESULTADOS DE LOS MODELOS 51
FIG. 17. MATRIZ DE CONFUSIÓN DEL MODELO SELECCIONADO 52
FIG. 18. COMPARATIVA DE ESCENARIOS PREDICHOS VS REALES 53
FIG. 19. ARQUITECTURA DEL PRODUCTO FINAL 54
FIG. 20. API VERSIÓN FINAL 57
FIG. 21. MÉTODO POST FUNCIONANDO CORRECTAMENTE 58
FIG. 22. INTERFACES FINALES EN FUNCIONAMIENTO 60

RESUMEN

En el presente estudio se desarrolló un modelo predictivo haciendo uso de técnicas de minería de datos para analizar el comportamiento del cliente, con la finalidad de lograr identificar y clasificar a los clientes con mayor riesgo a desertar en una empresa de telecomunicaciones y así, apoyar a la empresa en la toma de decisiones certísimas y la creación de estrategias de retención. Para lograr el objetivo principal, se analizaron las características algorítmicas de los principales algoritmos de minería de datos propuestos por la bibliográfica para determinar el que mejor logre adaptarse a la realidad presente, obteniendo el mejor desempeño en las métricas de evaluación propuestas con el algoritmo XGBoost, el cual, obtuvo un 83% de precisión para determinar a los potenciales clientes con riesgo a desertar. Para el desarrollo del módulo de predicción en base al algoritmo seleccionado, se empleó la metodología CRISP-DM para la construcción, evaluación y despliegue. El despliegue del modelo se realizó construyendo en base a los lenguajes de programación JavaScript y Python, empleando el Framework Flask, una interfaz web local, la cual, permite generar reportes específicos y globales al usuario final. Finalmente, se determinó el grado de usabilidad aceptable del modelo a partir de dos indicadores; su efectividad, demostrada en el grado de precisión obtenido de 83%, los resultados en las métricas de evaluación y el porcentaje de asertividad del 80%; y la eficiencia de la interfaz final, en términos de empleo y su desempeño en las pruebas de caja blanca y negra.

Palabras clave: minería de datos, algoritmos, aprendizaje supervisado, clasificación, telecomunicaciones, deserción de clientes.

ABSTRACT

In the present study, a predictive model was developed using data mining techniques to analyze customer behavior to identify and classify customers with the highest risk of deserting a telecommunications company and thus support the company in making very certain decisions and creating retention strategies. To achieve the main objective, the algorithmic characteristics of the main data mining algorithms proposed by the bibliography were analyzed to determine the one that best adapts to the present reality, obtaining the best performance in the evaluation metrics proposed with the XGBoost algorithm, which, obtained an 83% accuracy to determine the potential clients with risk of deserting. For the development of the prediction module based on the selected algorithm, the CRISP-DM methodology was used for the construction, evaluation and deployment. The deployment of the model was carried out based on the JavaScript and Python programming languages, using the Flask Framework, a local web interface, which allows the generation of specific and global reports to the end user. Finally, the degree of acceptable usability of the model was determined from two indicators; its effectiveness, demonstrated in the degree of precision obtained of 83%, the results in the evaluation metrics and the percentage of assertiveness of 80%; and the efficiency of the final interface, in terms of use and its performance in white and black box tests.

Keywords: data mining, algorithms, supervised learning, classification, telecommunications, customer churn.

I. INTRODUCCIÓN

En la actualidad, con la globalización de los servicios y la amplia competencia, los productos brindados por las organizaciones son cada vez más similares en calidad y precio [1], [2]. Por este motivo, las empresas se han transformado y han pasado de tener un enfoque orientado al producto, a un enfoque orientado al cliente [3], [4], [5]. Según [6], la existencia de una empresa se justifica en sus clientes, ya que estos representan el activo más importante dentro de la organización.

Con la amplia competencia en el mercado, los clientes tienen relativa facilidad en cambiar de un proveedor de servicios a otro y son las organizaciones las que se verán obligadas a diseñar un plan de contingencia que les permita fidelizar a sus clientes [7], [8]. Aquellos clientes que deciden desligarse de los servicios brindados por una empresa son denominados desertores e identificar a dicho clientes puede permitir a las organizaciones aplicar estrategias de fidelización y retención [9]. La fidelización o retención de los clientes es una estrategia empleada por las organizaciones para satisfacer las necesidades de sus clientes y así evitar que se desliguen de sus servicios [10], [11], [12].

En el pasado, la eficiencia en la adquisición de clientes en relación con el número de abandonos era equiparable y la deserción de los clientes no era un problema importante para las organizaciones; no obstante, con el crecimiento del mercado, la feroz competencia y la globalización de los servicios, los costos de adquisición de nuevos clientes aumentaron exponencialmente [13], [14]. Por otro lado, algunas investigaciones nos muestran que el costo de obtener de un nuevo cliente es de 5 a 7 veces mayor al costo de retención de uno antiguo [15]. Además, según [16] en su investigación nos menciona un dato extraído del libro “The loyalty effect” donde se demostró que el aumento del 5% en la tasa de retención de clientes logró aumentos del 35% y el 95% en el valor actual neto de los clientes en una empresa desarrolladora de software y una agencia de publicidad, respectivamente. Complementando esta idea según [17], incrementar la taza de retención de clientes en un 5% aumentaría las ganancias de una empresa de un 25% a un 85%.

La deserción de los clientes en la actualidad es un problema de suma relevancia en diversos sectores, así mismo, las áreas más destacadas por sus estudios realizados son la educación [18], [19], [20], y el sector financiero [21], [6]. El

sector de las telecomunicaciones no escapa de esta realidad, en las telecomunicaciones móviles, el término “deserción” hace referencia a la perdida de suscriptores que cambian de un proveedor a otro durante un periodo de tiempo. Según [5], la tasa media de abandono de las telecomunicaciones móviles es de aproximadamente 2.2% mensual, significando esto que uno de cada cincuenta suscriptores de una determinada empresa interrumpe sus servicios cada mes.

Centrándonos en la realidad de la empresa de telecomunicaciones en la cual se realizó la presente investigación, se pudo determinar mediante una encuesta realizada al analista de datos del equipo de inteligencia comercial (*Ver Anexo 03*) algunas de las posibles causas que originan la deserción de sus clientes, por ejemplo, la poca eficiencia al analizar la información recogida de sus clientes. Al ser una empresa top en el mercado de telecomunicaciones del país, la cantidad de clientes afiliados a los servicios de esta es gigante (aproximadamente de 20 millones), a su vez, la cantidad de información proporcionada o recogida por cada uno de estos clientes a través de sus distintos sistemas constituye una cantidad masiva de información que no puede ser manejada por los analistas humanos en su totalidad y al no contar con sistemas capaces de analizar estos grandes cúmulos de información con el objetivo de encontrar patrones y crear conocimiento a partir de ella, este proceso se vuelve prácticamente imposible, tomando semanas en poder completar una análisis de los clientes empleando análisis estadístico y otros mecanismos tradicionales de análisis. Otra de las posibles causas por las cuales un cliente decide desertar de la compañía es la disconformidad referente al servicio o la atención oportuna de sus reclamos. Ambas razones expresadas, por ejemplo, en reclamos directos a la compañía, ya sea por temas referentes a la suscripción del servicio, la calidad de este o cobros irregulares. He ahí la importancia de la minería de datos, siendo esta un mecanismo de explotación y análisis, consistente en la búsqueda y extracción de información valiosa en grandes volúmenes de datos [22]. En otras palabras, la minería de datos resulta una herramienta útil para las empresas debido a que les permite analizar su información desde diferentes perspectivas obteniendo de ella información valiosa a la brevedad apoyando la toma de decisiones efectivas y a su vez, sirviendo de ayuda para la construcción del perfil del cliente basado en su comportamiento.

La presente investigación, se justifica en la necesidad de la empresa de telecomunicaciones en la cual se desarrolló la presente, de obtener una mayor

efectividad en la retención de sus clientes, para ello, se necesita analizar e identificar los patrones de conducta de los mismos, para determinar cuál será la ratio de deserción futura e implementar soluciones de mejora para aumentar el nivel de fidelización. Por los motivos previamente mencionados, en este estudio se propone desarrollar un modelo predictivo haciendo uso de técnicas de minería de datos para analizar el comportamiento de los clientes y lograr identificar y clasificar a estos según su probabilidad de deserción y así apoyar a la empresa en la toma de decisiones certísimas y la creación de estrategias de retención para mejorar su nivel de fidelización. Para lograr cumplir este objetivo, se establecieron una serie de objetivos específicos como: analizar comparativamente las características algorítmicas de las técnicas de minería de datos, para determinar la que mejor se adapte a las variables del modelo propuesto; elaborar en base al algoritmo seleccionado el módulo de predicción considerando las variables que logran definir el comportamiento fidelizado del cliente; reportar los patrones de conducta en base a las variables del modelo que definan escenarios del comportamiento del cliente como apoyo en la toma de decisiones estratégicas de fidelización, y determinar el grado de usabilidad aceptable del modelo para garantizar la satisfacción del cliente de la empresa de telecomunicaciones.

II. REVISIÓN DE LA LITERATURA

2.1. Antecedentes

Como se mencionó previamente, la deserción de los clientes es un problema que suscita en diversos sectores, no obstante, en el ámbito nacional no se han desarrollado estudios referentes a la implementación de modelos predictivos como alternativa de solución para este problema; sin embargo, en otros sectores como la banca y la educación existen investigaciones relevantes que pueden adaptarse a la realidad del sector de las telecomunicaciones y han sido utilizadas de base para el desarrollo de la presente investigación; por otro lado, se han considerado en el ámbito internacional investigaciones realizadas tanto en el sector de las telecomunicaciones como en sectores afines con el objetivo de tener como base los modelos empleados para el desarrollo de la presente investigación.

2.1.1. Antecedentes internacionales

En la investigación denominada "Customer churn prediction for a motor insurance company" publicada por IEE Access [23], se

planteó como objetivo desarrollar un modelo predictivo que pueda identificar los factores que influyen en la deserción de los clientes en una empresa de seguros de automóviles y así poder establecer los clientes propensos a desertar en un periodo de tiempo determinado. El modelo predictivo se llevó a cabo empleando árboles de decisión obteniendo como resultado un modelo eficaz para la identificación de clientes propensos a desertar teniendo un 91.18% de precisión, así como también se logró modelar el tiempo estimado para el abandono de los clientes con alto riesgo a desertar lo cual ayuda a la empresa a realizar campañas de marketing destinadas a reducir las tasas de abandono en la empresa. Finalmente, el autor concluye que el enfoque y el estudio realizado puede aplicarse y adaptarse a distintas realidades que compartan el mismo problema de la deserción de clientes. Este antecedente fue considerado debido al algoritmo de minería de datos empleado, así como su enfoque aplicable a distintas realidades.

En el estudio denominado “Diseño de un modelo predictivo de fuga de clientes utilizando algoritmos machine learning” publicado por la Universidad ECCI [24], se planteó el análisis y modelamiento, en base a técnicas de machine learning y la metodología CRISP-DM, modelos de predicción para la deserción de clientes en una empresa prestadora del servicio de telecomunicaciones. Se emplearon diversos algoritmos, entre los cuales destacan los algoritmos de boosting y el algoritmo de random forest, además se empleó GridSearch como validación cruzada para obtener los parámetros que mejor se adapten a la realidad presente. Finalmente, se obtuvo la confiabilidad aceptable del modelo en base a su nivel de precisión del 79.5% al clasificar correctamente a los posibles clientes desertores; además, el autor agrega que la efectividad del mismo se verá reflejada en los estudios posteriores de algún tipo de campaña de retención realizada por la compañía utilizando como herramienta el modelo para la identificación de clientes desertores. Esta investigación fue considerada para la selección de algoritmos utilizados en la presente, así como para la elección de la metodología

a utilizar. A su vez, debido a su relación con el cuarto objetivo de esta investigación, ya que el autor presenta los resultados de la implementación como muestra de la eficacia obtenida por el modelo y por ende la usabilidad del mismo al determinar a los potenciales desertores.

En el estudio denominado “Modelos de predicción de deserción de clientes para una administradora de fondos ecuatoriana” publicado por la Revista Compendium [6], se empleó de técnicas de minería de datos para la creación de modelos de predicción de deserción de clientes, los cuales, puedan ser utilizados dentro del mercado de desintermediación financiera. Se emplearon modelos tales como: arboles de decisión, random forest, entre otros. Estos modelos fueron evaluados en base a su nivel de precisión, obteniendo los mejores resultados con el algoritmo de random forest, el cual, obtuvo un porcentaje del 93% en la identificación de clientes desertores. Esta investigación fue considerada por el empleo y comparación de distintos algoritmos para predecir la deserción de los clientes, con lo cual, sirvieron de guía para la selección de los algoritmos empleados en la presente.

En la investigación denominada “Predicción de abandono de clientes en telecomunicaciones mediante el Aprendizaje Automático” publicada por la Universidad Jorge Tadeo Lozano [25], presentó el empleo de técnicas de minería de datos, basándose en datos históricos, para determinar patrones que puedan identificar posibles deserciones. El autor presenta el empleo de distintos algoritmos de minería tales como: arboles de decisión, random forest, xgboost, entre otros. Finalmente, se obtiene como resultado un modelo predictivo creado a partir del algoritmo XGBoost, el cual, obtuvo los mejores resultados en las métricas de evaluación propuestas obteniendo un grado de precisión del 74% y un valor AUC del 79% comprobando así su nivel de efectividad al predecir a los potenciales clientes desertores. Este antecedente fue considerado por el análisis comparativo de algoritmos empleados, lo cual, sirvió de guía para la comparación de los algoritmos empleados en la presente

investigación. Así mismo, la forma de presentar los resultados obtenidos y la evaluación de los modelos desarrollados sirvió de base para la presente.

2.1.2. Antecedentes nacionales

En la tesis denominada "Modelo de análisis predictivo para determinar clientes con tendencia a la deserción en bancos peruanos" publicada por la Universidad Peruana de Ciencias Aplicadas (UPC) [26], se planteó como objetivo la implementación de un modelo que permita determinar a los clientes con tendencias a la deserción en los bancos del Perú con la finalidad de servir de soporte a la toma de decisiones. Entre las principales causas del problema de investigación, este estudio identifica la imposibilidad de los modelos actuales de poder analizar grandes cúmulos de información, así como la falta de foco en el cliente por parte de los modelos actuales. Con la propuesta del modelo predictivo para la retención de clientes se cumplió el propósito de identificar a los clientes propensos a la deserción a través del análisis del comportamiento obteniendo un 93.20% en el modelo implementado. Concluyendo así que los resultados obtenidos fueron alentadores sustentando de dicha forma que el empleo de un modelo de predicción puede alcanzar grandes índices de precisión para la identificación de clientes con riesgo a desertar. Esta investigación fue considerada debido a su forma de presentar los resultados finales del modelo y fue tomada en cuenta para la realización del reporte final. Así mismo, brindó como solución a la problemática de la imposibilidad de los modelos actuales de analizar grandes cúmulos de información, la implementación de un modelo predictivo que permita aumentar los índices de eficiencia al realizar esta actividad, por ende, haciendo referencia a la usabilidad del modelo al precisar la identificación de clientes con riesgo a desertar en grandes cúmulos de datos.

En la tesis denominada "Modelo para automatizar el proceso de predicción de la deserción en estudiantes universitarios en el primer año de estudio" publicada por la Universidad Peruana de Ciencias Aplicadas (UPC) [27], se planteó como objetivo la implementación

de un modelo que permita determinar a los estudiantes con tendencias a la deserción en su primer año de estudio con la finalidad de brindar a las instituciones educativas una mayor visibilidad y oportunidad de acción frente a la problemática de la deserción. El autor desarrolló un modelo de análisis predictivo, en base al análisis de 15 variables de relevancia y el empleo de diversos algoritmos de minería de datos con el objetivo de encontrar el que mejor logre adaptarse a la realidad de estudio. Finalmente, con la propuesta desarrollada, se obtiene un 67.10% en la precisión del modelo para determinar a los potenciales alumnos desertores, así como la detección de las variables que más influyen en la deserción de los alumnos. Esta investigación fue considerada debido a los algoritmos empleados, los cuales, sirvieron de guía para la selección de los algoritmos utilizados en la presente y a su vez, por la forma en la que se presentaron los resultados finales, los cuales, fueron tomados en cuenta para la realización del reporte final.

En la tesis de doctoral denominada "Modelo predictivos de la deserción estudiantil en una universidad privada del Perú" publicada por la Universidad Nacional Mayor de San Marcos (UNMSN) [28], se planteó identificar la contribución de los modelos predictivos al determinar la deserción de los alumnos en asignaturas críticas; así como determinar el grado de confiabilidad de la implementación del mismo. La investigación se realizó como alternativa de solución a la problemática de la deserción estudiantil, desarrollándose un modelo predictivo basado en las características de los estudiantes para determinar los estudiantes propensos a desaprobar un curso lo cual puede llevarlos luego a la deserción de sus estudios universitarios. La implementación de este modelo favoreció a la institución a tomar acciones pertinentes frente a esta problemática para ayudar a los alumnos con mayor riesgo a desertar, logrando así reducir la cantidad de alumnos desaprobados en un curso de un 40% a 50% en comparación a los años anteriores en los que no se utilizó un modelo predictivo, demostrando a su vez el grado de confiabilidad del modelo predictivo para la certísima identificación de los alumnos

con mayor riesgo a desertar en base a sus características como estudiante. Esta investigación fue considerada por la variedad de algoritmos empleados, los cuales, sirvieron de guía para la selección de los algoritmos utilizados en la presente. Así mismo, el empleo de la metodología CRISP-DM, sirvió de guía para ser seleccionada como base para el desarrollo del modelo propuesto.

2.1.3. Antecedentes locales

Luego de realizarse una exhaustiva búsqueda en diversos repositorios de investigación y bases de datos de tesis, no se encontraron estudios realizados a nivel local ni en localidades cercanas (Piura o La Libertad) que cumplan con el requerimiento de antigüedad establecido (mayor a 2017) para la presente investigación.

2.2. Bases teórico científicas

En este apartado se presentan algunos conceptos esenciales con el objetivo de familiarizar al lector con el contexto de la presente investigación.

2.2.1. Dato e información

2.2.1.1. Dato

Según [29], un dato son registros puros de los estados del mundo, de fácil estructuración. Es decir, un dato es la descripción empírica de un hecho o suceso que podamos apreciar en nuestra realidad.

2.2.1.2. Información

Según [30], la información está conformada por un conjunto de datos pertinentes y con un propósito en específico. Es decir, la información representa la estructuración y agrupación con sentido de los datos lo cual dota a dicha agrupación de un significado.

2.2.1.3. Conocimiento

Según [19], el conocimiento representa información con valor resultantes después de una reflexión, síntesis, evaluación y conversión de la información, así como la adecuación a un contexto en específico. En pocas palabras, el conocimiento se origina a través de la síntesis de la información generando el saber “¿qué hacer?” o “¿cómo responder?” frente a un determinado suceso.

2.2.2. Minería de Datos

2.2.2.1. Definición

Según [22], la minería de datos es un mecanismo de explotación y análisis, consistente en la creación de conocimiento a partir de grandes cúmulos de información. Esta definición da énfasis en que debido a las grandes cantidades de información y el sin número de herramientas informáticas existentes se ha transformado el concepto del análisis de datos y se ha orientado al manejo de técnicas especializadas denominadas con el nombre de minería de datos.

A su vez, [29] define a la minería de datos como un proceso de identificación de patrones, potencialmente usables y comprensibles. Indicando también que la minería de datos a partir de grandes volúmenes de datos también denominados como Data Warehouse comprende un conjunto de técnicas enfocadas en la descripción y predicción. Finalmente, podemos concluir que la minería de datos es la disciplina que estudia el análisis de grandes cúmulos de información apoyándose en algoritmos especializados con el objetivo de obtener conocimiento a partir de ella. La visión de la minería de datos según [30] puede representarse en Fig. I.

Fig. 1. Visión de minería de datos

2.2.2.2. Tipos de Datos

Según [31], la minería de datos establece dos tipos de datos:

a. Magnitudes

Hace referencia a datos cuantitativos. Todo aquel dato que puede medirse o contarse, es decir, datos cuantificables. Los datos cuantitativos están conformados por dos subclases, los datos discretos y los datos continuos. Los datos discretos, son aquellos datos que poseen un conteo preciso, por ejemplo, la edad, el número de horas, número de días, etc. Por otro lado, los datos continuos, son

aquellos datos que pueden tomar cualquier valor dentro de un intervalo, por ejemplo, el peso, grados de precisión, tiempo, etc.

b. Categóricos

Hace referencia a datos cualitativos. Todo aquel dato que puede asignarse a una categoría correspondiente. Los datos categóricos están conformados por dos subclases, los datos nominales y los datos ordinales. Los datos nominales, son aquello datos para los cuales existe una asignación arbitraria como por ejemplo los colores, el sexo, tipo de servicio, etc. Por otro lado, los datos ordinarios hacen referencia a aquellos datos que poseen una relación de orden entre sus categorías, por ejemplo, grados académico o números de reclamo.

2.2.2.3. Grupos de Métodos

a. Métodos supervisados

Según [32], son aquellos métodos que predicen un dato o un conjunto de ellos, a partir de otros datos conocidos. En otras palabras, este modelo está basando en un variable objetivo y variables secundarias que ayudan a predecir la variable de salida, existiendo una dependencia entre las variables de entrada con las de salida.

b. Métodos no supervisados

Según [32], son aquellos métodos en los que se descubren patrones y tendencias a partir de los datos en general. En otras palabras, este modelo identifica patrones o estructuras a partir de todos los datos, ya que todos poseen la misma importancia. En este tipo de modelos se conocen los datos de entrada, pero no de salida y a diferencia de los métodos supervisados no requieren de una variable objetivo ya que buscan la interdependencia de las variables.

2.2.3. Modelos predictivos

Según [26], un modelo predictivo es una colección de técnicas matemáticas con el objetivo de encontrar relación entre una variable objetivo y otras variables independientes, con la finalidad de predecir valores futuros de la variable objetivo. A su vez, nos plantea que, para la identificación de los factores influyentes en la

predicción, podemos agruparlos por el nivel de afectación al resultado, desde poca afectación, media afectación y los de clara o mayor afectación.

2.2.3.1. Modelos basados en clasificación

Según [22], consiste en identificar características similares de un objeto con la finalidad de determinarle una categoría definida. Esta forma de análisis puede ser empleada para la construcción de modelos tanto descriptivos como predictivos. Entre algunos métodos basados en clasificación tenemos los árboles de decisión y las redes neuronales.

2.2.4. Métricas de evaluación

2.2.4.1. Matriz de confusión

La matriz de confusión es empleada para evaluar el desempeño de un clasificador en dónde la salida puede ser de dos a más clases. Está métrica de evaluación se utiliza para mostrar en una tabla los tipos de errores que se comenten en el modelo [33]. A partir de la matriz de confusión es que pueden construirse otras métricas de evaluación como: precision, acurracy, recall y F1 score.

2.2.4.2. Precision (Precisión)

La precisión es la métrica que representa la proporción de verdaderos positivos que son correctamente identificados en comparación con el número total de valores positivos que el modelo predijo [33].

2.2.4.3. Acurracy (Exactitud)

La exactitud es la métrica que indica el número de elementos clasificados correctamente en comparación con la totalidad de elementos [33].

2.2.4.4. Recall (Exhaustividad/Sensibilidad)

La sensibilidad es la métrica que indica la tasa de clasificación positiva correcta, es decir, la proporción de positivos que el modelo ha clasificado correctamente en función del número total de muestras positivas [33].

2.2.4.5. AUC (Área bajo la curva)

El área bajo la curva es la métrica que se utiliza para medir el acierto en la predicción de eventos binarios; es decir, eventos que ocurren o no ocurren. Refleja la relación entre la tasa de verdaderos positivos y la tasa de falsos positivos para diferentes niveles de umbral [33].

2.2.4.6. F1 Score (Valor F1)

El valor F1 es la métrica consistente en una combinación entre la métrica de sensibilidad y precisión y se emplea como un balance entre ellas. Esta métrica obtiene un valor alto, si ambas métricas (sensibilidad y precisión) son altas [33].

III. MATERIALES Y MÉTODOS

3.1. Tipo de investigación

Basándonos en lo presentado por [34], la presente investigación posee un enfoque de investigación experimental aplicada de diseño preexperimental con preprueba y postprueba. La investigación se realizó con una población específica y se contrasta el antes y el después de haber aplicado esta solución.

3.2. Métodos de investigación

Los métodos de investigación empleados serán los siguientes:

TABLA I
MÉTODOS DE INVESTIGACIÓN

Método	Descripción
Analítico	Estudio y análisis del problema que presenta la organización
Deductivo	Estrategia para el desarrollo de la solución al problema
Implementación	Se llevará acabo la ejecución la propuesta de solución

3.3. Técnicas e instrumentos de recolección de datos

En la siguiente tabla se visualizan las técnicas e instrumentos que serán empleados para la recolección de datos.

TABLA II
TÉCNICAS E INSTRUMENTOS DE RECOLECCIÓN DE DATOS

Técnicas	Instrumentos	Elementos de la población	Propósito
Fuentes internas	Reportes (Data Excel)	Clientes	Conocer el perfil de los clientes
Revisión de la literatura	Artículos e investigaciones científicas	Investigadores en el campo	Conocer acerca de las alternativas de solución desarrolladas en estudios previos
Entrevista	Guía de entrevista (*Anexo 03)*	Analista de Datos/ Trabajador del equipo de inteligencia comercial	Conocer el contexto y las variables de análisis

3.4. Procedimientos

3.4.1. Metodología de desarrollo

A continuación, se mencionan las actividades que se realizaron en cada una de las fases de la metodología CRISP-DM:

1. **Fase #1: Comprensión del negocio**

   En esta fase se desarrollaron las siguientes actividades:

   - ✓ Determinar los objetivos del negocio.
   - ✓ Evaluar la situación.
   - ✓ Determinar el objetivo de minería de datos.
   - ✓ Desarrollar el plan de proyecto.

2. **Fase #2: Comprensión de datos**

   En esta fase se desarrollaron las siguientes actividades:

   - ✓ Obtener los datos iniciales.
   - ✓ Describir los datos.
   - ✓ Explorar los datos.
   - ✓ Verificar la calidad de los datos.

3. **Fase #3: Preparación de los datos**

   En esta fase se desarrollaron las siguientes actividades:

   - ✓ Seleccionar los datos.
   - ✓ Limpieza de datos.
   - ✓ Construcción de los datos.
   - ✓ Integración de los datos.
   - ✓ Dar formato a los datos.

4. **Fase #4: Modelamiento**

   En esta fase se desarrollaron las siguientes actividades:

   - ✓ Seleccionar técnica de modelamiento.

- ✓ Generar el diseño de prueba.
- ✓ Construir el modelo.
- ✓ Evaluación del modelo.

5. **Fase #5: Evaluación**

   En esta fase se desarrollaron las siguientes actividades:

   - ✓ Evaluar resultados.
   - ✓ Revisar el proceso.
   - ✓ Determinar los siguientes pasos.

6. **Fase #6: Despliegue**

   En esta fase se desarrollaron las siguientes actividades:

   - ✓ Desplegar el plan.
   - ✓ Monitorear y mantener.
   - ✓ Desarrollar el reporte final.
   - ✓ Revisión del proyecto

3.4.2. Producto acreditable

1. **Interfaces**

   Se construyeron las interfaces del sistema vinculado al modelo predictivo empleando el framework Flask y haciendo uso de los lenguajes de programación Python y JavaScript, así como el leguaje de marcado HTML5, las misma se presentan en el ***ítem 4.1.6. Fase #6: Despliegue, sección Desarrollar el reporte final, en el Capítulo IV. Resultados.***

2. **Arquitectura**

   Se diseñó una arquitectura idónea para el funcionamiento del sistema vinculado al modelo predictivo, el cual se detalla en el ***ítem 4.1.5. Fase #5: Evaluación, sección Determinar los siguientes pasos, en el Capítulo IV. Resultados.***

3. **Infraestructura tecnológica**

   Considerando la arquitectura anteriormente descrita, se definen los componentes necesarios en el ***ítem 4.1.6. Fase #5: Evaluación, sección Determinar los siguientes pasos, en el Capítulo IV. Resultados.***

3.4.3. Manual de usuario

Se elaboró un manual de usuario con la finalidad de ayudar a los usuarios en el uso de la interfaz web que se implementó para el uso del modelo predictivo, la cual se muestra en el *Anexo Nº 04.*

3.5. Matriz de consistencia

TABLA III
MATRIZ DE CONSISTENCIA

FORMULACIÓN DEL PROBLEMA		MÉTODOLOGÍA DE INVESTIGACIÓN			
La deserción de los clientes en las empresas que brindan los servicios de telecomunicaciones.		TIPO DE INVESTIGACIÓN Investigación aplicada Preexperimental			
OBJETIVO GENERAL		MÉTODO	DESCRIPCIÓN		
Desarrollar un modelo predictivo haciendo uso de técnicas de minería de datos para analizar el comportamiento del cliente que mejore su nivel de fidelización en una empresa de telecomunicaciones.		Analítico	Estudio y análisis del problema que presenta la organización		
		Deductivo	Estrategia para el desarrollo de la propuesta de solución al problema		
		Implementación	Se llevará a cabo la ejecución de la propuesta de solución		
		TÉCNICAS	INSTRUMENTOS	ELEMENTOS DE LA POBLACIÓN	PROPÓSITO
		Fuentes internas	Reportes (Data Excel)	Clientes	Conocer el perfil de los clientes
		Revisión de la literatura	Artículos e investigaciones científicas	Investigadores en el campo	Conocer acerca de las alternativas de solución desarrolladas en estudios previos
		Entrevista	Guía de entrevista (*Anexo 03*)	Analista de Datos/ Trabajador del equipo de inteligencia comercial	Conocer el contexto y las variables de análisis
OBJETIVOS ESPECÍFICOS	DESCRIPCIÓN DEL LOGRO DE LOS OBJETIVOS ESPECÍFICOS			INDICADORES	
Analizar comparativamente las características algorítmicas de las técnicas de minería de datos, para determinar la que mejor se adapte a las variables del modelo propuesto.	Se desarrollarán y compararán modelos predictivos funcionales en base a diversos algoritmos de clasificación.			Resultados en las métricas de evaluación Grado de precisión en la identificación de los potenciales clientes a desertar	

Elaborar en base al algoritmo seleccionado el módulo de predicción considerando las variables que logran definir el comportamiento fidelizado del cliente.	Se obtendrá un módulo de predicción funcional en base al algoritmo seleccionado.	Construcción del módulo de predicción funcional Grado de precisión en la identificación de los potenciales clientes a desertar.
Reportar los patrones de conducta en base a las variables del modelo que definan escenarios del comportamiento del cliente como apoyo en la toma de decisiones estratégicas de fidelización.	La interfaz del sistema reportará los patrones de conducta en base a la información brindada por el usuario y definirá el comportamiento del cliente.	Reporte global del comportamiento y la segmentación de clientes (posibles desertores o sin riesgo a desertar)
Determinar el grado de usabilidad aceptable del modelo para garantizar la satisfacción del cliente de la empresa de telecomunicaciones.	El modelo obtendrá un grado aprobatorio en usabilidad calculada a través de los beneficios que otorguen los resultados del empleo del modelo en aspectos como la efectividad y eficiencia.	Grado de precisión en la identificación de los potenciales clientes a desertar. Grado de usabilidad de los resultados del modelo

3.6. Consideraciones éticas

Se tuvo en consideración las restricciones establecidas por la empresa por la obligación de salvaguardar el secreto de las telecomunicaciones y a mantener la confidencialidad de los datos personales de sus abonados y usuarios de acuerdo con la Constitución Política del Perú y las normas legales aplicables.

A continuación, se listan los aspectos que se han considerado para la protección y bienestar de los participantes de esta investigación, en este caso la organización y sus clientes, así como de la seguridad (resguardo) de los datos:

- ✓ Evaluación antiplagio empleando el software turnitin.
- ✓ Referencias a toda la información obtenida por parte de libros, tesis u artículos empleados para la redacción de la presente investigación en formato IEEE.
- ✓ Anonimato de la empresa de Telecomunicaciones en cuestión.
- ✓ Censura de RUC’s en los reportes publicados.
- ✓ Resguardo de los datos y secreto de la información.

IV. RESULTADOS Y DISCUSIÓN

4.1. En base a la metodología utilizada

En el presente trabajo de investigación se ha utilizado la metodología CRISP-DM como guía para el desarrollo del modelo de minería de datos. Dicha metodología sirvió de marco de referencia durante las fases de: Comprensión del negocio de las telecomunicaciones, comprensión de los datos de entrada, el modelamiento del módulo de predicción, su evaluación y el despliegue final.

4.1.1. Fase #1. Comprensión del negocio

i. Determinar los objetivos del negocio

a. Objetivo principal

- Asegurar la consolidación financiera de la compañía y acelerar la transformación para la captura de valor a largo plazo.

b. Objetivos secundarios

- Convergencia y retención
  - Fidelización y captura de clientes.
  - Crecimiento de ingresos en nuevas necesidades.
  - Retención de clientes afectados.
- Experiencia Cliente
  - Adaptarnos a la nueva normalidad: Digitalización.
  - Cumplimiento de la promesa cliente.
- Eficiencia Operativa
  - Maximizar la caja.
  - Venta de activos.
  - Acelerar la transformación: Automatización y Digitalización.

ii. Evaluar la situación

La empresa del sector de las telecomunicaciones en cuestión desea obtener una mayor efectividad en la retención de sus clientes, para ello, necesita analizar e identificar los patrones de conducta de sus clientes, para determinar cuál será la ratio de deserción futura e implementar soluciones de mejora para aumentar el nivel de fidelización de sus clientes. El término “deserción”, en el sector de

las telecomunicaciones, hace referencia a la pérdida de suscriptores que cambian de un proveedor a otro durante un periodo de tiempo determinado. Según una entrevista realizada al analista de datos del equipo de inteligencia comercial de la empresa (*Anexo 03*) se pudo determinar algunas posibles causas que originaron la deserción de sus clientes, por ejemplo, la poca eficiencia al analizar la información recogida de sus clientes. Al ser una empresa top en el mercado de telecomunicaciones del país, la cantidad de clientes afiliados a los servicios de esta es gigante (aproximadamente de 20 millones), a su vez, la cantidad de información proporcionada o recogida por cada uno de estos clientes a través de sus distintos sistemas constituye una cantidad masiva de información que no puede ser manejada por los analistas humanos en su totalidad y al no contar con sistemas capaces de analizar estos grandes cúmulos de información con el objetivo de encontrar patrones y crear conocimiento a partir de ella, este proceso se vuelve prácticamente imposible, tomando semanas en poder completar una análisis de los clientes empleando análisis estadístico y otros mecanismos tradicionales de análisis. Otra de las posibles causas por las cuales un cliente decide desertar de la compañía es la disconformidad referente al servicio o la atención oportuna de sus reclamos. Ambas razones expresadas, por ejemplo, en reclamos directos a la compañía, ya sea por temas referentes a la suscripción del servicio, la calidad de este o cobros irregulares. Teniendo en cuenta esta realidad es que la empresa mantiene la necesidad de contar con un sistema o modelo que le permita automatizar el análisis de la información de la compañía para encontrar patrones de comportamiento basado en la información de sus clientes y reducir tiempos, con lo cual, se identifiquen a los posibles clientes con riesgo a desertar y de esta forma, implementar medidas o campañas de fidelización oportunas para mantener a sus clientes en la compañía.

iii. Determinación del objetivo de minería de datos

El objetivo presentado anteriormente, se puede traducir en estos objetivos de minería de datos:

- **a.** Utilizar información histórica de los clientes para identificar patrones, variables o tendencias relevantes que puedan explicar los índices de deserción presentados.
- **b.** Analizar la relación entre los patrones de conducta, las variables o tendencias manifestadas en la data histórica de las transacciones de los clientes.
- **c.** Elaboración de un modelo predictivo utilizando los datos disponibles del comportamiento de los clientes para pronosticar las posibilidades de abandono de cada cliente.
- **d.** Agrupar y reportar a los clientes en base a su potencialidad y probabilidad de deserción.

iv. Desarrollar el plan de proyecto

TABLA IV
PLAN DE PROYECTO

Fase	Tiempo promedio	Recursos	Riesgo
Comprensión del negocio	2 semanas	Desarrollador del proyecto	Problemas para la contextualización de los objetivos del negocio. Disponibilidad del tiempo de los involucrados.
Comprensión de los datos	4 semanas	Desarrollador del proyecto	Problemas referentes a la comprensión de los datos.
Preparación de los datos	4 semanas	Desarrollador del proyecto	Problemas referentes a la limpieza e integración de los datos.
Modelado	5 semanas	Desarrollador del proyecto	Incapacidad para encontrar un modelo adecuado.
Evaluación	3 semanas	Desarrollador del proyecto	Incapacidad para implementar los resultados.
Despliegue	4 semanas	Desarrollador del proyecto	Problemas al desplegar el modelo o al desarrollar el reporte final.

4.1.2. Fase #2. Comprensión de los datos

i. Obtener datos iniciales

Para esta parte de la metodología la principal fuente de datos se obtiene del data set brindado por la empresa de telecomunicaciones, el cual, proporciona una data histórica que representa el comportamiento del cliente en la empresa y su grado de satisfacción con el servicio recibido.

A su vez, se realizó un estudio del negocio en base a una entrevista con el analista de datos del área de inteligencia comercial (*Anexo 03*) para comprender la naturaleza de los datos brindados y obtener una mayor comprensión de cuáles son los datos que en base a sus estudios, influyen más en la deserción de los clientes.

ii. Describir los datos

a. Cantidad de datos

El data set brindado por la empresa de telecomunicaciones se encuentra en un formato ccv el cual es un tipo de documento en formato abierto sencillo para representar datos en forma de tabla, en las que las columnas se separan por comas y las filas por saltos de línea. A su vez, posee una variedad de variables (13 columnas) que representan un punto importante en el negocio, específicamente del área comercial, así como un total de 65150 registros (filas).

Para la visualización de los datos en un formato más comprensible y manejable se empleó la herramienta Google Colab como se aprecia en la Figura 2; cabe resaltar que no solo se empleó para visualizar los datos, sino también se empleó posteriormente para trabajar los datos.

CANT_LLAM_SAL_TOT	MIN_SAL_TOT	CANT_LLAM_ENT_TOT	MIN_ENT_TOT	CANT_SMS_TOT	MB_TOTAL	PERMANENCIA	Q_MOVILES	SEGMENTO	Q_RECLAMOS
25038	37038.85	16254	19691.06	1671	309558.49	18	4	PREMIUM	1
4193	2574.26	1891	1382.99	30	0.02	23	5	PREMIUM	0
18501	41636.40	12965	31261.24	3839	126826.34	145	3	PREMIUM	0
6374	9847.05	3926	5865.86	195	57012.74	71	6	PREMIUM	0
726	2136.31	460	1310.98	5	108374.79	13	1	PREMIUM	3
129	175.22	126	189.91	4	402.58	1	4	PREMIUM	0

Fig. 2. Visualización de datos en Google Colab

b. Calidad de datos

El set de datos brindado por la empresa de telecomunicaciones cuenta con una serie de variables que representan importancia dentro del área comercial. Para obtener una comprensión completa de cada variable en particular y evitar futuras complicaciones en la fase de modelado se procede a definirlas, así como identificar sus tipos de datos en la Tabla V.

TABLA V
DICCIONARIO DE DATOS

Variable	Tipo	Escala de Medida	Descripción
RUC	Cualitativa Nominal	Razón	Identificador único de cliente
CANT_LLAM_SAL_TOT	Cuantitativa Discreta	Razón	Indica la cantidad total de llamadas salientes del cliente
MIN_SAL_TOT	Cuantitativa Continua	Razón	Indica los minutos totales salientes que el cliente utilizó
CANT_LLAM_ENT_TOT	Cuantitativa Discreta	Razón	Indica la cantidad total de llamadas entrantes del cliente
MIN_ENT_TOT	Cuantitativa Continua	Razón	Indica los minutos totales entrantes que el cliente recibió
CANT_SMS_TOT	Cuantitativa Discreta	Razón	Indica la cantidad total de mensajes que el cliente realizó
MB_TOTAL	Cuantitativa Continua	Razón	Indica el consumo total de Gigabytes utilizados por el cliente.
PERMANENCIA	Cuantitativa Discreta	Razón	Indica la permanencia en meses del cliente en la empresa
Q_MOVILES	Cuantitativa Discreta	Razón	Indica la cantidad total de líneas móviles del cliente en la empresa
SEGMENTO	Cualitativa Nominal	Nominal	Indica el sector al cual pertenece el cliente
Q_RECLAMOS	Cuantitativa Discreta	Razón	Indica la cantidad de reclamos y llamadas por averías realizadas por el cliente en su permanencia con la operadora.
FACTURACION	Cuantitativa Continua	Razón	Indica la cantidad total a pagar en soles por consumo del cliente
CLIENTE	Cuantitativa Discreta	Razón	Indica si el cliente ha desertado o no

iii. Explorar los datos

En esta fase se utilizó el apoyo visual de Google Colab para poder explorar la data desde diferentes puntos vista que nos permita profundizar el conocimiento de la misma. Inicialmente, como buena práctica se verificó el nombre de las columnas para estar seguro de que no existan complicaciones a futuro en el momento de trabajar con ellas. Sin embargo, no hubo mayores complicaciones en este punto ya que todos los campos se encontraron debidamente nombrados; posiblemente dado a la fuente de donde provienen estos datos (Base de datos SQL). Dicho

proceso se realizó con el propósito de documentar el grado de limpieza de la información y dejar constancia que no se realizó ninguna modificación a la misma.

El siguiente paso fue el de verificar el tipo de dato asignado a cada una de las variables por el entorno de trabajo para ello nos apoyamos de la función .dtypes.

Al realizarse la verificación de los tipos de datos asignados se pudo identificar de que la variable RUC había sido mal asignada como una variable numérica y se procedió a su reasignación.

El siguiente paso que se consideró, fue agrupar las variables en dos subgrupos, variables numéricas (11) y variables categóricas (1); esto con el objetivo de darles un procesamiento distinto al momento de verificar la calidad de los datos y su posterior preparación. Dicho paso se puede apreciar en la Figura 3.

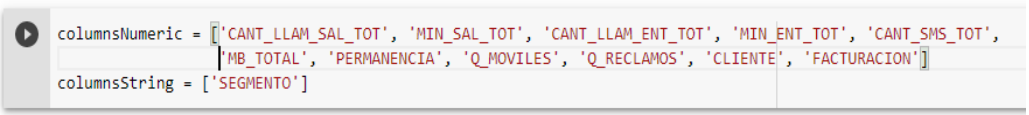


Fig. 3. Agrupación de los datos numéricos y categóricos

Habiéndose agrupado las variables en 2 subgrupos se procedió a hacer un análisis de las variables categóricas. En primera instancia, se determinó que el número de RUC no era una variable que representara relevancia en el análisis así que se descartó y se trabajó únicamente con la variable categórica “SEGMENTO”. Se utilizó la función .describe() de la librería pandas para obtener un análisis de estadísticas descriptivas sobre dicha variable. En dicho análisis se obtuvo un conteo de los datos, la cantidad de valores únicos, la moda (donde se visualizó que gran parte de la información brindada pertenece a un segmento en específico); datos que sirvieron más adelante en un análisis profundo de la información para la construcción de los datos.

Para poder visualizar la distribución de los valores en dichas variables se empleó una función creada para imprimir el tamaño de cada grupo único dentro de la variable segmento. Al hacer esto, se pudo confirmar que existe una gran cantidad de registros

pertenecientes al segmento PREMIUM y los demás registros repartidos entre los segmentos ALTO VALOR y EMPRESAS.

Como siguiente paso se procedió a realizar el análisis de las variables numéricas. Se utilizó la función .describe() de la librería pandas para obtener un análisis de estadísticas descriptivas sobre dichas variables. En dicho análisis se obtuvo un conteo de los datos, el promedio, valores mínimos y máximos, los cuartiles (1,2,3) y la desviación típica; datos que permiten visualizar que tan bien distribuidos están los datos. Dicho proceso se puede visualizar en la Figura 4.

```
df[columnsNumeric].describe()
```

	CANT_LLAM_SAL_TOT	MIN_SAL_TOT	CANT_LLAM_ENT_TOT	MIN_ENT_TOT	CANT_SMS_TOT	MB_TOTAL	PERMANENCIA	Q_MOVILES	Q_RECLAMOS	CLIENTE	FACTURAC
count	6.515000e+04	6.515000e+04	6.515000e+04	6.515000e+04	6.515000e+04	6.515000e+04	65150.000000	65150.000000	65150.000000	65150.000000	6.083300e
mean	1.336310e+04	2.493804e+04	1.043769e+04	1.898191e+04	5.110299e+03	2.831003e+05	72.500998	21.065188	4.632111	0.352694	5.695739e
std	1.838444e+05	3.824434e+05	1.480688e+05	3.093757e+05	3.616353e+05	6.931101e+06	65.419361	326.964882	96.674792	0.477812	2.054340e
min	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	-4.000000	0.000000	0.000000	0.000000	3.500000
25%	1.264000e+03	2.356020e+03	9.770000e+02	1.863170e+03	4.200000e+01	1.367482e+04	14.000000	1.000000	0.000000	0.000000	2.440500e
50%	3.556000e+03	6.821355e+03	2.751500e+03	5.331300e+03	2.240000e+02	5.328357e+04	60.000000	3.000000	1.000000	0.000000	6.592900e
75%	8.215000e+03	1.579297e+04	6.442750e+03	1.209727e+04	9.410000e+02	1.544706e+05	111.000000	7.000000	3.000000	1.000000	2.083520e
max	3.830478e+07	8.290605e+07	3.110132e+07	6.784097e+07	8.798316e+07	1.440268e+09	336.000000	45856.000000	22951.000000	1.000000	9.961980e

Fig. 4. Análisis descriptivo de las variables numéricas

Con la visualización del análisis descriptivo se pudo obtener una idea de cómo están distribuidos los valores en cada una de las variables numéricas. Tenemos por ejemplo variables como Q_MOVILES, MB_TOTAL, Q_RECLAMOS donde se puede visualizar la amplia diferencia entre los valores mínimos y máximos en cada una de esas variables, lo que representa un cúmulo de datos dispersos en donde ciertos valores se disparan de un valor promedio, lo que en conclusión da la idea de la existencia de valores atípicos dentro de dichas variables. También se puede apreciar data no tan dispersa como en el caso de PERMANENCIA donde el valor promedio es cercano al valor del 2 cuartil (50%) lo que indica una dispersión controlada de los datos. También podemos visualizar datos con valores iguales a 0 en distintas variables, lo cual nos da a entender que pueden existir datos missing o atípicos que generan una dispersión muy grande en la data. Teniendo en cuenta el conocimiento obtenido en este acercamiento a la data, se pudo marcar el camino a seguir en los

siguientes pasos teniendo en cuenta esta primera concepción de la data que ayudará a la próxima fase de limpieza de datos. En primera instancia, se comprobó la nula existencia de errores en digitación por lo cual, no se tuvo que realizar un análisis específico para este control; en segunda instancia se compró la existencia de datos nulos y el grado de dispersión de los datos, lo cual, facilitó la selección de estrategias correctas en la siguiente sección. Finalmente, se estableció la existencia de una variable categórica (SEGMENTO) la cual fue recodificada para poder emplearla en el desarrollar del modelo.

iv. Verificar la calidad de los datos

Para este punto se analizó inicialmente la presencia de datos missing tanto en las variables numéricas como categóricas.

Posterior al previo análisis, se pudo determinar que la data presentada es una data relativamente limpia, sin embargo, una de las variables tiene un número considerable de datos missing. Se identificó una cantidad de 0 datos missing en lo que respecta a la variable categórica analizada (SEGMENTO) y en el caso de las variables numéricas analizadas se pudo determinar la existencia de datos missing en la variable FACTURACIÓN. A su vez, luego de haber realizado el acercamiento a la data en el punto pasado, se pudo concluir que no existen errores visibles en la digitación de la data presentada, teniendo valores netamente reales en la variable categórica a analizar y valores netamente numéricos en las variables que corresponden a valores numéricos. Este relativo buen grado de calidad en la información con respecto a la existencia de valores missing o errores de digitación se pudo concluir que se debe a la fuente de donde se extrajo la data por parte de la organización, así como los buenos procesos del manejo de la información. Debido a esto, nos permite manejar la data original sin necesidad de optar por realizar ciertas modificaciones que podrían influenciar en los resultados finales en la fase de evaluación.

4.1.3. Fase #3. Preparación de datos

i. Seleccionar los datos

Para este punto, habiendo reflexionado en base a lo visualizado previamente en la fase de comprensión de los datos con respecto al comportamiento de las variables, la encuesta realizada (*Ver Anexo 03*) y previo al análisis detallado de las variables que tienen mayor grado de impacto en la deserción de los clientes realizado por la empresa, en colaboración con esta investigación, se consideró pertinente optar por todas las variables numéricas y categóricas, a excepción del RUC, para el desarrollo del modelo final ya que basado en el análisis previo se pudo concluir que son variables de valor para poder encontrar patrones relevantes que aporten criterios interesantes para el análisis.

ii. Limpieza de datos

En este apartado, como mencionamos anteriormente, se determinó la presencia de registros missing en la variable numérica FACTURACIÓN. Teniendo en consideración esto, se decidió tomar la medida de imputar estos registros vacíos empleando la estrategia de utilizar la mediana como factor de remplazo debido a que por la forma en la que se encontraban distribuidos los valores en esta variable, los datos atípicos podrían ser un problema, y al utilizar la mediana, esta no se verá influenciada por estos valores ya que ella trabaja principalmente con los valores centrales. Dicho proceso se puede apreciar en la Figura 5.

```
[ ]  #Se usa el metodo de imputacion:
     from sklearn.impute import SimpleImputer

     #Se genera el imputador iterativo - Imputacion Univariada Numerica
     imp_univ_num = SimpleImputer(missing_values=np.nan, strategy='median')
```

```
[ ]  data_impt_num = df[columnas_input]
```

```
[ ]  #Se realiza la imputación univariada en una nueva base de datos - Variables Numericas
     imp_univ_num.fit(data_impt_num) #entendimiento de lo que se desea hacer o entrenamiento o ajuste
     data_imp = pd.DataFrame(data=imp_univ_num.transform(data_impt_num),
                             columns=data_impt_num.columns,dtype='float')
```

Fig. 5. Imputación de valores nulos en la variable FACTURACION

Como se puede apreciar en la imagen para la imputación de los valores missing en la variable numérica, se empleó una librería denominada SimpleImputer que ayudó a generar un imputador de

datos que tuvo como parámetros la expresión de valor missing (NaN) en dicha variable y la estrategia de imputación, que como se mencionó anteriormente, se empleó la mediana para la imputación de datos. El siguiente paso fue correr el imputador previamente generado teniendo como valores el extracto de valores missing dentro de las variables numéricas (almacenado en la variable data_imp). Finalmente, se comprobó la cantidad de valores vacíos dentro de la variable imputada y se verificó que se pudo remplazar satisfactoriamente cada uno de los registros vacíos. Este proceso de limpieza se realizó principalmente para evitar tener un análisis incompleto de la data, así como también para evitar vacíos de información que influyan en gran medida a resultados desfavorables del modelo final.

ii. Construcción de los datos

En este apartado se procedió en primera instancia a recodificar los datos categóricos a valores numéricos que puedan ser empleados por el futuro modelo. Para ello, teniendo en cuenta el previo acercamiento a la data, se identificó que la variable categórica a recodificar es la variable SEGMENTO. Una vez identificada dicha variable se empleó la librería LabelEncoder para codificar cada uno de los registros dándole un valor numérico a cada valor único dentro de dicha variable almacenando los resultados de la codificación en la variable data_recod (posteriormente df_imp) como puede apreciarse en la Figura 6.

```
[77] #LABEL ENCODER PARA RECODIFICAR LOS DATOS
     from sklearn.preprocessing import LabelEncoder
     # Preprocesamiento con LabelEncoderfrom
     for c in columnsString:
         print(str(c))
         le = LabelEncoder()
         le.fit(data_recod[str(c)])
         data_recod[str(c)]=le.transform(data_recod[str(c)])

     SUB_SEGMENTO
     /usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:8: SettingWithCopyWarning:
     A value is trying to be set on a copy of a slice from a DataFrame.
     Try using .loc[row_indexer,col_indexer] = value instead
```

Fig. 6. Recodificación de la variable categórica SEGMENTO

iii. Integración de los datos

En este apartado se procedió a eliminar las variables FACTURACIÓN y SEGMENTO de la data inicial ya que estas fueron la variable con las que se construyó nueva data con el fin de recodificar o imputar la información vacía en ellas. Posterior a este proceso de eliminación se unificó el resultado de la eliminación de la data inicial con los fragmentos finales de data recodificada e imputada pertenecientes a las variables antes mencionadas (data_imp, df_imp), con el objetivo de tener el consolidado final de la data a utilizar para la construcción del modelo que veremos en el siguiente apartado. Al no alterarse el orden de los registros en los procesos anteriores, la adhesión se pudo realizar sin problemas. El proceso de integración puede visualizarse en la Figura 7.

```
[ ]  df = df.drop('FACTURACION',axis=1)

[ ]  df = df.drop('SEGMENTO',axis=1)

[ ]  # Consolidamos los subset!
     df_fin = pd.concat([df,df_imp,data_imp],axis=1)
```

Fig. 7. Consolidación de la data final

iv. Dar formato a los datos

Al realizar un análisis de la distribución de los datos en la variable target, se pudo evidenciar que la proporción de clientes identificados como no desertores era mayor que los clientes identificados como desertores, por lo cual, para no incurrir en problemas de desbalance que pueda afectar los resultados del modelo, se empleó el algoritmo ADASYN, el cual, es capaz de crear información sintética a partir de la data proporcionada con el objetivo de reducir el desequilibrio presente en la distribución de valores de la variable target. Dicho proceso puede apreciarse en la Figura 8.

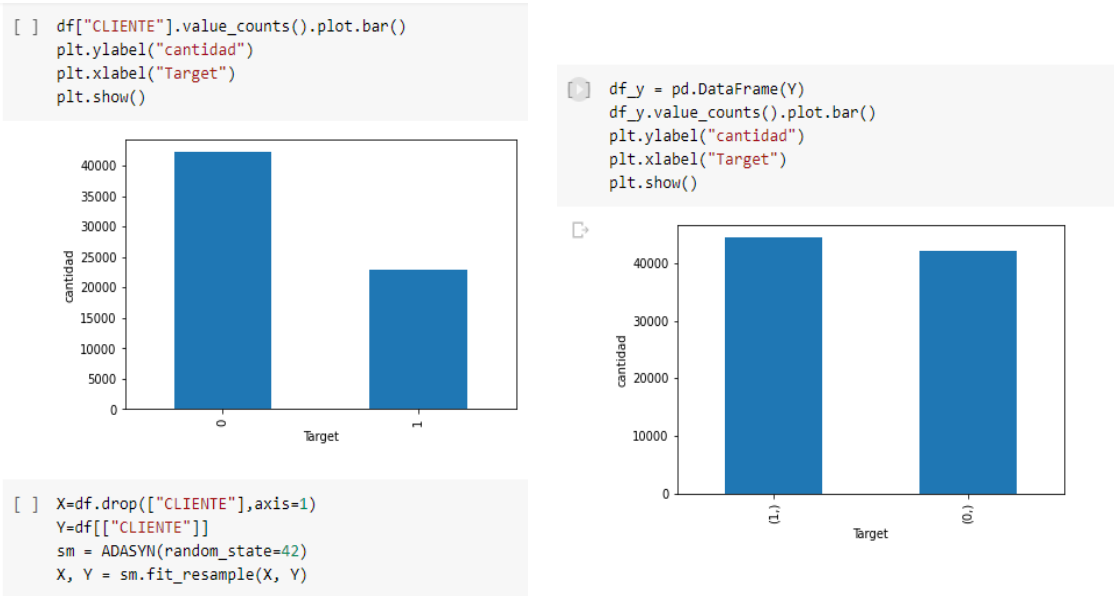


Fig. 8. Equilibrado de la data

El siguiente paso fue dividir la data en data de entrenamiento y data de prueba, es decir, los datos que utilizaremos para entrenar el modelo predictivo y los datos que emplearemos para comprobar si el modelo que hemos generado a partir de los datos de entrenamiento obtiene buenos resultados. En la presente se consideró para la data de entrenamiento y la data de prueba una proporción de 80% y 20% respectivamente, como lo propone [35]. Este proceso puede visualizarse en la Figura 9.

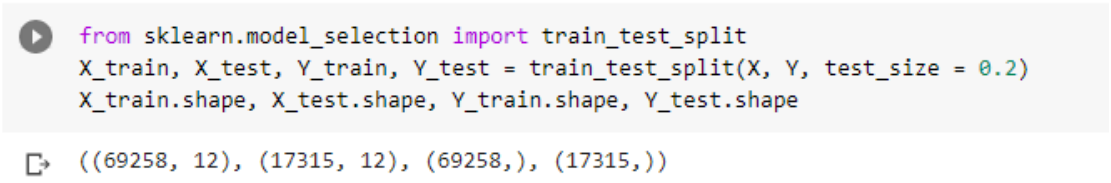


Fig. 9. Determinar la data de entrenamiento y prueba

Finalmente, cabe mencionar que los algoritmos seleccionados para el desarrollo del presente proyecto no requieren un formato especifico de la data ya que ellos son capaces de tratar la información sin un formato en específico, teniendo en consideración esto, se procedió con la fase de modelamiento.

4.1.4. Fase #4. Modelamiento

i. Seleccionar técnica de modelamiento

Iniciando con la etapa de modelado, se seleccionaron las técnicas de modelamiento a emplear teniendo en consideración 3 aspectos importantes, los tipos de datos disponibles para la generación del

modelo, los objetivos de minería de datos previamente presentados y los requisitos específicos previos de los modelos a escoger.

Primero, debido a la naturaleza de la data presentada, la cual, posee una variable objetivo (target) que indica si el cliente desertó o no de la compañía y un conjunto de variables independientes que representan el comportamiento del cliente, se determinó que el modelo a seguir debe estar basado en métodos supervisados, ya que estos poseen la característica de trabajar en base a una variable objetivo, prediciendo los resultados de esta, ayudado del comportamiento de las variables independientes [32].

Para el segundo punto, en congruencia a lo expuesto por [22], [24] y [25], se determinó que los métodos de clasificación son los más alineados con los objetivos de minería de datos en la presente investigación, ya que estos métodos son ampliamente utilizados en realidades en las que se requiere predecir una salida determinada en una variable objetivo debido a su alto grado de precisión.

Finalmente, y siguiendo la línea de lo previamente considerado, se tiene en cuenta que los métodos de clasificación emplean por lo general, dividir el conjunto de datos en 2 conjuntos al azar para ser trabajados uno como conjunto de entrenamiento y el otro como conjunto de evaluación. Teniendo en consideración el previo análisis, la experiencia propia y las investigaciones de [6], [23], [24], [25], [26], [27] y [28], se consideró pertinente para esta investigación utilizar los modelos mencionados a continuación, siendo en gran parte, los más empleados por los autores en las distintas bibliografías y considerando las características expuestas por [36], [37], [38] y [39].

TABLA VI
CARACTERÍSTICAS COMPARATIVAS DE LOS ALGORITMOS EMPLEADOS

Algoritmo	Dificultad	Cantidad de registros	Nivel	Tipo de aprendizaje
Decision Tree	Fácil	Alto	Medio	Supervisado
Bagging	Media	Media	Medio	Supervisado
Adabost	Media	Media	Medio	Supervisado
Random Forest	Media	Alto	Medio	Supervisado
XGBoost	Media	Alto	Medio	Supervisado
SVM	Difícil	Alto	Alto	Supervisado

ii. Generar el diseño de prueba

Para este apartado se detalla el cómo se midió inicialmente el rendimiento de los modelos implementados con el objetivo de poder recalibrar los distintos parámetros de cada modelo para obtener mejor resultados, así como para decidir finalmente cual será el modelo que mejor se adapte a la realidad presente. Teniendo en consideración las investigaciones de [6], [23], [24], , [25], [26], [27] y [28]; así como lo expuesto por [33], se consideró pertinente para evaluar el buen desempeño del modelo las siguientes métricas de evaluación:

TP: Positivos Verdaderos

TN: Negativos Verdaderos

FP: Positivos Falsos

FN: Negativos Falsos

- **Precision:** Esta métrica representa el número de verdaderos positivos que son realmente positivos en comparación con el número total de valores positivos predichos. Nos permite medir la calidad del modelo en tareas de clasificación.

$$\boldsymbol{precision} = \frac{\boldsymbol{TP}}{\boldsymbol{TP} + \boldsymbol{FP}}$$

- **Acurracy:** La exactitud mide el porcentaje de casos que el modelo ha acertado.

$$\boldsymbol{acurracy} = \frac{\boldsymbol{TP} + \boldsymbol{TN}}{\boldsymbol{TP} + \boldsymbol{TN} + \boldsymbol{FP} + \boldsymbol{FN}}$$

- **Recall:** La métrica de exhaustividad nos informa sobre la cantidad que el modelo es capaz de identificar.

$$\boldsymbol{recall} = \frac{\boldsymbol{TP}}{\boldsymbol{TP} + \boldsymbol{FN}}$$

- **AUC (Área bajo la curva)**: Refleja la relación entre la tasa de verdaderos positivos y la tasa de falsos positivos para diferentes niveles de umbral.

$$\mathbf{1} - \boldsymbol{especificidad} = \frac{\boldsymbol{FP}}{\boldsymbol{TN} + \boldsymbol{FP}}$$

- **F1 Score:** El valor F1 se utiliza para combinar las medidas de precision y recall en un sólo valor. Esta métrica es realmente práctica ya que permite comparar el rendimiento combinado de la precisión y la exhaustividad entre varias soluciones.

$$F1 = 2 * \frac{precision * recall}{precision + recall}$$

- **Matriz de confusión:** Es una herramienta que permite visualizar el desempeño del modelo empleado en el aprendizaje supervisado. Permite ver qué tipos de aciertos y errores que está teniendo el modelo a la hora de pasar por el proceso de aprendizaje con los datos.

TABLA VII
MATRIZ DE CONFUSIÓN

		Predicción	
		Positivos	**Negativos**
Observación	**Positivos**	Verdaderos Positivos (TP)	Falsos Negativos (FN)
	Negativos	Falsos Positivos (FP)	Verdaderos Negativos (TN)

iii. Construir el modelo

En este apartado se detallan los diferentes métodos que se emplearon para el modelo, considerando en primera instancia los resultados de la implementación de estos con los parámetros iniciales; para luego de su previa evaluación, recalibrar los métodos que obtuvieron los más altos resultados con el objetivo de mejorar su grado de efectividad al obtener los resultados.

a. Arboles de Decisión

Inicialmente, se importaron las librerías referentes a este algoritmo y se entrenó empleando la data de entrenamiento y prueba previamente obtenida en la sección pasada, teniendo en consideración la métrica de Accuracy para evaluar el resultado de su implementación, tal y como puede apreciarse en la Figura 10.

```
# Testearemos la profundidad de 1 a cantidad de atributos +1
for depth in depth_range:
    fold_accuracy = []
    tree_model = tree.DecisionTreeClassifier(criterion='gini',
                                             min_samples_split=20,
                                             min_samples_leaf=5,
                                             max_depth = depth,
                                             class_weight={1:3.5})
    for train_fold, valid_fold in cv.split(df):
        f_train = df.loc[train_fold]
        f_valid = df.loc[valid_fold]
# Mostramos los resultados obtenidos
do = pd.DataFrame({"Max Depth": depth_range, "Average Accuracy": accuracies})
do = do[["Max Depth", "Average Accuracy"]]
print(do.to_string(index=False))
```

```
 Max Depth  Average Accuracy
         1          0.352694
         2          0.364927
         3          0.505495
         4          0.515257
         5          0.566708
         6          0.546600
         7          0.590959
         8          0.569194
         9          0.589578
        10          0.596086
        11          0.602226
        12          0.603684
        13          0.604712
```

Fig. 10. Algoritmo Decision Tree

Como se puede apreciar en la imagen al implementar este método con los parámetros básicos (criterio Gini que se empleó para medir la calidad de cada división del árbol, min_sample_split con un número mínimo de muestras de 20 para dividir un nodo, min_samples_left de 5 como número mínimo de muestras necesarias de nodo hoja y una profundidad max_depth máxima), se obtuvieron resultados bastante bajos que fueron creciendo a mayor profundidad del árbol implementado, dándonos como mejor resultado un 0.60 de accuracy en su punto más profundo.

Con el objetivo de mejorar los resultados de este algoritmo, se empleó una serie de algoritmos de ensamble diseñados para ayudar a mejorar el rendimiento de los modelos a través de construir aleatoriamente una cantidad mayor del mismo modelo para comparar resultados.

El primero de dichos algoritmos de ensamble a utilizar fue el de Random Forest. Para este algoritmo, al igual que el anterior, inicialmente se importaron las librerías a emplear y se ejecutó el método con los parámetros por defecto dándole como premisa la creación de 500 árboles (n_estimators) para obtener distintos resultados y quedarnos con el mejor de ellos, dichos resultados se

evaluaron según las métricas vistas en la sección anterior. Este proceso se puede visualizar en la Figura 11.

Fig. 11. Algoritmo Random Forest

Como se puede apreciar en la imagen, la implementación de este algoritmo dio resultados bastante óptimos ya que se basa en construir distintos árboles de decisión para obtener los mejores resultados.

El segundo algoritmo de ensamble a utilizar fue Bagging. Siguiendo el mismo proceso, para este algoritmo, al igual que el anterior, inicialmente se importaron las librerías a emplear y se ejecutó el método con los parámetros por defecto con el objetivo de visualizar en primera instancia los resultados obtenidos. A su vez, este algoritmo trabaja en base a un algoritmo previo, por dicho motivo, se empleó como núcleo el algoritmo de árbol de decisión previamente implementado. Dicho proceso puede visualizarse en la Figura 12.

```
[ ] from sklearn.ensemble import BaggingClassifier
    from sklearn import metrics

[ ] bg = BaggingClassifier(tree.DecisionTreeClassifier(), max_samples=0.5, max_features=1.0, n_estimators=100)
    bg.fit(X_train, Y_train)

    BaggingClassifier(base_estimator=DecisionTreeClassifier(ccp_alpha=0.0,
                                                            class_weight=None,
                                                            criterion='gini',
                                                            max_depth=None,
                                                            max_features=None,
                                                            max_leaf_nodes=None,
                                                            min_impurity_decrease=0.0,
                                                            min_impurity_split=None,
                                                            min_samples_leaf=1,
                                                            min_samples_split=2,
                                                            min_weight_fraction_leaf=0.0,
                                                            presort='deprecated',
                                                            random_state=None,
                                                            splitter='best'),
                      bootstrap=True, bootstrap_features=False, max_features=1.0,
                      max_samples=0.5, n_estimators=100, n_jobs=None,
                      oob_score=False, random_state=None, verbose=0,
                      warm_start=False)

    print("Accuracy:",metrics.accuracy_score(Y_test, Y_pred))
    print("Precision:",metrics.precision_score(Y_test,Y_pred))
    print("Recall:",metrics.recall_score(Y_test,Y_pred))
    print("F1_SCORE:",metrics.f1_score(Y_test,Y_pred))
    print("AUC:",metrics.roc_auc_score(Y_test,Y_pred))

    Accuracy: 0.789489
    Precision: 0.839775
    Recall: 0.733727
    F1_SCORE: 0.783178
    AUC: 0.791591
```

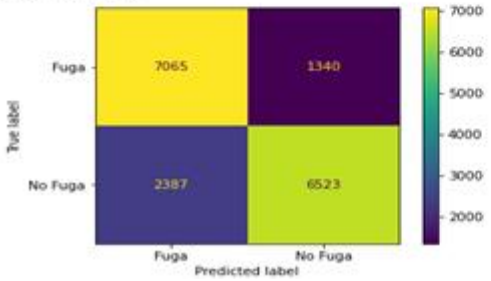


Fig. 12. Algoritmo Bagging

Como se puede apreciar en la imagen, este algoritmo recalibró los resultados iniciales y consiguió un resultado final optimo en termino de los resultados de las métricas de evaluación utilizadas; no obstante, el nivel de precisión final del modelo no se ve reflejado en la matriz de consistencia, en donde podemos visualizar que en distintos escenarios no logra cumplir las expectativas ideales.

El siguiente algoritmo por implementar para mejorar el modelo inicial fue AdaBoost. Siguiendo el mismo proceso, este algoritmo se trabajó en base algoritmo de Decision Tree con el objetivo de mejorar sus resultados en base a probar distintos escenarios. Para este algoritmo inicialmente se utilizaron los parámetros por efectos para visualizar los resultados obtenidos**.** Dicho proceso puede apreciarse en la Figura 13.

```
[ ] from sklearn.ensemble import AdaBoostClassifier
    from sklearn import metrics
    from sklearn.metrics import plot_confusion_matrix

[ ] adb = AdaBoostClassifier(tree.DecisionTreeClassifier(), n_estimators=5, learning_rate=1)
    adb.fit(X_train, Y_train)

    AdaBoostClassifier(algorithm='SAMME.R',
                       base_estimator=DecisionTreeClassifier(ccp_alpha=0.0,
                                                             class_weight=None,
                                                             criterion='gini',
                                                             max_depth=None,
                                                             max_features=None,
                                                             max_leaf_nodes=None,
                                                             min_impurity_decrease=0.0,
                                                             min_impurity_split=None,
                                                             min_samples_leaf=1,
                                                             min_samples_split=2,
                                                             min_weight_fraction_leaf=0.0,
                                                             presort='deprecated',
                                                             random_state=None,
                                                             splitter='best'),
                       learning_rate=1, n_estimators=5, random_state=None)

    Accuracy: 0.726364
    Precision: 0.734701
    Recall: 0.738631
    F1_SCORE: 0.736661
    AUC: 0.725902
```

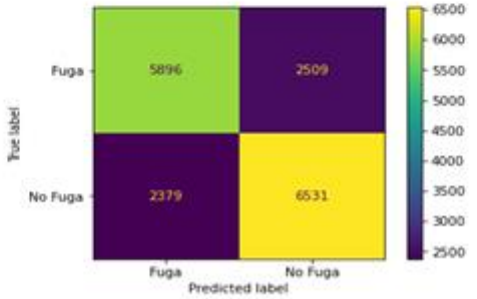


Fig. 13. Algoritmo AdaBoost

Como se puede apreciar en la imagen, este modelo no mejoró los resultados previos obtenidos, sino por el contrario, disminuyó su efectividad siendo esto demostrado en las métricas de evaluación. Por lo tanto, se decidió implementar un modelo de ensamble más con el objetivo de mejorar dichos resultados.

Finalmente, se empleó el algoritmo de ensamble más completo denominado “XGBoost”. Este algoritmo, al igual que los anteriores algoritmos de ensamble, trabaja en base a un modelo previo para mejorar los resultados, en este caso se trabajó en base al modelo de árbol de decisión para mejorar los resultados obtenidos. Al ser este un algoritmo mucho más preciso que sus antecesores, se decidió ajustar los parámetros del modelo desde el inicio, generando en primera instancia, una lista con los parámetros más empleados por este modelo y sus posibles valores; esta lista fue utilizada con una función denominada “GridSearchCV”, la cual, se encarga de

probar el modelo con los distintos parámetros y escenarios con el objetivo de conseguir los parámetros óptimos a utilizar. Una vez brindada esta lista con los parámetros más relevantes del modelo como, por ejemplo, “nthread” que representa el valor de subprocesos en paralelo a ejecutar que en este escenario se escogió uno a 1a vez por la carga computacional, “objetive” que representa el objetivo de aprendizaje de modelo que en este caso debido a la realidad presente se decidió escoger logística binaria. Una ratio de aprendizaje (learning_rate) entre 0.05 y 1 que es para reducir el tamaño de cada interacción del modelo para evitar el sobreajuste, el número de estimadores que para este caso se consideró entre 100 y 200 como buen escenario, entre un par de parámetros más con menos relevancia. Al finalizar este proceso, se obtuvo una lista con los mejores parámetros a escoger utilizando la métrica de accuracy como referencia para esta evaluación y dichos parámetros fueron empleados para desarrollar el ajuste final de este modelo. Finalmente, se decidió ejecutar el modelo. Dicho proceso puede apreciarse en la Figura 14.

```
[ ] import xgboost
```

```
[ ] xgb = xgboost.XGBClassifier()
```

```
[ ] from sklearn.model_selection import GridSearchCV
    from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score,roc_auc_score
    from sklearn.metrics import plot_confusion_matrix
```

```
[ ] fit_params = [
                  {'nthreads': [1],
                  'objective': ['binary:logistic'],
                  'learning_rate': [0.05, 0.1],
                  'n_estimators': [100,200],
                  'early_stopping_rounds': [10],
                  'eval_metric': ['logloss'],
                  'eval_set': [(X_test, Y_test)]}
                  ]
```

```
[ ] clf = GridSearchCV(xgb, fit_params,
                       cv=10, scoring='accuracy')
```

```
[ ] clf.fit(X_train, Y_train)
    print("accuracy:" ,accuracy_score(Y_test, Y_pred))
    print("precision:" ,precision_score(Y_test, Y_pred))
    print("recall:" ,recall_score(Y_test, Y_pred))
    print("f1_score:", f1_score(Y_test, Y_pred))
    print("auc:", roc_auc_score(Y_test,Y_pred))
```

```
Accuracy: 0.798727
Precision: 0.828131
Recall: 0.771734
F1_SCORE: 0.798938
AUC: 0.799747
```

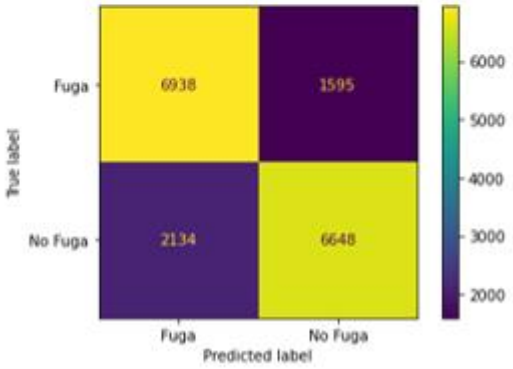


Fig. 14. Algoritmo XGBoost

Como se puede apreciar en la imagen, este modelo obtuvo unos resultados muy elevados en las métricas de evaluación escogidas lo cual se ve reflejado en el grado de predicción que se puede apreciar en la matriz de confusión diseñada, por tal motivo, es un modelo con gran oportunidad de ser tomado en cuenta al final de esta etapa.

Finalmente, se decidió implementar el algoritmo Sector Vector Machine, el cual, es un algoritmo de mayor alcance a comparación a los anteriores previamente implementados con el objetivo de

obtener una visión diferente del modelo implementado. Para ello, se inició importando las librerías y empleando una función de escalado para homogeneizar la información y así, ser empleada por el algoritmo en cuestión. El método seleccionado para escalar la data fue la función por excelencia denominada StandardScaler(). Una vez obtenida la data escalada a utilizar, se implementó el algoritmo con los parámetros por defecto para visualizar sus resultados, sin embargo, los resultados al implementar dicho algoritmo fueron considerablemente bajos comparado a los resultados de los modelos previamente implementados; no obstante, considerando la efectividad del algoritmo en cuestión, se decidió por implementar una técnica para recalibrar los parámetros del modelo.

La técnica empleada consiste en utilizar la función GridSearchCV utilizada en el modelo previo, la cual, se encarga de probar el modelo con las distintas opciones y escenarios con el objetivo de conseguir los parámetros óptimos a utilizar. Para dicho proceso, se tomó en consideración los parámetros más utilizados según la bibliografía de este algoritmo los cuales son los siguientes: 'C' encargado de regulación del modelo, "Gamma" que representa el coeficiente para el kernel escogido y "Kernel", siendo 'rbf' el escogido ya que es la función kernel más popular para la implementación de este modelo. Gracias a esta función se obtuvieron los mejores parámetros para el modelo, los cuales, fueron empleados para desarrollar el modelo recalibrado obteniéndose los resultados que pueden visualizarse en la Figura 15.

```
[58] sc = StandardScaler()
     X_train_scaled = sc.fit_transform(X_train)
     X_test_scaled = sc.transform(X_test)
```

```
[59] clf_svm = SVC(random_state=42, kernel= 'linear')
     clf_svm.fit(X_train_scaled, Y_train)
     Y_pred = clf_svm.predict(X_test_scaled)
     plot_confusion_matrix(clf_svm,
                           X_test_scaled,
                           Y_test,
                           values_format='d',
                           display_labels=["Fuga","No Fuga"])
```

```
print("Accuracy:",metrics.accuracy_score(Y_test, Y_pred))
print("Precision:",metrics.precision_score(Y_test,Y_pred))
print("Recall:",metrics.recall_score(Y_test,Y_pred))
print("F1_SCORE:",metrics.f1_score(Y_test,Y_pred))
print("AUC:",metrics.roc_auc_score(Y_test,Y_pred))
```

```
Accuracy: 0.632862
Precision: 0.639556
Recall: 0.667856
F1_SCORE: 0.653399
AUC: 0.631543
```

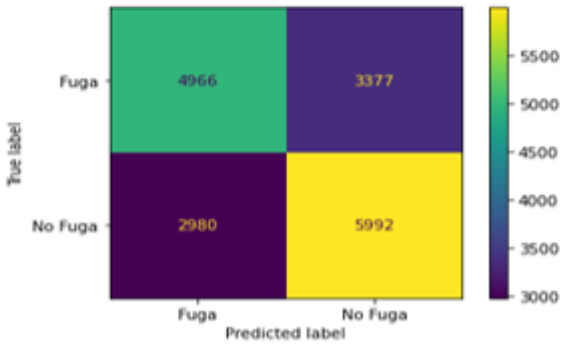


Fig. 15. Algoritmo SVM recalibrado

Como se puede apreciar en la imagen, al haber recalibrado los parámetros de dicho algoritmo se pudo obtener mejores resultados, no obstante, el algoritmo no se adaptó de la mejor manera a la data empleada, por lo cual, no obtuvo resultados resaltantes.

v. Evaluación del modelo

Para esta actividad, se realizó una tabla comparativa de los resultados obtenidos por cada uno de los modelos previamente implementados con el objetivo de comparar y seleccionar el algoritmo que mejor logró adaptarse a la realidad de la organización obteniendo la mayor efectividad. Dicho proceso puede apreciarse en la Figura 16.

```
comparison = {'accuracy score': (acc_rf,acc_adb,acc_bg,acc_xg,acc_svc),'auc score': (auc_rf,auc_adb,a
comparison_df = pd.DataFrame(comparison)
comparison_df.index = ['RANDOM_FOREST','ADA_BOOST','BAGGING','XG_BOOST','SVM']
comparison_df
```

	Accuracy	AUC	F1 score	Precision	Recall
XGBoost	0.798729	0.799747	0.798938	0.828131	0.771734
Random Forest	0.786601	0.787977	0.784611	0.822437	0.750111
Baggin	0.789489	0.791591	0.783178	0.839775	0.733727
AdaBoost	0.726364	0.725902	0.736661	0.734701	0.738631
SVM	0.632862	0.631543	0.653399	0.639556	0.667856

Fig. 16. Comparación de los resultados de los modelos

Como se puede apreciar en la imagen, se tomó en consideración las métricas de evaluación previamente definidas para evaluar los resultados del modelo, siendo los modelos más destacables los que emplearon los algoritmos de ensamble Random Forest, Bagging y XGBoost. Sin embargo, en el contraste previo realizado entre los resultados de los valores obtenidos por cada métrica de evaluación y los resultados obtenidos en sus respectivas matrices de confusión, el modelo que obtuvo el mejor desempeño y que logró adaptarse de mejor manera a la realidad de la data presente fue el modelo que empleo el algoritmo de XGBoost; siendo este el modelo a utilizar para la posterior fase de evaluación y despliegue.

4.1.5. Fase #5. Evaluación

i. Evaluar resultados

Según lo presentado en los puntos previos y visualizando los resultados presentados de cada uno de los modelos implementados, se pudo concluir que el modelo que mejor se adaptó a la información y a la realidad expuesta en el presente proyecto fue el modelo desarrollado en base al algoritmo de XGBoost en comparación a los otros modelos desarrollados, obteniendo un accuracy de 80%, un AUC de 80% y un grado de precision del 83%. A continuación, en la Figura 17, se presentará la matriz de confusión resultante al evaluar este modelo y se dará la respectiva conclusión a partir de ella.

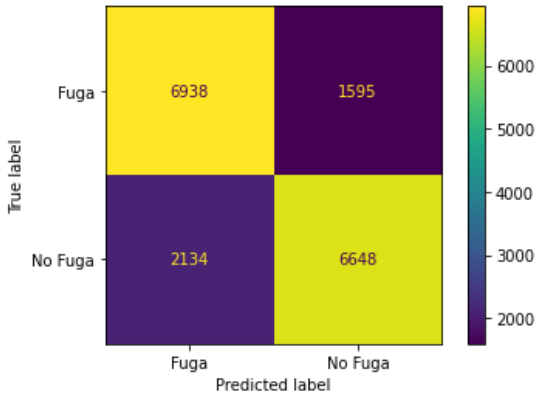


Fig. 17. Matriz de confusión del modelo seleccionado

Según lo resultante en la matriz de confusión, se puede justificar que el modelo clasificó correctamente 6938 que siguen siendo clientes mientras que 1595 fueron la cantidad de clientes clasificados erróneamente. De la misma forma, clasificó correctamente 6648 clientes que fugaron a la competencia y 2134 fueron catalogados como que se quedaron en la compañía, pero realmente se fueron con la competencia.

ii. Revisar el proceso

Para constatar que los resultados del modelo fueron favorables, se realizó un código de QA (verificación de la calidad), para el cual, inicialmente se hizo una comparativa con una porción de los registros de los valores reales de la variable CHURN y los valores predichos por el modelo; luego, se crearon escenarios aleatorios considerando los valores de cada variable en la data entregada por parte de la organización y se le asignaron dichos escenarios al modelo para evaluar la predicción en cada uno de ellos. El mencionado proceso de puede apreciar en la Figura 18 y 19.

```
y_pred = clf.predict(X_test)
```

```
y_pred = clf.predict(X_test)
df = pd.DataFrame({'Actual': Y_test, 'Predicted': y_pred})
df1 = df.head(60)
df1
```

	Actual	Predicted
0	1	1
1	0	0
2	0	0
3	1	0
4	0	0
5	1	0
6	0	0
7	1	1
8	1	0
9	1	1

```
cliente = np.array([25038,37038.85,16254,19691.06,1671,309558.49,18,4,1,2,5619.46,0])
cliente2 = np.array([4193,2574.26,1891,1382.99,30,0.02,23,5,0,2,195.21,1])
cliente3 = np.array([693,908.71,444,695.85,1,0,126,1,0,0,79.65,1])
```

```
model = pickle.load(open("/content/ml_model_xgbost.pkl", "rb"))
```

```
model
```

```
model.predict([cliente])
```

```
array([0])
```

```
model.predict([cliente2])
```

```
array([1])
```

```
model.predict([cliente3])
```

```
array([1])
```

Fig. 18. Comparativa de escenarios predichos vs reales

Como se puede apreciar en los resultados de la Figura 18, el modelo logra predecir correctamente en un 83%, el churn recibiendo escenarios aleatorios de la base proporcionada por la organización y contrastándolo con los resultados reales, confirmando los valores resultantes en la fase de evaluación Habiendo revisado el proceso correctamente, se procede a determinar los siguientes pasos a realizar.

iii. Determinar los siguientes pasos

Una vez evaluado y seleccionado el modelo que se adapta a la realidad presente, se desarrolló la arquitectura necesaria para su despliegue y funcionamiento en la web. Este paso, ayudó a integrar los resultados del modelo final con el proceso comercial de la organización y a producir un informe de resultados para el usuario

final. La fase de despliegue fue desarrollada en base a los siguientes componentes: La generación del archivo ejecutable del modelo entrenado, la construcción y subida de la API a la web y finalmente el desarrollo de la interfaz final para el usuario; la arquitectura empleada se detalla a continuación en la Figura 19.

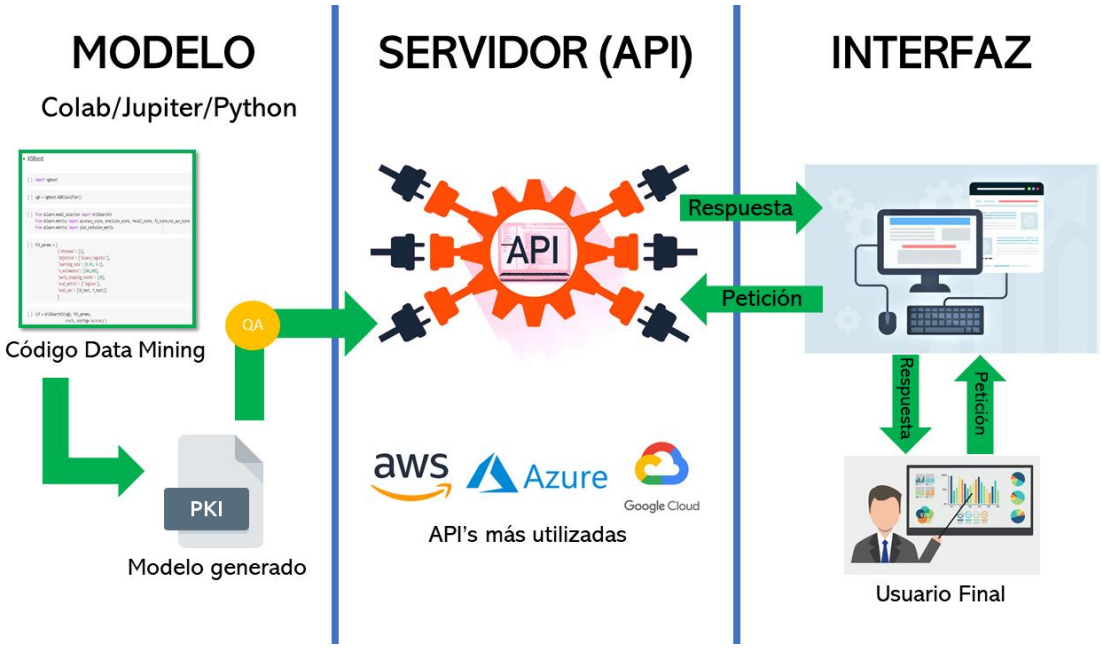


Fig. 19. Arquitectura del producto final

La fase uno del despliegue consiste en la elaboración, entrenamiento y evaluación del modelo predictivo para su final exportación en forma de archivo ejecutable, dicho archivo será la base de todo el sistema y del cual se basará la API web para dar respuesta a las peticiones del cliente y generar las predicciones solicitadas. Cabe resaltar que todo el proceso de elaboración, entrenamiento y evaluación ya fue realizado en los capítulos previos y únicamente queda exportarse a un archivo ejecutable.

La fase dos del despliegue consiste en la construcción y carga al servidor web de la API que servirá como nexo entre el modelo y la interfaz final que visualizará el cliente, su objetivo principal es escuchar las solicitudes del cliente y dar respuesta a cada una de ellas a través del modelo generado previamente.

Finalmente, la fase tres del despliegue consiste en la elaboración de la interfaz de usuario y su conexión directa a la API, para que el usuario pueda realizar sus solicitudes directamente desde la

interfaz y recibir las predicciones requeridas, así como los reporteres solicitados.

Para la implementación de esta arquitectura se requiere la siguiente infraestructura tecnológica:

- Hardware
  - Ordenador: Para el desarrollo del módulo de predicción se requiere un ordenador con la suficiente capacidad computacional para desarrollar y ejecutar el módulo de predicción.
  - Servidor Web HTTP: Para la presente implementación se empleó el servidor web local de desarrollo proporcionado por el framework Flask y será el encargado de alojar la API desarrollada para la comunicación entre el modelo y el usuario final.
  - Conexión a internet (Wi-FI o cableada): La conexión a internet será un pilar fundamental al momento de escalar el producto final a un servidor Web remoto, no obstante, para fines del presente proyecto no constituyó gran relevancia al haberse desarrollado en un servidor web local.
- Software
  - Sistema Operativo: El presente producto final al ser desarrollado en una plataforma de código abierto para Python es adaptable a cualquier sistema operativo.
  - Programas informáticos: Para el desarrollo y correcto funcionamiento del presente proyecto se emplearon programas gratuitos como Google Colab para el desarrollo del módulo de predicción, Visual Studio Code para el desarrollo de la API y la interfaz final y Microsoft Excel para el ingreso de data al sistema, así como la exportación final de reportes en dicho formato.
  - Lenguajes de programación: Para el desarrollo del módulo de predicción, así como la API encargada de recibir las peticiones de la interfaz final y emitir una respuesta al usuario

se empleó el lenguaje Python. Así mismo, se empleó el lenguaje de programación interpretado JavaScript para agregar ciertas funcionalidades a la interfaz final y el lenguaje de marcado HTML para la construcción de la misma.

4.1.6. Fase #6. Despliegue

i. Desplegar el plan

La primera fase del despliegue consistió en generar el archivo ejecutable de modelo entrenado, para ello, se empleó la librería pickle de Python para guardar el modelo seleccionado una vez entrenado y validado en los procesos previos.

La segunda fase del despliegue se basó en la construcción de la API que sirvió como nexo entre la interfaz de usuario y el modelo generado; para ello, se decidió por optar por los frameworks de desarrollo Fast API y Flask por sus diversas funcionalidades y facilidad para la creación de Web API's. Para este apartado, se empleó dos versiones de la API final, para la versión I, se empleó el framework Fast API con el objetivo de emplear su herramienta de prueba de peticiones en línea para comprobar el correcto funcionamiento de la API sin la necesidad de haber construido previamente una interfaz; y la versión II empleando el framework Flask con el objetivo de emplear las herramientas que brinda para facilitar la conexión de la API con la interfaz final. Una vez seleccionadas las herramientas a utilizar, se procedió con la construcción de la API empleando inicialmente Fast API, para esto, se basó en 2 métodos importantes, el método GET que será el encargado de dar respuesta a cualquier solicitud directa a la API, y el método POST que será el método empleado para la comunicación y el envió de datos desde el formulario a la API. Empleando este método se estructurará toda la lógica y parámetros requeridos por el modelo para realizar la predicción, así como los reportes de evaluación de predicción global de clientes. Una vez

construida la API se procedió inicialmente a cargar la misma en un servidor web local empleando la librería uvicorn.

Se empleó la herramienta de prueba de peticiones en línea que brinda FastApi para hacer la validación de los métodos creados y la conexión de la API con el modelo y comprobar que esté funcionando todo correctamente.

Una vez comprobado que la estructura de la API, así como la conexión del modelo a la misma funciona correctamente, se procedió a construir la versión final de la API empleando el framework Flask, esto, con el objetivo de poder facilitar la conexión de la API final, a la interfaz que se desarrolle para el proyecto final. Esta versión final de la API a diferencia de la anterior tendrá el método GET vinculado directamente a la interfaz final y esta servirá como medio de comunicación entre el cliente y la API. La construcción de esta API empleando los 2 métodos fundamentales se puede apreciar en la Figura 20.

Fig. 20. API versión final

ii. Monitorear y mantener

Para este apartado se monitoreó inicialmente que los métodos GET y POST de la API se hayan subido al servidor correctamente, para esto, se empleó la herramienta de prueba de peticiones en línea que nos proporcionó la versión I de la API.

El siguiente paso es comprobar el correcto funcionamiento de los métodos en el servidor, para ello, se procede a realizar peticiones POST con ciertos clientes escogidos al azar de la data empleada para el entrenamiento del modelo, con el objetivo de comprobar que se realice la predicción correcta y nos asegure el correcto funcionamiento del modelo (en términos de precisión) y la API en el servidor. Estos procesos pueden visualizarse en la Figura 21.

Fig. 21. Método POST funcionando correctamente

Como se puede apreciar en la imagen, el método POST empleado para la predicción del modelo, se encuentra funcionando correctamente; y a su vez; la API responde correctamente a los escenarios propuestos aleatoriamente comprobando la correcta comunicación entre el modelo entrenado y la API subida al servidor web.

iii. Desarrollar el reporte final

Para el desarrollo de la interfaz final, se empleó el framework Flask y el lenguaje de programación Python, así como lenguaje de marcado puro HTML, lenguaje de diseño como CSS, la biblioteca multiplataforma para diseños de Bootstrap 4 y el lenguaje de programación interpretado de JavaScript para algunas funcionalidades dentro de la interfaz.

Otros factores que se tomaron en cuenta para el desarrollo de la interfaz final, es que sea una interfaz simple, amigable a la vista, intuitiva y que automatice el proceso de digitación para el usuario final. A su vez, la interfaz permite generar reportes específicos y globales (de uno o varios clientes) en formato Excel, según el

usuario final lo decida. El producto final se puede apreciar en la Figura 22.

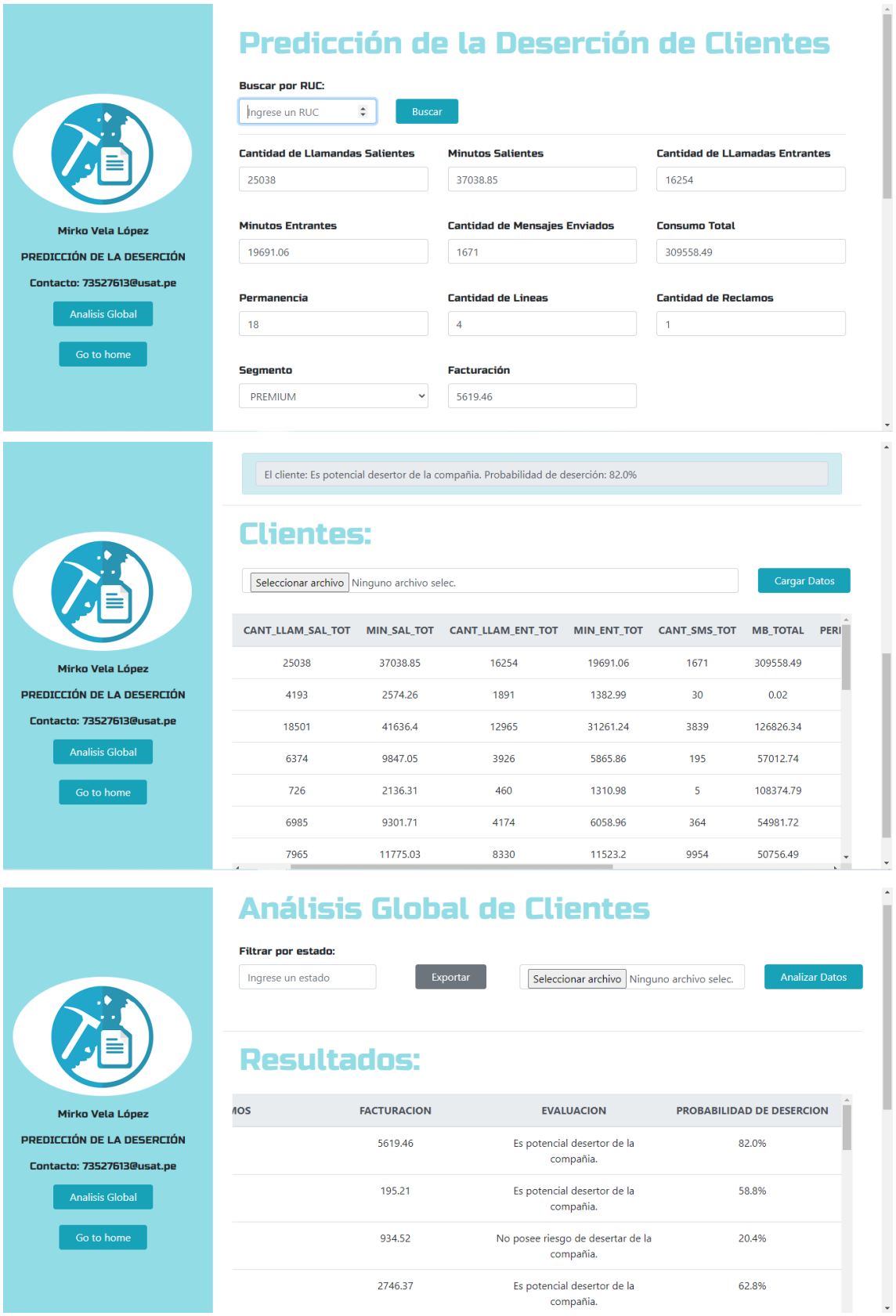


Fig. 22. Interfaces finales en funcionamiento

Como se puede apreciar en las imágenes mostradas se consiguió construir una interfaz amigable y simple para el empleo del modelo entrenado por parte del usuario final. La interfaz final cuenta con 2 apartados; el apartado principal, denominado “Home”, brinda la posibilidad de evaluar y predecir en base a la probabilidad de deserción, a un único cliente seleccionado de la data brindada; por otro lado, el apartado denominado “Análisis Global”, brinda la posibilidad de evaluar y predecir en base a la probabilidad de deserción, a un cumulo de clientes proporcionados en base a la data brindada; a su vez, este apartado muestra los pesos de cada variable analizada en torno a la relación que tienen con el resultado predicho (posible desertor o sin riesgo a desertar), además, este apartado cuenta con la posibilidad de generar reportes en formato Excel específicos y globales (de uno o varios clientes) según el usuario final lo requiera.

La interfaz final desarrollada cuenta con la posibilidad de cargar la data de los clientes identificados como posibles desertores por el Área de Inteligencia Comercial y contrastar los resultados de su análisis manual con el resultado más preciso que brinda el modelo predictivo desarrollado en base a la data histórica y la experiencia; de esta forma, se obtienen resultados más precisos y certeros de la realidad presente en los clientes y se apoya a la organización en la identificación temprana de la insatisfacción del cliente para la toma

de decisiones oportunas que ayuden a desarrollar estrategias focalizadas en pro de lograr fidelizar a los clientes con riesgo a desertar de la compañía.

iv. Revisión del proyecto

El proyecto se logró culminar habiendo comparados distintos modelos y seleccionando el modelo predictivo que mejor logró adaptarse a la realidad presente en la compañía (basado en el algoritmo XGBoost) obteniendo un grado de precisión del 83% y un acurracy del 80%; valores que se contrastan con la efectividad lograda por el modelo para predecir los escenarios de posibles clientes desertores. A su vez, el presente proyecto desarrolló toda la etapa de implementación del modelo y su puesta en producción para el despliegue final, con el objetivo de que el usuario final, pueda hacer uso de las predicciones del modelo a través de una interfaz web; esta interfaz, permite al usuario cargar la data de escenarios presentes y comprobar la posibilidad de deserción de cada uno de los clientes seleccionados con mayor eficacia y eficiencia que la del análisis manual y estadístico realizado el área de Inteligencia Comercial de la compañía. Además, la interfaz final brinda la capacidad al usuario final de analizar a los clientes de manera específica y global, brindándole un porcentaje de probabilidad de deserción que le permita centrarse en los clientes con el porcentaje de probabilidad de deserción dentro del rango establecido como señal de emergencia por la organización. Así mismo, la interfaz le permite conocer el grado de importancia de las variables con respecto a la predicción, para que el usuario final pueda tener en cuenta cuales variables implican mayor relevancia para determinar la predicción.

La precisión que brinda este sistema para determinar los posibles clientes desertores de la compañía permitirá a la organización reducir la lista de clientes identificados como potenciales desertores y de esta forma guiar la dirección de estrategias de fidelización a los clientes que lo requieran con mayor urgencia (Clientes con riesgo a desertar alto); además, permitirá anticiparse

y poder tomar decisiones oportunas para la fidelización de los clientes, logrando así impactar directamente en el NPS, en la retención de los clientes y finalmente reduciendo la perdida móvil, así como recortando gastos innecesarios logrando redireccionar los gastos empleados en campañas de fidelización en los clientes que realmente necesitan ser atendidos.

Para finalizar, se realizaron pruebas de caja blanca y caja negra al producto final las cuales pueden apreciarse en los *Anexos 05 y 06.*

4.2. Impactos esperados

Los impactos esperados con la implementación del modelo predictivo son los siguientes:

4.2.1. Impactos económicos

Se espera que la implementación del modelo predictivo en la organización favorezca la planificación de estrategias de retención enfocada a los clientes con real riesgo a desertar y, a su vez, consiga un aumento considerable en la taza de retención lo que ocasione el aumento en la rentabilidad, la disminución en costes de adquisición de nuevos clientes, ya que la perdida de los mismos será menor y el aumento de las ganancias en la gestión; así como también la reducción de los costos por perdida de clientes.

4.2.2. Impactos sociales

Se espera que la implementación del modelo pueda ayudar a identificar los factores que influyen en la insatisfacción y la deserción de clientes con el objetivo de desarrollar estrategias para brindar soluciones oportunas, que mejoren la calidad de servicio y trato al cliente, creando así un ambiente propicio entre la empresa y el consumidor del servicio.

4.2.3. Impactos en tecnología

Se busca que la presente investigación sirva de precedente para que futuras investigaciones puedan aplicar estrategias de minería de datos para predecir la deserción de clientes en diversos sectores a nivel nacional; contribuyendo así a la falta de estudios referente a la implementación de modelos predictivos a nivel nacional.

4.2.4. Impactos ambientales

Los impactos ambientales que pueden producirse al implementar soluciones tecnológicas suelen referirse a costos eléctricos, no obstante, la presente investigación no requiere de mucha implementación tecnológica para su desarrollo. Por lo tanto, la amenaza medioambiental que representaría en este ámbito sería muy baja. A su vez, la facilidad de la interfaz final de generar reportes descargables específicos y globales segmentando a los clientes según su riesgo a desertar, favorece al ahorro del papel de impresión y de esta forma contribuye a la protección del medio ambiente, así como la reducción de la deforestación que reduce el impacto en la contaminación.

En la presente investigación con el fin de analizar comparativamente las características algorítmicas de distintas técnicas de minería de datos para determinar la que mejor se adapte a la realidad presente, se realizó una comparativa de los resultados en las métricas de evaluación planteadas de distintos modelos basados en los principales algoritmos de clasificación utilizados por la bibliografía y se obtuvo un puntaje de 83% de precisión en el modelo basado en el algoritmo de XGBoost, el cual, tuvo como estructura base al algoritmo de Decision Tree, esto demostró que dicho algoritmo es el que mejor logró adaptarse a la realidad presente y a las variables en la data brindada. El algoritmo de Decision Tree fue empleado por [6], [23], para el desarrollo de sus modelos predictivos en sus respectivos contextos demostrando la efectividad del mismo y su facilidad para la interpretación de resultados, no obstante, a diferencia de esta investigación, la cual, cuenta con una data considerablemente más grande, se requirió emplear el algoritmo de ensamble XGBoost, tal como en la investigación de [25], para mejorar los resultados obtenidos en la predicción del modelo basado en Decision Tree, desarrollando varios modelos basados en este algoritmo para obtener los mejores resultados. Una vez analizados los principales algoritmos de clasificación utilizados por la bibliografía, se puede afirmar que el algoritmo de Decision Tree es bastante útil para los problemas de clasificación y que este puede potenciar sus resultados apoyándose en algoritmos de ensamble como lo es el algoritmo de XGBoost.

Con respecto a elaborar en base al algoritmo seleccionado el módulo de predicción considerando las variables que logran definir el comportamiento fidelizado del cliente, se desarrolló e implementó el módulo de predicción, teniendo como entrada distintas variables de relevancia (tráfico móvil, cantidad de líneas, cantidad de reclamos, facturación, entre otras) que definen el comportamiento de los clientes dentro de la organización. Este módulo de predicción de desarrolló en base al algoritmo XGBoost y siguiendo el marco que propone la metodología CRISP-DM, la cual, determina seis etapas para el desarrollo de modelos predictivos y brinda una guía en las tareas a realizar en cada una de las etapas. La metodología CRISP-DM fue empleada por [24] y [28] como base para el desarrollo de sus propuestas debido a su versatilidad y por ser la guía de referencia con más amplia trayectoria para el desarrollo de proyectos de minería de datos. A su vez, [6] consideró implementar distintos modelos tanto de clasificación como

de regresión con el objetivo de encontrar el que mejor se adapte a la problemática de la deserción de clientes en una administradora de fondos; en contraste con la presente investigación, la cual, se enfocó en desarrollar modelos predictivos basados en aprendizaje supervisado, como lo son los algoritmos de clasificación, debido a las características de la data de entrada para el modelo, así como del contexto de la necesidad de clasificar a los clientes con riesgo a desertar. Finalizando, se puede afirmar que la metodología CRISP-DM sirve como guía para el desarrollo y evaluación de un modelo basado en algoritmos de minería de datos, así como brindar las pautas a seguir para el despliegue del mismo.

En relación con reportar los patrones de conducta en base a las variables del modelo que definan escenarios del cliente como apoyo en la toma de decisiones estratégicas de fidelización, se desarrolló una interfaz web siguiendo las pautas para el despliegue del modelo brindadas por la metodología CRISP-DM, empleando el lenguaje de marcado HTML, así como los lenguajes de programación JavaScript y Python empleando el Framework de desarrollo Flask. Se logró así una interfaz intuitiva que permite la interacción entre el encargado de inteligencia comercial de la organización y el modelo predictivo cumpliendo con las expectativas de funcionamiento y facilitándole la detección certísima de clientes con potencial riesgo a desertar. En las investigaciones presentadas por [6], [23], [24], [27] y [28] no se llegó a desplegar el modelo quedándose únicamente en el modelado y evaluación del mismo, por lo cual, proponen la base a seguir para el desarrollo de una interfaz que pueda servir de nexo entre el modelo desarrollado y el usuario final. Por otro lado, las investigaciones de [25] y [26] concluyen en la implementación de una interfaz que se alimenta de la información de los clientes para realizar las predicciones correspondientes cada que el usuario final requiera un reporte. Teniendo como base a estas investigaciones se desarrolló una interfaz web local que sirva como nexo entre el usuario final y el modelo predictivo y que cuente con la facilidad de realizar reportes específicos y globales según lo requiera el usuario final responsable de la evaluación de los clientes. Además, como valor diferencial a estas investigaciones, la interfaz final desarrollada en la presente investigación cuenta con la capacidad de asignar un porcentaje de probabilidad de deserción, el cual, permitirá al usuario que realice la evaluación de los clientes tener una perspectiva de cuáles son los clientes con mayor riesgo a los que deberían enfocar las estrategias de retención.

Con respecto a determinar el grado de usabilidad aceptable del modelo para garantizar la satisfacción del cliente de la empresa de telecomunicaciones, se consideró dos indicadores, la efectividad y la eficiencia. Para determinar la efectividad, se presentaron los resultados obtenidos en las diferentes métricas de evaluación planteadas por el modelo predictivo seleccionado basado en el algoritmo de XGBoost, el cual, obtuvo un 0.80 de casos positivos detectados correctamente lo que equivale a que, de 50 clientes evaluados como posibles desertores, el modelo logró predecir de forma acertada 40 de ellos, errando únicamente en 10 predicciones que catalogó como clientes sin riesgo a desertar, lo que representa un asertividad en los resultados de aproximadamente el 80%, si el modelo hubiera sido implementado con anterioridad. A su vez, el modelo logró obtener un 0.83 de casos positivos detectados, lo cual representa la precisión del modelo para predecir a los clientes con riesgo a desertar de la compañía. En comparación a la investigación presentada por [28], la cual, nos presenta la aplicación del modelo en el contexto de la deserción estudiantil; logrando reducir la cantidad de alumnos desaprobados en un curso de un 40% a 50% en comparación a los años anteriores en los que no se empleó el modelo predictivo; y la investigación presentada por [24], la cual, nos indica que la efectividad de su modelo construido con una precisión del 79.5% se verá reflejada en los estudios posteriores de algún tipo de campaña de retención realizada por la compañía utilizando como herramienta el modelo para la identificación de clientes desertores; se puede afirmar que habiendo obtenido altos índices de precisión al detectar a los posibles clientes desertores se consiguió comprobar la efectividad del modelo y a su vez, se espera con certeza que ayude a reducir la tasa de abandono en el contexto aplicable. Con respecto a la eficiencia, en comparación a lo presentado por [26], el cual, con su modelo permitió la posibilidad de poder analizar grandes cúmulos de información en un menor tiempo; se logró implementar y presentar a la organización una interfaz web sencilla e intuitiva con la cual el usuario final pueda interactuar rápidamente con opciones de autocompletado en el formulario principal y la capacidad de realizar una inspección global en cuestión de segundos (o minutos dependiendo la cantidad de información proporcionada y la carga computacional) en base a la data brindada del comportamiento de los clientes para obtener como resultado la evaluación de los mismos y obtener una respuesta con un alto nivel de precisión con respecto a

la situación actual del cliente, es decir, su potencial riesgo a desertar, mejorando la situación actual de la organización, en la cual, el análisis de la información de clientes realizada de forma manual podía tardar semanas según lo expresado en la entrevista con él analista de datos (*Ver Anexo 03*). Además, se realizaron pruebas de caja blanca y caja negra (*Ver Anexo 05 y 06*) para garantizar el correcto funcionamiento de los flujos de ejecución y verificar la correcta funcionalidad de la interfaz fina. Teniendo en cuenta estas consideraciones, además de la evaluación y aceptación del modelo por parte del analista de datos de la empresa (*Ver Anexo 02*) , se afirma que se logrará reducir los tiempos de manejo de la interfaz, así como se reducirá la carga de trabajo del área de inteligencia comercial de la compañía con respecto al análisis de la información para determinar la potencial deserción de los clientes, en los tiempos estimados dentro de la planificación, ya que encontrarán en el modelo, una herramienta útil y precisa de evaluación para predecir el comportamiento de los clientes.

V. CONCLUSIONES

Mediante la implementación del modelo de minería de datos para predecir la deserción de clientes en una empresa de telecomunicaciones, se concluyó:

1. Al analizar comparativamente las características algorítmicas de las técnicas de minería de datos basándonos en los principales algoritmos de clasificación empleados por la bibliografía, se pudo identificar al algoritmo XGBoost como el que mejor se adaptó a la realidad presente obteniendo un 83% de precisión para predecir a clientes con posible riesgo a desertar de la compañía, así como un 80% de sensibilidad que representa la proporción de verdaderos positivos que el modelo pudo predecir correctamente, es decir, el porcentaje de predicciones realizadas correctamente, con lo cual, se puede afirmar que este algoritmo logró adaptarse a las necesidades de la organización.

2. Se logró construir en base al algoritmo seleccionado el módulo de predicción considerando las variables que logran definir el comportamiento del cliente empleando la metodología CRISP-DM como guía de desarrollo para la etapa de construcción, evaluación y despliegue del modelo. A su vez, se estableció la conexión entre el módulo de predicción y la interfaz final a través de una API desarrollada con el framework Flask, para que el usuario

final pueda emplear el modelo de forma intuitiva y amigable, manteniendo el grado de precisión del 83% en los casos positivos detectados correctamente, conseguido por el modelo.

3. Se lograron reportar los patrones de conducta en base a las variables del modelo que definan escenarios del comportamiento del cliente a través de la implementación del modelo predictivo en una interfaz web local intuitiva construida a partir del lenguaje de marcado HTML y los lenguajes de programación JavaScript y Python, la cual, posee la capacidad de generar reportes descargables específicos y globales según lo requiera el usuario final responsable de la evaluación de los clientes. Además, la interfaz final proporciona la capacidad de asignar un porcentaje de probabilidad de deserción, el cual, permitirá al usuario obtener una perspectiva de cuáles son los clientes con mayor riesgo a los que deberían enfocar las estrategias de retención.
4. Finalmente, se pudo determinar el grado de usabilidad aceptable del modelo en base a la efectividad y eficiencia del mismo. Con respecto a la efectividad, el modelo obtuvo un 83% de precisión para la predicción de los clientes con riesgo a desertar de la compañía y un 0.80 de casos positivos detectados correctamente, lo que representa un grado de asertividad en los resultados de aproximadamente el 80%, si el modelo hubiera sido implementado con anterioridad, con lo cual, se espera que ayude a reducir la tasa de abandono en el contexto aplicable. Con respecto a la eficiencia, se logró construir una interfaz web sencilla e intuitiva, con la cual, el usuario final pueda interactuar rápidamente y realizar una evaluación de los clientes en base a la data brindada para obtener una respuesta en cuestión de segundos (o minutos dependiendo la cantidad de información proporcionada y la carga computacional) con respecto a la situación actual del cliente, es decir, su potencial riesgo a desertar; además, se realizaron pruebas de caja blanca y caja negra para garantizar el correcto funcionamiento de los flujos de ejecución y verificar la correcta funcionalidad de la interfaz final, así como también se consideró la evaluación y aceptación del producto final por parte del analista de datos de la empresa.

VI. RECOMENDACIONES

1. Se recomienda mejorar el modelo planteado haciendo uso de distintos parámetros que puedan recalibrar los resultados del mismo para obtener un mayor grado de precisión y exactitud para la detección de clientes con potencial riesgo a desertar.
2. Para mejorar los resultados obtenidos en la presente investigación, se propone emplear algoritmos de Deep Learning ya que son algoritmos mayormente empleados para cúmulos de información muy grandes y por su capacidad de aprender progresivamente de la data estudiada.
3. En cuanto al desarrollo de la interfaz se recomienda agregar un módulo que permita visualizar gráficamente el peso de cada una de las variables al momento de determinar al cliente con potencial riesgo a desertar. Esto podría apoyar a la identificación de la causa principal por la cual el cliente es determinado como propenso a desertar y así dar un seguimiento a dicha causa.
4. Por último, se considera apropiado para futuras investigaciones, considerar diversos criterios para la generación de reportes, entre los cuales, se recomienda adaptar registros de fechas para la elaboración de futuros modelos.

REFERENCIAS

[1] H. Arellano Díaz, «La calidad en el servicio como ventaja competitiva,» *Dominio de las Ciencias,* vol. III, pp. 72-83, 2017.

[2] D. P. Puerto Becerra, «La globalización y el crecimiento empresarial a través de estrategias de internacionalización,» *Pensamiento & Gestión,* nº 28, pp. 171-195, 2010.

[3] P. A. Fuentes Jiménez, «LA ORIENTACIÓN AL MERCADO: EVOLUCIÓN Y MEDICIÓN DE UN ENFOQUE DE GESTIÓN QUE TRASCIENDE AL MARKETING,» *PERSPECTIVAS,* nº 25, pp. 25-83, 2010.

[4] L. M. Valenzuela Fernández y C. A. Martínez Troncoso, «Orientación al Cliente, Tecnologías de Información y Desempeño Organizacional: Caso empresa de consumo masivo en Chile,» *Revista Venezolana de Gerencia,* vol. XX, nº 70, pp. 334-352, 2015.

[5] N. Lu, H. Lin, J. Lu y G. Zhang, «A Customer Churn Prediction Model in Telecom Industry Using Boosting,» *IEEE Transactions on Industrial Informatics,* vol. 10, nº 2, pp. 1659 - 1665, 2014.

[6] M. Bohorquez, J. Torys y M. Paredes Aguirre, «MODELOS DE PREDICCIÓN DE DESERCIÓN DE CLIENTES PARA UNA ADMINISTRADORA DE FONDOS ECUATORIANA,» *Revista Compendium: Cuadernos de Economia y Administración,* vol. VII, nº 1, pp. 1-11, 2020.

[7] E. Baena, J. J. Sanchez y O. Montoya Suárez, «EL ENTORNO EMPRESARIAL Y LA TEORÍA DE LAS CINCO FUERZAS COMPETITIVAS,» *Scientia et Technica,* vol. III, nº 23, 2003.

[8] I. Sánchez García, «Why some satisfied customers want to switch service providers?,» *Universia Business Review,* vol. XXXI, nº 12-41, 2011.

[9] N. Guangli, Z. Lingling, L. Xingsen y S. Yong, «The Analysis on the Customers Churn of Charge Email Based on Data Mining Take One Internet Company for Example,» *Institute of Electrical and Electronics Engineers,* nº 843-847, 2006.

[10] S. D. R. Pierrend Hernández, «Customer Loyalty and Customer Retention: Trend Required Today,» *Gestión en el Tercer Milenio,* vol. XXIII, nº 45, pp. 5-13, 2020.

[11] S. Peña Escobar, G. S. Ramírez Reyes y J. C. Osorio Gómez, «Evaluación de una estrategia de fidelización de clientes,» *Ingenierías Universidad de Medellín,* vol. XIV, nº 26, pp. 87-104, 2015.

[12] J. C. Fandos, M. Estrada, D. Monferrer y L. Callarisa, «ESTUDIO DEL PROCESO DE FIDELIZACIÓN DEL CONSUMIDOR FINAL,» *Revista Brasileira de Marketing,* vol. 12, nº IV, pp. 108-127, 2013.

[13] X. Zhang, G. Feng y H. Hui, «Customer-Churn Research Based on Customer Segmentation,» *2009 International Conference on Electronic Commerce and Business Intelligence,* pp. 443-446, 2009.

[14] L. Duen-Ren, Ya-Yueh y Shih, «Hybrid approaches to product recommendation based on customer lifetime value and purchase preferences,» *The journal of systems and software,* vol. LXXVII, nº 2, p. 181–191, 2005.

[15] J. Rozum, «Defining and Understanding Software Measurement Data,» *Software Engineering Institute Carnegie Mellon University*, 2002.

[16] J. Ahn, J. Hwang, D. Kim, H. Choi y S. Kang, «A Survey on Churn Analysis in Various Business Domains,» *IEEE Access*, vol. 8, pp. 220816-220839, 2020.

[17] R. Feinberg y M. Trotter, «Immaculate deception: the unintended negative effects of the CRM revolution: maybe we would be better off without customer relations management.,» *Defying the Limits*, pp. 26-31, 2001.

[18] Á. Hernández Prados, J. S. Álvarez Muñoz y A. Aranda Martínez, «EL PROBLEMA DE LA DESERCIÓN ESCOLAR EN LA PRODUCCIÓN CIENTÍFICA EDUCATIVA,» *Revista Internacional de Ciencias Sociales y Humanidades SOCIOTAM*, vol. XXVI, nº 1, pp. 89-112, 2017.

[19] R. Berumen y F. Lydie, «Deserción escolar en la educación superior en México: revisión de literatura,» *Revista Iberoamericana para la Investigación y el Desarrollo Educativo*, vol. XXI, nº 22, 2021.

[20] J. D. Torres González, D. Acevedo Correa y L. A. Gallo García, «CAUSAS Y CONSECUENCIAS DE LA DESERCIÓN Y REPITENCIA ESCOLAR: UNA VISIÓN GENERAL EN EL CONTEXTO LATINOAMERICANO,» *Cultura Educación y Sociedad*, vol. VI, nº 2, pp. 157-187, 2015.

[21] J. Lara Rubio, F. J. Liébana Cabanillas y M. Martínez Fiestas, «Lealtad bancaria y la medida del riesgo de abandono de los clientes de las entidades financieras,» *Harvard Deusto Business Research*, vol. II, nº 1, pp. 67-87, 2013.

[22] I. Wilford Rivera, «MINERÍA DE DATOS: HERRAMIENTA DE APOYO EN LA SELECCIÓN DE EQUIPOS DE PROYECTOS INFORMÁTICOS,» *Industrial*, vol. 27, nº 3, 2006.

[23] M. Spiteri y G. Azzopardi, «Customer Churn Prediction for a Motor Insurance Company,» *Thirteenth International Conference on Digital Information Management (ICDIM)*, pp. 173-178, 2018.

[24] D. A. Pinto Galindo, «Diseño de un Modelo Predictivo de Fuga de Clientes Utilizando Algoritmos Machine Learning,» Universidad ECCI, Bogotá, 2020.

[25] J. D. Falla Arango, «Predicción de abandono de clientes en telecomunicaciones mediante el Aprendizaje Automático,» Universidad Jorge Tadeo Lozano, Bogotá, 2021.

[26] R. A. Barrueta Meza y E. J. P. Castillo Villarreal, «Modelo de análisis predictivo para determinar clientes con tendencia a la deserción en bancos peruanos,» Universidad Peruana de Ciencias Aplicadas (UPC), Lima, 2018.

[27] E. N. Cevallos Medina y C. J. Barahona Chunga, «Modelo para automatizar el proceso de predicción de la deserción en estudiantes universitarios en el primer año de estudio,» UNIVERSIDAD PERUANA DE CIENCIAS APLICADAS, Lima, 2021.

[28] O. Sifuentes Bitocchi, «Modelos predictivos de la deserción estudiantil en una universidad privada del Perú,» Universidad Nacional Mayor de San Marcos, Lima, 2018.

[29] M. I. Mejía Rocha y M. Colín Salgado, «GESTIÓN DEL CONOCIMIENTO Y SU IMPORTANCIA EN LAS ORGANIZACIONES,» *Revista TRILOGÍA,* nº 9, pp. 25-35, 2013.

[30] J. Lara, Minería de Datos, Madrid: Ediciones CEF, 2014.

[31] M. Perez, Minería de Datos a través de ejemplos, Ciudad de México: Editorial Alfaomega, 2015.

[32] J. L. Sánchez Ramírez, «Fundamentos de la Minería de Datos,» Chalco de Díaz Covarrubias, 2018.

[33] A. Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 2nd Edition, O'Reilly Media, Inc., 2019.

[34] K. Reyes, «Investigación en Ingeniería,» *Dossier académico,* 2020.

[35] A. C. Müller y S. Guido, Introduction to Machine Learning with Python: A Guide for Data Scientists, O'Reilly Media, Inc, 2016.

[36] J. C Riquelme, R. Ruiz y K. Gilbert, «Minería de Datos: Conceptos y Tendencias,» *Inteligencia Artificial. Revista Iberoamericana,* vol. X, nº 29, pp. 11-18, 2006.

[37] D. Verma y N. Rakesh, «Data Mining: Next Generation Challengues and Future Directions,» *International Journal of Modeling and Optimization,* vol. II, nº 5, 2012.

[38] S. Mitra y T. Achayra, Data mining: Multimedia, Soft Computing, and Bioinformatics, John Wiley & Sons, 2003.

[39] A. Burkov, The Hundred-Page Machine Learning Book, doi:10.1111/j.1468-0394.1988.tb00341.x, 2019.

VII. ANEXOS

ANEXO N° 01. CARTA DE ACEPTACIÓN DE LA ENTIDAD PARA LA EJECUCIÓN DEL PROYECTO

Lima, 30 de mayo del 2022

Alejandro Tadayuki Moritani Diaz
Commercial Manager B2B Mid-Market Segment
Telefónica del Perú S.A.A

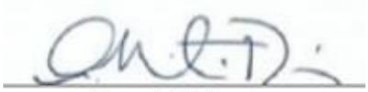
Presente.-

De mi especial consideración:

En referencia a la solicitud realizada por el estudiante MIRKO BRUNO VELA LÓPEZ identificado con DNI N° 73527613 y código universitario 171CV70519 de la ESCUELA DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN en la que se solicita facilidades para desarrollar y dar continuidad de su trabajo de investigación denominado "IMPLEMENTACIÓN DE UN MODELO DE MINERIA DE DATOS PARA PREDECIR LA DESERCIÓN DE LOS CLIENTES EN UNA EMPRESA DE TELECOMUNICACIONES" me es grato comunicar que **se acepta su petición**, solicitando exhortar al estudiante a coordinar previa y anticipadamente las fechas de entrega y puntos específicos necesarios para el desarrollo de su trabajo de investigación teniendo en consideración las restricciones establecidas por la empresa por la obligación de salvaguardar el secreto de las telecomunicaciones y a mantener la confidencialidad de los datos personales de sus abonados y usuarios de acuerdo con la Constitución Política del Perú y las normas legales aplicables.

Agradeciendo de antemano su atención a la presente, me despido expresando mi estima personal.

Atentamente.



Firma del Referente

Alejandro Tadayuki Moritani Diaz
Commercial Manager B2B Mid-Market Segment

Empresas

ANEXO N° 02. CONSTANCIA DE APROBACIÓN DEL PRODUCTO ACREDITABLE DE LA ENTIDAD DONDE SE EJECUTÓ LA TESIS

Lima, 20 de mayo 2022

Carlos Alberto Trigoso Valeriano
Analista de Datos – Ejecutivo en Movistar Empresas
Telefónica del Perú S.A.A

Presente. -

De mi especial consideración:

En referencia a la solicitud realizada por el estudiante MIRKO BRUNO VELA LÓPEZ identificado con DNI N° 73527613 y código universitario 171CV70519 de la ESCUELA DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN en la que solicita la evaluación del cumplimiento de su producto acreditable presentado en la tesis denominada "IMPLEMENTACIÓN DE UN MODELO DE MINERIA DE DATOS PARA PREDECIR LA DESERCIÓN DE LOS CLIENTES EN UNA EMPRESA DE TELECOMUNICACIONES" me es grato comunicar que **se aprueba el cumplimiento y funcionamiento del producto acreditable,** así como el correcto desempeño en base a la información de prueba empleada por el estudiante para realizar las predicciones de potenciales clientes desertores en base al comportamiento de los clientes. Se exhorta al estudiante a seguir desarrollando mecanismos que posibiliten la optimización de procesos de evaluación que puedan aplicarse en el contexto real de las telecomunicaciones.

Agradeciendo de antemano su atención a la presente, me despido expresando mi estima personal.

Atentamente.

Firma del referente

Carlos Alberto Trigoso Valeriano
Ejecutivo en Movistar Empresas

Empresas

ANEXO N° 03. GUIA DE ENTREVISTA: COMPRENSIÓN DEL CONTEXTO Y LAS VARIABLES DE ANÁLISIS

ENCUESTADO: Analista de Datos – Inteligencia Comercial

ENTREVISTADOR: Mirko Bruno Vela López

OBJETIVO: Conocer el contexto y las variables que se analizan para determinar la deserción de los clientes.

FECHA: 05-04-2021

PREGUNTAS:

1. ¿Qué variables consideran que influyen en la deserción de los clientes?
2. ¿Qué análisis se realizó para determinar dichas variables?
3. ¿Cuál consideran que es el principal problema al analizar la información del comportamiento de los clientes?
4. ¿Existe una segmentación definida en los clientes?
5. ¿Cuentan con data histórica de la deserción de los clientes?
6. ¿Qué porcentaje en la data histórica que manejan representa a clientes desertores?
7. ¿Manejan información completa en todas las variables de análisis del comportamiento de los clientes?
8. ¿En qué tipo de clientes se enfoca con mayor importancia el análisis de su potencial deserción?
9. ¿Cada cuánto tiempo realizan el análisis del comportamiento de los clientes para determinar su potencial deserción?
10. ¿Existe algún mecanismo tecnológico para analizar el comportamiento de los clientes para determinar los potenciales riesgos de deserción?
11. ¿Considera necesario conocer la probabilidad de deserción de los clientes?
12. ¿Cuánto tiempo toma evaluar a los clientes para determinar su potencial deserción con los procesos actuales?

ANEXO N° 04. MANUAL DE USUARIO

MANUAL DE USUARIO DE LA INTERFAZ FINAL

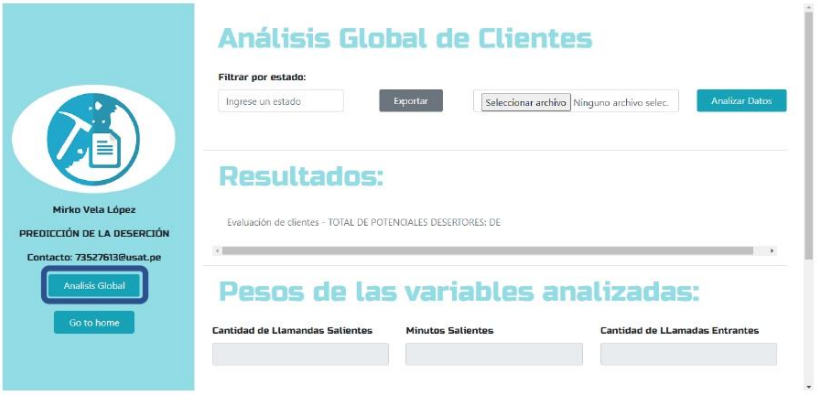
IMPLEMENTACIÓN DE UN MODELO DE MINERÍA DE DATOS PARA PREDECIR LA DESERCIÓN DE LOS CLIENTES EN UNA EMPRESA DE TELECOMUNICACIONES

AUTOR: MIRKO BRUNO VELA LÓPEZ

1. NAVEGACIÓN EN LA INTERFAZ FINAL

La interfaz final cuenta con dos secciones, la sección principal y que se carga al inicializar el sistema es la sección "Home", la cual, se emplea para realizar la predicción de la potencialidad de deserción de un cliente en específico.

La siguiente sección del sistema es "Análisis Global", la cual, se emplea para realizar la predicción de la potencialidad de deserción de un conjunto de clientes en simultaneo; así mismo, esta sección muestra el porcentaje de importancia de cada variable analizada para dar como resultado la predicción, es decir, e porcentaje de influencia de cada variable en el resultado predicho. Para acceder a este apartado se requiere dar clic en el botón "Análisis Global" situado en la barra lateral izquierda.

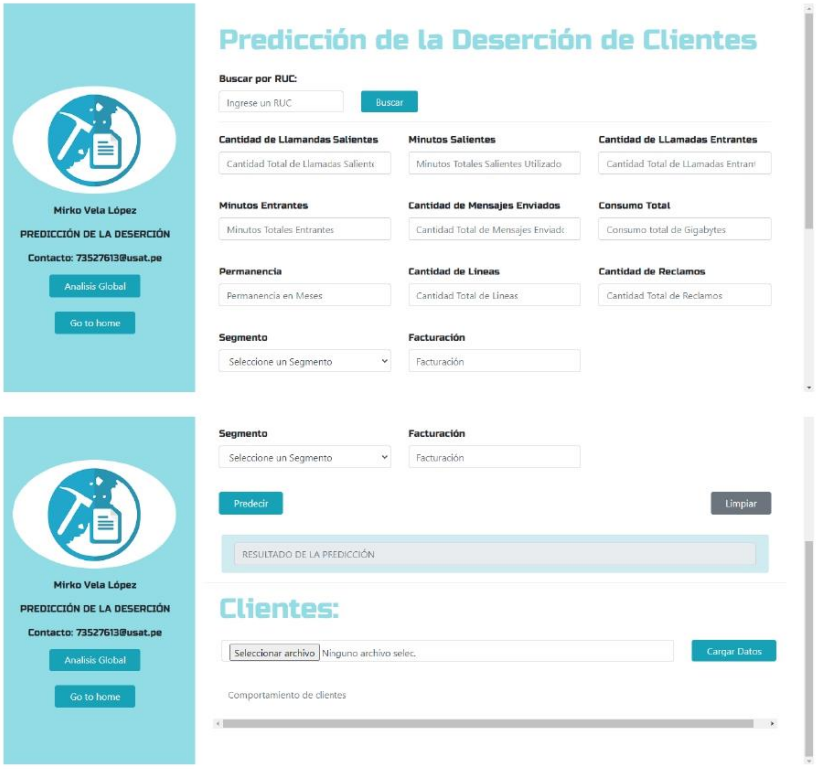


Para regresar al apartado "Home" únicamente se requiere dar clic en el botón "Go to home".

2. FUNCIONAMIENTO DEL APARTADO HOME

a. Visualización inicial

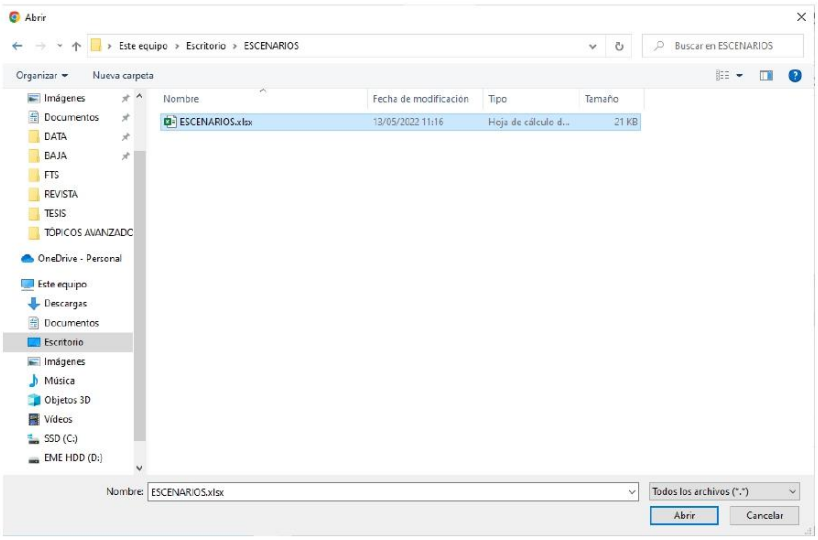
En el apartado Home, se puede apreciar inicialmente las variables requeridas para el análisis de los clientes, dichas variables definen el comportamiento del cliente dentro de la organización y son las empleadas por el equipo de inteligencia comercial para evaluar a los clientes entorno a su potencial deserción. Como se puede apreciar, esta interfaz inicial cuenta con un apartado para cargar los datos del cliente en formato Excel y ser empleados como base para la predicción (Los campos que debe contener la base en Excel pueden apreciarse en los Anexos). En este primer apartado, se realiza la predicción de la potencialidad de deserción de un único cliente seleccionado en base a su probabilidad de deserción y dicho resultado aparecerá en la casilla de "RESULTADO DE LA PREDICCIÓN" como se verá en las secciones siguientes.



b. Cargar la data

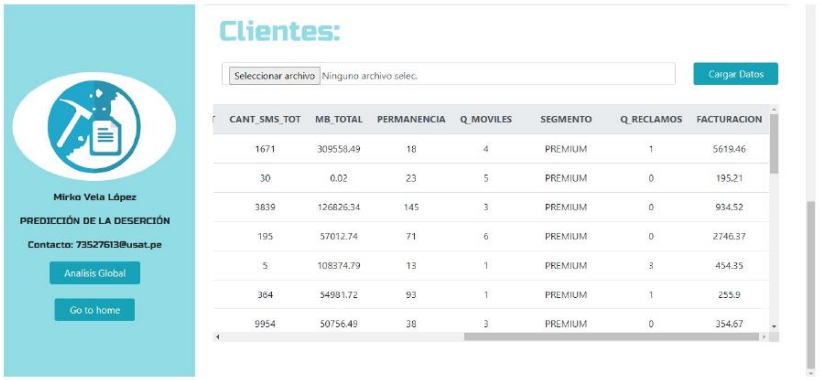
Para realizar la carga de la data se debe dar clic en el botón "seleccionar archivo".

Una vez presionado el botón "Seleccionar archivo", se abrirá el panel de archivos local y se deberá seleccionar la base de datos a emplear para el análisis predictivo.



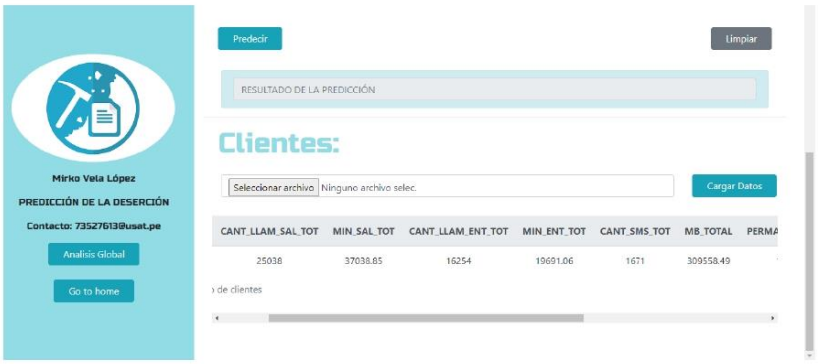
Una vez seleccionada la base de datos en Excel a emplear, se almacenará automáticamente en la interfaz y aparecerá el nombre del archivo seleccionado. Una vez seleccionada la base de datos, se procede a dar clic en el botón cargar datos, para visualizar la información almacenada en la interfaz.

La interfaz mostrará una visualización en formato tabla de los datos proporcionados en la fuente Excel.

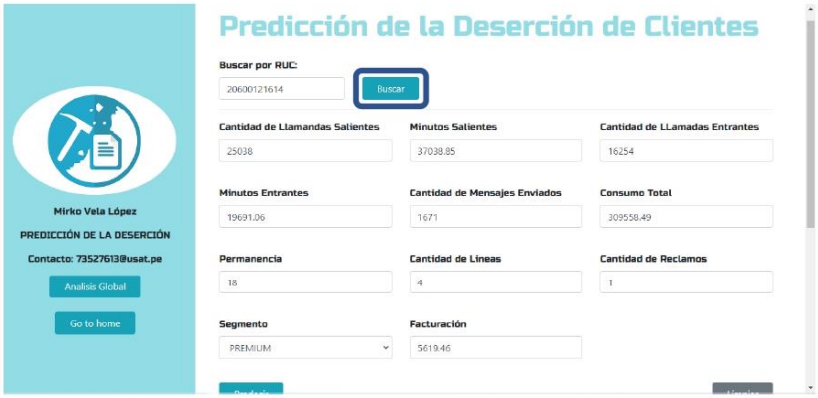


Una vez cargada la data, el usuario puede identificar al cliente sobre el cual desea realizar la evaluación y predicción; una vez identificado se debe digitar el numero de RUC (identificador único de cliente) en el apartado de "Buscar por RUC" (Por protección de los datos del cliente, para este ejemplo se empleará un RUC ficticio).

Una vez digitado el RUC del cliente que se desea evaluar, la tabla se filtrará con la información de ese único cliente.

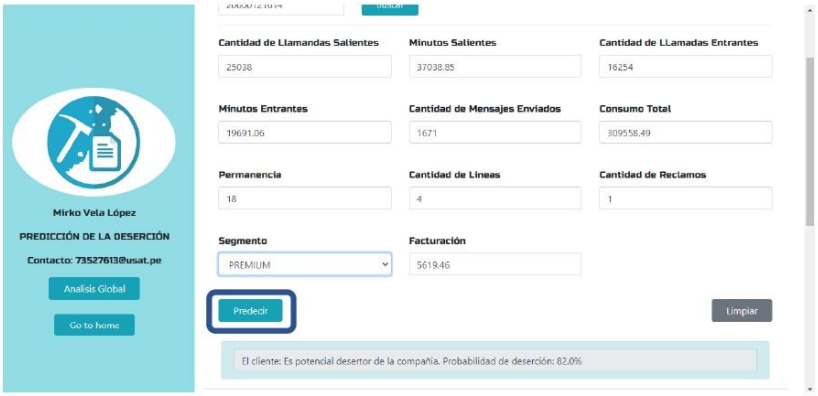


Para realizar la carga de los datos en el formulario de evaluación previo a realizar la predicción, se debe dar clic en el botón "Buscar" y automáticamente el formulario se llenará con los datos de la tabla.



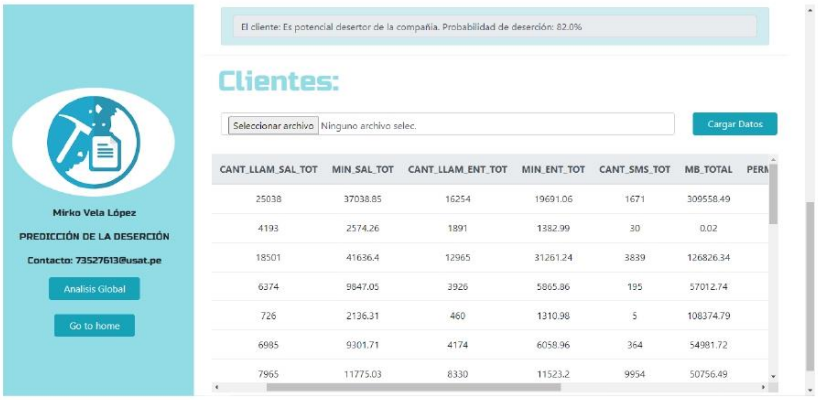
c. Ejecutar la predicción

Una vez cargada la data en el formulario, para realizar la evaluación y predicción de la potencialidad de deserción del cliente seleccionado, se debe dar clic en el botón "Predecir" y se ejecutará la predicción, indicando si el cliente posee o no riesgo a desertar y el porcentaje de probabilidad de deserción (El modelo evalúa como posible desertor a todo cliente que supere el 50% de riesgo de deserción).



Como se puede apreciar, el cliente evaluado fue catalogado por el modelo predictivo como un potencial desertor de la compañía con una probabilidad de deserción del 82.0%

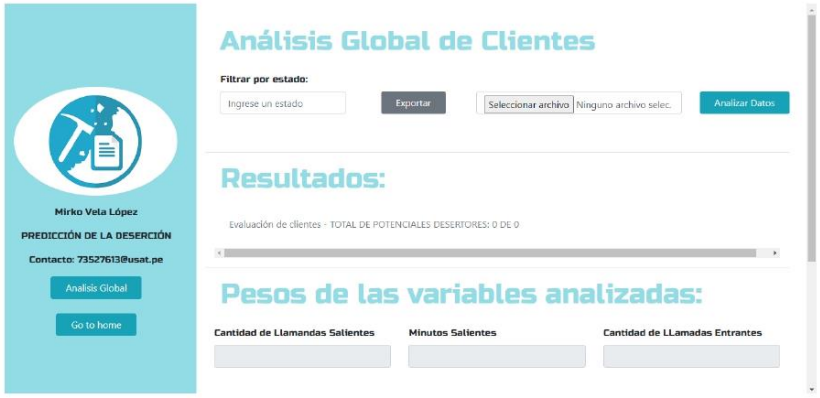
Al realizar la predicción la visualización de los clientes en la tabla del apartado de “Clientes” en la interfaz vuelve a mostrar la totalidad de clientes cargados para volver a realizar el proceso; para ello, previamente se emplea el botón “Limpiar” para borrar los datos del formulario y poder realizar una nueva predicción.



3. FUNCIONAMIENTO DEL APARTADO ANALISIS GLOBAL

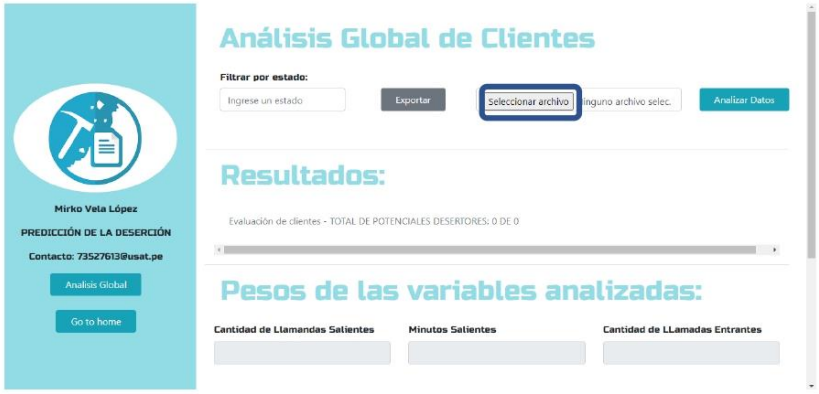
a. Visualización inicial

En el apartado Análisis Global, se puede apreciar inicialmente el apartado de carga para la base de datos en Excel a emplear para evaluar y predecir la potencialidad de deserción de los cliente (Los campos que debe contener la base en Excel pueden apreciarse en los Anexos), el apartado de "Resultados" en dónde se mostrarán los registros cargados desde la fuente Excel, así como los resultados de la predicción en termino de clasificación en "Potencial Desertor" o "Sin Riesgo a Desertar" y su porcentaje de probabilidad de deserción; finalmente en el apartado de "Pesos de las variables analizadas" se brindará el porcentaje de importancia de cada variable analizada para dar como resultado la predicción, es decir, el porcentaje de influencia de cada variable en el resultado predicho.

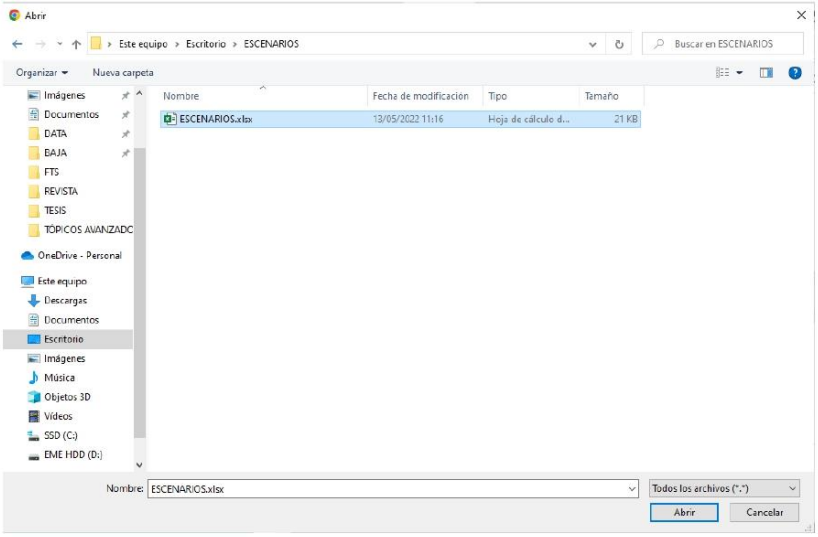


b. Cargar la data

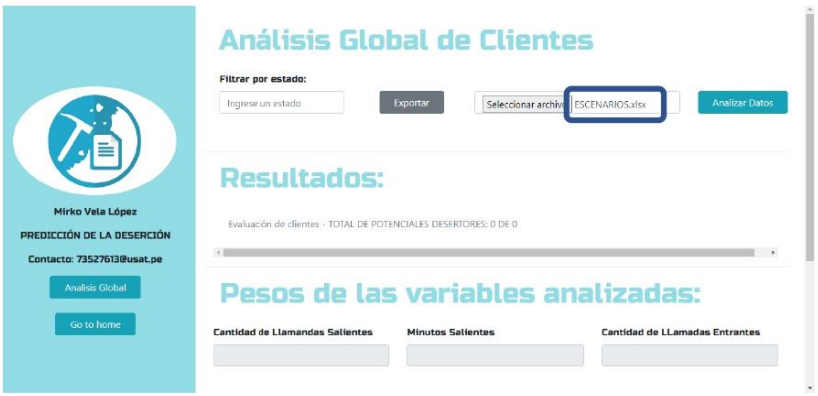
Para realizar la carga de la data se debe dar clic en el botón "seleccionar archivo".



Una vez presionado el botón "Seleccionar archivo", se abrirá el panel de archivos local y se deberá seleccionar la base de datos a emplear para el análisis predictivo.



Una vez seleccionada la base de datos en Excel a emplear, se almacenará automáticamente en la interfaz y aparecerá el nombre del archivo seleccionado.



c. Ejecutar la predicción

Una vez cargada la data en el formulario, para realizar la evaluación y predicción de la potencialidad de deserción de los clientes brindados en la base de datos Excel seleccionada, se debe dar clic en el botón "Analizar Datos" y se ejecutará la el modelo, indicando la potencialidad de deserción de cada cliente brindado en la fuente de datos en términos de "Potencial Desertor" o "Sin Riesgo a Desertar", así mismo, se brindará el porcentaje de probabilidad de deserción (El modelo evalúa como posible desertor a todo cliente que supere el 50% de riesgo de deserción); además, en el aparado de "Pesos de las variables analizadas" se visualizará el porcentaje de importancia de cada variable con respecto a la predicción, es decir, el porcentaje de influencia de cada variable en el resultado predicho (Este valor porcentual es igual para la evaluación de cada cliente).

Además, la tabla de Resultados muestra la cantidad de clientes identificados como potenciales desertores del total de clientes analizados.

4. CONLUSIONES

Esta interfaz final permite al usuario encargado de evaluar a los potenciales clientes desertores hacer uso del modelo predictivo desarrollado y obtener resultados con un alto índice de precisión al determinar a los potenciales clientes desertores, brindándole la capacidad de analizar a los clientes de manera específica y global, además de brindarle un porcentaje de probabilidad de deserción que le permita centrarse en los clientes con el porcentaje de probabilidad de deserción dentro del rango establecido como señal de emergencia por la organización. Así mismo, la interfaz le permite conocer el grado de importancia de las variables con respecto a la predicción, para que el usuario final pueda tener en cuenta cuales variables implican mayor relevancia para determinar la predicción. Finalmente, al contrastar los resultados obtenidos por el análisis tradicional, con los resultados precisos del modelo predictivo, se puede reducir la lista de clientes identificados como potenciales desertores y de esta forma guiar la dirección de estrategias de fidelización a los clientes que lo requieran con mayor urgencia (Clientes con riesgo a desertar alto).

5. ANEXOS

El archivo Excel brindado como base de datos para la interfaz final debe tener la siguiente estructura sin varear el orden de las variables (Dicha estructura es la empleada por el área comercial para el análisis).

***Los RUC's fueron censurados por protección a la información del cliente.**

RUC	CANT_LLAM_SAL_TOT	MIN_SAL_TOT	CANT_LLAM_ENT_TOT	MIN_ENT_TOT	CANT_SMS_TOT	MB_TOTAL	PERMANENCIA	O_MOVILES	SEGMENTO	O_RECLAMOS	FACTURACION
2.06E+10	25038	37038.85	16254	19691.06	1671	309558.49	18	4	PREMIUM	1	5619.46
2.0482E+10	4198	2574.26	1891	1382.99	80	0.02	23	5	PREMIUM	0	195.21
2.0528E+10	18501	41636.4	12965	31261.24	3839	126826.34	145	3	PREMIUM	0	934.52
2.0449E+10	6974	9847.05	3926	5865.86	195	57012.74	71	6	PREMIUM	0	2746.37
2.0603E+10	726	2138.31	480	1810.98	5	108874.79	13	1	PREMIUM	3	454.35
2.0542E+10	6985	9301.71	4174	6058.96	364	54981.72	93	1	PREMIUM	1	255.9
2.0522E+10	7965	11775.08	8330	11523.2	9954	50756.49	38	3	PREMIUM	0	354.67
2.0352E+10	12705	31556.55	9351	18092.9	3435	96173.89	223	6	PREMIUM	0	756.72
2.0111E+10	5619	7486.73	2835	4776.21	116	4948.51	98	4	PREMIUM	3	812.51
2.0582E+10	8157	9282.72	10055	11172.85	1009	57081.45	186	3	PREMIUM	1	1366.8
2.0486E+10	693	938.71	444	695.85	1	0	126	1	PREMIUM	0	79.65
2.0602E+10	1857	4708.44	660	1458.81	809	78356.76	13	10	PREMIUM	3	5512.86
2.06E+10	2669	11547.48	2514	15690.13	268	89372.06	93	2	PREMIUM	1	383.09
2.0545E+10	3240	3754.81	7116	9270.47	234	16462.67	89	1	PREMIUM	0	1165.96
2.0455E+10	2226	3015.74	1891	2587.29	132	29542.6	125	3	PREMIUM	5	950.53
2.0487E+10	8279	19227.5	3421	7643.56	1227	249927.12	8	4	PREMIUM	3	1650.46
2.0524E+10	3107	4940.26	3692	4006.06	265	87087.77	29	1	PREMIUM	4	389.57
2.06E+10	13926	21334.65	9623	14768.47	1173	67321.26	144	3	PREMIUM	0	287.08
2.0549E+10	115	142.37	12	9.5	5	34156.59	29	5	PREMIUM	0	41.52
2.0185E+10	2023	2169.04	1824	2314.43	71	4005.9	37	2	PREMIUM	0	118.63
2.0508E+10	505	949.68	156	501.24	3	5917.36	206	2	PREMIUM	1	3889027
2.0584E+10	19822	41885.18	17550	40525.57	3858	147109.68	118	4	PREMIUM	18	25618461
2.06E+10	4679	9375.42	4506	10928.87	281	44578.29	51	1	PREMIUM	3	563.89

ANEXO N° 05. PRUEBAS DE CAJA BLANCA

DESCRIPCIÓN DE PRUEBAS DE CAJA BLANCA		
Requisito		
Módulo / Área Funcional / Subproceso ´Predicción	**Tipo de requisito** Funcional	**Código de historia de usuario** HU-10
Descripción del requisito Validar si todas y cada una de las líneas del código se ejecutan al menos una vez.		
Caso de prueba		
Código de prueba PCN10	**Caso de prueba** Predicción	**Fecha de prueba** 20/11/2021
Funcionalidad / Característica a evaluar Cobertura de la declaración.	**Datos de entrada / Acciones de entrada** Código fuente.	**Resultado esperado** Garantizar que se ejecuten todas y cada una de las líneas del código al menos una vez.
Requerimientos de ambiente de pruebas Equipo: Procesador: 1.0 GHz Memoria: 256 Mb RAM Navegador web (Todos los navegadores, sin embargo, se recomiendan aquellos basados en Chromium) Conexión a internet: 5Mbps como mínimo de velocidad.		**Condiciones / Restricciones** Ninguna.
Seguimiento		
Resultado obtenido La ejecución de todas y cada una de las líneas de código al menos una vez se ha realizado exitosamente.	**Estado actual** Conforme	**Observaciones** Ninguna
Correcciones		
Fecha de cambio de estado	**Observaciones**	

DESCRIPCIÓN DE PRUEBAS DE CAJA BLANCA		
Requisito		
Módulo / Área Funcional / Subproceso ´Estructura General	**Tipo de requisito** Funcional	**Código de historia de usuario** HU-20
Descripción del requisito Verificar que cada unidad de la estructura funciona correctamente. (Módulo de predicción, API, interfaz)		
Caso de prueba		
Código de prueba PCN20	**Caso de prueba** Estructura general.	**Fecha de prueba** 20/11/2021
Funcionalidad / Característica a evaluar Prueba unitaria.	**Datos de entrada / Acciones de entrada** Código fuente.	**Resultado esperado** Garantizar que cada unidad estructural funcione correctamente de forma independiente.
Requerimientos de ambiente de pruebas Equipo: Procesador: 1.0 GHz Memoria: 256 Mb RAM Navegador web (Todos los navegadores, sin embargo, se recomiendan aquellos basados en Chromium) Conexión a internet: 5Mbps como mínimo de velocidad.		**Condiciones / Restricciones** Ninguna.
Seguimiento		
Resultado obtenido La ejecución correcta de todas las unidades estructurales se ha realizado exitosamente.	**Estado actual** Conforme	**Observaciones** Ninguna
Correcciones		
Fecha de cambio de estado	**Observaciones**	

DESCRIPCIÓN DE PRUEBAS DE CAJA BLANCA		
Requisito		
Módulo / Área Funcional / Subproceso ´Estructura General	**Tipo de requisito** Funcional	**Código de historia de usuario** HU-30
Descripción del requisito Verificar que la combinación de unidades funcione correctamente como grupo. (Módulo de predicción, API, interfaz)		
Caso de prueba		
Código de prueba PCN30	**Caso de prueba** Estructura general.	**Fecha de prueba** 20/11/2021
Funcionalidad / Característica a evaluar Prueba de integración.	**Datos de entrada / Acciones de entrada** Código fuente.	**Resultado esperado** Garantizar que todas las unidades estructurales funcionen correctamente de forma integral sin exponer ningún tipo de error en la interacción de componentes.
Requerimientos de ambiente de pruebas Equipo: Procesador: 1.0 GHz Memoria: 256 Mb RAM Navegador web (Todos los navegadores, sin embargo, se recomiendan aquellos basados en Chromium) Conexión a internet: 5Mbps como mínimo de velocidad.		**Condiciones / Restricciones** Ninguna.
Seguimiento		
Resultado obtenido La ejecución correcta de todas las unidades estructurales de forma integral se ha realizado exitosamente.	**Estado actual** Conforme	**Observaciones** Ninguna
Correcciones		
Fecha de cambio de estado	**Observaciones**	

DESCRIPCIÓN DE PRUEBAS DE CAJA BLANCA			
Requisito			
Módulo / Área Funcional / Subproceso API	**Tipo de requisito** Funcional	**Código de historia de usuario** HU-40	
Descripción del requisito Validar si cada sucursal se ejecuta al menos una vez.			
Caso de prueba			
Código de prueba PCN40	**Caso de prueba** API	**Fecha de prueba** 20/11/2021	
Funcionalidad / Característica a evaluar Cobertura de sucursales.	**Datos de entrada / Acciones de entrada** Código fuente..	**Resultado esperado** Garantizar que se ejecuten todas y cada una de las ramas desde cada punto de decisión.	
Requerimientos de ambiente de pruebas Equipo: Procesador: 1.0 GHz Memoria: 256 Mb RAM Navegador web (Todos los navegadores, sin embargo, se recomiendan aquellos basados en Chromium) Conexión a internet: 5Mbps como mínimo de velocidad.		**Condiciones / Restricciones** Ninguna.	
Seguimiento			
Resultado obtenido La ejecución de todas y cada una de las ramas desde cada punto de decisión se ha realizado exitosamente.	**Estado actual** Conforme	**Observaciones** Ninguna	
Correcciones			
Fecha de cambio de estado	**Observaciones**		

DESCRIPCIÓN DE PRUEBAS DE CAJA BLANCA		
Requisito		
Módulo / Área Funcional / Subproceso ´Predicción	**Tipo de requisito** Funcional	**Código de historia de usuario** HU-50
Descripción del requisito El sistema cuenta con un procedimiento que permite predecir la posibilidad de deserción del cliente evaluado, habiendo cargado primero la información comportamiento del cliente (Cantidad de Llamadas Salientes, Minutos Salientes, Cantidad de Llamadas Entrantes, Minutos Entrantes, Cantidad de Mensajes Enviados, Consumo Total, Permanencia, Cantidad de Líneas, Cantidad de Reclamos, Segmento y Facturación).		
Caso de prueba		
Código de prueba PCN50	**Caso de prueba** Estructura de predicción	**Fecha de prueba** 20/11/2021
Funcionalidad / Característica a evaluar Predecir la posibilidad de deserción del cliente.	**Datos de entrada / Acciones de entrada** Comportamiento del cliente.	**Resultado esperado** Correctas validaciones de los campos de entrada y ejecución de la estructura de predicción.
Requerimientos de ambiente de pruebas Equipo: Procesador: 1.0 GHz Memoria: 256 Mb RAM Navegador web (Todos los navegadores, sin embargo, se recomiendan aquellos basados en Chromium) Conexión a internet: 5Mbps como mínimo de velocidad.	**Condiciones / Restricciones** Ninguna.	
Seguimiento		
Resultado obtenido La validación de los campos de entrada y todo el proceso de predicción se ejecuta exitosamente.	**Estado actual** Conforme	**Observaciones** Ninguna
Correcciones		
Fecha de cambio de estado	**Observaciones**	

ANEXO N° 06. PRUEBAS DE CAJA NEGRA

DESCRIPCIÓN DE PRUEBAS DE CAJA NEGRA			
Requisito			
Módulo / Área Funcional / Subproceso ´API	**Tipo de requisito** Funcional	**Código de historia de usuario** HU-10	
Descripción del requisito Validar la conexión de la API al modelo predictivo.			
Caso de prueba			
Código de prueba PCN10	**Caso de prueba** API	**Fecha de prueba** 20/11/2021	
Funcionalidad / Característica a evaluar Peticiones de la API al modelo predictivo.	**Datos de entrada / Acciones de entrada** Comportamiento del cliente.	**Resultado esperado** La API logra comunicarse con el modelo predictivo y predice el estado actual del cliente en términos de su probabilidad de deserción.	
Requerimientos de ambiente de pruebas Equipo: Procesador: 1.0 GHz Memoria: 256 Mb RAM Navegador web (Todos los navegadores, sin embargo, se recomiendan aquellos basados en Chromium) Conexión a internet: 5Mbps como mínimo de velocidad.		**Condiciones / Restricciones** Ninguna.	
Seguimiento			
Resultado obtenido La comunicación entre la API y el modelo predictivo se ha realizado exitosamente.	**Estado actual** Conforme	**Observaciones** Ninguna	
Correcciones			
Fecha de cambio de estado	**Observaciones**		

DESCRIPCIÓN DE PRUEBAS DE CAJA NEGRA		
Requisito		
Módulo / Área Funcional / Subproceso ´Predicción	**Tipo de requisito** Funcional	**Código de historia de usuario** HU-20
Descripción del requisito Validar la data permitida como entrada (Valores netamente numéricos).		
Caso de prueba		
Código de prueba PCN20	**Caso de prueba** Predicción	**Fecha de prueba** 20/11/2021
Funcionalidad / Característica a evaluar Bloqueo del ingreso de datos no permitidos.	**Datos de entrada / Acciones de entrada** Comportamiento del cliente.	**Resultado esperado** La interfaz no permite el ingreso de valores que no sean de carácter numérico. (Muestra mensajes de alerta y deniega el ingreso)
Requerimientos de ambiente de pruebas Equipo: Procesador: 1.0 GHz Memoria: 256 Mb RAM Navegador web (Todos los navegadores, sin embargo, se recomiendan aquellos basados en Chromium) Conexión a internet: 5Mbps como mínimo de velocidad.		**Condiciones / Restricciones** Ninguna.
Seguimiento		
Resultado obtenido El bloqueo del ingreso de entradas de carácter no numérico se ha realizado exitosamente.	**Estado actual** Conforme	**Observaciones** La interfaz no debe permitir el ingreso de entradas de carácter no numérico que generen problemas en el modelo predictivo.
Correcciones		
Fecha de cambio de estado	**Observaciones**	

DESCRIPCIÓN DE PRUEBAS DE CAJA NEGRA		
Requisito		
Módulo / Área Funcional / Subproceso ´Predicción	**Tipo de requisito** Funcional	**Código de historia de usuario** HU-30
Descripción del requisito Validar el ingreso de campos vacíos como datos de entrada.		
Caso de prueba		
Código de prueba PCN30	**Caso de prueba** Predicción	**Fecha de prueba** 20/11/2021
Funcionalidad / Característica a evaluar Bloqueo del ingreso de campos vacíos.	**Datos de entrada / Acciones de entrada** Comportamiento del cliente.	**Resultado esperado** La interfaz no permite el ingreso campos vacíos. (Muestra mensajes de alerta)
Requerimientos de ambiente de pruebas Equipo: Procesador: 1.0 GHz Memoria: 256 Mb RAM Navegador web (Todos los navegadores, sin embargo, se recomiendan aquellos basados en Chromium) Conexión a internet: 5Mbps como mínimo de velocidad.	**Condiciones / Restricciones** Ninguna.	
Seguimiento		
Resultado obtenido El bloqueo del ingreso de campos vacíos se ha realizado exitosamente.	**Estado actual** Conforme	**Observaciones** La interfaz no debe permitir el ingreso de campos vacíos que generen problemas en el modelo predictivo.
Correcciones		
Fecha de cambio de estado	**Observaciones**	

DESCRIPCIÓN DE PRUEBAS DE CAJA NEGRA			
Requisito			
Módulo / Área Funcional / Subproceso ´Predicción	**Tipo de requisito** Funcional	**Código de historia de usuario** HU-40	
Descripción del requisito Predecir la posibilidad de deserción del cliente evaluado.			
Caso de prueba			
Código de prueba PCN40	**Caso de prueba** Predicción	**Fecha de prueba** 20/11/2021	
Funcionalidad / Característica a evaluar Predicción de la posibilidad de deserción.	**Datos de entrada / Acciones de entrada** Comportamiento del cliente.	**Resultado esperado** La interfaz final predice el estado actual del cliente en términos de su probabilidad de deserción.	
Requerimientos de ambiente de pruebas Equipo: Procesador: 1.0 GHz Memoria: 256 Mb RAM Navegador web (Todos los navegadores, sin embargo, se recomiendan aquellos basados en Chromium) Conexión a internet: 5Mbps como mínimo de velocidad.		**Condiciones / Restricciones** Ninguna.	
Seguimiento			
Resultado obtenido La predicción de la posibilidad de deserción del cliente se ha realizado exitosamente.	**Estado actual** Conforme	**Observaciones** Ninguna	
Correcciones			
Fecha de cambio de estado	**Observaciones**		

DESCRIPCIÓN DE PRUEBAS DE CAJA NEGRA		
Requisito		
Módulo / Área Funcional / Subproceso ´Predicción	**Tipo de requisito** Funcional	**Código de historia de usuario** HU-50
Descripción del requisito Cargar la data del comportamiento del cliente para poder emplearse en la interfaz.		
Caso de prueba		
Código de prueba PCN50	**Caso de prueba** Predicción	**Fecha de prueba** 20/11/2021
Funcionalidad / Característica a evaluar Cargar la data del comportamiento del cliente	**Datos de entrada / Acciones de entrada** Comportamiento del cliente.	**Resultado esperado** La interfaz permite cargar la data del comportamiento del cliente para ser empleada por el módulo de predicción.
Requerimientos de ambiente de pruebas Equipo: Procesador: 1.0 GHz Memoria: 256 Mb RAM Navegador web (Todos los navegadores, sin embargo, se recomiendan aquellos basados en Chromium) Conexión a internet: 5Mbps como mínimo de velocidad.		**Condiciones / Restricciones** Ninguna.
Seguimiento		
Resultado obtenido La carga de la data del comportamiento del cliente a la interfaz se ha realizado exitosamente.	**Estado actual** Conforme	**Observaciones** La data cargada debe tener el formato específico que usa la compañía para consolidar la data del comportamiento del cliente.
Correcciones		
Fecha de cambio de estado	**Observaciones**	