

**FACULTAD DE INGENIERÍA**

**CARRERA: INGENIERÍA INFORMÁTICA ADMINISTRATIVA**

**MODALIDAD DE GRADUACIÓN**

**PROYECTO DE GRADO**

**MODELO DE APRENDIZAJE AUTOMÁTICO PARA LA PREDICCIÓN DEL ABANDONO DE CLIENTES EN ENTIDADES BANCARIAS**

**ENRIQUE ARCE ALEJANDRO**

**Santa Cruz de la sierra – Bolivia**

**2023**

Logotipo, nombre de la empresa

Descripción generada automáticamente

**FACULTAD DE INGENIERÍA**

**CARRERA: INGENIERÍA INFORMÁTICA ADMINISTRATIVA**

**MODALIDAD DE GRADUACIÓN**

**PROYECTO DE GRADO**

**MODELO DE APRENDIZAJE AUTOMÁTICO PARA LA PREDICCIÓN DEL ABANDONO DE CLIENTES EN ENTIDADES BANCARIAS**

**Proyecto de Grado para optar al título de**

**Licenciado(a) en Ingeniería Informática Administrativa**

**ENRIQUE ARCE ALEJANDRO**

**Reg.: 2019112108**

**Santa Cruz de la sierra – Bolivia**

**2023**

**AGRADECIMIENTOS**

Agradecer primeramente a Dios, por haberme sostenido y levantado en los momentos más difíciles, él es quien ha sido mi fortaleza y mi ayuda indispensable durante en esta etapa de mi vida, a él sea toda la gloria y honra.

**ABSTRACT**

|  |  |
| --- | --- |
| **TITULO** | MODELO DE APRENDIZAJE AUTOMÁTICO PARA LA PREDICCIÓN DEL ABANDONO DE CLIENTES EN ENTIDADES BANCARIA |
| **AUTOR** | ENRIQUE ARCE ALEJANDRO |

**PROBLEMÁTICA**

La entidad financiera no capaz de prevenir a tiempo el abandono de un cliente de la empresa; esta situación hace que los clientes sean irregulares, que los ingresos disminuyan o que la reputación de la entidad bancaria quede en duda.

**OBJETIVO GENERAL**

Hacer uso de un modelo de aprendizaje automático que sea capaz de predecir el abandono de clientes en una entidad financiera.

**CONTENIDO**

El presente trabajo presenta el siguiente contenido:

- Introducción al proyecto

- Análisis de Requisitos

- Análisis y Diseño

- Implementación

- Seguridad

- Pruebas

- Conclusión y Recomendaciones

- Bibliografía

|  |  |
| --- | --- |
| **CARRERA:** | Ingeniería Informática Administrativa |
| **GUÍA:** | Dr. Javier Alanoca |
| **DESCRIPTORES:** | Modelo de Aprendizaje Automático, Análisis de Datos |
| **CORREO:** | [ingenriquearce@gmail.com](mailto:ingenriquearce@gmail.com) |
| **FECHA:** | Santa Cruz de la sierra, 2023 |

**TABLA DE CONTENIDO**

[1. CAPÍTULO 1 - DEFINICIÓN DEL PROYECTO DE INVESTIGACIÓN 10](#_Toc138083261)

[1.1. DEFINICIÓN DEL PROBLEMA 10](#_Toc138083262)

[1.1.1. Descripción del problema 10](#_Toc138083263)

[1.1.2. Situación Problemática 10](#_Toc138083264)

[1.1.3. Situación Deseada 11](#_Toc138083265)

[1.2. OBJETO DE LA INVESTIGACIÓN 11](#_Toc138083266)

[1.3. OBJETIVOS 12](#_Toc138083267)

[1.3.1. Objetivo General 12](#_Toc138083268)

[1.3.2. Objetivos Específicos 12](#_Toc138083269)

[1.4. METODOLOGÍA 13](#_Toc138083270)

[1.5. ALCANCE 13](#_Toc138083271)

[2. CAPÍTULO 2 – DESCRIPCIÓN DEL NEGOCIO 14](#_Toc138083272)

[2.1. DESCRIPCIÓN DEL RUBRO 14](#_Toc138083273)

[2.2. DESCRIPCIÓN DEL MERCADO DEL RUBRO 14](#_Toc138083274)

[2.3. DESCRIPCIÓN DE LOS SERVICIOS Y PRODUCTOS DE LAS ENTIDADES BANCARIAS EN BOLIVIA 14](#_Toc138083275)

[3. Capítulo 3 - Aprendizaje automático para la predicción de abandono de clientes 16](#_Toc138083276)

[3.1. APRENDIZAJE AUTOMÁTICO 16](#_Toc138083277)

[3.2. CARACTERÍSTICAS DE UN SOFTWARE QUE UTILIZA APRENDIZAJE AUTOMÁTICO 16](#_Toc138083278)

[3.3. TIPOS DE MODELOS DE APRENDIZAJE AUTOMÁTICO 17](#_Toc138083279)

[3.4. MODELO DE APRENDIZAJE AUTOMÁTICO ESPECÍFICO PARA LA PREDICCIÓN DE ABANDONO DE CLIENTES 18](#_Toc138083280)

[3.5. METODOLOGÍA CRISP-DM 18](#_Toc138083281)

[3.6. HERRAMIENTAS 20](#_Toc138083282)

[3.6.1. Python 20](#_Toc138083283)

[3.6.2. Scikit-learn 20](#_Toc138083284)

[3.6.3. Google Colab 20](#_Toc138083285)

[4. Capítulo 4 - Análisis de requisitos 20](#_Toc138083286)

[4.1. Introducción 20](#_Toc138083287)

[4.1.1. Propósito 20](#_Toc138083288)

[4.1.2. Definiciones, abreviaturas y acrónimos 21](#_Toc138083289)

[4.1.3. Visión general de la especificación 22](#_Toc138083290)

[4.2. Descripción general del modelo 22](#_Toc138083291)

[4.2.1. Perspectiva del producto 22](#_Toc138083292)

[4.2.2. Funciones del producto 22](#_Toc138083293)

[4.2.3. Características de los usuarios 22](#_Toc138083294)

[4.2.4. Suposiciones y dependencias 24](#_Toc138083295)

[4.3. Especificación de requisitos 24](#_Toc138083296)

[4.3.1. Interfaces externas 24](#_Toc138083297)

[4.3.2. Requisitos funcionales 25](#_Toc138083298)

[4.3.3. Requisitos no funcionales 25](#_Toc138083299)

[4.3.4. Restricciones del modelo 26](#_Toc138083300)

[4.3.5. Atributos del modelo 26](#_Toc138083301)

[4.3.6. MODELO DE DOMINIO 27](#_Toc138083302)

[4.3.7. Diagrama de estados de transición 28](#_Toc138083303)

[5. CAPÍTULO 5 – ANÁLISIS Y DISEÑO 28](#_Toc138083304)

[5.2. DICCIONARIO DE DATOS 29](#_Toc138083305)

[5.3. DIAGRAMA DE DESPLIEGUE 30](#_Toc138083306)

[6. CAPÍTULO 6 – IMPLEMENTACIÓN 31](#_Toc138083307)

[6.1. ANÁLISIS DE ELECCIÓN DE LA HERRAMIENTA 31](#_Toc138083308)

[6.2. ANÁLISIS DE LAS TECNOLOGÍAS UTILIZADAS 31](#_Toc138083309)

[6.2.1. Tecnologías 31](#_Toc138083310)

[6.2.2. Base de Datos 31](#_Toc138083311)

[6.3. ENTRENAMIENTO DEL MODELO 31](#_Toc138083312)

[6.3.1. Definición del modelo de entrenamiento 31](#_Toc138083313)

[6.3.2. Definición de intentos 31](#_Toc138083314)

[6.3.3. Definición de entidades 31](#_Toc138083315)

[7. CAPÍTULO 7 – SEGURIDAD 31](#_Toc138083316)

[7.1. INTRODUCCIÓN 31](#_Toc138083317)

[7.2. INTEGRIDAD 31](#_Toc138083318)

[7.3. ROLES 31](#_Toc138083319)

[7.4. DISPONIBILIDAD 31](#_Toc138083320)

[7.5. CONFIABILIDAD 31](#_Toc138083321)

[8. CAPÍTULO 8 – PRUEBAS 31](#_Toc138083322)

[8.1. ESTRATEGIA DE PRUEBAS 31](#_Toc138083323)

[8.1.1. Pruebas de funcionalidad 31](#_Toc138083324)

[8.2. RESULTADOS PRUEBAS FUNCIONALES 31](#_Toc138083325)

[CONCLUSIONES 31](#_Toc138083326)

[RECOMENDACIONES 31](#_Toc138083327)

[BIBLIOGRAFÍA 31](#_Toc138083328)

**TABLA DE ILUSTRACIONES**

[Ilustración 1. Representación de la situación deseada. 2](#_Toc145687264)

[Ilustración 2. Representación del objeto de investigación 3](#_Toc145687265)

[Ilustración 3. Diagrama de estados de transición 19](#_Toc145687266)

[Ilustración 4. Diagrama de caso de uso - Gestionar Resultado 21](#_Toc145687267)

[Ilustración 5. Diagrama de Clases 21](#_Toc145687268)

[Ilustración 6. Diccionario de datos 22](#_Toc145687269)

[Ilustración 7. Diagrama de despliegue 22](#_Toc145687270)

[Ilustración 8. Implementación de la selección de modelo 29](#_Toc145687271)

[Ilustración 9. Precisión de los modelos 30](#_Toc145687272)

[Ilustración 10. Implementación de la visualización de los resultados del modelo 31](#_Toc145687273)

[Ilustración 11. Resultados del modelo 31](#_Toc145687274)

**PARTE I**

**INTRODUCCIÓN AL PROYECTO**

# CAPÍTULO 1 - DEFINICIÓN DEL PROYECTO DE INVESTIGACIÓN

## DEFINICIÓN DEL PROBLEMA

### Descripción del problema

En una entidad bancaria, el abandono de clientes representa un impacto negativo en los ingresos de la empresa, más aún cuando se trata de clientes regulares, que están afiliados a dicha entidad financiera; ya que el pilar de sus ingresos se basa en la actividad de sus clientes. Es así que cada empresa dentro de este rubro, busca la fidelización de sus clientes a través de estrategias de marketing, es decir, procesos analizados por especialistas, dirigidos a los clientes a los cuales dicho análisis concluyó que se debería fidelizar, este proceso tiene su margen de error ya que se basa en las conclusiones humanas a través de los datos observables, sin embargo gracias a los modelos de aprendizaje automático que existen hoy en día, se puede llegar a estas conclusiones de una manera más precisa y óptima.

### Situación Problemática

El abandono de clientes en una entidad bancaria representa un impacto negativo en los ingresos de esta, es un problema que, el proceso para percatarse del abandono de un cliente depende de la observación humana, e incluso si esta fuese precisa, se llega a ejecutar un plan de acción cuando el cliente ya ha abandonado la empresa y eso convierte a dicho plan, en un proceso ineficiente.

### Situación Deseada

La empresa será capaz de predecir el abandono de clientes a tiempo, ya que si consiguen averiguar qué clientes van a abandonar, podrán tomar decisiones basadas en estadística computacional y ejecutar estrategias de fidelización de los clientes a tiempo.

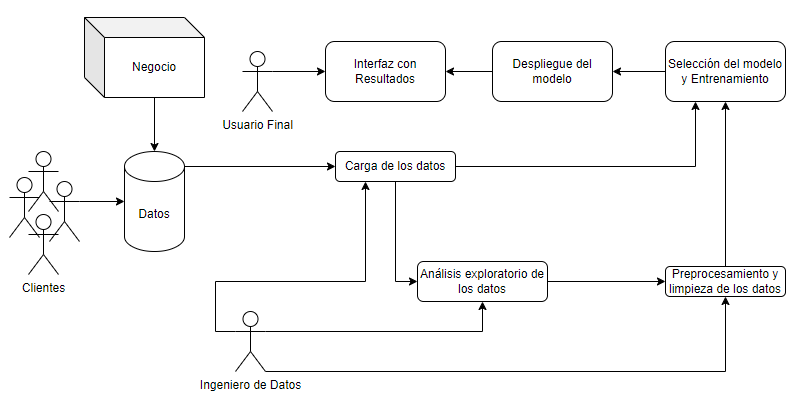


Ilustración . Representación de la situación deseada.

## OBJETO DE LA INVESTIGACIÓN

En el presente las entidades bancarias que desean llevar acabo un análisis de las entradas y salidas de sus clientes, manejan reportes de estos para analizar el crecimiento o decrecimiento de dicha cantidad de manera periódica, en muchos casos estos reportes se generan a través de diferentes herramientas de reportes que trabajan con datos obtenidos de la base de datos de la empresa.

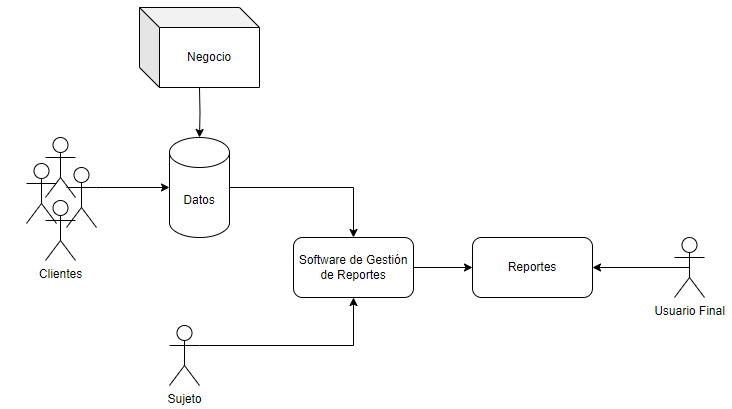


Ilustración . Representación del objeto de investigación

## OBJETIVOS

### Objetivo General

Elaborar un software basado en un modelo de aprendizaje automático para predecir el abandono de clientes en una entidad bancaria.

### Objetivos Específicos

* Analizar el entorno del objeto de estudio.
* Analizar el negocio.
* Analizar los datos disponibles.
* Analizar los requisitos a cumplir por el modelo.
* Diseñar los diagramas correspondientes.
* Limpiar los datos conforme lo requiera el modelo.
* Preprocesar los datos obtenidos.
* Seleccionar los modelos de aprendizaje automático adecuados.
* Entrenar los modelos seleccionados.
* Desarrollar un prototipo para validar los resultados.
* Implementar una interfaz gráfica para mostrar los resultados del modelo.
* Realizar una evaluación de los resultados del modelo y sus parámetros.
* Desplegar el modelo en la interfaz gráfica.

Poner en el documento porque no uso BI.

## METODOLOGÍA

La metodología empleada en el área IT Data, y la escogida para esta investigación es según Shearer (2000), CRISP-DM, esta metodología se utilizará asegurándonos la coherencia de resultados y el empleo de un procedimiento estándar en los proyectos de datos realizados.

CRISP-DM es una metodología estándar para el desarrollo de proyectos de minería de datos que ha sido ampliamente adoptada en la industria. Fue desarrollada por un grupo de expertos en minería de datos en 1996, y desde entonces ha sido revisada y actualizada varias veces.

La metodología CRISP-DM consta de seis fases principales: comprensión del negocio, comprensión de los datos, preparación de los datos, modelado, evaluación y despliegue. Cada fase es importante y tiene su propio conjunto de tareas y objetivos.

## ALCANCE

• El proyecto analizará el entorno del negocio, los datos disponibles y las etapas descritas por la metodología.

• Se documentará el proceso de análisis del negocio, tratamiento de los datos, selección y uso del modelo y los resultados a través de una interfaz gráfica.

• Se desarrollará un prototipo del modelo de aprendizaje automático seleccionado para validar los resultados.

• El uso del modelo contempla las entidades bancarias que cumplan con los parámetros que acepta el modelo de datos.

# CAPÍTULO 2 – DESCRIPCIÓN DEL NEGOCIO

## DESCRIPCIÓN DEL RUBRO

Predecir el abandono de clientes en bancos puede ser un proceso complejo debido a que existen muchos factores que influyen en la decisión del cliente de abandonar el banco, como pueden ser las tasas de interés, la calidad del servicio al cliente, la competencia, la satisfacción con los productos y servicios ofrecidos, entre otros.

## DESCRIPCIÓN DEL MERCADO DEL RUBRO

El mercado bancario es altamente competitivo y está en constante cambio debido a la evolución tecnológica, los cambios en la regulación gubernamental y la dinámica del mercado global. En consecuencia, los bancos están constantemente buscando nuevas formas de diferenciarse y mantenerse relevantes para sus clientes.

## DESCRIPCIÓN DE LOS SERVICIOS Y PRODUCTOS DE LAS ENTIDADES BANCARIAS EN BOLIVIA

Los bancos en Bolivia ofrecen una amplia gama de productos y servicios financieros para satisfacer las necesidades de sus clientes y mantener su competitividad en el mercado. A continuación, se menciona algunos de los servicios y productos más importantes de este rubro:

Cuentas de ahorro: Las cuentas de ahorro son una forma de depositar y mantener dinero en una entidad bancaria con el fin de generar intereses y ahorrar dinero. Estas cuentas pueden ser de diferentes tipos, como cuentas de ahorro a la vista o a plazo fijo.

Cuentas corrientes: Las cuentas corrientes son cuentas bancarias que permiten a los clientes realizar depósitos, retiros y pagos utilizando cheques o tarjetas de débito. Estas cuentas son adecuadas para empresas y personas que realizan una gran cantidad de transacciones.

Tarjetas de crédito: Las tarjetas de crédito son una forma de financiamiento que permite a los clientes realizar compras y pagos a crédito. Las tarjetas de crédito ofrecen diferentes límites de crédito y diferentes tasas de interés, y pueden incluir recompensas y beneficios adicionales.

Préstamos: Los préstamos son una forma de financiamiento en la que un cliente recibe una cantidad de dinero de una entidad bancaria y se compromete a devolverlo en un plazo determinado con intereses. Los préstamos pueden ser de diferentes tipos, como préstamos personales, préstamos para vivienda o préstamos para vehículos.

Seguros: Las entidades bancarias también ofrecen seguros para proteger a sus clientes de diferentes riesgos, como seguros de vida, seguros de salud, seguros para vehículos, entre otros.

Servicios de banca en línea: La banca en línea permite a los clientes realizar transacciones bancarias a través de internet, incluyendo transferencias, pagos de servicios, consulta de saldo, entre otros.

Servicios de asesoramiento financiero: Muchos bancos en Bolivia también ofrecen servicios de asesoramiento financiero para ayudar a sus clientes a tomar decisiones financieras informadas y planificar sus finanzas.

# CAPÍTULO 3 - APRENDIZAJE AUTOMÁTICO PARA LA PREDICCIÓN DE ABANDONO DE CLIENTES

## APRENDIZAJE AUTOMÁTICO

Según Alpaydin (2010), el aprendizaje automático es "un conjunto de métodos computacionales que permiten a un sistema cambiar su comportamiento de manera adaptativa basado en sus experiencias pasadas".

Es así como se sintetiza que un software de aprendizaje automático se caracteriza por su capacidad de aprender de los datos que utiliza para obtener resultados funcionales para cualquier aplicación o programa, ya sea detectando fraudes financieros, clasificando imágenes o prediciendo comportamientos.

## CARACTERÍSTICAS DE UN SOFTWARE QUE UTILIZA APRENDIZAJE AUTOMÁTICO

Según Kelleher (2018), algunas de las principales características de un software que utiliza aprendizaje automático son:

La capacidad para aprender a partir de datos, es decir que el software utiliza algoritmos y modelos estadísticos para aprender patrones y tendencias a partir de los datos proporcionados.

La adaptabilidad que puedan tener, es decir que el software es capaz de ajustar su comportamiento a medida que se le proporcionan más datos y aprende de sus experiencias pasadas.

La capacidad para generalizar, es decir que el software es capaz de aplicar sus conocimientos a nuevos datos y situaciones, en lugar de simplemente recordar los datos específicos que ha visto antes.

La automatización que permite que el software pueda automatizar tareas que antes requerían la intervención humana, como la identificación de patrones en grandes conjuntos de datos.

La capacidad para mejorar con el tiempo que hace que el software puede mejorar su precisión y rendimiento a medida que se le proporciona más datos y se ajustan sus modelos.

## TIPOS DE MODELOS DE APRENDIZAJE AUTOMÁTICO

Según Hastie (2009), existen tres tipos de modelos de aprendizaje automático:

Modelos supervisados: Estos modelos utilizan datos etiquetados para aprender a predecir una variable objetivo. Por ejemplo, un modelo supervisado podría aprender a predecir el precio de una casa en función de sus características, como el número de habitaciones y baños.

Modelos no supervisados: Estos modelos se utilizan para encontrar patrones y estructuras en los datos no etiquetados. Por ejemplo, un modelo no supervisado podría agrupar datos de clientes en función de sus comportamientos de compra.

Modelos de aprendizaje por refuerzo: Estos modelos aprenden a tomar decisiones en función de las recompensas y castigos recibidos por sus acciones. Por ejemplo, un modelo de aprendizaje por refuerzo podría aprender a jugar un juego al recibir recompensas por tomar decisiones correctas y castigos por tomar decisiones incorrectas.

## MODELOS DE APRENDIZAJE AUTOMÁTICO ESPECÍFICO PARA LA PREDICCIÓN DE ABANDONO DE CLIENTES

### Regresión Logística

Según Kelleher (2018), la regresión logística es un modelo supervisado que se utiliza para predecir una variable binaria, como la probabilidad de que un cliente abandone o no una empresa.

Entonces se puede concluir que de los 3 tipos de modelos ya vistos será el supervisado que utilizaremos y más en específico uno de regresión logística para dar como resultado una solución binaria.

### Árboles de Decisión

Según Breiman (1984), los árboles de decisión son un modelo de aprendizaje automático que se utiliza en problemas de clasificación y regresión. Estos modelos representan decisiones y sus posibles consecuencias en una estructura de árbol. Comienzan con un nodo raíz que contiene el atributo más relevante, y se ramifican en nodos secundarios que representan los valores posibles de ese atributo. Este proceso se repite recursivamente, dividiendo el conjunto de datos en subconjuntos más pequeños hasta que se alcanza una decisión final o un valor de regresión.

### Máquinas de Soporte Vectorial (SVM)

Según Cortes (1995), las Máquinas de Soporte Vectorial (SVM) son un conjunto de algoritmos de aprendizaje automático que se utilizan para el análisis de datos, especialmente en problemas de clasificación y regresión. El objetivo principal de las SVM es encontrar el hiperplano de separación óptimo en un espacio de alta dimensión para maximizar la distancia entre las clases de datos. Este enfoque las hace efectivas en la clasificación de datos tanto lineales como no lineales, lo que las convierte en una herramienta versátil para diversas aplicaciones, desde reconocimiento de patrones hasta análisis de imágenes y procesamiento de texto.

### Gradient Boosting

Según Friedman (2001), Gradient Boosting es un conjunto de técnicas de aprendizaje automático que se utilizan en problemas de clasificación y regresión. Se basa en la construcción secuencial de árboles de decisión ponderados, donde cada árbol se enfoca en corregir los errores del modelo anterior. En cada iteración, se ajusta un nuevo árbol para predecir los errores residuales del modelo actual, y luego se agregan estos árboles para obtener una predicción final. Esta técnica es conocida por su capacidad para manejar datos ruidosos y realizar predicciones precisas en una variedad de aplicaciones.

## METODOLOGÍA CRISP-DM

Según Shearer (2000), CRISP-DM (Cross Industry Standard Process for Data Mining) es un enfoque estándar y ampliamente utilizado para proyectos de minería de datos y aprendizaje automático. Proporciona una estructura paso a paso para guiar a los profesionales en la realización de proyectos de análisis de datos y extracción de conocimiento.

CRISP-DM consta de seis etapas principales:

### Comprensión del negocio

En esta etapa, se busca comprender los objetivos y requisitos del proyecto desde la perspectiva del negocio. Se establecen los objetivos de la minería de datos y se definen los criterios de éxito.

### Comprensión de los datos

En esta etapa, se recopilan y exploran los datos disponibles. Se realiza un análisis inicial de los datos para comprender su calidad, estructura y distribución. También se identifican las variables relevantes y se determina si se requiere algún tipo de transformación o limpieza de los datos.

### Preparación de los datos

En esta etapa, se realiza el procesamiento y la preparación de los datos para su posterior análisis. Esto incluye tareas como la selección de variables, la codificación de variables categóricas, el tratamiento de valores faltantes y la normalización de datos, entre otros.

### Modelado

En esta etapa, se seleccionan y aplican técnicas de modelado y aprendizaje automático para construir modelos predictivos o descriptivos. Se exploran diferentes algoritmos y se ajustan sus parámetros para obtener el mejor rendimiento. Luego, se evalúan los modelos utilizando medidas de rendimiento adecuadas.

### Evaluación

En esta etapa, se evalúan y validan los modelos construidos. Se verifica si los modelos cumplen con los objetivos del negocio y se realiza una evaluación detallada de su rendimiento. También se pueden realizar ajustes y mejoras en función de los resultados obtenidos.

### Despliegue

En esta etapa, se implementan los modelos en un entorno operativo. Esto puede implicar la integración con sistemas existentes, la automatización de la generación de predicciones o la generación de informes para su uso en la toma de decisiones. Además, se elabora un plan de monitoreo y mantenimiento continuo para asegurar que los modelos sigan siendo efectivos con el tiempo.

Es importante destacar que CRISP-DM es un enfoque iterativo, lo que significa que puede ser necesario volver a etapas anteriores a medida que se obtiene un mayor conocimiento sobre los datos o se refina el modelo. Esta metodología proporciona una guía estructurada para gestionar proyectos de minería de datos y facilita la colaboración entre los diferentes equipos involucrados.

## HERRAMIENTAS

### Python

Según Guttag (2016), Python es un lenguaje de programación de alto nivel y de propósito general que se caracteriza por su sintaxis legible y su enfoque en la legibilidad del código. Es un lenguaje interpretado y dinámicamente tipado, lo que significa que no requiere de una etapa de compilación y permite una programación más interactiva y flexible.

### Scikit-learn

Según Pedregosa (2011), scikit-learn es una biblioteca de aprendizaje automático de código abierto para Python que proporciona una amplia gama de algoritmos y herramientas para construir y aplicar modelos de aprendizaje automático. Está diseñada para ser fácil de usar y eficiente, y se utiliza ampliamente en la comunidad de aprendizaje automático para tareas de clasificación, regresión, agrupamiento y más.

### Visual Studio Code

Según Microsoft (2021), Visual Studio Code es un entorno de desarrollo integrado (IDE) de código abierto desarrollado por Microsoft. Es ampliamente utilizado por programadores y desarrolladores de software para escribir, depurar y administrar código en una variedad de lenguajes de programación. VS Code es conocido por su velocidad, extensibilidad y características de edición avanzadas.

**PARTE II**

**ANÁLISIS Y DISEÑO**

# CAPÍTULO 4 - ANÁLISIS DE REQUISITOS

## INTRODUCCIÓN

### Propósito

El propósito de este capítulo es la descripción a detalle de todos los requisitos

que tendrá el sistema propuesto. Esta descripción esta adecuada al formato

IEEE830.

### Definiciones, abreviaturas y acrónimos

Datos: cualquier información o estadística que se recopila, almacena y procesa en un formato estructurado o no estructurado.

Análisis de datos: el proceso de examinar y transformar datos para obtener información valiosa y conocimientos significativos.

Minería de datos: la exploración y análisis de grandes conjuntos de datos para descubrir patrones y tendencias ocultos.

Visualización de datos: la presentación gráfica de datos y estadísticas para facilitar la comprensión y la toma de decisiones.

Inteligencia empresarial (BI): el proceso de recopilar, analizar y presentar información empresarial para tomar decisiones informadas.

ETL: Extracción, Transformación y Carga (por sus siglas en inglés). Un proceso utilizado para integrar datos de diferentes fuentes y transformarlos en un formato común para su análisis.

CSV: Comma-Separated Values. Un formato de archivo común utilizado para almacenar datos tabulares.

JSON: Notación de objeto de JavaScript. Un formato de archivo utilizado para almacenar y transferir datos estructurados.

### Visión general de la especificación

En este capítulo se pretende describir todas las principales características que posee el software basado en un modelo de aprendizaje automático, además se mostrará la lista de los diferentes tipos de requisitos que tiene la aplicación del modelo determinado y se podrá evidenciar las restricciones que tendrá.

## DESCRIPCIÓN GENERAL DEL MODELO

### Perspectiva del producto

El software que se pretende implementar es totalmente independiente a los a una sola base de datos. Por políticas de seguridad, las entidades bancarias se niegan a compartir dichos datos, por ello se optó por aplicar un modelo totalmente independiente a un sistema bancario en específico y se enfoca en los parámetros de los datos que dicho modelo acepte para realizar la predicción. En resumen, el modelo soporta un conjunto de datos con determinadas columnas, el mismo que recibe el software para obtener los resultados del modelo.

### Funciones del producto

El software soportará la predicción de probabilidad de abandono de clientes en una entidad bancaria, podrá generar los resultados solicitados en una interfaz gráfica.

### Características de los usuarios

La interacción del usuario final con el modelo es a través de una interfaz gráfica interactiva capaz de mostrar los resultados del modelo. Por otro lado, existe el un archivo de tipo jupyter notebook donde se detalla todo el proceso de datos, este brinda un detalle más amplio del funcionamiento del modelo, este es diseñado para interactuar con los analistas de datos en busca del mantenimiento y exploración de los datos y el modelo.

A continuación, las características importantes del usuario:

Experiencia técnica: En primera instancia la interfaz le entrega los resultados del modelo intuitivamente sin ningún requerimiento técnico, pero si el usuario desea más detalles e interacción profunda con los datos y el modelo, podrá acceder al modelo en el archivo jupyter notebook y para ello requiere de conocimiento de ingeniería de datos.

Conocimiento del negocio: Los usuarios del modelo deben tener un buen conocimiento del negocio y de la industria en la que opera la empresa. Esto les permitirá interpretar los resultados del modelo y tomar decisiones informadas basadas en ellos.

Comprensión de los datos: Para el uso de la interfaz no es necesario, pero los usuarios que busquen profundizar en el jupyter notebook del modelo deben entender los datos que se utilizan para entrenar el modelo y los datos que se utilizan para hacer predicciones. Esto les permitirá evaluar la calidad de las predicciones y hacer ajustes si es necesario.

Habilidad para interpretar los resultados: Los usuarios del modelo deben ser capaces de interpretar los resultados de las predicciones y tomar decisiones basadas en ellos. Esto puede requerir habilidades analíticas y de pensamiento crítico.

Actitud hacia el modelo: Los usuarios deben tener una actitud positiva hacia el modelo y estar dispuestos a utilizarlo para tomar decisiones informadas. Si los usuarios no confían en el modelo, es posible que no lo utilicen de manera efectiva.

### Suposiciones y dependencias

Se necesitará cambiar los requisitos propuestos en caso de que:

* La empresa decida conectar una base de datos con parámetros distintos al modelo propuesto.
* La empresa desee algún tipo de información adicional a la predicción de abandono clientes.

## ESPECIFICACIÓN DE REQUISITOS

### Interfaces externas

#### Interfaces de usuario

Se trata de una interfaz gráfica interactiva para obtener los resultados del modelo.

#### Interfaces de hardware

Para la ejecución del modelo, será necesario disponer de una computadora que soporte la ejecución del modelo.

#### Interfaces de software

Sistema Operativo Computadora: Windows 10 o superior.

Editor de código: Visual Studio Code o Google Colab.

Navegador: La interfaz se ejecutará en el localhost.

#### Interfaces de comunicación

El modelo y una interfaz de usuario se comunicarán entre sí a fin de mostrar los resultados del modelo.

### Requisitos funcionales

* Como cliente espero que se solucionen los problemas que influirían en mi decisión de cambiarme a otra entidad financiera para no realizar un proceso de cambio de entidad financiera.
* Como usuario final que accede a la interfaz interactiva espero resultados entendibles para no tener problemas en entender los resultados y que el software sea útil.
* Como analista de datos, necesito fuentes de resultados del modelo y no solo de resultados con la finalidad de generar visualizaciones si fuese necesario.
* Como gerente general, se espera resultados concretos y útiles con la finalidad de tomar decisiones eficientes.

### Requisitos no funcionales

* Escalabilidad: El modelo debe ser capaz de manejar grandes volúmenes de datos y aumentar su capacidad de procesamiento a medida que la cantidad de datos aumente.
* Seguridad: El modelo debe ser seguro y proteger los datos confidenciales de los clientes, así como los datos de la empresa.
* Fiabilidad: El modelo debe ser confiable y producir resultados precisos y coherentes en todo momento.
* Mantenibilidad: El modelo debe ser fácil de mantener y actualizar para garantizar que siga siendo efectivo a medida que cambien las necesidades y los requisitos de la empresa.

### Restricciones del modelo

Por el momento no existen restricciones en el modelo.

### Atributos del modelo

* Precisión: El modelo debe ser preciso y tener una alta tasa de aciertos en la predicción del abandono de clientes. Esto significa que el modelo debe ser capaz de identificar a los clientes que tienen más probabilidades de abandonar la empresa.
* Especificidad: El modelo también debe ser específico, es decir, debe ser capaz de identificar correctamente a aquellos clientes que tienen intención de abandonar la empresa.
* Facilidad de uso: El modelo debe ser fácil de usar para los analistas de datos y los responsables de la toma de decisiones.
* Interpretabilidad: El modelo debe ser interpretable, lo que significa que los resultados de la predicción deben poder ser explicados y entendidos fácilmente por los usuarios.
* Coste efectivo: El modelo debe ser capaz de proporcionar una buena relación calidad-precio, es decir, debe ser eficiente en términos de coste y proporcionar resultados valiosos para la empresa.

## COMPRENSIÓN DEL NEGOCIO

### Objetivos del negocio

El principal objetivo del negocio con el proyecto es la reducción de la taza de abandono de los clientes en la entidad bancaria determinada. El propósito del proyecto es ser un soporte para la fidelización de los clientes que cuentan con un nivel de probabilidad de abandono de clientes en las entidades bancarias, esto guarda una estrecha relación con los objetivos de cualquier entidad bancaria que es, evitar el abandono de clientes.

### Contexto del negocio

La situación actual de los bancos es que hay mayor competencia debido a las nuevas entidades bancarias que van emergiendo con el pasar de los años. Algunos de los factores del mercado que mayor impacto tienen en el abandono de clientes son, las tasas de interés en tarjetas de crédito y en las cuentas bancarias, es decir, se basan en el consumo que el cliente.

### Necesidades del negocio

El negocio necesita tener un departamento o equipo de personas para tomar acción acerca de los resultados de la predicción del abandono de clientes.

### Identificación de Casos de Uso

|  |  |
| --- | --- |
| Nro. CU | Descripción |
| 1 | Gestionar Resultados |
| 2 | Gestionar Análisis Exploratorio de Datos |
| 3 | Gestionar Carga de Datos |
| 4 | Gestionar Sugerencia de Modelo a Utilizar |
| 5 | Gestionar Búsqueda de Cliente |

### Mapeo de Requerimientos

|  |  |  |  |
| --- | --- | --- | --- |
| Nro. Req. | Descripción del requerimiento | Nro. CU | Nombre del CU |
| 1 | Como usuario necesito una interfaz gráfica con la cual interactuar para observar los resultados. | 1 | Gestionar Resultados |
| 2 | Como usuario quiero efectuar la limpieza de los datos que estoy cargando y ver los resultados de esta limpieza a través de un análisis exploratorio de datos. | 2 | Gestionar Análisis Exploratorio de Datos |
| 3 | Como usuario deseo que el software me sugiera o indique cual modelo es más preciso según el conjunto de datos que estoy utilizando. | 3 | Gestionar Carga de Datos |
| 4 | Como usuario deseo que el software me sugiera o indique cual modelo es más preciso. | 4 | Gestionar Sugerencia de Modelo a Utilizar |
| 5 | Como usuario, al conocer que cliente tienen alta probabilidad de abandonar, necesito buscar los datos de estos clientes en específico. | 5 | Gestionar Búsqueda de Cliente |

## CARACTERÍSTICAS DEL CONJUNTO DE DATOS

El modelo de dominio para nuestro proyecto de predicción de abandono de clientes incluye las siguientes variables:

RowNumber (Número de Fila): Este parece ser un número de fila o índice que identifica de manera única cada fila en tu conjunto de datos.

CustomerId (ID del Cliente): Este es un identificador único para cada cliente en tu conjunto de datos. Se utiliza para distinguir a cada cliente.

Surname (Apellido): Esta columna contiene los apellidos de los clientes. Indica el apellido de cada cliente en el conjunto de datos.

CreditScore (Puntuación de Crédito): Representa la puntuación de crédito de cada cliente, que generalmente se utiliza para evaluar la solvencia crediticia.

Geography (Geografía): Indica la ubicación geográfica o el país de residencia de cada cliente.

Gender (Género): Esta columna contiene el género de cada cliente, es decir, si son masculinos o femeninos.

Age (Edad): Representa la edad de cada cliente en años.

Tenure (Tenencia): Indica la cantidad de años que un cliente ha sido titular de una cuenta o ha estado en relación con la entidad bancaria.

Balance (Saldo): Esta columna puede representar el saldo actual en la cuenta de cada cliente en la entidad bancaria.

NumOfProducts (Número de Productos): Indica la cantidad de productos o servicios financieros que el cliente tiene con la entidad bancaria.

HasCrCard (Tiene Tarjeta de Crédito): Puede indicar si el cliente posee o no una tarjeta de crédito (1 para sí, 0 para no).

IsActiveMember (Miembro Activo): Esta columna podría indicar si el cliente es un miembro activo de la entidad bancaria (1 para sí, 0 para no).

EstimatedSalary (Salario Estimado): Representa el salario estimado o ingresos del cliente.

Exited (Abandonó): Esta columna podría indicar si el cliente ha abandonado o cerrado su cuenta con la entidad bancaria (1 para sí, 0 para no).

Estas variables capturan diferentes aspectos que pueden influir en la probabilidad de abandono de un cliente. Es importante tener en cuenta que este modelo de dominio es una simplificación y que pueden existir otras variables o factores que también podrían ser relevantes.

# CAPÍTULO 6 – ANÁLISIS Y DISEÑO

## DESCRIPCIÓN DE LOS CASOS DE USO

La descripción de cada caso de uso siguiendo la metodología de desarrollo basada en casos de uso, la cual debe describir los siguientes componentes:

- Descripción del Caso de Uso

- Diagrama de Caso de Uso

A continuación, se describe cada caso de uso, del software siguiendo el orden detallado en el mapeo de requerimientos:

### Descripción Caso de Uso – 1

|  |  |  |  |
| --- | --- | --- | --- |
| **Caso de Uso ID** | 1 | | |
| **Nombre del caso de uso** | Gestionar Resultados | | |
| **Creado por** | Enrique Arce | **Última actualización** | Enrique Arce |
| **Fecha de creación** | 04/09/2023 | **Última revisión** | 06/10/2023 |
| **Actores** | Usuario | | |
| **Descripción** | El usuario solicita los resultados del modelo. | | |
| **Precondiciones** | Haber cargado y limpiado los datos. Opcionalmente el usuario puede ver que modelo es más preciso para ver los resultados. | | |
| **Postcondiciones** | La información de los resultados debe ser desplegada. | | |
| **Flujo normal** | 1. El usuario selecciona la cantidad top de clientes que desea ver en el resultado. 2. El usuario escoge que modelo desea y presiona el botón de dicho modelo. 3. El software despliega los resultados del modelo seleccionado. | | |
| **Excepciones** | 1. El resultado del modelo no está listo | | |
| **Flujo alternativo** | 1. El software notifica el motivo de la ausencia de un resultado del modelo. | | |
| **Frecuencia de uso** | Muy frecuente | | |

Diagrama

Descripción generada automáticamente

Ilustración . Diagrama de caso de uso - Gestionar Resultado

### Descripción Caso de Uso – 2

|  |  |  |  |
| --- | --- | --- | --- |
| **Caso de Uso ID** | 2 | | |
| **Nombre del caso de uso** | Gestionar Análisis Exploratorio de Datos | | |
| **Creado por** | Enrique Arce | **Última actualización** | Enrique Arce |
| **Fecha de creación** | 04/09/2023 | **Última revisión** | 13/11/2023 |
| **Actores** | Usuario | | |
| **Descripción** | El usuario con un botón genera un análisis exploratorio de datos que se limpian previo al análisis. | | |
| **Precondiciones** | Debe existir el archivo cargado. | | |
| **Postcondiciones** | La información del análisis exploratorio de datos debe ser desplegada. | | |
| **Flujo normal** | 1. El usuario presiona el botón de ver análisis exploratorio de datos. 2. El software despliega los resultados de la limpieza y del análisis de datos. | | |
| **Excepciones** | 1. El análisis o la limpieza no se despliega. | | |
| **Flujo alternativo** | 1. El software notifica la ausencia de los resultados del análisis o de la limpieza. | | |
| **Frecuencia de uso** | Frecuente | | |

Diagrama

Descripción generada automáticamente

Ilustración . Diagrama de caso de uso - Gestionar Análisis Exploratorio de Datos

### Descripción Caso de Uso – 3

|  |  |  |  |
| --- | --- | --- | --- |
| **Caso de Uso ID** | 4 | | |
| **Nombre del caso de uso** | Gestionar Carga de Datos | | |
| **Creado por** | Enrique Arce | **Última actualización** | Enrique Arce |
| **Fecha de creación** | 14/10/2023 | **Última revisión** | 13/11/2023 |
| **Actores** | Usuario | | |
| **Descripción** | El usuario carga el archivo con los datos que le indica software. | | |
| **Precondiciones** | Visibilidad de el ejemplo de cómo debe estar organizado el archivo de datos. | | |
| **Postcondiciones** | Debe aparecer el nombre del archivo de datos subido. | | |
| **Flujo normal** | 1. El usuario presiona el botón de cargar datos. 2. El usuario selecciona, desde el explorador de archivos, el archivo que contiene los datos a utilizarse. | | |
| **Excepciones** | 1. El archivo no cumple los requisitos del software. 2. El archivo no puede ser leído. | | |
| **Flujo alternativo** | 1. El software notifica que el archivo no cumple los requisitos o no se encuentra en el formato adecuado. | | |
| **Frecuencia de uso** | Siempre | | |

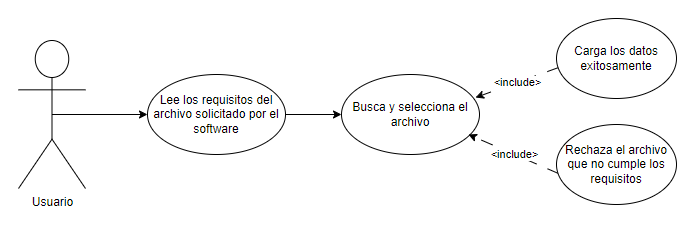


Ilustración 6. Diagrama de casos de uso – Gestionar Carga de Datos

### Descripción Caso de Uso – 4

|  |  |  |  |
| --- | --- | --- | --- |
| **Caso de Uso ID** | 4 | | |
| **Nombre del caso de uso** | Gestionar Sugerencia de Modelo a Utilizar | | |
| **Creado por** | Enrique Arce | **Última actualización** | Enrique Arce |
| **Fecha de creación** | 04/09/2023 | **Última revisión** | 06/10/2023 |
| **Actores** | Usuario | | |
| **Descripción** | El usuario solicita la sugerencia de cual modelo es más preciso para los datos. | | |
| **Precondiciones** | Haber realizado la limpieza de datos. | | |
| **Postcondiciones** | La sugerencia de cual modelo utilizar, debe ser desplegada. | | |
| **Flujo normal** | 1. El usuario presiona el botón de mostrar sugerencia. 2. El software despliega la sugerencia con un gráfico. | | |
| **Excepciones** | 1. La gráfica de sugerencia no puede desplegarse | | |
| **Flujo alternativo** | 1. El software notifica el error de la ausencia de un resultado gráfico. | | |
| **Frecuencia de uso** | Opcional – Frecuente | | |

Diagrama

Descripción generada automáticamente

Ilustración . Diagrama de casos de uso – Gestionar Sugerencia de Modelo a Utilizar

### Descripción Caso de Uso - 5

|  |  |  |  |
| --- | --- | --- | --- |
| **Caso de Uso ID** | 5 | | |
| **Nombre del caso de uso** | Gestionar Búsqueda de Cliente | | |
| **Creado por** | Enrique Arce | **Última actualización** | Enrique Arce |
| **Fecha de creación** | 13/11/2023 | **Última revisión** | 17/11/2023 |
| **Actores** | Usuario | | |
| **Descripción** | El usuario digita el id de cliente que le proporciona la gráfica de resultados para obtener todos los datos de dicho cliente. | | |
| **Precondiciones** | Haber solicitado los resultados. | | |
| **Postcondiciones** | Se debe mostrar todos los datos del id del cliente solicitado. | | |
| **Flujo normal** | 1. El usuario digita el id del cliente que desea obtener. 2. El software despliega los datos del cliente solicitado. | | |
| **Excepciones** | 1. Los datos del cliente no se despliegan | | |
| **Flujo alternativo** | 1. El software notifica la ausencia del cliente solicitado. | | |
| **Frecuencia de uso** | Opcional | | |

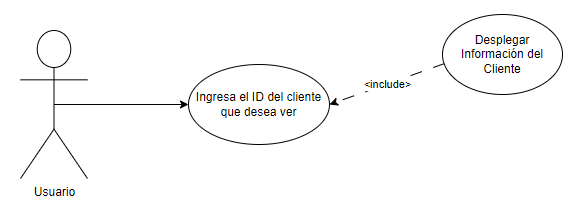


Ilustración 6. Diagrama de casos de uso – Gestionar Búsqueda de Cliente

## COMPRENSIÓN DE LOS DATOS

### Análisis de Datos Iniciales

#### Diccionario de Datos

Tabla

Descripción generada automáticamente

Ilustración 8. Diccionario de datos

#### Diagrama de Estados de Transición

Diagrama

Descripción generada automáticamente

Ilustración 7. Diagrama de clases ejemplo de una entidad bancaria

### Análisis de Variables y Relaciones

### Preparación de la Variable Objetivo

## DIAGRAMA DE DESPLIEGUE

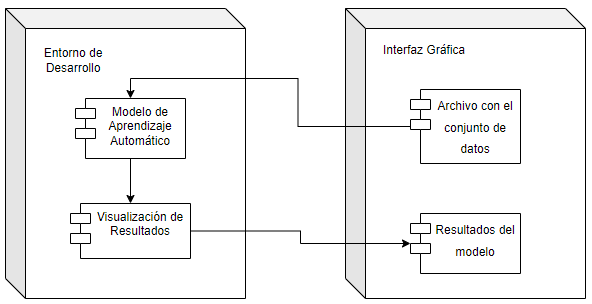


Ilustración . Diagrama de despliegue

PARTE III

IMPLEMENTACIÓN Y PRUEBAS

# CAPÍTULO 6 – IMPLEMENTACIÓN

## ANÁLISIS DE ELECCIÓN DE LA HERRAMIENTA

Se procedió a investigar qué tipo herramienta se utilizará, para ello se planteó posibles alternativas. Herramientas en las que se podrá crear, entrenar y desplegar el modelo.

Para ello se investigó acerca de Python y R como lenguajes de programación; Anaconda, Google Colab y Visual Studio como entornos de trabajo; Scikit-Learn (sklearn), Statsmodels y glm() en R como librerías y funciones para trabajar con machine learning; matplotlib, seaborn, pandas y ggplot2 como librerías de análisis y visualización de datos; Streamlit, Dash y Pywebio como librerías de Python para la construcci[on de la interfaz gráfica.

### Lenguajes de Programación

|  |  |  |
| --- | --- | --- |
| **Criterio** | **Python** | **R** |
| Popularidad | Muy popular en la comunidad de ciencia de datos y machine learning. Amplia adopción en la industria. | Ampliamente utilizado en estadísticas y análisis de datos, con una comunidad sólida. |
| Bibliotecas y Ecosistema | Bibliotecas populares como NumPy, pandas, scikit-learn, TensorFlow y PyTorch para análisis y machine learning. | Bibliotecas robustas como dplyr, ggplot2, caret, xgboost y Shiny para análisis y machine learning. |
| Facilidad de Aprendizaje | Fácil de aprender, con una curva de aprendizaje suave, especialmente para principiantes. | Requiere una curva de aprendizaje moderada, especialmente si tienes experiencia en programación. |
| Flexibilidad | Altamente flexible y versátil, adecuado para una amplia gama de aplicaciones y dominios. | Flexible y potente en estadísticas y análisis de datos, pero menos versátil para aplicaciones generales. |
| Visualización de Datos | Ofrece una variedad de bibliotecas como Matplotlib, Seaborn y Plotly para visualización de datos. | Tiene un sistema de visualización sólido con ggplot2, lattice y otras bibliotecas. |
| Comunidad y Soporte | Tiene una comunidad grande y activa, con una gran cantidad de recursos y soluciones disponibles en línea. | Tiene una comunidad sólida, especialmente en estadísticas y análisis de datos. |
| Integración con Big Data | Tiene herramientas y bibliotecas como Apache Spark y Dask para trabajar con big data. | Menos opciones de integración con big data en comparación con Python. |
| Desarrollo Web | Ofrece opciones como Flask y Django para el desarrollo web. | No es tan fuerte en desarrollo web como Python. Shiny es una opción para aplicaciones web interactivas. |
| Machine Learning | Tiene bibliotecas bien establecidas como scikit-learn, TensorFlow y PyTorch para machine learning. | Ofrece paquetes como caret y xgboost para machine learning, aunque no tan extensos como Python. |

### Librerías para utilizar aprendizaje automático

|  |  |  |  |
| --- | --- | --- | --- |
| **Característica** | **Scikit-Learn** | **Statsmodels** | **glm() en R** |
| Lenguaje de Programación | Python | Python | R |
| Facilidad de Uso | Fácil de usar y tiene una amplia comunidad de usuarios. | Relativamente fácil de usar, pero puede requerir más código para análisis detallados. | Ampliamente utilizado para estadísticas avanzadas, pero puede tener una curva de aprendizaje. |
| Flexibilidad y Personalización | Ofrece muchas opciones de personalización y modelos de clasificación además de la regresión logística. | Principalmente enfocado en estadísticas, lo que lo hace menos flexible para otros tipos de modelos. | Principalmente utilizado para estadísticas avanzadas, pero puede ser personalizado para otros tipos de modelos. |
| Resultados Detallados | Proporciona resultados simplificados, pero puede ser complementado con otras librerías para análisis más detallados. | Proporciona resultados detallados y estadísticas para el análisis. | Proporciona resultados detallados y estadísticas para el análisis. |
| Comunidad y Soporte | Tiene una gran comunidad de usuarios y amplia documentación. | Cuenta con una comunidad activa y documentación. | R tiene una comunidad sólida y recursos en línea abundantes. |
| Integración con otras Librerías | Se integra bien con otras librerías de machine learning y análisis de datos en Python. | Puede integrarse con otras librerías de Python, pero con menos facilidad que Scikit-Learn. | Puede integrarse con otras librerías de R para análisis de datos y gráficos. |
| Curva de Aprendizaje | Relativamente baja, especialmente para tareas comunes. | Baja para tareas comunes, pero mayor para análisis detallados. | Puede tener una curva de aprendizaje moderada para usuarios nuevos en R. |

### Librerías de análisis y visualización de datos

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Librería** | **Dificultad de Uso** | **Claridad** | **Precisión** | **Eficiencia de Código** |
| Matplotlib (Python) | Moderada a Alta | Menos Clara | Alta | Menos Eficiente |
| Seaborn (Python) | Moderada | Más Clara | Alta | Más Eficiente |
| Pandas (Python) | Baja a Moderada | Menos Clara | Adecuada | Más Eficiente |
| ggplot2 (R) | Moderada | Más Clara | Alta | Eficiente |

### Librerías para desarrollo de la interfaz gráfica

|  |  |  |  |
| --- | --- | --- | --- |
| **Criterio** | **Streamlit** | **Dash** | **Pywebio** |
| Escalabilidad | Buen soporte para aplicaciones pequeñas y medianas. Puede volverse lento en aplicaciones grandes. | Escalabilidad adecuada para aplicaciones de tamaño mediano a grande. | Diseñado para aplicaciones pequeñas y medianas. Escalabilidad limitada para aplicaciones grandes. |
| Bajo Acoplamiento | Permite una estructura de código simple y menos acoplada. No necesita un código base complejo. | Requiere una estructura más compleja de componentes interconectados. | Ofrece una estructura modular y componentes independientes, lo que facilita el bajo acoplamiento. |
| Seguridad | La seguridad depende del desarrollador para implementar medidas adecuadas. | Ofrece opciones para autenticación y autorización, lo que permite un mayor control de seguridad. | Proporciona características de seguridad básicas, pero depende en gran medida de las prácticas de seguridad del desarrollador. |
| Permisos de Usuarios | Depende de la implementación personalizada para el control de acceso. | Ofrece un mayor control de acceso y autorización de usuarios. | Proporciona funciones básicas para el control de acceso de usuarios. |
| Facilidad de Uso | Muy fácil de aprender y usar, con una curva de aprendizaje suave. Ideal para prototipado rápido. | Tiene una curva de aprendizaje más pronunciada debido a su estructura más compleja. | Relativamente fácil de usar, pero puede ser menos intuitivo que Streamlit. |
| Rendimiento | Rendimiento moderado adecuado para aplicaciones de tamaño mediano. | Rendimiento sólido, especialmente para aplicaciones más grandes. | Rendimiento moderado adecuado para aplicaciones de tamaño mediano. |
| Comunidad | Comunidad activa y creciente, con una amplia variedad de extensiones y paquetes disponibles. | Comunidad activa y en crecimiento, con un ecosistema robusto de complementos. | Comunidad en crecimiento, pero más pequeña en comparación con Streamlit y Dash. |
| Documentación | Documentación detallada y abundante con ejemplos útiles. | Documentación completa y ejemplos útiles disponibles. | Documentación suficiente, pero no tan extensa como las otras dos. |

Según la investigación se concluye el uso de las siguientes tecnologías:

* Python
* Pandas y matplotlib
* Scikit-Learn
* Streamlit

## ANÁLISIS DE LAS TECNOLOGÍAS A UTILIZAR

El proceso de implementación del modelo de aprendizaje automático al software correspondiente se realiza mediante un proceso de integración entre el archivo donde se aloja el modelo y la interfaz gráfica de acceso para el usuario. Tanto el modelo como la interfaz gráfica son desarrollados en el entorno de desarrollo en la nube. A continuación, se describirán las tecnologías a utilizarse.

### Aplicación Web

Con fines de mostrar los resultados del modelo y la interacción con el mismo, se utiliza Python para desarrollar una aplicación web.

### Entorno de Desarrollo

Los notebooks de Google Colab se ejecutan en servidores de Google, lo que permite ejecutar código en la nube sin necesidad de configurar un entorno local. Google Colab es un servicio en línea proporcionado por Google que ofrece un entorno de desarrollo y ejecución de notebooks de Jupyter de forma gratuita. Es aquí donde correrá nuestro modelo y se alojará el código de la interfaz gráfica.

## PREPARACIÓN DE LOS DATOS

### Limpieza de Datos

### Preprocesamiento de Datos

### Selección de Características

### División de Conjuntos

### Transformación de datos

## MODELADO

### Selección de Modelo

### Entrenamiento de Modelo

# CAPÍTULO 7 – PRUEBAS

## EVALUACIÓN

### Métricas de Evaluación

### Validación Cruzada

### Interpretación de Resultados

## ESTRATEGIA DE PRUEBAS

En este capítulo, se describe en detalle la estrategia de pruebas diseñada y ejecutada para evaluar el rendimiento y la efectividad del modelo de aprendizaje automático desarrollado en el proyecto. Se explican los diferentes tipos de pruebas realizadas y los criterios de éxito establecidos para cada una.

### Pruebas de funcionalidad

En este subapartado, se profundiza en las pruebas de funcionalidad realizadas para asegurarse de que el modelo cumple con los requisitos establecidos y que funciona correctamente. Se describen los casos de prueba diseñados, los datos de entrada utilizados y los resultados esperados. Además, se discute cómo se han identificado y gestionado los errores y las desviaciones de los resultados esperados.

## RESULTADOS PRUEBAS FUNCIONALES

Se presentan los resultados de las pruebas de funcionalidad realizadas. Se muestran las métricas clave utilizadas para evaluar el rendimiento del modelo, como la precisión, la sensibilidad y la especificidad. Se incluyen gráficos y visualizaciones que ayuden a comprender el comportamiento del modelo durante las pruebas. Además, se discuten las implicaciones de los resultados y se destacarán las áreas de mejora identificadas.

# CAPITULO 8 – DESPLIEGUE

## IMPLEMENTACIÓN DE INTERFAZ GRÁFICA

# CONCLUSIONES

# RECOMENDACIONES

# BIBLIOGRAFÍA

Alpaydin, E. (2010). Introduction to machine learning (2nd ed.). Cambridge, MA: MIT Press. Recuperado el 22 de marzo de 2023

Kelleher, J. D., & Tierney, B. (2018). Data science: An introduction (2nd ed.). Boca Raton, FL: CRC Press. Recuperado el 12 de marzo de 2023

Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning: Data mining, inference, and prediction (2nd ed.). New York, NY: Springer. Recuperado el 12 de marzo de 2023

Microsoft. (2021). Visual Studio Code. <https://code.visualstudio.com/>. Recuperado el 10 de octubre de 2023

Kelleher, J. D., & Tierney, B. (2018). Data science: An introduction (2nd ed.). Boca Raton, FL: CRC Press. Recuperado el 22 de marzo de 2023

Kelleher, J. D., & Tierney, B. (2018). Data science: An introduction (2nd ed.). Boca Raton, FL: CRC Press. Recuperado el 28 de marzo de 2023

Pressman, R. S. (2014). Ingeniería del software: un enfoque práctico. McGraw-Hill. Recuperado el 28 de marzo de 2023

Tanenbaum, A. S. y Bos, H. (2015). Sistemas operativos modernos. Pearson Educación. Recuperado el 28 de marzo de 2023

Sommerville, I. (2011). Ingeniería de software. Pearson Educación. Recuperado el 28 de marzo de 2023

Shearer, C. (2000). The CRISP-DM model: the new blueprint for data mining. Journal of data warehousing, 5(4). Recuperado el 30 de marzo de 2023

Guttag, J. V. (2016). Introduction to Computation and Programming Using Python: With Application to Understanding Data. MIT Press. Recuperado el 11 de abril de 2023

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Vanderplas, J. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research. Recuperado el 21 de mayo de 2023

Rajpurkar, P., Irvin, J., Bagul, A., Ding, D., Duan, T., Mehta, H., ... & Patel, B. (2019). Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. Recuperado el 21 de Agosto de 2023

Cortes, C., & Vapnik, V. (1995). Support-Vector Networks. Machine Learning, 20(3), 273-297. Recuperado el 11 de noviembre de 2023

Friedman, J. H. (2001). Greedy Function Approximation: A Gradient Boosting Machine. Annals of Statistics, 29(5), 1189-1232. Recuperado el 11 de noviembre de 2023