UNIVERSIDAD NACIONAL DE SAN JUAN

Facultad de Ciencias Exactas, Físicas y Naturales

Departamento de Informática

LICENCIATURA EN SISTEMAS DE INFORMACION

Trabajo Final

"Hacia la construcción de un modelo estadístico que permita identificar potenciales nuevos clientes tomadores de préstamos en instituciones financieras"

Autor: Pujado, Horacio Matias

Asesor: Mag. Leonel Ganga

San Juan

2026

Índice de contenido

Índice de figuras

Dedicatoria

A mi madre, por ser mi mayor fuente de inspiración, ejemplo de resiliencia y presencia constante en cada paso de mi vida.

A mi familia, por enseñarme que el esfuerzo silencioso construye los logros más duraderos.

A quienes me acompañaron en este camino, incluso sin comprender del todo lo que hacía, pero confiando siempre en que valía la pena.

Y a mi hermana, con la esperanza de que este trabajo le recuerde que el esfuerzo abre puertas inimaginables y que el conocimiento no tiene límites: nos transforma, nos eleva y nos conecta con mundos nuevos.

Agradecimientos

Quiero expresar mi más profundo agradecimiento al Mag. Leonel Ganga, por su guía académica, su paciencia y su compromiso constante durante el desarrollo de este trabajo. Su acompañamiento fue clave para transformar una idea en una propuesta sólida y rigurosa.

Agradezco también a la Universidad Nacional de San Juan, por brindarme las herramientas necesarias para crecer profesionalmente y por fomentar un entorno de formación crítica y aplicada.

A mis colegas, por los intercambios enriquecedores, las discusiones metodológicas y el apoyo mutuo que hicieron de este camino una experiencia compartida.

Finalmente, a mi familia y seres queridos, por su apoyo emocional, por respetar mis tiempos y por sostenerme en los momentos de mayor exigencia. Este logro es también fruto de su presencia constante.

Resumen

Este trabajo final propone el desarrollo y evaluación de modelos predictivos orientados a identificar clientes con alta probabilidad de aceptar un préstamo personal en instituciones financieras. A partir de un conjunto de datos de acceso público, se aplicaron técnicas de ciencia de datos que incluyeron análisis exploratorio de datos, preprocesamiento, balanceo de clases y modelado supervisado mediante regresión logística, árboles de clasificación y redes neuronales.

El estudio se centró en la comparación del desempeño predictivo de estos modelos, utilizando métricas apropiadas para problemas de clasificación con clases desbalanceadas. Los resultados obtenidos evidencian que es posible construir modelos capaces de identificar clientes con mayor probabilidad de aceptar un producto financiero, lo que puede contribuir a optimizar estrategias de captación comercial y mejorar la asignación de recursos dentro de las instituciones financieras.

Este trabajo se enmarca en la Licenciatura en Sistemas de Información de la Universidad Nacional de San Juan, y busca aportar valor tanto desde una perspectiva académica como aplicada al campo de la analítica financiera.

Abstract

This final project proposes the development and evaluation of predictive models aimed at identifying customers with a high probability of accepting a personal loan within financial institutions. Using a publicly available dataset, several data science techniques were applied, including exploratory data analysis, data preprocessing, class balancing, and supervised modeling through logistic regression, decision trees, and artificial neural networks.

The study focused on comparing the predictive performance of these models using evaluation metrics appropriate for classification problems with imbalanced classes. The results demonstrate that it is possible to build models capable of identifying customers with a higher likelihood of accepting a financial product, which can contribute to optimizing customer acquisition strategies and improving resource allocation within financial institutions.

This work was developed within the framework of the Bachelor's Degree in Information Systems at the National University of San Juan, aiming to contribute both academically and practically to the field of financial analytics.

1. Introducción

El presente proyecto se desarrolla como trabajo final para la obtención del título de Licenciatura en Sistemas de Información de la Facultad de Ciencias Exactas, Físicas y Naturales (FCEFyN) de la Universidad Nacional de San Juan.

El objetivo central de la investigación consiste en desarrollar y evaluar comparativamente tres modelos predictivos basados en Regresión Logística, Árboles de Decisión y Redes Neuronales Artificiales. El propósito de este análisis comparativo es identificar cuál de estos enfoques presenta el mejor desempeño para detectar clientes con alta probabilidad de aceptar un préstamo personal, contribuyendo así a mejorar las estrategias de captación comercial en instituciones financieras.

Debido a que las instituciones financieras suelen proteger su información mediante estrictos acuerdos de confidencialidad, el acceso a datos reales resulta limitado para fines académicos. Por esta razón, el presente trabajo utiliza un conjunto de datos de acceso público, lo cual permite reproducir las principales etapas de un proyecto de ciencia de datos: exploración de datos, preprocesamiento, construcción de modelos predictivos y evaluación de su desempeño.

Con el propósito de orientar el desarrollo del trabajo, se plantean diversos interrogantes de investigación, entre ellos: ¿Qué variables resultan más relevantes para explicar la aceptación de un préstamo personal? ¿Qué técnicas de modelado permiten identificar con mayor precisión a los clientes potencialmente interesados? ¿Cómo pueden utilizarse los datos disponibles para mejorar las estrategias de captación en el sector financiero?

Finalmente, es importante señalar que el alcance de este estudio se centra en comparar el desempeño de distintos enfoques de modelado utilizados en problemas de clasificación, tales como la regresión logística, los árboles de decisión y las redes neuronales. Si bien cada una de estas técnicas admite niveles más avanzados de optimización y análisis metodológico, el objetivo del trabajo consiste en evaluar su capacidad predictiva relativa mediante métricas apropiadas, más que en desarrollar una optimización exhaustiva de cada algoritmo individual.

2. Formulación del Problema y Justificación

En la industria financiera actual, la competencia y la necesidad de maximizar las oportunidades de negocio han llevado a las instituciones a buscar no solo la gestión eficiente del riesgo, sino también la optimización de sus estrategias para identificar y captar clientes potenciales. Es fundamental lograr una convergencia entre el vasto volumen de datos disponibles y la identificación de información clave para mejorar la toma de decisiones estratégicas para potenciar el crecimiento de la cartera de clientes.

La ciencia de datos emerge como una herramienta clave en este contexto, ya que posee la capacidad de analizar grandes conjuntos de datos de alta dimensionalidad, descubrir patrones ocultos y aplicar algoritmos de aprendizaje automático para predecir comportamientos de los clientes. No obstante, a pesar de los avances tecnológicos, existen desafíos significativos para implementar estos modelos de manera efectiva, tanto desde el punto de vista técnico, como en lo que respecta a cuestiones éticas relacionadas con el uso y manejo de la información

El estudio se enfoca principalmente en identificar a los clientes con alta probabilidad para solicitar un préstamo, detectando qué factores son más determinantes e influyentes a la hora de generar un modelo estadístico. Aunque la tecnología ha avanzado, la aplicación eficiente de estas técnicas en la práctica aún presenta retos, como la correcta interpretación de los datos, la elección adecuada de los algoritmos y la implementación ética de los modelos.

Este estudio responde a la creciente demanda por parte de las instituciones financieras de optimizar sus estrategias de captación de clientes en un entorno cada vez más competitivo y cambiante. La capacidad de identificar con precisión a aquellos clientes con alta probabilidad de aceptar un préstamo es crucial para mejorar la eficacia de las campañas de marketing y para gestionar los recursos de manera más eficiente. Mediante el desarrollo y la validación de un modelo estadístico, se pretende identificar las variables más importantes y los patrones de comportamiento que mejor predicen el interés de los clientes por productos financieros específicos a través de técnicas de ciencia de datos, con el objetivo de mejorar la capacidad predictiva y ofrecer soluciones más personalizadas que se adapten a las necesidades del mercado actual.

3. Objetivos de la Investigación

3.1. Objetivo General

- Desarrollar y evaluar modelos predictivos basados en técnicas de aprendizaje automático que permitan identificar clientes con alta probabilidad de aceptar un préstamo personal en instituciones financieras.

3.2. Objetivos específicos:

- Seleccionar y aplicar distintos algoritmos de aprendizaje supervisado para construir modelos predictivos orientados a estimar la propensión de aceptación de préstamos personales
- Comparar el desempeño de distintos modelos predictivos mediante métricas de evaluación apropiadas para determinar cuál presenta mejor capacidad predictiva en el contexto analizado.
- Interpretar los resultados obtenidos para identificar patrones y tendencias en el comportamiento de los clientes.

4. Marco teórico

4.1. Introducción

El credit scoring comenzó a utilizarse de manera sistemática en la década de 1960 como una herramienta estadística destinada a evaluar la probabilidad de incumplimiento de los solicitantes de crédito. Su objetivo principal consistía en estimar el riesgo asociado a una operación crediticia y apoyar la toma de decisiones en la concesión de préstamos. Con el tiempo, estos modelos se consolidaron como uno de los instrumentos fundamentales para la gestión del riesgo en las instituciones financieras.

Sin embargo, el desarrollo de nuevas tecnologías, el crecimiento del crédito al consumo y la creciente disponibilidad de datos han ampliado el alcance de estas herramientas analíticas. En la actualidad, las instituciones financieras no solo utilizan modelos predictivos para mitigar riesgos, sino también para identificar oportunidades comerciales y optimizar estrategias de captación de clientes. En este contexto, el uso de técnicas de analítica avanzada permite pasar

de una lógica centrada exclusivamente en la evaluación del riesgo hacia un enfoque más amplio orientado a la generación de valor y a la mejora de la eficiencia comercial.

En la literatura especializada se distinguen distintos tipos de modelos basados en scoring. Por un lado, el credit scoring se orienta a estimar la probabilidad de incumplimiento de un cliente frente a una obligación crediticia. Por otro lado, los modelos conocidos como propensity scoring o modelos de activación comercial buscan estimar la probabilidad de que un cliente decida contratar un producto financiero determinado. Aunque ambos enfoques utilizan técnicas estadísticas similares, la diferencia fundamental radica en la variable objetivo que se pretende modelar: mientras el primero se enfoca en el riesgo de impago, el segundo se centra en la probabilidad de adopción de un producto.

El presente trabajo se inscribe dentro de esta segunda línea de investigación. En particular, se propone desarrollar un modelo predictivo orientado a identificar clientes con alta probabilidad de aceptar un préstamo personal, utilizando técnicas de aprendizaje supervisado aplicadas a datos históricos de comportamiento financiero. Este tipo de modelos resulta especialmente relevante para el diseño de campañas comerciales más eficientes, ya que permite focalizar los esfuerzos de marketing en aquellos clientes con mayor probabilidad de conversión.

En este contexto, la ciencia de datos emerge como una herramienta clave para transformar grandes volúmenes de información en conocimiento accionable. Mediante el uso de técnicas de modelización estadística y aprendizaje automático, es posible identificar patrones de comportamiento que permitan anticipar decisiones de los clientes y mejorar la eficiencia de las estrategias comerciales en el sector financiero.

Finalmente, el presente trabajo propone el desarrollo y evaluación de distintos modelos predictivos orientados a estimar la probabilidad de aceptación de un préstamo personal. El objetivo central no radica en la optimización exhaustiva de cada técnica de modelado, sino en analizar comparativamente el desempeño de enfoques representativos utilizados en problemas de clasificación, tales como la regresión logística, los árboles de decisión y las redes neuronales. Si bien es posible profundizar en ajustes más avanzados o en configuraciones más complejas de cada algoritmo, ello excedería el alcance de este estudio. En consecuencia, el foco del trabajo se orienta a evaluar la capacidad predictiva de estos modelos mediante métricas apropiadas para

problemas con clases desbalanceadas, permitiendo identificar cuál de ellos resulta más adecuado para apoyar estrategias comerciales en el ámbito bancario.

4.2. Conceptos Fundamentales

4.2.1. ¿Qué es el crédito?

La palabra "crédito" proviene del antiguo término latino "credo", que significa "confiar en" o "depender de". Si prestas algo a alguien, entonces debes tener confianza en que esa persona cumplirá con la obligación (Anderson, 2007).

Un crédito es una operación de financiación donde una persona llamada 'acreedor' (normalmente una entidad financiera), presta una cierta cifra monetaria a otro, llamado 'deudor', quien a partir de ese momento, garantiza al acreedor que retornará esta cantidad solicitada en el tiempo previamente estipulado más una cantidad adicional, llamada 'intereses' (Javier Montes de Oca, 2015).

4.2.2. ¿Qué es el scoring?

El scoring experto o estadístico es una técnica de evaluación que nos permite predecir la probabilidad de que un cliente cumpla con sus obligaciones financieras en una ventana de tiempo determinada. Esta metodología analiza de manera exhaustiva la información personal y crediticia de cada individuo, como su edad, ingresos e historial de pagos, para crear un perfil de riesgo personalizado. De esta forma, podemos identificar a aquellos clientes con mayor potencial de éxito (Anderson, 2007).

En términos simples, es un proceso que, a través de modelos matemáticos, convierte la información de un solicitante en una puntuación numérica. Esta puntuación refleja la probabilidad de que la persona cumpla con sus obligaciones de pago. En esencia, es como traducir un lenguaje complejo (los datos del cliente) a un código sencillo (el score) que las instituciones financieras pueden entender fácilmente.

El scoring proporciona una base sólida para tomar decisiones informadas sobre la aprobación o denegación de solicitudes de crédito. Al emplear modelos estadísticos, se reduce la subjetividad en el proceso de evaluación del riesgo crediticio. Aunque no puede predecir con

certeza el comportamiento de un individuo específico, el scoring permite estimar de manera confiable cómo se comportará, en promedio, un grupo de personas con características similares.

La innovación en el scoring crediticio está impulsando una transformación en la forma en que las instituciones financieras evalúan a sus clientes. Al aprovechar una gama más amplia de datos y aplicar modelos analíticos avanzados, las entidades pueden construir perfiles de cliente más completos y personalizados, optimizar sus procesos operativos y tomar decisiones estratégicas más acertadas. Esto se traduce en una mayor eficiencia, una mejor experiencia del cliente y un crecimiento sostenible del negocio.

4.2.3. Tipos de Modelos de Credit Scoring

Inicialmente, el scoring crediticio se limitaba a evaluar nuevas solicitudes de crédito, sirviendo como un filtro para decidir si se otorga o no un préstamo. Sin embargo, en el siglo XXI, este concepto ha evolucionado significativamente, ampliando su alcance para abarcar toda la gestión del crédito englobando cualquier modelo cuantitativo utilizado para evaluar el riesgo crediticio.

Estos modelos reciben diferentes nombres dependiendo de la fuente de información; el objetivo o lo que está midiendo (Anderson, 2007). Siguiendo la división planteada por Raymond Anderson (2007), los más comunes son:

- Application Score: Utilizado para dar origen a nuevos negocios, y combina datos del cliente, tratos anteriores y las agencias de crédito.
- Behavioural Score: Utilizado para la gestión de cuentas (establecimiento de límites, gestión de sobrepasos, autorizaciones), y generalmente se enfoca en el comportamiento de una cuenta individual. En su mayoría utilizan datos referentes al rendimiento del producto crediticio en cuestión. Con el tiempo, fue posible enriquecer estos modelos utilizando información demográfica, de agencia de informes crediticios o entidad de información crediticia. Estos modelos suelen ser a medida, desarrollados internamente en la entidad.
- Collections Score: utilizado como parte del proceso de cobranzas, generalmente para impulsar acciones en los centros de llamadas. Suele combinar información propia del

proceso de cobranzas (como promesa de pago, historial de contactos), con información de agencia de informes crediticios o entidad de información crediticia y otros productos.

- Customer Score: a diferencia del Behavioural Score y Collections Score, que evalúan a nivel de cuenta, éste combina la información de todas las cuentas de un cliente. Suele ser útil para identificar el perfil de riesgo global de un cliente para mejorar la oferta y realizar gestiones de cuenta más apropiadas. Suele utilizarse para cross-selling.
- Bureau Score: a diferencia del Behavioural Score y Collections Score, proporciona una evaluación integral del cliente. Esta visión holística es crucial para optimizar la gestión de cuentas, ya que permite identificar oportunidades de venta cruzada y realizar acciones de marketing más personalizadas

Cada modelo de scoring se nutre de una combinación única de datos, provenientes tanto del cliente como de fuentes internas y externas. Las nuevas tecnologías han transformado los modelos de scoring, permitiendo integrar una gran variedad de datos, desde el historial de transacciones hasta las interacciones en redes sociales. Esta diversidad de información permite obtener una visión más completa del comportamiento crediticio del cliente y, en consecuencia, tomar decisiones más precisas y personalizadas

4.2.4. Propensity scoring (evaluación de propensiones)

Propensity scoring (evaluación de propensiones) se refiere al uso de técnicas estadísticas para medir la propensión o inclinación de los clientes a comportarse de ciertas maneras específicas. En el contexto del crédito y la gestión de relaciones con los clientes, el propensity scoring se utiliza para predecir la probabilidad de que un cliente realice una acción particular o muestre cierto tipo de comportamiento.

Raymond Anderson (2007) detalla 4 dimensiones son fundamentales para entender y gestionar el comportamiento del cliente en el contexto del crédito y la toma de decisiones financieras. Las 4Rs: Risk (Riesgo), Response (Respuesta), Retention (Retención), Revenue (Ingresos) (Anderson, 2007).

- Riesgo: ¿El cliente hará algo que nos ponga en riesgo de pérdida financiera?
- Respuesta: ¿El cliente responderá a una oferta?

- Retención: ¿El cliente se quedará o se marchará?
- Ingresos: ¿Cuánto ingreso se espera?

a) Riesgo

El riesgo es el más conocido y ponderante en el puntaje crediticio, cubre no solo la probabilidad de pérdida, sino también la gravedad de la pérdida. Existen tres tipos básicos de puntaje de riesgo que utilizan las empresas:

- Puntaje de crédito: Los puntajes de riesgo crediticio se usan principalmente para predecir morosidades e incluyen la mayoría de los puntajes de solicitudes, comportamiento, clientes, cobros y burós. A menudo son los únicos puntajes utilizados para tomar decisiones.
- Puntaje de fraude: se considera un riesgo operativo y se trata de forma totalmente independiente. Se busca identificar a quienes no tienen intención de pagar
- Puntaje de seguros: Se busca predecir el reclamo de seguro a corto plazo. Si bien no está relacionado con los datos de crédito se ha demostrado que existe una fuerte correlación entre los datos de crédito y los reclamos a las aseguradoras.

b) Respuesta: Los Scores de Respuesta buscan lograr un mejor direccionamiento de las campañas de tal manera de limitar las mismas a aquellas personas que tienen mayor probabilidad de convertirse en clientes de la entidad o de aceptar un producto específico. Esto permite reducir costos de campaña y mejorar la tasa de aceptación.

c) Retención: La retención de clientes es un factor clave para la sostenibilidad de cualquier negocio. Evaluar la probabilidad de que un cliente se mantenga activo con nosotros, especialmente después de una oferta especial, es crucial para garantizar la rentabilidad de nuestras operaciones. Los modelos de scoring de deserción nos permiten identificar aquellos clientes con mayor riesgo de cancelación y diseñar estrategias proactivas para retenerlos

d) Ingresos: La optimización de los ingresos es un objetivo clave para cualquier negocio. Los modelos de scoring de ingresos nos permiten identificar a los clientes más rentables y a aquellos con mayor potencial de crecimiento. Al analizar variables como el historial

de transacciones, el comportamiento de navegación y la respuesta a las campañas de marketing, podemos diseñar estrategias personalizadas para maximizar el valor de vida del cliente.

4.2.5. Técnicas de desarrollo de Scoring

La elección de la técnica adecuada dependerá del tipo de datos disponibles, el tamaño de la muestra y el objetivo específico del modelo. Estas técnicas se dividen en paramétricas, ya que requieren suposiciones sobre el comportamiento estadístico de los datos, y no paramétricas, que no requieren ningún supuesto.

La selección del modelo estadístico más apropiado para un problema de clasificación depende en gran medida de las características del conjunto de datos y los objetivos del análisis. Modelos lineales como la regresión logística y el análisis discriminante son útiles cuando las relaciones entre las variables son lineales. Sin embargo, cuando las relaciones son más complejas, modelos no lineales como los árboles de decisión o las redes neuronales, algoritmos genéticos, k-vecino más cercano, pueden ofrecer un mejor ajuste. Además, factores como la interpretabilidad del modelo, la eficiencia computacional y la capacidad de manejar grandes volúmenes de datos también influyen en la elección final.

Tradicionalmente, el análisis discriminante y la regresión lineal dominaron el campo del Credit Scoring debido a su simplicidad y accesibilidad en software estadístico (Anderson, 2007). Sin embargo, la regresión logística ha emergido como la técnica preferida. Su capacidad para modelar variables dependientes binarias, la facilidad de interpretación de sus resultados y su robustez en situaciones de desequilibrio de clases la convierten en una herramienta poderosa para evaluar el riesgo crediticio.

4.2.6. Técnicas Paramétricas

Estas técnicas asumen una relación específica, generalmente lineal, entre las variables que influyen en el resultado de interés. Estos modelos se caracterizan por tener un número fijo de parámetros que deben ser estimados a partir de los datos. Su principal ventaja radica en su facilidad de interpretación y eficiencia computacional, lo que los hace ideales para situaciones donde la relación entre las variables es relativamente simple. Sin embargo, su principal

desventaja es que pueden ser demasiado restrictivos, limitando su capacidad para capturar relaciones más complejas que puedan existir en los datos.

4.2.6.1. Regresión Lineal:

La regresión lineal es una técnica estadística fundamental que busca modelar la relación lineal entre una variable dependiente y una o más variables independientes. Al asumir una relación lineal entre estas variables, la regresión lineal proporciona un modelo sencillo e interpretable para entender y predecir fenómenos (Montgomery et al., 2021) . Aunque su simplicidad puede limitarla en algunos casos, la regresión lineal sigue siendo una herramienta valiosa en diversos campos, desde las ciencias sociales hasta la ingeniería, debido a su capacidad para identificar patrones básicos y realizar inferencias estadísticas.

El modelo de regresión lineal (Legendre, Gauss, Galton y Pearson) considera que, dado un conjunto de observaciones $\{y_i, x_{i1}, \dots, x_{np}\}_{i=1}^{n}$, la media μ de la variable respuesta y se relaciona de forma lineal con la o las variables regresoras $x_1, \dots, x_p$ acorde a la ecuación (Amat Rodrigo, 2023) :

$$\mu_y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p$$

El resultado de esta ecuación se conoce como la línea de regresión poblacional, y recoge la relación entre los predictores y la media de la variable respuesta.

Otra definición que se encuentra con frecuencia en los libros de estadística es:

$$y_y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p + \epsilon_i$$

En este caso, y_y representa el valor observado de la variable respuesta para la i-ésima observación. El término ϵ_i corresponde al error aleatorio asociado a dicha observación, el cual captura la diferencia entre el valor observado y su valor esperado. Dado que una observación puntual difícilmente coincide exactamente con su media condicional, se incorpora este término aleatorio en el modelo.

En ambos casos, la interpretación de los elementos del modelo es la misma:

β_0: es la ordenada en el origen, se corresponde con el valor promedio de la variable respuesta y cuando todos los predictores son cero.

β_p: es el efecto promedio que tiene sobre la variable respuesta el incremento en una unidad de la variable predictora x_p, manteniéndose constantes el resto de variables. Se conocen como coeficientes parciales de regresión.

ϵ_i: es el residuo o error, la diferencia entre el valor observado y el estimado por el modelo. Recoge el efecto de todas aquellas variables que influyen en y pero que no se incluyen en el modelo como predictores.

En la gran mayoría de casos, los valores β_0y β_p poblacionales se desconocen, por lo que, a partir de una muestra, se obtienen sus estimaciones y $\widehat{\beta_{p0}}$ y $\widehat{\beta_p}$:. Ajustar el modelo consiste en estimar, a partir de los datos disponibles, los valores de los coeficientes de regresión que maximizan la verosimilitud (likelihood), es decir, los que dan lugar al modelo que con mayor probabilidad puede haber generado los datos observados.

El método empleado con más frecuencia es el ajuste por mínimos cuadrados ordinarios (OLS), que identifica como mejor modelo la recta (o plano si es regresión múltiple) que minimiza la suma de las desviaciones verticales entre cada dato de entrenamiento y la recta, elevadas al cuadrado.

El término "lineal" en los modelos de regresión hace referencia al hecho de que los parámetros se incorporan en la ecuación de forma lineal, no a que necesariamente la relación entre cada predictor y la variable respuesta tenga que seguir un patrón de recta (Amat Rodrigo, 2023).

Aunque la **regresión lineal** se utilizó para la mayoría de los primeros modelos de scoring, se sabía que tenía limitaciones, en gran parte porque la variable objetivo es binaria. La **regresión logística** es más adecuada para resultados binarios y, por lo tanto, para la mayoría de los modelos de **scoring crediticio**.

4.2.6.2. Regresión logística

Mientras que la regresión lineal se utilizó para proporcionar la mayor parte de los primeros modelos de puntuación, se sabía que tenía limitaciones, principalmente porque la variable objetivo es binaria. La regresión logística es más apropiada para resultados binarios y, por lo tanto, para la mayoría de la evaluación de crédito.

La regresión logística es una técnica estadística utilizada para modelar la probabilidad de que ocurra un evento binario (es decir, que tenga dos posibles resultados, como sí/no, éxito/fracaso) en función de una o más variables predictoras.

En esencia, la regresión logística busca establecer una relación entre variables independientes (también conocidas como predictores) y una variable dependiente binaria. A diferencia de la regresión lineal, que modela una relación lineal entre variables continuas, la regresión logística se enfoca en predecir la probabilidad de que ocurra un evento, y su resultado siempre está entre 0 y 1.

Por lo tanto, la regresión logística es un modelo estadístico de clasificación binaria que entrega la probabilidad de pertenencia a uno de los dos grupos definidos, utilizando para ello un conjunto de regresores (variables) $x_i \in R^n \; con \; i = \{1 \dots N\}$ y N el número de observaciones.

David Cox, en 1958, introdujo la regresión logística como un método para transformar los resultados continuos de una regresión lineal en probabilidades que oscilan entre 0 y 1. Esta transformación se logra típicamente mediante la función logística, también llamada sigmoide, que asigna cualquier valor real a un valor entre 0 y 1."

$$sigmoide = \; \sigma(y) = \frac{1}{1 + e^{-y}}$$

Para valores de y positivos muy grandes, e^{-y} es aproximadamente 0, por lo que el valor de la función sigmoide es 1. Para valores de y negativos muy grandes, e^{-y} tiende a infinito, por lo que el valor de la función sigmoide es 0.

Sustituyendo la y de la ecuación por la función de un modelo lineal $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p + \epsilon_i$ se obtiene que:

$$P(y = 1|X = x) = \frac{e^{(\beta_0+\beta_1 x_1+\cdots+\beta_p x_p)}}{1 + e^{(\beta_0+\beta_1 x_1+\cdots+\beta_p x_p)}}$$

Donde $\mathrm{P(y = 1|X = x)}$ puede interpretarse como la probabilidad de que la variable respuesta y adquiera el valor 1 (clase de referencia), dado los predictores $x_1, \ldots\ldots, x_p$.

La expresión obtenida tiene la característica de ser siempre positiva, ya que la función exponencial solo toma valores positivos y, el cociente de valores positivos, es siempre positivo. Esto hace posible aplicarle el logaritmo:

$$ln\left(\frac{p(y = 1|X = x)}{p(y = 0|X = x)}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p$$

Al realizar la transformación, en el lado derecho se obtiene la ecuación de un modelo lineal. El término de la izquierda resulta ser el logaritmo de un cociente de probabilidades, lo que se conoce como razón de probabilidad (log of odds).

Como resultado de este proceso se consigue convertir un problema de clasificación no lineal, en un problema de regresión lineal que sí puede ajustarse mediante los métodos convencionales.

Una vez obtenidos los coeficientes del modelo $(\beta_0, \beta_1, \ldots, \beta_p)$ se puede obtener la probabilidad de que una nueva observación pertenezca a la clase $y = 1$ con la ecuación:

$$p(y = 1|X = x) = \frac{e^{(\beta_0+\beta_1 x_1+\cdots+\beta_p x_p)}}{1 + e^{(\beta_0+\beta_1 x_1+\cdots+\beta_p x_p)}}$$

Modelar el *log of odds* es la estrategia matemática que permite encontrar los valores de los coeficientes (ajustar el modelo).

4.2.6.3. Interpretación del modelo

Los principales elementos que hay que interpretar en un modelo de regresión logística son los coeficientes de los predictores (Amat Rodrigo, 2023) :

- β_0 es la ordenada en el origen o intercept. Se corresponde con el valor esperado del logaritmo de odds cuando todos los predictores son cero. Puede transformarse a

probabilidad con la ecuación $\frac{e^{(\beta_0)}}{1+e^{(\beta_0)}}$ Tras la transformación, su valor se corresponde con la probabilidad esperada de pertenecer a la clase 1 cuando todos los predictores son cero.

- β_p los coeficientes de regresión parcial de cada predictor indican el cambio promedio del logaritmo de odds al incrementar en una unidad la variable predictora x_p, manteniéndose constantes el resto de variables. Esto equivale a decir que, por cada unidad que se incrementa x_p, se multiplican los odds por e^{β_p}.

Aunque en los modelos lineales los coeficientes de regresión suelen ser el primer punto de análisis, la interpretación completa del modelo abarca muchos otros factores. Entre estos se incluyen la significancia global del modelo, que evalúa si el modelo en su totalidad es útil para explicar la variabilidad en la variable de respuesta, y la significancia individual de cada predictor, que permite identificar cuáles de las variables predictoras tienen una relación estadísticamente relevante con la variable objetivo.

Cuando el único objetivo del modelo es realizar predicciones, estos aspectos pueden pasar a un segundo plano, ya que el enfoque suele centrarse exclusivamente en la precisión de las predicciones generadas. Sin embargo, si el propósito va más allá de la predicción y se orienta hacia la inferencia, es decir, hacia la explicación de cómo y por qué los predictores afectan la variable de respuesta, estos elementos de interpretación cobran gran importancia. En estos casos, es fundamental entender no solo si el modelo predice con exactitud, sino también cómo se relacionan los predictores con la respuesta y qué papel juega cada uno en la estructura del modelo.

4.2.6.4. Predicciones

Una vez ajustado el modelo de regresión logística, se procede a realizar predicciones sobre el conjunto de datos de prueba. Estas predicciones representan probabilidades de que la variable dependiente adopte el valor 1, es decir, que se presente el evento de interés (por ejemplo, "default = Sí" en un modelo de riesgo crediticio). Para obtener una predicción categórica (0 o 1), se establece un umbral, típicamente de 0.5, aunque este valor puede modificarse para optimizar la sensibilidad o la especificidad, dependiendo de los objetivos específicos de la aplicación.

4.2.7. Técnicas No Paramétricas

Las técnicas no paramétricas son un conjunto de métodos estadísticos utilizados para analizar datos cuando no se cumplen los supuestos de las pruebas paramétricas tradicionales. A diferencia de las pruebas paramétricas, que suelen asumir una distribución normal de los datos y homogeneidad de varianzas, las pruebas no paramétricas son más flexibles y pueden aplicarse a una variedad más amplia de situaciones. Bajo esta categoría se incluyen Árboles de Decisión (RPA), Redes Neuronales (NNs), algoritmos genéticos y el método de K vecinos más cercanos (K-NN) entre otros.

4.2.7.1. Árboles de Decisión (RPA)

Los **árboles de decisión** entre la viabilidad de usos, también se emplean para la visualización de datos en problemas de clasificación y predicción. En su forma más primitiva, un árbol de decisión es un tipo de sistema experto, donde un conjunto de reglas es definido por personas con experiencia práctica .

Los árboles de decisión son modelos de aprendizaje automático supervisado no paramétrico que se representan visualmente como estructuras jerárquicas en forma de árbol. Estos modelos se utilizan para clasificar datos en categorías o predecir valores numéricos. Comenzando por un nodo raíz, el árbol se ramifica en múltiples ramas, cada una representando una posible decisión basada en las características de los datos. Este proceso continúa hasta llegar a los nodos hoja, que representan las clases o valores finales predichos. La estructura arborescente facilita la interpretación de las decisiones tomadas por el modelo (Tan et al., 2019) .

Los árboles de decisión construyen modelos predictivos mediante una estrategia de 'divide y vencerás'. Inician con un conjunto de datos completo y lo dividen repetidamente (modo recursivo) en subconjuntos más pequeños y homogéneos, basados en las características más relevantes. Este proceso continúa hasta que se alcanza un criterio de detención, como alcanzar un número máximo de niveles o lograr una pureza suficiente en los nodos hoja, es decir, puntos de datos en una sola clase (Amat Rodrigo, 2023) . Sin embargo, árboles demasiado complejos pueden sobreajustarse a los datos de entrenamiento, lo que afecta su capacidad de generalizar a nuevos datos haciendo que sea más difícil mantener la pureza del nodo. Como resultado, los árboles de decisión tienen preferencia por los árboles pequeños, lo cual es consistente con el

principio de parsimonia en la Navaja de Occam. Es decir, "las entidades no deben multiplicarse más allá de la necesidad", es decir que los árboles de decisión deben agregar complejidad solo si es necesario (IBM, 2023).

La medida de selección de atributos para subconjuntos más pequeños y homogéneos es una heurística que permite seleccionar el criterio de división de los datos de la mejor manera posible. También se conoce como reglas de división porque ayuda a determinar puntos de interrupción para tuplas en un nodo dado. Las medidas de selección más populares son Ganancia de información, Proporción de ganancia e Índice de Gini.

El sobreajuste u overfitting es lo que reduce la capacidad predictiva del modelo al aplicarlo a nuevos datos, para prevenirlo y mejorar la capacidad de generalización de los árboles de decisión, se recurre a la poda (pruning) o limitar el tamaño del árbol (parada temprana).

El proceso de poda consiste en eliminar aquellas ramas del árbol que aportan poca información o generan divisiones poco significativas en los datos. Una vez podado el árbol, se evalúa su desempeño mediante técnicas como la validación cruzada.

En el proceso de parada temprana el tamaño final que adquiere un árbol puede controlarse mediante reglas de parada que detengan la división de los nodos dependiendo de si se cumplen o no determinadas condiciones. El nombre de estas condiciones puede variar dependiendo del software o librería empleada (Amat Rodrigo, 2020) .

Otra estrategia para mejorar la precisión y robustez de los modelos basados en árboles es el uso de bosques aleatorios. Esta técnica combina múltiples árboles de decisión, cada uno construido a partir de una muestra aleatoria de los datos, lo que reduce el riesgo de sobreajuste y mejora la capacidad predictiva del modelo

4.2.7.2. Redes Neuronales (NNs)

Las redes neuronales son modelos computacionales inspirados en la estructura y funcionamiento del cerebro humano. Están diseñadas para reconocer patrones en datos, aprender de ellos y realizar tareas complejas como el reconocimiento de imágenes, el procesamiento del lenguaje natural y la toma de decisiones.

Existen varios paradigmas que pueden usarse para desarrollar redes neuronales. El más adecuado para el *credit scoring* es el perceptrón multicapa (MLP), o retropropagación, que tiene la ventaja de manejar fácilmente tanto la no linealidad como las interacciones. También se le conoce como un "clasificador universal", ya que, en teoría, puede modelar cualquier proceso de decisión. Otros paradigmas incluyen la Función de Base Radial (RBF), los Mapas Autoorganizados (SOM) y las Redes de Kohonen (Anderson, 2007)

Las redes neuronales tienen la capacidad de manejar grandes volúmenes de datos y descubrir patrones complejos, lo que las hace útiles para identificar relaciones no lineales entre los datos de clientes, como hábitos de compra, historial crediticio, comportamiento de pago, entre otros factores relevantes para el scoring crediticio.

En el scoring de clientes, los métodos más tradicionales, como la regresión logística y el análisis discriminante, suelen preferirse por su simplicidad, interpretabilidad y menor riesgo de sobreajuste. Pero, sin embargo, las redes neuronales pueden ser útiles en escenarios donde la precisión predictiva es más importante que la interpretabilidad, como en determinación de fraudes o cuando los datos son altamente complejos y no lineales.

4.2.7.3. Elementos de una red neuronal artificial

El cerebro humano se compone principalmente de células nerviosas llamadas neuronas, conectadas entre sí a través de fibras denominadas axones. Cada vez que una neurona se estimula (por ejemplo, en respuesta a un estímulo), transmite activaciones nerviosas a otras neuronas a través de los axones. Las neuronas receptoras captan estas activaciones a través de estructuras llamadas dendritas, que son extensiones del cuerpo celular de la neurona. La fuerza del punto de contacto entre una dendrita y un axón, conocido como sinapsis, determina la conectividad entre las neuronas (Tan et al., 2019).

Una red neuronal artificial, que modela la estructura del cerebro humano, consta de múltiples unidades de procesamiento, o nodos, y estas entidades están conectadas por enlaces dirigidos. Los nodos son una representación artificial de las neuronas reales que realizan cálculos básicos restando y sumando sus entradas, y los enlaces dirigidos son equivalentes a las conexiones entre las neuronas, también conocidas como axones y dendritas. Finalmente, el peso de un enlace dirigido entre dos nodos artificiales es una metáfora del peso sináptico, ya que refleja la fuerza

de la conexión entre dos neuronas reales. De manera similar al propósito de los sistemas nerviosos biológicos, el objetivo de las redes neuronales artificiales es ajustar los pesos a los valores correctos para reflejar las relaciones de las entradas y salidas.

El uso de un modelo de RNA se centra principalmente en extraer características útiles de los atributos originales que sean más relevantes para la clasificación, siendo capaces de extraer conjuntos de características mucho más ricos, lo que resulta en un buen rendimiento de clasificación. Tradicionalmente, la extracción de características se ha logrado mediante técnicas de reducción de dimensionalidad, como el PCA.

Para tener un entendimiento aproximado de su funcionamiento se examina el enfoque clásico para el aprendizaje de modelos de RNA, tomando como base el modelo más simple llamado perceptrones (García, 2021) .

El equivalente artificial de una neurona biológica es el nodo o neurona artificial U_j cuyo esquema se muestra en la figura 2. Se trata de una unidad de procesamiento que combina linealmente n señales de entrada x_i multiplicando cada una de ellas por un peso sináptico w_{ij} para producir una entrada neta v_j

$$v_j = w_{1j}.x_1 + w_{2j}.x_2 + .. + w_{nj}.x_n$$

que utilizando el símbolo de sumatoria queda expresada como:

$$v_j = \sum_{i=1}^{i=n} w_{ij}\,.\,x_i$$

Si la entrada neta iguala o supera un cierto potencial umbral θ, entonces la neurona dispara un potencial o activación a_j de valor 1 y si no lo alcanza, la neurona permanece inactiva y la activación se considera de valor 0. Este comportamiento corresponde al modelo de una neurona binaria y se describe matemáticamente mediante una función de activación escalón, del siguiente modo:

$$a_j = f(v_j) = \begin{cases} 1 \; si \; v_j \geq \theta \\ 0 \; si \; v_j < \theta \end{cases}$$

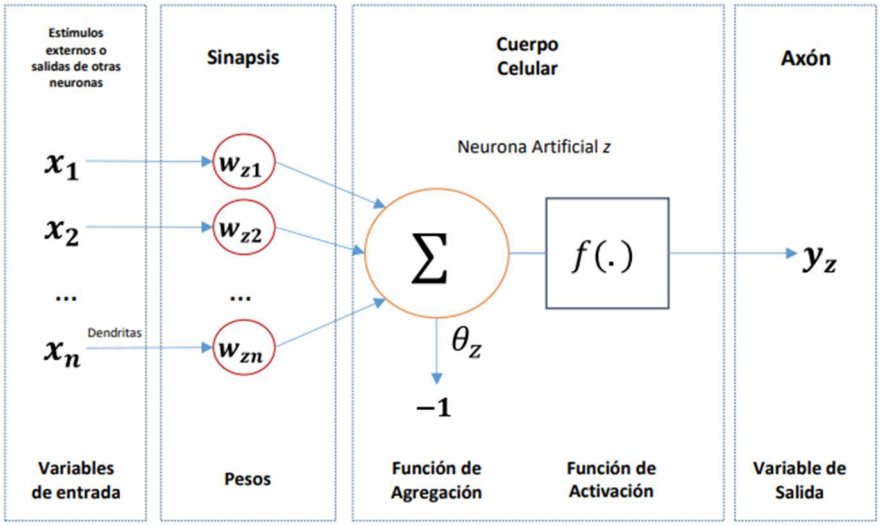


Figura 1: Modelo de neurona artificial

Fuente: Rico et al., 2009

En la mayor parte de modelos la función de activación $f(.)$ es monótona creciente y continua. En la figura 2, se expone las funciones de activación más usuales, en donde: x representa el potencial postsináptico y el estado de activación.

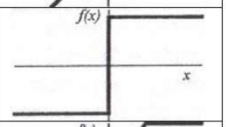
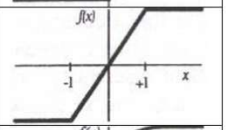
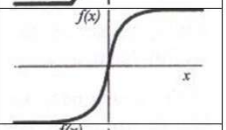
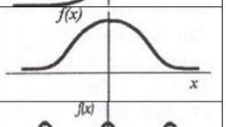
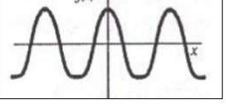
	Función	Rango	Gráfica
Identidad	$y = x$	$[-\infty, \infty]$	
Escalón	$y = \begin{cases} 1, & si\ x \geq 0 \\ 0, & si\ x < 0 \end{cases}$ $y = \begin{cases} 1, & si\ x \geq 0 \\ -1, & si\ x < 0 \end{cases}$	$[0, 1]$ $[-1, 1]$	
Lineal a tramos	$y = \begin{cases} 1, & si\ x > 1 \\ x, & si - 1 \leq x \leq 1 \\ -1, & x < -1 \end{cases}$	$[-1, 1]$	
Sigmoidea	$y = \frac{1}{1 + e^{-x}}$ $y = \tanh(x)$	$[0, 1]$ $[-1, 1]$	
Gaussiana	$y = Ae^{-Bx^2}$	$[0, A]$	
Sinusoidal	$y = A\sin(wx + \varphi)$	$[-A, A]$	

Figura 2: Funciones de activación habituales

Fuente: Palacios Burgos, 2024

4.2.8. Conceptos del proceso de modelización

4.2.8.1. Matriz de confusión

Una matriz de confusión, o matriz de error, es una herramienta de visualización que muestra los resultados de un modelo de clasificación, organizando en una tabla el número de instancias reales de una clase específica frente al número de instancias que el modelo ha clasificado en esa clase. Este recurso es fundamental en la evaluación de modelos de clasificación, ya que permite derivar métricas de rendimiento como la precisión y la concordancia, entre otras.

La matriz de confusión es aplicable a distintos algoritmos clasificadores, como Naïve Bayes, regresión logística y árboles de decisión. Debido a su utilidad en el ámbito de la ciencia de datos y el aprendizaje automático, muchas bibliotecas incluyen funciones para generar matrices de confusión, como el módulo sklearn.metrics de scikit-learn en Python (Jacob Murel et al., 2024)

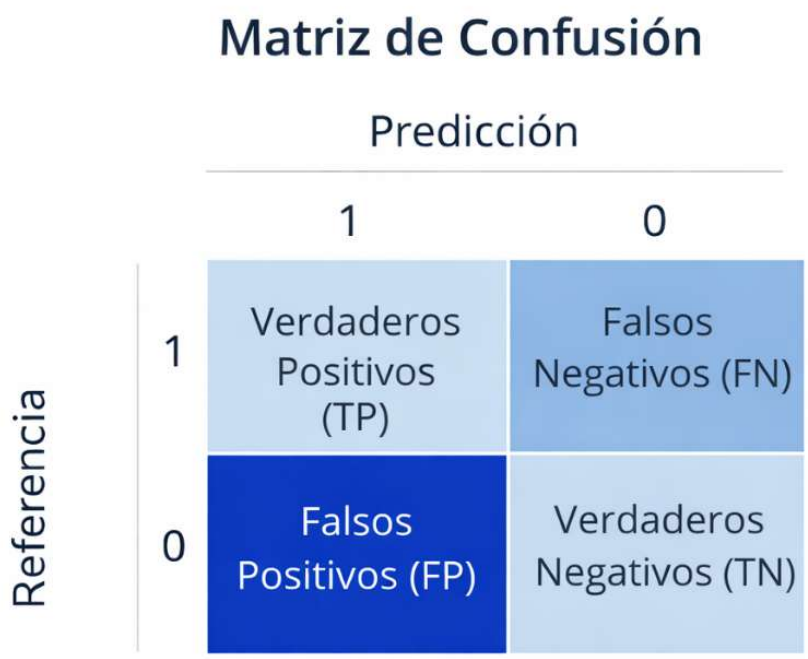


Figura 3: Matriz de confusión

Elaboración Propia

Cuadro **superior izquierdo (TP - Verdaderos Positivos)**: Representa el número de instancias en las que el modelo predijo correctamente la clase positiva. Es decir, predijo "Positivo" y la instancia realmente era "Positiva".

Cuadro superior derecho (FN - Falsos Negativos): Muestra los casos en los que el modelo falló al predecir la clase positiva, clasificando incorrectamente como "Negativo" cuando la instancia realmente era "Positiva".

Cuadro inferior izquierdo (FP - Falsos Positivos): Indica las instancias negativas que el modelo identificó incorrectamente como positivas. Estos se conocen como errores de tipo I en estadística.

Cuadro inferior derecho (TN - Verdaderos Negativos): Representa los casos en los que el modelo predijo correctamente la clase negativa. Es decir, predijo "Negativo" y la instancia realmente era "Negativa".

A partir de esta matriz, se calculan métricas de evaluación importantes para el modelo:

- **Exactitud (Accuracy)**: Mide la **proporción total de predicciones correctas** (tanto positivas como negativas) sobre el total de predicciones realizadas. Es una métrica general que indica qué tan bien está funcionando el modelo en su conjunto.

$$Exactitud = \frac{TP + TN}{TP + TN + FP + FN}$$

- **Sensibilidad (Recall o Tasa de Verdaderos Positivos)**: Proporción de verdaderos positivos sobre el total de observaciones positivas reales.

$$Sensibilidad = \frac{TP}{TP + FN}$$

- **Especificidad**: Proporción de verdaderos negativos sobre el total de observaciones negativas reales.

$$Especificidad = \frac{TN}{TN + FP}$$

- **Precisión (Precision)**: Mide la **proporción de predicciones positivas correctas** sobre todas las predicciones positivas hechas por el modelo. En otras palabras, es una medida de qué tan confiables son las predicciones positivas del modelo. Se usa cuando el costo de un falso positivo es alto.

$$Precisión = \frac{TP}{TP + FP}$$

4.2.8.2. Curva ROC y AUC (Área Bajo la Curva)

La curva ROC es una representación gráfica del rendimiento de un modelo de clasificación a través de distintos umbrales de decisión. Muestra la relación entre la **Tasa de Verdaderos Positivos** (TPR o Sensibilidad) y la **Tasa de Falsos Positivos** (FPR) para cada valor posible del umbral de decisión (Tan et al., 2019).

El área bajo la curva ROC (AUC, por sus siglas en inglés) es un valor numérico que resume el rendimiento de la curva ROC en un solo valor, proporcionando una métrica de evaluación única para el modelo. El AUC mide la capacidad del modelo para clasificar correctamente las instancias positivas y negativas de forma consistente.

Interpretación del AUC:

- **AUC = 1**: Indica un modelo perfecto que clasifica todas las instancias correctamente, sin falsos positivos ni falsos negativos.
- **AUC = 0.5**: Representa un modelo sin capacidad discriminativa, equivalente a hacer una predicción aleatoria. Esto significa que el modelo no es mejor que el azar.
- **0.5 < AUC < 1**: Un AUC entre 0.5 y 1 indica un modelo con cierto grado de poder predictivo. Cuanto más cercano esté el AUC a 1, mejor será la capacidad del modelo para separar las clases.

5. Estudio experimental

5.1. Set de datos

5.1.1. Descripción de los datos

Debido a los estrictos acuerdos de confidencialidad que protegen los datos en instituciones financieras, el presente estudio se basa en un conjunto de datos de acceso público. El dataset seleccionado fue obtenido de Kaggle, una plataforma reconocida a nivel mundial que permite a investigadores y profesionales compartir y acceder a datos para realizar análisis y desarrollar modelos predictivos. Este dataset, titulado Bank Loan Modelling, contiene información relevante sobre características socioeconómicas y demográficas de individuos, variables habitualmente empleadas en la construcción de modelos de predicción de préstamos (Walke, 2019). La elección de un conjunto de datos públicos facilita la replicabilidad de los resultados y asegura que el estudio se alinee con principios éticos y de transparencia, aspectos fundamentales en la investigación científica.

El set de datos contiene datos de 5000 clientes. Los datos incluyen información demográfica del cliente (edad, ingresos, etc.), la relación del cliente con el banco (hipoteca, cuenta de valores, etc.) y la respuesta del cliente a la última campaña de préstamo personal (Préstamo Personal). De estos 5000 clientes, solo 480 (= 9,6%) aceptaron el préstamo personal que se les ofreció en la campaña anterior.

Dominio: Banca

Fuente: https://www.kaggle.com/datasets/krantiswalke/bank-personal-loan-modelling

5.1.2. Contexto

Este caso se trata de un banco (Thera Bank) cuya gerencia quiere explorar formas de convertir a sus clientes de pasivos en clientes de préstamos personales (mientras los mantiene como depositantes). Una campaña que el banco llevó a cabo para clientes pasivos mostró una tasa de conversión saludable de más del 9% de éxito. Esto ha generado al departamento de marketing minorista idear campañas con un mejor marketing dirigido para aumentar la tasa de éxito con un presupuesto mínimo.

5.1.3. Información básica del conjunto de datos:

El conjunto de datos contiene datos de 5000 clientes.

- 6 variables numéricas: ID, Age, Experience, Income, CC_Avg, Mortgage

- 3 variables categóricas: Family, Education, ZIP_Code

- 5 variables booleanas: Personal_Loan, Securities_Account, CD_Account, Online, CreditCard

Si bien el conjunto de datos incluye variables que podrían permitir la incorporación de información externa para enriquecer el análisis, este tipo de integración no fue considerado en el presente estudio. La incorporación de fuentes adicionales podría aportar variables complementarias potencialmente relevantes para el modelado predictivo; sin embargo, su utilización implicaría procesos adicionales de obtención, integración y validación de datos que exceden el alcance de esta investigación. En consecuencia, el análisis se realizó utilizando exclusivamente las variables disponibles en el dataset original.

5.1.4. Información de atributos

A continuación, se presenta las variables incluidas en el conjunto de datos utilizado en el estudio, junto con su tipo y descripción

Variable	**Tipo**	**Descripción**
ID	Numérica	Identificador único del cliente.
Age	Numérica	Edad del cliente en años.
Experience	Numérica	Años de experiencia profesional del cliente.
Income	Numérica	Ingreso anual del cliente expresado en miles de dólares ($000).
ZIP_Code	Categórica	Código postal del domicilio del cliente.
Family	Categórica	Tamaño del grupo familiar del cliente.
CCAvg	Numérica	Gasto promedio mensual con tarjeta de crédito ($000).

Education	Categórica	Nivel educativo del cliente (1: Licenciatura, 2: Graduado, 3: Profesional/Avanzado).
Mortgage	Numérica	Valor de la hipoteca del cliente, si la posee ($000).
Personal_Loan	Binaria	Variable objetivo. Indica si el cliente aceptó el préstamo personal ofrecido en la última campaña.
Securities_Account	Binaria	Indica si el cliente posee una cuenta de valores en el banco.
CD_Account	Binaria	Indica si el cliente posee un certificado de depósito (CD) en el banco.
Online	Binaria	Indica si el cliente utiliza servicios de banca en línea.
CreditCard	Binaria	Indica si el cliente posee una tarjeta de crédito emitida por el banco.

5.2. Análisis exploratorio de datos (EDA)

5.2.1. Análisis descriptivo variables numéricas

Para comprender la distribución de los datos en nuestro estudio, se realizó un análisis descriptivo de las variables numéricas. A continuación, se presentan las estadísticas básicas, incluyendo media, mediana, desviación estándar, valores mínimo y máximo, así como los cuartiles (Q1 y Q3) y el rango intercuartílico (IQR).

	skim_variable <chr>	n_missing <int>	complete_rate <dbl>	mean <dbl>	sd <dbl>	p0 <dbl>	p25 <dbl>	p50 <dbl>	p75 <dbl>	p100 <dbl>	hist <chr>
1	Age	0	1	45.338400	11.463166	23	35.0	45.0	55.0	67	▆▆▆▅▃
2	Experience	0	1	20.104600	11.467954	-3	10.0	20.0	30.0	43	▆▆▆▅▃
3	Income	0	1	73.774200	46.033729	8	39.0	64.0	98.0	224	▇▆▃▂▁
4	Family	0	1	2.396400	1.147663	1	1.0	2.0	3.0	4	▇▇▁▁▂
5	CCAvg	0	1	1.937913	1.747666	0	0.7	1.5	2.5	10	▇▃▁▁▁
6	Mortgage	0	1	56.498800	101.713802	0	0.0	0.0	101.0	635	▇▁▁▁▁
7	ZIP_Code	0	1	93152.503000	2121.852197	9307	91911.0	93437.0	94608.0	96651	▁▁▁▁▇

7 rows

Figura 4: Estructura general del dataset
Fuente: Elaboración propia.

Las variables analizadas presentan una tasa de completitud del 100%, lo que indica ausencia total de valores faltantes y permite un análisis confiable.

- Edad (Age) y Experiencia (Experience) exhiben una distribución centrada, con medias en torno a 45 y 20 años respectivamente. Sin embargo, se observa un valor mínimo negativo en Experience (-3), lo cual sugiere una posible inconsistencia o error de carga en los datos.
- Income muestra una media elevada y una dispersión importante (desvío estándar de 46 mil), con valores extremos que alcanzan los 224 mil, lo que justifica el análisis de su asimetría y su impacto sobre el modelo.
- Mortgage presenta una distribución altamente asimétrica: la mediana y los percentiles inferiores se ubican en cero, mientras que el valor máximo llega a 635, revelando una gran proporción de individuos sin hipoteca y un grupo minoritario con montos significativamente altos.
- CCAvg (gasto promedio con tarjeta de crédito) y Family (tamaño del grupo familiar) mantienen distribuciones más compactas, aunque con cierta asimetría hacia valores altos en el caso de CCAvg (máximo de 10).
- ZIP_Code varía en un rango amplio pero esperable, sin presencia de valores atípicos evidentes.

5.2.2. Análisis descriptivo variables categóricas

El análisis de distribución de las variables categóricas permite caracterizar el perfil financiero y de comportamiento de los clientes en la muestra analizada. Variables como **CD_Account** y **Securities_Account** muestran una marcada concentración en la categoría negativa, evidenciando que la mayoría de los individuos no poseen estos instrumentos financieros. Esta baja puede señalar oportunidades de expansión comercial en productos de inversión.

La variable **CreditCard** también presenta una proporción mayoritaria de clientes sin tarjeta de crédito, aunque con una distribución más equilibrada en comparación con las anteriores. **Online** refleja un predominio claro de clientes con acceso a canales digitales, lo cual refuerza la necesidad de considerar el comportamiento digital como factor relevante en modelos de propensión y retención.

Respecto a **Education**, se identifica una distribución diversificada entre los distintos niveles educativos, con una ligera concentración en la categoría intermedia. Esta heterogeneidad permite evaluar el efecto del nivel formativo en la toma de decisiones financieras.

Finalmente, la variable **Personal_Loan**, que representa el objetivo de modelado, muestra un desbalance notorio con mayoría de casos negativos (no adquisición de préstamo). Esta asimetría deberá ser tratada adecuadamente en las etapas de entrenamiento del modelo, ya que puede impactar en la capacidad predictiva y en la sensibilidad del clasificador frente a la clase minoritaria.

Figura 5: Variables categóricas
Fuente: Elaboración propia.

5.2.3. Matriz de correlación

Con el fin de explorar la estructura interna de los datos y evaluar posibles relaciones lineales entre variables, se llevó a cabo un análisis de correlación. La matriz resultante permitió identificar patrones significativos de asociación entre las variables predictoras y la variable objetivo (Personal_Loan), así como entre los propios predictores.

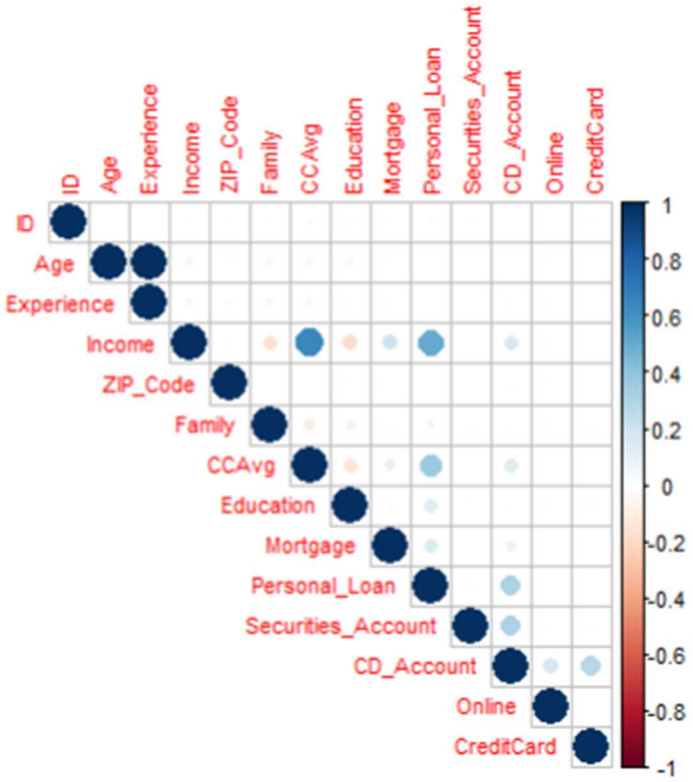


Figura 6: Matriz de correlación
Fuente: Elaboración propia.

El análisis de correlación permitió identificar relaciones significativas entre la variable objetivo Personal_Loan y ciertos atributos predictivos. En particular, se observaron correlaciones positivas de magnitud moderada con variables como CD_Account, Income, CCAvg y, en menor medida, Education. Estos resultados sugieren que la tenencia de productos financieros complementarios, un mayor nivel de ingresos y actividad transaccional, así como un mayor nivel educativo, se asocia con una mayor propensión a aceptar un préstamo personal. Dichas variables reflejan perfiles de bancarización y capacidad económica que resultan relevantes para la toma de decisiones crediticias.

Por otro lado, no se detectaron correlaciones excesivamente altas entre variables predictoras ($|r| > 0.9$), lo cual indica una baja colinealidad multivariante. Este hallazgo favorece la estabilidad de los modelos estadísticos y reduce el riesgo de distorsión en la estimación de parámetros, especialmente en algoritmos sensibles a la redundancia estructural.

Sin embargo, se identificó una correlación negativa inusualmente fuerte entre Age y Experience, lo que podría responder a una construcción artificial del dataset o a una codificación redundante. Esta relación debe ser evaluada con cautela, ya que podría introducir sesgos o duplicidades en el modelado predictivo si no se trata adecuadamente.

Finalmente, variables como ZIP_Code, Securities_Account e ID presentaron coeficientes de correlación cercanos a cero tanto respecto de la variable objetivo como entre sí. Esto sugiere una baja relevancia lineal, lo que podría justificar su exclusión en etapas posteriores de selección de atributos, salvo que demuestren aportes no lineales o sinérgicos mediante técnicas más complejas como árboles de decisión o redes neuronales.

En conjunto, este análisis proporciona un marco preliminar para la selección informada de variables, facilitando una interpretación más clara del fenómeno crediticio y contribuyendo al diseño de modelos predictivos más robustos, eficientes y alineados con la lógica comercial del negocio bancario.

5.2.4. Valores atípicos

Con el propósito de explorar la distribución estadística y posibles anomalías en las variables cuantitativas seleccionadas, se construyeron diagramas de caja (boxplots). Esta representación gráfica, acompañada de los límites teóricos de valores atípicos definidos por el rango intercuartílico (IQR). De esta manera se puede identificar la presencia y magnitud de observaciones fuera del rango esperado.

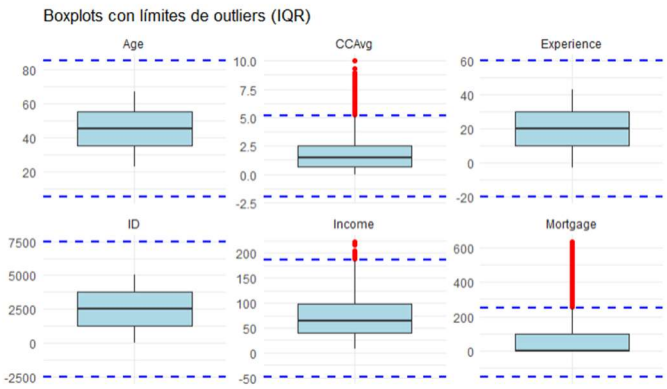


Figura 7: Boxplots valores atípicos
Fuente: Elaboración propia.

En particular, se evidencia una asimetría positiva en la distribución de tres variables Income, CCAvg y Mortgage.

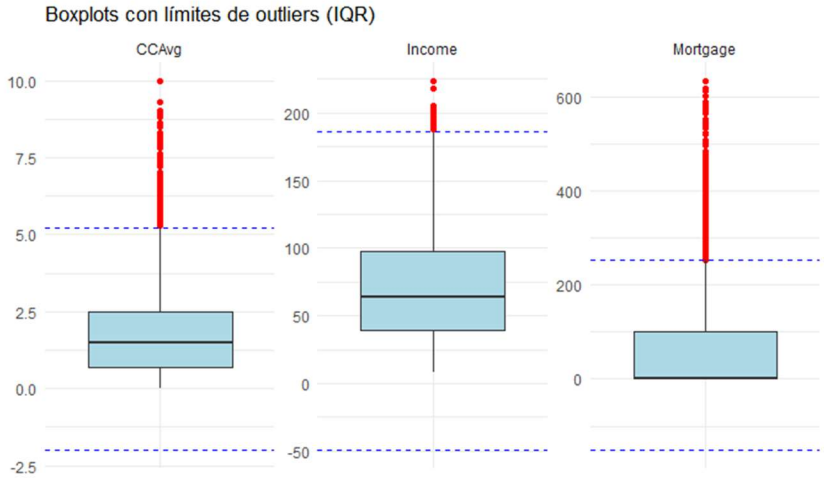


Figura 8: Boxplots valores atípicos con limites
Fuente: Elaboración propia.

Desde el punto de vista analítico, estas observaciones deben ser abordadas con cautela. Si bien pueden representar casos extremos legítimos asociados a clientes de alto perfil económico, también podrían influir desproporcionadamente en los modelos predictivos si no se controlan adecuadamente. Por tanto, resulta pertinente considerar estrategias como la transformación logarítmica, el recorte (winsorization) o la normalización robusta, dependiendo del enfoque metodológico adoptado en las etapas subsiguientes del análisis.

5.3. Pre-procesamiento de Datos

5.3.1. Depuración de Variables del Conjunto de Datos

Con el objetivo de optimizar la calidad del modelo y evitar la inclusión de variables con bajo aporte informativo o riesgo de generar ruido, se llevó a cabo un proceso de depuración del conjunto de datos. En primer lugar, la variable **ID** fue analizada durante la etapa exploratoria con el fin de comprender la estructura del dataset; sin embargo, posteriormente fue removida del proceso de modelado por tratarse de un identificador único de cada observación, carente de valor predictivo y sin relación estructural con la variable objetivo.

De forma similar, la variable **ZIP_Code** fue considerada en la fase de análisis exploratorio para evaluar su posible relevancia; no obstante, al tratarse de un atributo geográfico codificado numéricamente y no evidenciar un aporte explicativo significativo en el contexto del dataset utilizado, se decidió excluirla del conjunto final de variables utilizadas en el modelado.

Finalmente, la variable **Age** fue descartada debido a su alta correlación negativa con la variable **Experience**, lo que sugiere una redundancia estructural entre ambas. En virtud de ello, se decidió conservar sólo una de estas variables con el fin de evitar posibles problemas de multicolinealidad durante el proceso de modelado.

Esta selección de variables permitió trabajar con un conjunto de predictores más parsimonioso y focalizado, manteniendo aquellos atributos con mayor relevancia potencial en la predicción de la variable objetivo.

5.3.2. Tratamiento de Valores Faltantes

Como parte del análisis exploratorio de datos (EDA), se realizó una evaluación de la presencia de valores faltantes en el conjunto de datos utilizando la función skim () del paquete **skimr**.

Los resultados indican que todas las variables presentan **n_missing = 0** y una **complete_rate = 1**, lo que confirma que el dataset no contiene valores faltantes.

En consecuencia, no fue necesario aplicar técnicas de tratamiento o imputación de datos faltantes, permitiendo continuar directamente con el análisis exploratorio y el proceso de modelado.

	skim_variable <chr>	n_missing <int>	complete_rate <dbl>
1	ID	0	1
2	Age	0	1
3	Experience	0	1
4	Income	0	1
5	ZIP_Code	0	1
6	Family	0	1
7	CCAvg	0	1
8	Education	0	1
9	Mortgage	0	1
10	Personal_Loan	0	1

Figura 9 : Valores Faltantes

Fuente: Elaboración propia.

No obstante, es importante mencionar que en la literatura existen diversas técnicas comúnmente utilizadas para el tratamiento de valores faltantes. Entre las más empleadas se encuentran:

- **Eliminación de registros (Listwise Deletion):** consiste en eliminar las observaciones que contienen valores faltantes. Este método es adecuado cuando el número de datos faltantes es reducido y su eliminación no afecta significativamente el tamaño de la muestra.
- **Imputación mediante medidas de tendencia central:** se reemplazan los valores faltantes utilizando estadísticas como la media, mediana o moda de la variable correspondiente. Es una técnica simple y ampliamente utilizada en análisis exploratorios.
- **Imputación basada en modelos:** utiliza algoritmos estadísticos o de aprendizaje automático para estimar los valores faltantes, como regresión, K-Nearest Neighbors (KNN) o métodos de imputación múltiple.
- **Imputación múltiple:** genera varias estimaciones posibles para los valores faltantes y combina los resultados para obtener estimaciones más robustas, reduciendo el sesgo en el análisis.

5.3.3. Tratamiento de Valores Atípicos

La detección y el tratamiento de valores atípicos constituye una etapa crítica en la preparación de datos para modelos de aprendizaje supervisado. La presencia de observaciones extremas, especialmente en variables numéricas como Income, CCAvg y Mortgage, puede distorsionar las estimaciones de los modelos, afectar la estabilidad de los coeficientes y reducir la generalización del sistema predictivo.

El análisis exploratorio realizado previamente permite identificar la distribución altamente asimétrica en las mencionadas variables, con colas superiores extendidas que reflejan la existencia de un subconjunto de individuos con características económicas atípicamente elevadas.

Para mitigar su impacto, se aplicó la técnica de Winsorización (Winsorizing), que consiste en limitar los valores extremos a determinados percentiles (típicamente el 1% y el 99%), conservando la estructura general de la variable sin eliminar registros. Esta estrategia permite atenuar la influencia de los outliers severos sin incurrir en una pérdida de información valiosa.

5.3.4. Balanceo de Clases

Como el objetivo principal es identificar a aquellos clientes con alta probabilidad de aceptar un préstamo personal, a partir de sus características demográficas y financieras. Se analiza inicialmente la distribución de la variable objetivo Personal_Loan, que indica si el cliente aceptó (valor 1) o no aceptó (valor 0) el préstamo ofrecido durante una campaña anterior del banco.

Del total de 5.000 observaciones, se registraron 4.520 casos en los que el cliente no aceptó el préstamo (90,4%) y solamente 480 casos positivos (9,6%). Esta distribución revela un fuerte desbalance de clases, situación común en problemas de clasificación binaria donde el evento de interés (aceptación del préstamo) es poco frecuente.

Este desbalance implica que el modelo predictivo podría sesgar hacia la clase mayoritaria si no se toman medidas específicas durante el entrenamiento. Es decir, un modelo podría alcanzar una precisión global (accuracy) alta simplemente prediciendo siempre la clase "no" (0), pero sin lograr identificar correctamente a los clientes con mayor propensión a aceptar un préstamo, que son precisamente los casos de mayor interés para el banco.

Para atacar este problema, se usan técnicas de tratamiento de desbalance con el objetivo de mejorar la capacidad del modelo para detectar correctamente la clase minoritaria:

a) *Aplicación de SMOTE (Synthetic Minority Over-sampling Technique)*: En problemas de clasificación donde la variable objetivo presenta un fuerte desbalance entre clases, existen diversas estrategias para mitigar este efecto. Entre las más comunes se encuentran el submuestreo de la clase mayoritaria (undersampling), que reduce el número de observaciones de la clase dominante; el sobremuestreo de la clase minoritaria (oversampling), que replica observaciones existentes para equilibrar la distribución; y

la ponderación de clases, que asigna mayor peso a los errores cometidos sobre la clase minoritaria durante el entrenamiento del modelo.

En el presente estudio se optó por utilizar la técnica de sobremuestreo sintético SMOTE, implementada mediante la librería DMwR, con el fin de igualar la proporción entre clases en el conjunto de entrenamiento. Este método genera nuevas observaciones sintéticas de la clase minoritaria a partir de combinaciones de vecinos cercanos, sin recurrir a duplicaciones simples de los registros existentes. Además, el balanceo se aplicó únicamente sobre el conjunto de entrenamiento, manteniendo el conjunto de test con su distribución original para evaluar el desempeño de manera realista.

Como resultado, se obtiene un conjunto de entrenamiento más balanceado que permite mejorar la capacidad del modelo para identificar correctamente la clase minoritaria, preservando al mismo tiempo la distribución original del conjunto de test para realizar una evaluación más realista del desempeño predictivo.

b) *Evaluación con métricas apropiadas para clases desbalanceadas*: Con este análisis se detecta que utilizar la métrica de precisión global (accuracy) como único criterio de rendimiento resulta inadecuado en contextos desbalanceados. En su lugar, se decidió evaluar el desempeño de los modelos mediante métricas más sensibles al comportamiento de la clase positiva, tales como:

    i. Recall (Sensibilidad): proporción de verdaderos positivos identificados correctamente.
    ii. Precisión (Precision): proporción de verdaderos positivos sobre el total de positivos predichos.
    iii. F1-score: media armónica entre precisión y recall.
    iv. AUC-ROC: medida global de discriminación del modelo entre clases, basada en la relación entre la tasa de verdaderos positivos y la tasa de falsos positivos a lo largo de distintos umbrales de clasificación.
    v. Kolmogorov–Smirnov (KS): métrica que mide la máxima distancia entre las distribuciones acumuladas de las probabilidades estimadas para las clases positiva y negativa, indicando la capacidad del modelo para discriminar entre clientes que aceptan y no aceptan el préstamo.

    Interpretación:

    - < 0.20 Modelo débil

- 0.20 – 0.40 Modelo aceptable
- 0.40 – 0.60 Modelo bueno
- > 0.60 Modelo muy fuerte

Estas métricas permitieron realizar una evaluación más robusta del desempeño de los modelos, alineada con el objetivo del estudio de identificar de manera eficiente a los clientes con mayor propensión a aceptar un préstamo personal.

5.4. Construcción del Modelo y Diagnóstico del Modelo

5.4.1. División del Conjunto de Datos en Entrenamiento y Prueba

Con el fin de entrenar y evaluar de forma objetiva los modelos predictivos, se divide el conjunto de datos en dos subconjuntos mutuamente excluyentes: uno destinado al entrenamiento del modelo (train) y otro reservado para su evaluación (test).

La partición se realiza con el procedimiento de muestreo estratificado provisto por la función sample.split() del paquete caTools, con una proporción del 70% de los casos asignados al conjunto de entrenamiento y el 30% restante al conjunto de prueba. Se fijó una semilla aleatoria (set.seed(123)) para garantizar la reproducibilidad del proceso.

Este enfoque permite estimar el desempeño del modelo sobre datos no vistos durante el entrenamiento, preservando la distribución original de la variable objetivo (Personal_Loan) en ambas particiones. De esta manera, se asegura una evaluación robusta y representativa de la capacidad generalizadora del modelo.

En contextos más complejos de modelado predictivo, es habitual dividir los datos en conjuntos adicionales, como un conjunto de validación (validation set) destinado a la optimización de hiperparámetros, o incluso utilizar esquemas de validación temporal (out-of-time, OOT), donde se reservan observaciones correspondientes a períodos posteriores para evaluar el comportamiento del modelo en escenarios más cercanos a condiciones reales de operación. Este tipo de validación resulta especialmente útil para detectar posibles cambios en la distribución de los datos (data drift) a lo largo del tiempo.

No obstante, dado el alcance del presente estudio y la naturaleza del dataset utilizado, se optó por una partición clásica en conjunto de entrenamiento y conjunto de prueba, suficiente para evaluar de manera objetiva el desempeño comparativo de los modelos propuestos

5.5. Modelos

Una vez completado el proceso de limpieza, transformación y balanceo de los datos, se procedió a la construcción de distintos modelos de aprendizaje supervisado con el objetivo de predecir si un cliente aceptará un préstamo personal. La selección del modelo adecuado es crucial, ya que incide directamente en la eficacia del sistema predictivo y, por ende, en la toma de decisiones estratégicas de la entidad financiera.

En este trabajo se han implementado tres modelos de clasificación, representativos de enfoques distintos dentro del aprendizaje supervisado:

- Regresión logística.
- Árbol de decisión.
- Red neuronal artificial.

5.5.1. Modelo 1: Regresión Logística

La Regresión Logística fue elegida como modelo base debido a su facilidad de interpretación y su capacidad para identificar la relación entre variables predictoras y la aceptación de préstamos personales.

5.5.1.1. Resultados del Modelo de Regresión Logística

```
Call:
glm(formula = Personal_Loan ~ ., family = "binomial", data = datos_balanceados)

Coefficients:
                     Estimate Std. Error z value Pr(>|z|)
(Intercept)        -1.284e+01  4.225e-01 -30.402  < 2e-16 ***
Experience          2.911e-03  4.402e-03   0.661   0.5084
Income              5.948e-02  1.976e-03  30.100  < 2e-16 ***
Family              6.752e-01  4.786e-02  14.108  < 2e-16 ***
CCAvg               3.152e-01  3.307e-02   9.531  < 2e-16 ***
Education           1.928e+00  8.411e-02  22.917  < 2e-16 ***
Mortgage            7.955e-04  4.205e-04   1.892   0.0585 .
Securities_Account -1.378e+00  2.334e-01  -5.903 3.56e-09 ***
CD_Account          5.030e+00  2.708e-01  18.575  < 2e-16 ***
Online             -6.525e-01  1.134e-01  -5.756 8.62e-09 ***
CreditCard         -1.129e+00  1.369e-01  -8.249  < 2e-16 ***
```

```
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 8575.2  on 6187  degrees of freedom
Residual deviance: 2735.2  on 6177  degrees of freedom
AIC: 2757.2

Number of Fisher Scoring iterations: 7
```

El modelo obtenido presentó un ajuste adecuado, con una deviance residual de 2735,2 y un AIC de 2757,2, indicando una mejora considerable respecto a la deviance nula (8575,2). Se utilizaron los valores estimados de los coeficientes, sus errores estándar, y el valor z para evaluar la significancia estadística de cada predictor.

Entre las variables más influyentes en la probabilidad de aceptación de un préstamo se destacan:

- Ingreso (Income): coeficiente positivo significativo ($\beta = 0.059$, $p < 0.001$), lo cual indica que, a mayor nivel de ingresos, mayor es la probabilidad de aceptar un préstamo.
- Tamaño de la familia (Family): también presenta un efecto positivo significativo ($\beta = 0.675$, $p < 0.001$).
- Gasto promedio con tarjeta de crédito (CCAvg): influencia positiva ($\beta = 0.315$, $p < 0.001$), sugiriendo que los clientes con mayor gasto mensual en tarjetas están más predispuestos a solicitar préstamos personales.
- Nivel educativo (Education): resultado altamente significativo ($\beta = 1.928$, $p < 0.001$), lo que refleja una asociación positiva entre nivel educativo más alto y la propensión a aceptar préstamos.
- Cuenta de certificado de depósito (CD_Account): tuvo el mayor coeficiente estimado ($\beta = 5.03$, $p < 0.001$), indicando que los clientes con este tipo de cuenta tienen una alta probabilidad de aceptar un préstamo.

Por otro lado, se observaron relaciones negativas significativas en:

- Cuenta de valores (Securities_Account): coeficiente negativo ($\beta = -1.378$, $p < 0.001$), indicando menor probabilidad de aceptación entre quienes poseen esta cuenta.

- Uso de banca en línea (Online): ($\beta = -0.653$, $p < 0.001$).
- Uso de tarjeta de crédito del banco (CreditCard): ($\beta = -1.129$, $p < 0.001$).

Algunas variables, como Experience y Mortgage, no resultaron significativas a niveles convencionales ($p > 0.05$), lo cual sugiere que su influencia en la decisión de aceptar un préstamo es limitada en el contexto de este modelo.

Este modelo permitió identificar un conjunto de variables predictoras estadísticamente significativas, interpretables desde el punto de vista del negocio, y que ofrecen información valiosa para el direccionamiento de campañas de marketing enfocadas en la promoción de préstamos personales.

5.5.1.2. Matriz de Confusión:

La siguiente matriz de confusión muestra el desempeño del modelo en términos de predicciones correctas e incorrectas:

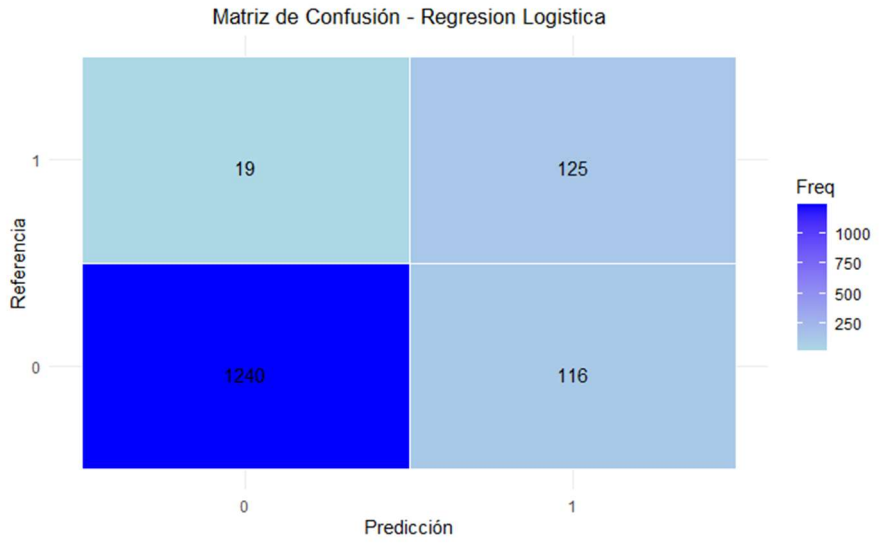


Figura 10: Matriz de confusión - Regresión Logística
Fuente: Elaboración propia.

- Verdaderos Negativos (TN = 1240)
    - Son los clientes que no aceptaron el préstamo y fueron correctamente clasificados por el modelo (no aceptaron).

    - Esto indica que el modelo fue eficaz para detectar perfiles que no estaban interesados en adquirir un préstamo, lo que puede ayudar a reducir esfuerzos de marketing innecesarios en estos casos.
- Falsos Negativos (FN = 19)
    - Son los clientes que sí aceptaron el préstamo, pero el modelo predijo que no lo harían.
    - Este es un aspecto crítico, ya que significa que potenciales clientes interesados fueron ignorados, lo cual representa una pérdida directa de oportunidad comercial para el banco.
- Falsos Positivos (FP = 116)
    - Son los clientes que no aceptaron el préstamo, pero el modelo predijo que sí lo harían.
    - En la práctica, esto implica que el banco podría destinar recursos en acciones de marketing hacia clientes con baja probabilidad real de conversión, generando costos innecesarios.
- Verdaderos Positivos (TP = 125)
    - Son los clientes que sí aceptaron el préstamo y el modelo los clasificó correctamente.
    - Este es el grupo de mayor interés para el banco, ya que representa a los clientes reales con alta probabilidad de conversión. Sin embargo, la cantidad identificada por el modelo es baja, lo que sugiere una limitada capacidad predictiva sobre la clase positiva.

5.5.1.3. Métricas de desempeño

El modelo presentó un rendimiento global destacado, con buena capacidad para identificar correctamente tanto clientes propensos como no propensos a aceptar un préstamo personal. Las métricas obtenidas fueron:

- Exactitud → 91.00%
- Intervalo de confianza (95%) → (0.8944, 0.9240)
- Kappa → 0.6015 (Concordancia moderada entre predicciones y valores reales)
- Sensibilidad (Recall Positivo) → 86.81%

- Especificidad (Recall Negativo) → 91.45%
- Valor predictivo positivo (PPV - Precisión en clase positiva- Precision) → 51.87%
- Valor predictivo negativo (NPV - Precisión en clase negativa) → 98.49%
- F1-score → 64.95%
- Balanced Accuracy → 89.13%

5.5.1.4. Interpretación del desempeño

El modelo de regresión logística presenta un desempeño general sólido, con elevada exactitud y capacidad para discriminar entre las clases. La alta sensibilidad (86.81%) indica que el modelo identifica correctamente a la mayoría de los clientes que aceptan el préstamo, lo cual resulta especialmente valioso para estrategias de captación o marketing dirigido.

La especificidad del 91.45% respalda la fiabilidad del modelo en la identificación de clientes no interesados, lo que permite reducir falsos positivos y asignar recursos de forma más eficiente. Sin embargo, la precisión en la clase positiva (51.87%) evidencia un número significativo de falsos positivos, es decir, clientes identificados como potenciales receptores del préstamo que finalmente no lo aceptan.

Para capturar de forma integrada el compromiso entre sensibilidad y precisión, se calculó el F1-score, que arrojó un valor de 64.95%. Esta métrica armoniza ambas dimensiones, ofreciendo una visión más realista del rendimiento del modelo ante clases desbalanceadas.

El valor predictivo negativo (98.49%) fue sobresaliente, lo que implica una alta confiabilidad al descartar falsos negativos. Por otra parte, el resultado estadísticamente significativo del McNemar's Test ($p < 2e\text{-}16$) indica que existen diferencias sistemáticas entre los errores tipo I y tipo II, lo cual podría reflejar desequilibrios residuales o limitaciones del modelo para capturar matices complejos en la estructura de los datos.

En conjunto, estos resultados sugieren que, si bien el modelo es competitivo, podrían explorarse mejoras orientadas a aumentar la precisión sin sacrificar sensibilidad. Esto incluye ajustes de umbral, ampliación del conjunto de variables o incorporación de modelos más sofisticados que permitan capturar interacciones no lineales.

5.5.1.5. Evaluación mediante AUC-PR

El modelo de regresión logística fue evaluado mediante la curva Precision-Recall, técnica especialmente útil en contextos de clases desbalanceadas. El resultado obtenido fue un AUC-PR de 0.8005, lo que indica una buena capacidad del modelo para equilibrar precisión y sensibilidad en la identificación de la clase positiva (clientes que aceptan un préstamo).

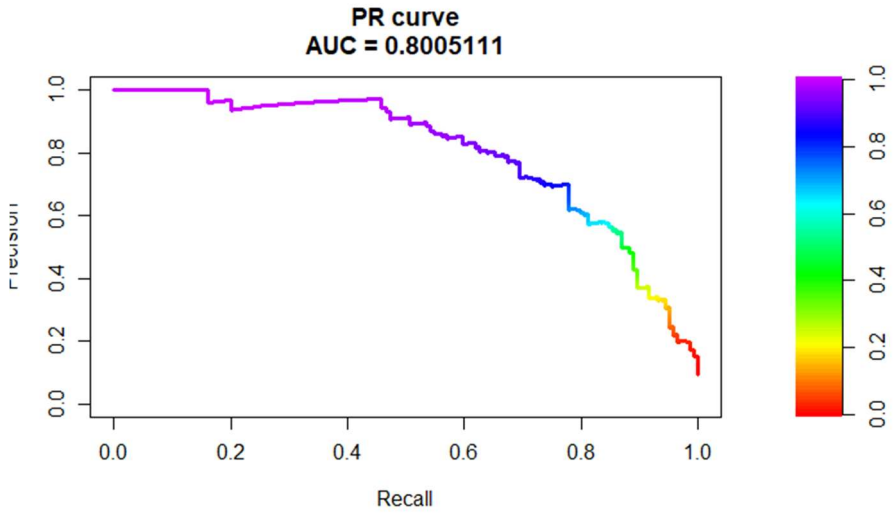


Figura 11: Curva de Precision-Recall (AUC-PR) – Regresión Logística
Fuente: Elaboración propia.

Este valor refuerza la robustez del modelo, ya que:

- Un AUC cercano a 1.0 representa un alto desempeño discriminativo.
- A diferencia del AUC-ROC, el AUC-PR es más sensible a la distribución de clases y permite evaluar de forma más específica la capacidad del modelo en problemas donde la clase minoritaria es la más relevante desde el punto de vista operativo.

En conjunto con las métricas previas (F1-score de 64.95%, sensibilidad del 86.8%), el valor del AUC-PR evidencia que el modelo no solo tiene buen comportamiento general, sino que también mantiene consistencia en su desempeño sobre la clase minoritaria, clave en contextos de negocio enfocados en conversión efectiva.

5.5.2. Modelo 2: Árboles de Clasificación

El modelo de Árbol de Clasificación se utilizó por su capacidad para generar reglas de decisión interpretables y segmentar los clientes según sus características. Este enfoque es particularmente útil en aplicaciones bancarias, donde las decisiones deben ser explicables para la toma de acciones estratégicas.

5.5.2.1. Resultados del Árbol de Clasificación

Se aplicó un Árbol de Clasificación utilizando con criterio de división basado en el índice de impureza de Gini. El modelo fue entrenado sobre el conjunto de datos previamente balanceado, compuesto por 6188 observaciones.

```
Call:
rpart(formula = Personal_Loan ~ ., data = datos_balanceados,
    method = "class", parms = list(split = "gini"))
  n= 6188

          CP nsplit  rel error     xerror        xstd
1 0.74173280      0 1.00000000 1.00000000 0.013003258
2 0.11970899      1 0.25826720 0.26058201 0.008671676
3 0.04662698      2 0.13855820 0.13988095 0.006564670
4 0.01223545      3 0.09193122 0.09358466 0.005434333
5 0.01000000      5 0.06746032 0.07308201 0.004827444

Variable importance
    Income      CCAvg  Education     Family CD_Account CreditCard
        39         24         16         11          6          3
```

Durante el proceso de construcción, el árbol evaluó múltiples divisiones potenciales, seleccionando como primera variable de mayor poder discriminativo a Income

La arquitectura final del árbol incluye 5 divisiones principales, con una disminución progresiva del error relativo desde 1.00 hasta 0.067. La poda se guió según el parámetro de complejidad (CP), preservando el equilibrio entre ajuste y generalización.

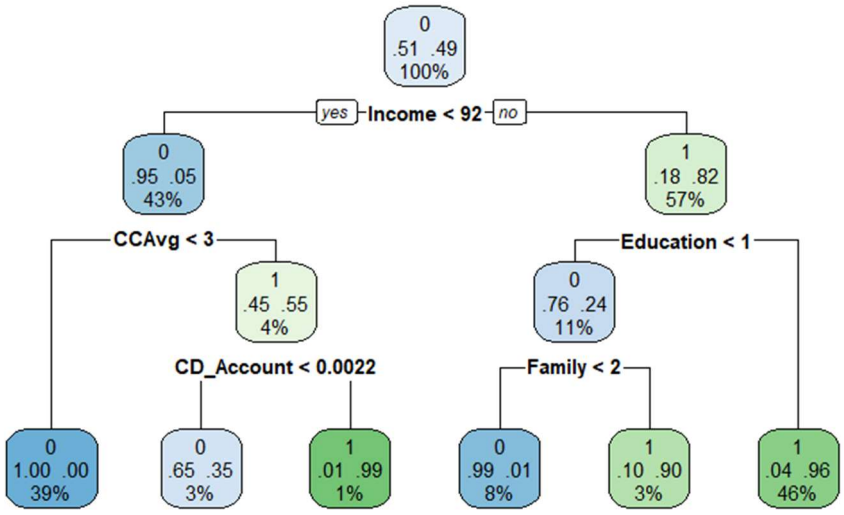


Figura 12: Árbol de decisión
Fuente: Elaboración propia.

El modelo identificó subgrupos con alta pureza en la clasificación:

- En el nodo 7, por ejemplo, se agruparon 2845 observaciones donde el 95.8% fueron clasificadas correctamente como clientes que aceptan el préstamo.
- De forma análoga, nodos como el 4 y el 12 reflejan predicciones acertadas del 100% y 98.8% para la clase negativa, respectivamente.

Este nivel de segmentación permite obtener reglas interpretables de decisión como, por ejemplo:

> Si Income > 92 mil y Education > nivel medio y CD_Account = 1, entonces la probabilidad de aceptar un préstamo personal es superior al 95%.

5.5.2.2. Matriz de Confusión:

La matriz de confusión del modelo refleja una mejora en la clasificación respecto a la regresión logística, mostrando una reducción en los errores de predicción

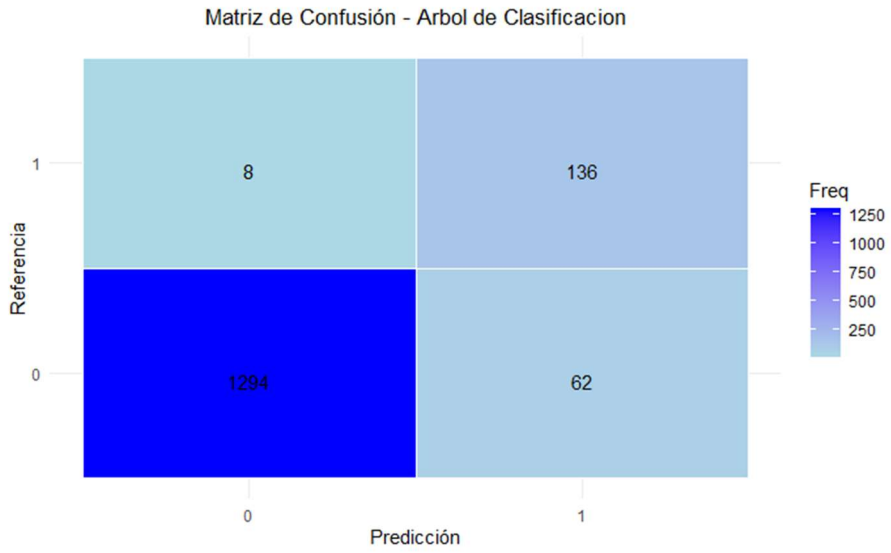


Figura 13: Matriz de confusión - Árbol de Clasificación
Fuente: Elaboración propia.

- Verdaderos Negativos (TN = 1294)
    - Clientes que no aceptaron el préstamo y fueron correctamente clasificados por el modelo.
    - El árbol fue eficiente para identificar perfiles no interesados, lo cual puede ayudar a optimizar recursos al reducir esfuerzos comerciales sobre esta población.
- Falsos Negativos (FN = 8)
    - Clientes que sí aceptaron el préstamo, pero el modelo predijo que no lo harían.
    - Esta es una de las mayores limitaciones del modelo: falló en detectar una gran proporción de clientes interesados, lo que implica una pérdida de oportunidades comerciales importantes para el banco.
- Falsos Positivos (FP = 62)
    - Clientes que no aceptaron el préstamo, pero el modelo predijo que sí lo harían.
    - Esto implica un posible gasto de recursos en usuarios con baja probabilidad real de conversión, aunque el número de falsos positivos fue moderado en comparación con otros modelos.

- Verdaderos Positivos (TP = 136)
    - Clientes que sí aceptaron el préstamo y fueron correctamente clasificados.
    - Este valor resulta muy bajo en relación al total de interesados reales, lo que revela una baja sensibilidad del modelo frente a la clase minoritaria.

5.5.2.3. Métricas de Desempeño

El modelo de Árbol de Decisión mostró un desempeño sólido en términos de exactitud y especificidad, logrando identificar correctamente a la mayoría de los clientes que no aceptaron el préstamo. Si bien la sensibilidad fue elevada, la precisión en la clase positiva (clientes que aceptan el préstamo) resultó más moderada, indicando la presencia de falsos positivos. No obstante, el valor predictivo negativo fue sobresaliente, evidenciando una alta confiabilidad al descartar correctamente a los clientes no interesados. Las métricas obtenidas fueron:

- Exactitud → 95.33%
- Intervalo de confianza (95%) → (0.9414, 0.9634)
- Kappa → 0.7697 (concordancia sustancial entre predicciones y valores reales)
- Sensibilidad (Recall positivo) → 94.44%
- Especificidad (Recall negativo) → 95.43%
- Valor predictivo positivo (PPV – Precisión en clase positiva) → 68.69%
- Valor predictivo negativo (NPV – Precisión en clase negativa) → 99.39%
- Precisión (Precision) → 68.69%
- F1-score → 79.22%
- Balanced Accuracy → 94.94%
- Nota: El F1-score fue calculado como la media armónica entre la precisión (68.69%) y la sensibilidad (94.44%), reflejando el equilibrio del modelo en la identificación de la clase positiva.

5.5.2.4. Interpretación del desempeño

El modelo de Árbol de Clasificación evidenció un comportamiento altamente satisfactorio, destacándose por su elevada capacidad de discriminación entre las clases. La sensibilidad (94.44%) indica que el modelo identifica correctamente a la mayoría de los clientes que

aceptaron un préstamo personal, lo cual lo hace especialmente útil desde una perspectiva operativa y comercial.

La especificidad del 95.43% y el valor predictivo negativo del 99.39% refuerzan la confiabilidad del modelo para descartar correctamente a quienes no aceptarán un préstamo, optimizando así el uso de recursos en campañas segmentadas y reduciendo el riesgo de contactar perfiles con baja probabilidad de conversión.

En cuanto a la clase positiva, el modelo alcanzó una precisión del 68.69%, superior a la obtenida por la regresión logística, lo cual representa una mejora en la reducción de falsos positivos. El F1-score (79.22%), al equilibrar precisión y sensibilidad, refleja un buen compromiso global del modelo en la predicción de la clase minoritaria.

Además, el test de McNemar ($p < 0.001$) reveló diferencias significativas entre los errores tipo I y tipo II, lo que podría atenderse mediante ajustes adicionales en la estructura del árbol (poda, complejidad) o estrategias de ensamblado (boosting o bagging).

En conjunto, el árbol no solo proporciona un rendimiento superior en términos cuantitativos, sino que además entrega reglas de decisión interpretables y operativamente accionables, lo cual constituye una ventaja relevante en contextos donde la transparencia del modelo es un requisito clave.

5.5.2.5. Evaluación mediante AUC-PR

La evaluación del modelo mediante la curva Precision-Recall (PR) arrojó un AUC-PR de 0.7051, lo que indica una capacidad moderada pero consistente del modelo para identificar correctamente a los clientes que aceptan un préstamo (se enfoca en el rendimiento sobre la clase positiva), sin incurrir en un exceso de falsos positivos.

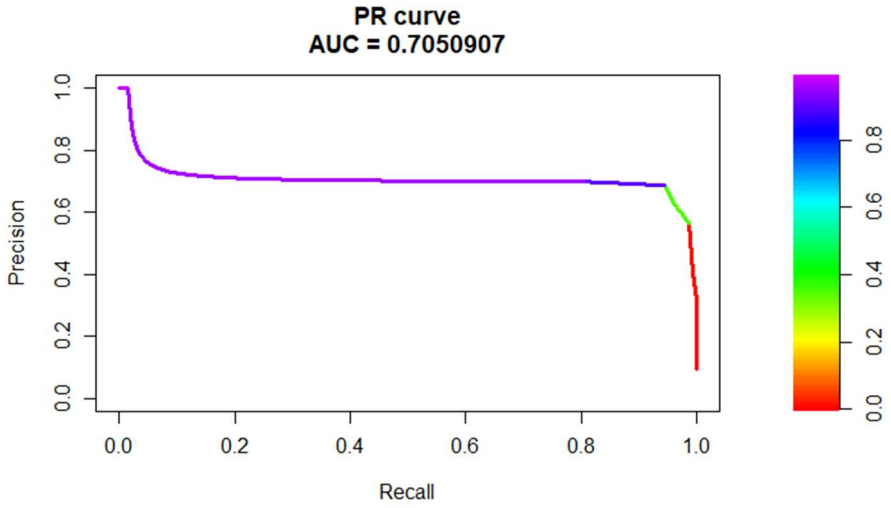


Figura 14: Curva de Precision-Recall (AUC-PR) - Árbol de Clasificación
Fuente: Elaboración propia.

El valor obtenido sugiere que el modelo logra un equilibrio aceptable entre precisión y sensibilidad a lo largo de distintos umbrales de clasificación, aunque todavía con espacio de mejora respecto a modelos más complejos o calibrados.

La forma de la curva (ascendente, con tramos prolongados de precisión estable) indica que el modelo mantiene su desempeño incluso cuando se buscan configuraciones más conservadoras (por ejemplo, priorizando menor cantidad de falsos positivos sin perder demasiados verdaderos).

Este análisis refuerza la conclusión de que el árbol de clasificación no solo ofrece reglas interpretables y buena exactitud, sino que también preserva un rendimiento competitivo en la detección de casos relevantes desde una perspectiva operacional.

5.5.3. Modelo 3: Red Neuronal

La Red Neuronal Artificial fue evaluada como modelo avanzado para la clasificación de clientes interesados en préstamos personales. Su potencial reside en la capacidad de captar relaciones complejas en los datos y modelar patrones no lineales.

Dado que los algoritmos de redes neuronales son sensibles a la escala de las variables de entrada, se procedió a normalizar todos los predictores numéricos utilizando la técnica de normalización Min-Max, que transforma los valores al rango [0, 1]. Este procedimiento garantiza una convergencia más estable durante el entrenamiento, evita que variables con mayor magnitud dominen el proceso de aprendizaje y mejora la eficiencia computacional del modelo.

5.5.3.1. Resultados de la Red Neuronal

El modelo de red neuronal se entrenó correctamente con un total de 4664 iteraciones, alcanzando un error global final de 9.0079, lo que refleja una convergencia aceptable para problemas de clasificación binaria con múltiples variables predictoras.

Desde el punto de vista operativo, la red aprendió relaciones no lineales entre las variables predictoras (como Income, CCAvg, Education, etc.) y la probabilidad de que un cliente acepte un préstamo. Aunque el error final es relativamente bajo, será necesario analizar métricas adicionales

La red neuronal posee:

- 10 neuronas de entrada, correspondientes a las variables predictoras: Experience, Income, Family, CCAvg, Education, Mortgage, Securities_Account, CD_Account, Online y CreditCard.
- Dos capas ocultas (estructura tipo hidden = c(10, 5)), con 10 y 5 neuronas respectivamente.
- Una neurona de salida, que representa la variable Personal_Loan y devuelve una probabilidad de aceptación del préstamo.

La visualización de la red permite observar la densidad de conexiones y la magnitud de los pesos asignados en cada capa. Los valores azules indican pesos positivos y los negros, negativos. Esto permite interpretar la influencia relativa de cada variable sobre la activación final del nodo de salida.

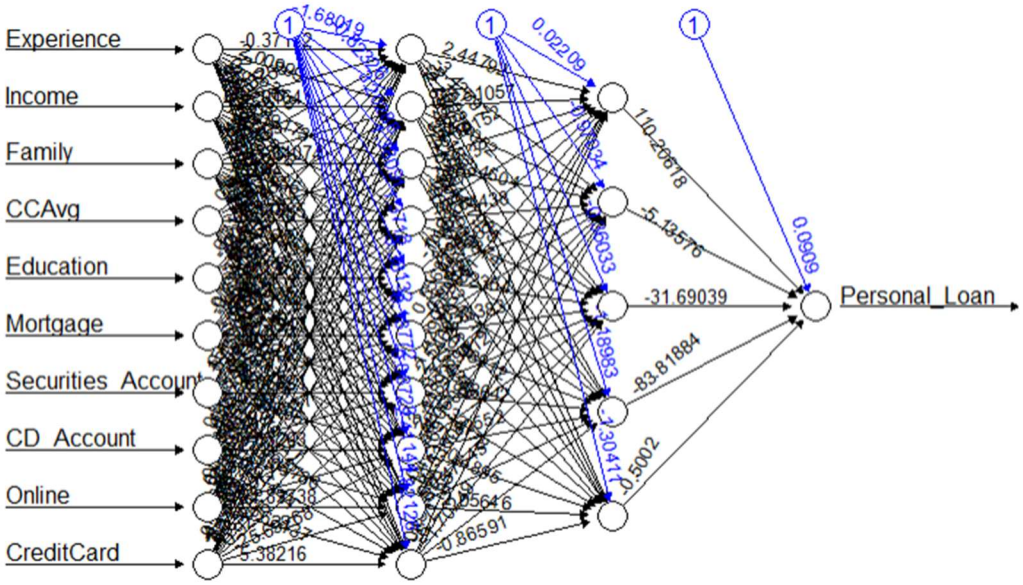


Figura 15: Grafica de Red Neuronal
Fuente: Elaboración propia.

Funciones utilizadas

- Función de activación: logistic (sigmoidea) en las capas ocultas y de salida.
- Función de error: sse (suma de errores al cuadrado), adecuada para tareas de clasificación binaria.
- Configuración de salida: linear.output = FALSE, lo que implica que la salida se interpreta como una probabilidad y no como una variable continua.

Resultado del entrenamiento

- Error total del modelo: 9.007864
- Número de pasos hasta convergencia: 4664

5.5.3.2. Matriz de Confusión:

Para evaluar el desempeño del modelo de red neuronal en la clasificación de los clientes según su probabilidad de aceptar un préstamo personal, al igual que los modelos anteriores, se utilizó la matriz de confusión:

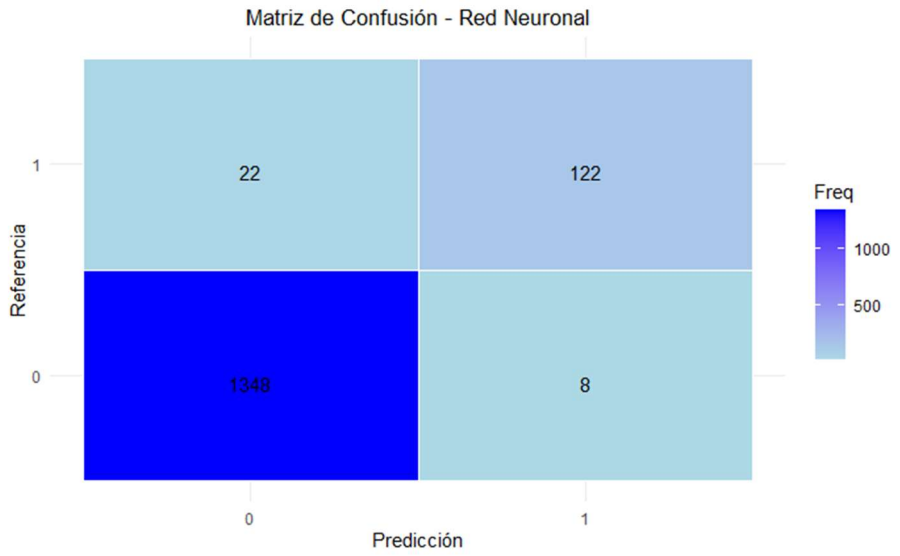


Figura 16: Matriz de confusión - Red Neuronal
Fuente: Elaboración propia.

- Verdaderos Negativos (TN = 1348)
  - Son los clientes que no aceptaron el préstamo y fueron correctamente clasificados.
  - El modelo fue altamente eficaz en reconocer perfiles no interesados, lo cual puede resultar útil para reducir costos en campañas innecesarias.
- Falsos Negativos (FN = 22)
  - Son los clientes que sí aceptaron el préstamo, pero fueron clasificados como si no lo hicieran.
  - Esta es una limitación relevante del modelo, ya que implica que muchos clientes con interés real no fueron detectados, generando pérdidas potenciales de negocio.
- Falsos Positivos (FP = 8)
  - Son los clientes que no aceptaron el préstamo, pero fueron clasificados como si lo hicieran.
  - La baja cantidad de falsos positivos es un aspecto positivo, ya que el modelo minimiza el riesgo de malgastar recursos en clientes no interesados.
- Verdaderos Positivos (TP = 122)

- Son los clientes que sí aceptaron el préstamo y fueron correctamente identificados.
- Aunque superior a algunos modelos previos, el valor sigue siendo bajo en términos absolutos, lo que sugiere una sensibilidad limitada hacia la clase positiva.

5.5.3.3. Métricas de Desempeño

El modelo de red neuronal muestra un rendimiento sobresaliente sobre el conjunto de prueba, superando los resultados obtenidos por la regresión logística y el árbol de clasificación en múltiples indicadores clave. Las métricas reportadas fueron:

- Exactitud → 98.00% • Intervalo de confianza (95%) → (0.9716, 0.9865)
- Kappa → 0.8795 (muy buena concordancia entre predicciones y valores reales)
- Sensibilidad (Recall positivo) → 84.72%
- Especificidad (Recall negativo) → 99.41%
- Valor predictivo positivo (PPV – Precisión en clase positiva) → 93.85%
- Valor predictivo negativo (NPV – Precisión en clase negativa) → 98.39%
- Precisión (Precision) → 93.85%
- F1-score → 88.89%
- Balanced Accuracy → 92.07%

El F1-score, calculado como media armónica entre sensibilidad y precisión, alcanzó un valor de 88.89%, lo que confirma que el modelo logró un equilibrio notable entre identificar correctamente la clase positiva y minimizar falsos positivos.

El valor-p del test de McNemar (p = 0.0176) indica que aún existen diferencias significativas entre errores tipo I y tipo II, aunque en menor medida que en modelos anteriores, lo cual evidencia un sesgo residual leve que podría refinarse con ajustes en la arquitectura o en el umbral de decisión.

5.5.3.4. Interpretación de Resultados

El modelo de red neuronal evidenció un rendimiento sobresaliente tanto en la clasificación general como en la detección de clientes propensos a aceptar un préstamo personal. La exactitud

del 98% y el valor de Kappa (0.8795) reflejan una muy buena concordancia entre las predicciones del modelo y los valores reales, superando ampliamente los resultados de modelos anteriores.

Desde la perspectiva del negocio, la precisión en la clase positiva (93.85%) representa una mejora sustancial respecto a los falsos positivos detectados previamente con modelos más simples. Esto implica que una gran parte de los clientes identificados como potenciales tomadores de préstamo efectivamente corresponden a esa categoría, optimizando así la asignación de recursos en campañas de marketing o retención.

La sensibilidad del 84.72% confirma que el modelo mantiene una alta capacidad de capturar casos relevantes, mientras que el F1-score (88.89%) señala un excelente equilibrio entre precisión y recall, incluso en un entorno desbalanceado.

Por otra parte, la especificidad del 99.41% y el valor predictivo negativo (98.39%) consolidan la confiabilidad del modelo en la identificación de clientes no interesados, lo cual reduce significativamente las probabilidades de incurrir en falsos negativos.

Finalmente, aunque el McNemar's Test ($p = 0.0176$) indicó una leve diferencia entre errores tipo I y tipo II, esta resulta menor en comparación con los modelos anteriores y no compromete el rendimiento general de la red.

En conjunto, estos resultados confirman que la red neuronal no solo presenta el mejor desempeño cuantitativo, sino que también ofrece una excelente capacidad predictiva para estrategias comerciales dirigidas, manteniendo un nivel bajo de error y alta precisión operativa.

5.5.3.5. Evaluación mediante AUC-PR

La curva Precision-Recall obtenida para el modelo de red neuronal exhibe un AUC-PR de 0.9346, lo que representa un desempeño excepcional en la identificación de la clase positiva (clientes que aceptan el préstamo).

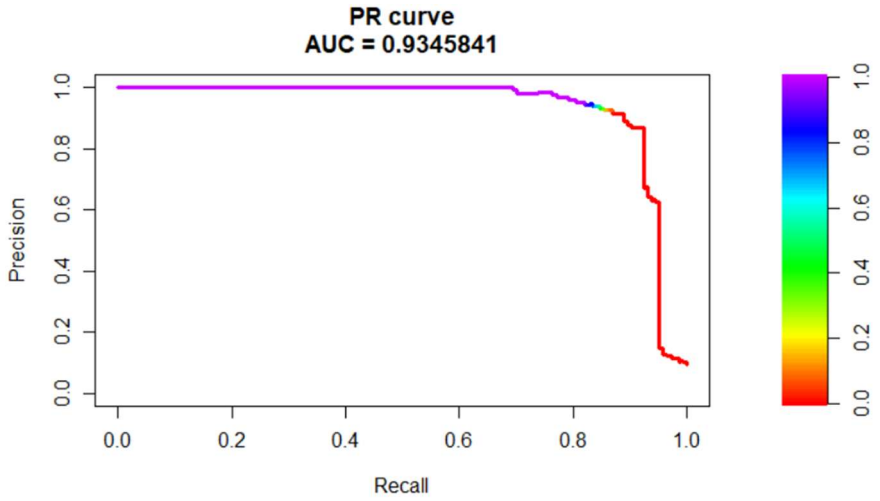


Figura 17: Curva de Precision-Recall (AUC-PR) - Red Neuronal
Fuente: Elaboración propia.

Este valor tan elevado implica que el modelo mantiene altos niveles de precisión (baja proporción de falsos positivos) incluso cuando se incrementa el recall (capacidad de capturar verdaderos positivos), lo que es especialmente valioso en escenarios donde la clase relevante tiene baja prevalencia, como en este caso.

Visualmente, la forma de la curva (ascendente con una meseta sostenida en niveles altos) sugiere que el modelo logra mantener un balance muy favorable entre precisión y sensibilidad en un amplio rango de umbrales de decisión. Esto le otorga flexibilidad operativa: se pueden ajustar estrategias según se priorice capturar más clientes o minimizar contactos innecesarios.

En conjunto con el F1-score (88.89%) y la precisión positiva (93.85%), esta curva reafirma que la red neuronal no solo alcanza el mejor rendimiento cuantitativo, sino también una gran robustez en su capacidad discriminativa frente a desequilibrios de clase.

5.5.4. Evaluación adicional mediante la métrica Kolmogorov-Smirnov (KS)

Con el objetivo de complementar la evaluación de los modelos desarrollados, se incorporó la métrica Kolmogorov-Smirnov (KS), ampliamente utilizada en contextos de scoring crediticio y análisis de riesgo financiero.

Esta métrica mide la máxima distancia entre las distribuciones acumuladas de las probabilidades predichas para las clases positiva y negativa, lo que permite evaluar la capacidad

del modelo para distinguir entre clientes que activan el préstamo personal y aquellos que no lo hacen. A diferencia de métricas como el F1-score, el estadístico KS no depende de un umbral de clasificación específico, ya que se basa en la separación entre ambas distribuciones a lo largo de todas las probabilidades predichas.

Los valores obtenidos para los modelos analizados fueron los siguientes:

Modelo	KS
Regresión Logística	0.7914
Árbol de Decisión	0.9114
Red Neuronal	0.9089

Los resultados indican que los tres modelos presentan una alta capacidad de discriminación, siendo el árbol de decisión el modelo con mayor separación entre las distribuciones de ambas clases, seguido muy de cerca por la red neuronal. La regresión logística también presenta un desempeño sólido, con un valor de KS superior a 0.79.

En términos generales, valores elevados de KS indican que el modelo logra diferenciar de manera efectiva entre clientes potenciales que aceptarían el préstamo y aquellos que no, lo cual resulta particularmente relevante en aplicaciones de segmentación y modelos de propensión en el sector financiero.

5.6. Comparación de Modelos

Con el propósito de evaluar cuál de los modelos implementados ofrece un mejor desempeño para predecir la probabilidad de aceptación de un préstamo personal, se desarrollaron y compararon tres algoritmos de aprendizaje supervisado: Regresión Logística, Árbol de Clasificación y Red Neuronal. Cada uno de ellos fue entrenado y evaluado sobre un conjunto de datos balanceado, para corregir el fuerte desbalance de clases presente en la variable objetivo.

La comparación se lleva a cabo utilizando un conjunto de métricas comunes en clasificación binaria, incluyendo exactitud (accuracy), sensibilidad (recall), especificidad, precisión, F1-score y AUC-PR (área bajo la curva Precisión-Recall).

Métrica	Regresión Logística	Árbol de Clasificación	Red Neuronal
Exactitud (Accuracy)	91,00 %	95,33 %	98,00 %
Índice de Kappa	0,6015	0,7697	0,8795
Sensibilidad (Recall +)	86,81 %	94,44 %	84,72 %
Especificidad (Recall -)	91,45 %	95,43 %	99,41 %
Precisión (Precision +)	51,87 %	68,69 %	93,85 %
F1-Score	64,95 %	79,22 %	88,89 %
AUC - Curva Precisión-Recall	0,8005	0,7051	0,9346
Exactitud Balanceada	89,13 %	94,94 %	92,07 %

La regresión logística ofreció una base sólida en términos interpretativos y un buen nivel de sensibilidad (86,81 %), lo que indica que el modelo fue capaz de identificar correctamente a una proporción significativa de clientes que efectivamente aceptaron el préstamo. No obstante, su precisión fue baja (51,87 %), lo que sugiere una tasa considerable de falsos positivos, es decir, casos donde el modelo predijo que un cliente aceptaría un préstamo cuando en realidad no lo hizo. Esta limitación impacta directamente en la eficiencia de campañas dirigidas, ya que implica un uso no óptimo de recursos sobre clientes no interesados.

El árbol de clasificación mejoró significativamente el desempeño con respecto al modelo anterior. Mostró una sensibilidad elevada (94,44 %) y una precisión superior (68,69 %), logrando un mejor balance entre detección de casos positivos y reducción de errores. Además, su estructura jerárquica permite una interpretación directa de las reglas de decisión, lo cual representa una ventaja en entornos donde la trazabilidad y la explicabilidad son relevantes. Sin embargo, su AUC-PR fue inferior al de los otros modelos (0,7051), lo que indica que su rendimiento se ve más afectado ante variaciones en el umbral de decisión.

La red neuronal, por su parte, se posicionó como el modelo con mejor desempeño general. Alcanzó los valores más altos en casi todas las métricas evaluadas, especialmente en F1-score (88,89 %), precisión (93,85 %) y AUC-PR (0,9346). Estos resultados reflejan un equilibrio altamente efectivo entre la identificación correcta de clientes interesados (verdaderos positivos) y la minimización de errores tipo I y tipo II. Su excelente capacidad discriminativa demuestra

su potencial como herramienta predictiva robusta, aunque con la desventaja de una menor interpretabilidad frente a modelos como la regresión logística o el árbol de decisión.

5.6.1. Conclusión de la comparación

El análisis comparativo entre modelos demuestra que la red neuronal es la alternativa más eficiente y precisa para predecir la aceptación de préstamos personales en el contexto planteado por el banco. Su superioridad en métricas clave como F1-score y AUC-PR la convierte en la opción más adecuada para maximizar el retorno de las campañas de marketing, minimizando errores y focalizando esfuerzos sobre los clientes con mayor probabilidad de conversión.

Sin embargo, la elección del modelo debe considerar también otros factores como la interpretabilidad, el tiempo de entrenamiento, la facilidad de implementación y la aceptación por parte del equipo comercial o de gestión. En contextos donde se requiere transparencia o justificación de las decisiones automatizadas, modelos como la regresión logística o el árbol pueden seguir cumpliendo un rol complementario.

6. Conclusiones Generales y Trabajos Futuros

6.1. Conclusiones Generales

El presente trabajo logró desarrollar y evaluar modelos predictivos capaces de identificar clientes con alta propensión a aceptar préstamos personales, utilizando técnicas de aprendizaje supervisado sobre un conjunto de datos bancarios públicos. A lo largo del estudio se abordaron todas las etapas clave del proceso de modelización: análisis exploratorio, preprocesamiento, balanceo de clases, construcción de modelos y evaluación comparativa.

Se implementaron tres enfoques representativos: regresión logística, árbol de clasificación y red neuronal artificial. Cada uno aportó fortalezas específicas en términos de interpretabilidad, capacidad discriminativa y precisión operativa. La regresión logística ofreció una base sólida y fácilmente explicable, el árbol de decisión permitió generar reglas accionables y segmentaciones claras, mientras que la red neuronal demostró el mejor desempeño cuantitativo, destacándose en métricas como F1-score (88.89%) y AUC-PR (0.9346).

El análisis de correlación previo permitió identificar variables clave como Income, CCAvg, CD_Account, Education y Family, todas asociadas positivamente con la aceptación de préstamos. Estas variables reflejan perfiles de bancarización, capacidad económica y comportamiento transaccional, fundamentales para la toma de decisiones comerciales.

Asimismo, el proceso de preparación de los datos incluyó técnicas como normalización de variables, tratamiento de valores atípicos mediante winsorización y balanceo de clases, lo que permitió mejorar la estabilidad de los modelos y su capacidad para identificar correctamente la clase minoritaria.

No obstante, es importante señalar algunas limitaciones del estudio. En primer lugar, el análisis se realizó sobre un conjunto de datos de acceso público, lo cual implica ciertas restricciones respecto a la disponibilidad de variables que podrían enriquecer el modelo en un contexto real de aplicación. Asimismo, el enfoque del trabajo se centró en la comparación del desempeño de distintos modelos predictivos representativos, sin profundizar en procesos exhaustivos de optimización de hiperparámetros o en estrategias más avanzadas de validación temporal. Estas decisiones responden al alcance del estudio, aunque abren oportunidades para futuras investigaciones que permitan ampliar y profundizar el análisis.

En conjunto, los resultados obtenidos evidencian la viabilidad de aplicar técnicas de ciencia de datos para apoyar procesos de identificación de clientes con mayor probabilidad de aceptar productos financieros. Este tipo de enfoques puede contribuir a mejorar la focalización de campañas comerciales y la asignación eficiente de recursos dentro de las instituciones financieras.

6.2. Trabajos Futuros

Si bien los resultados obtenidos en este trabajo evidencian un desempeño satisfactorio en la identificación de clientes con alta probabilidad de aceptar préstamos personales, existen diversas líneas de investigación que podrían ampliar y profundizar los hallazgos alcanzados.

En primer lugar, sería interesante evaluar modelos más avanzados de aprendizaje automático, como técnicas de ensamblado (por ejemplo, **Random Forest o Gradient Boosting**), que

podrían mejorar la capacidad predictiva del sistema al capturar relaciones más complejas entre las variables.

Otra posible línea de trabajo consiste en enriquecer el modelo mediante la incorporación de variables de comportamiento más dinámicas, como patrones transaccionales, evolución de ingresos o historial de interacción con productos financieros. La integración de este tipo de información podría mejorar aún más la capacidad del modelo para identificar oportunidades comerciales y fortalecer su aplicación práctica dentro de estrategias de marketing y gestión de clientes en instituciones financieras.

Finalmente, en aplicaciones reales del sector financiero también resulta recomendable considerar **esquemas de validación temporal**, como conjuntos de datos *out-of-time (OOT)*, que permitan evaluar el desempeño del modelo sobre datos correspondientes a períodos posteriores. Este tipo de validación facilita la detección de posibles cambios en la distribución de los datos (*data drift*) y contribuye a garantizar la estabilidad del modelo a lo largo del tiempo.

7. Anexo: Código R

Data Set

```
library(readxl)
library(data.table)
library(dplyr)
library(corrplot)
library(ggplot2)
library(tidyr)
library(readxl)
library(rsample)
library(pROC)
library(caret)
library(gridExtra)
library(fmsb)
library(reshape2)
library(skimr)
library(GGally)

datos <- read_excel(path = "C: DataSet/Bank_Personal_Loan_Modelling/Bank_Personal_Loan_Mode
lling.xlsx", sheet = 'Data')
```

```
# Renombrar las columnas con espacios
datos <- datos %>%
  rename(ID = ID,
         Age = Age,
         Experience = Experience,
         Income = Income,
         ZIP_Code = `ZIP Code`,  # Renombrar columna con espacio
         Family = Family,
         CCAvg = CCAvg,
         Education = Education,
         Mortgage = Mortgage,
         Personal_Loan = `Personal Loan`,  # Renombrar columna con espacio
         Securities_Account = `Securities Account`,  # Renombrar columna con espacio
         CD_Account = `CD Account`,  # Renombrar columna con espacio
         Online = Online,
         CreditCard = CreditCard)
```

```
# Ver las primeras filas
head(datos)

## # A tibble: 6 × 14
##      ID   Age Experience Income ZIP_Code Family CCAvg Education Mortgage
##   <dbl> <dbl>      <dbl>  <dbl>    <dbl>  <dbl> <dbl>     <dbl>    <dbl>
## 1     1    25          1     49    91107      4   1.6         1        0
## 2     2    45         19     34    90089      3   1.5         1        0
## 3     3    39         15     11    94720      1   1           1        0
## 4     4    35          9    100    94112      1   2.7         2        0
## 5     5    35          8     45    91330      4   1           2        0
## 6     6    37         13     29    92121      4   0.4         2      155
## # i 5 more variables: Personal_Loan <dbl>, Securities_Account <dbl>,
## #   CD_Account <dbl>, Online <dbl>, CreditCard <dbl>
```

```
skim(datos)
```

Data summary

Name	datos
Number of rows	5000
Number of columns	14
_____	
Column type frequency:	
numeric	14
_____	
Group variables	None

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
ID	0	1	2500.50	1443.52	1	1250.75	2500.5	3750.25	5000	▇▇▇▇
Age	0	1	45.34	11.46	23	35.00	45.0	55.00	67	▆▇▇▆
Experience	0	1	20.10	11.47	-3	10.00	20.0	30.00	43	▆▇▇▆
Income	0	1	73.77	46.03	8	39.00	64.0	98.00	224	▇▅▂▁
ZIP_Code	0	1	93152.50	2121.85	9307	91911.00	93437.0	94608.00	96651	▁▁▁▁▇
Family	0	1	2.40	1.15	1	1.00	2.0	3.00	4	▇▇▁▇
CCAvg	0	1	1.94	1.75	0	0.70	1.5	2.50	10	▇▃▁▁
Education	0	1	1.88	0.84	1	1.00	2.0	3.00	3	▇▁▇▇
Mortgage	0	1	56.50	101.71	0	0.00	0.0	101.00	635	▇▁▁▁
Personal_Loan	0	1	0.10	0.29	0	0.00	0.0	0.00	1	▇▁▁▁
Securities_Account	0	1	0.10	0.31	0	0.00	0.0	0.00	1	▇▁▁▁

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
CD_Account	0	1	0.06	0.24	0	0.00	0.0	0.00	1	
Online	0	1	0.60	0.49	0	0.00	1.0	1.00	1	
CreditCard	0	1	0.29	0.46	0	0.00	0.0	1.00	1	

Exploración básica del dataset

```
# Estructura general del dataset
str(datos)

## tibble [5,000 × 14] (S3: tbl_df/tbl/data.frame)
##  $ ID                : num [1:5000] 1 2 3 4 5 6 7 8 9 10 ...
##  $ Age               : num [1:5000] 25 45 39 35 35 37 53 50 35 34 ...
##  $ Experience        : num [1:5000] 1 19 15 9 8 13 27 24 10 9 ...
##  $ Income            : num [1:5000] 49 34 11 100 45 29 72 22 81 180 ...
##  $ ZIP_Code          : num [1:5000] 91107 90089 94720 94112 91330 ...
##  $ Family            : num [1:5000] 4 3 1 1 4 4 2 1 3 1 ...
##  $ CCAvg             : num [1:5000] 1.6 1.5 1 2.7 1 0.4 1.5 0.3 0.6 8.9 ...
##  $ Education         : num [1:5000] 1 1 1 2 2 2 2 3 2 3 ...
##  $ Mortgage          : num [1:5000] 0 0 0 0 0 155 0 0 104 0 ...
##  $ Personal_Loan     : num [1:5000] 0 0 0 0 0 0 0 0 0 1 ...
##  $ Securities_Account: num [1:5000] 1 1 0 0 0 0 0 0 0 0 ...
##  $ CD_Account        : num [1:5000] 0 0 0 0 0 0 0 0 0 0 ...
##  $ Online            : num [1:5000] 0 0 0 0 0 1 1 0 1 0 ...
##  $ CreditCard        : num [1:5000] 0 0 0 0 1 0 0 1 0 0 ...

datos_sub <- datos %>%
  select(Age,Experience,Income,Family,CCAvg,Mortgage,ZIP_Code)

skim(datos_sub)
```

Data summary

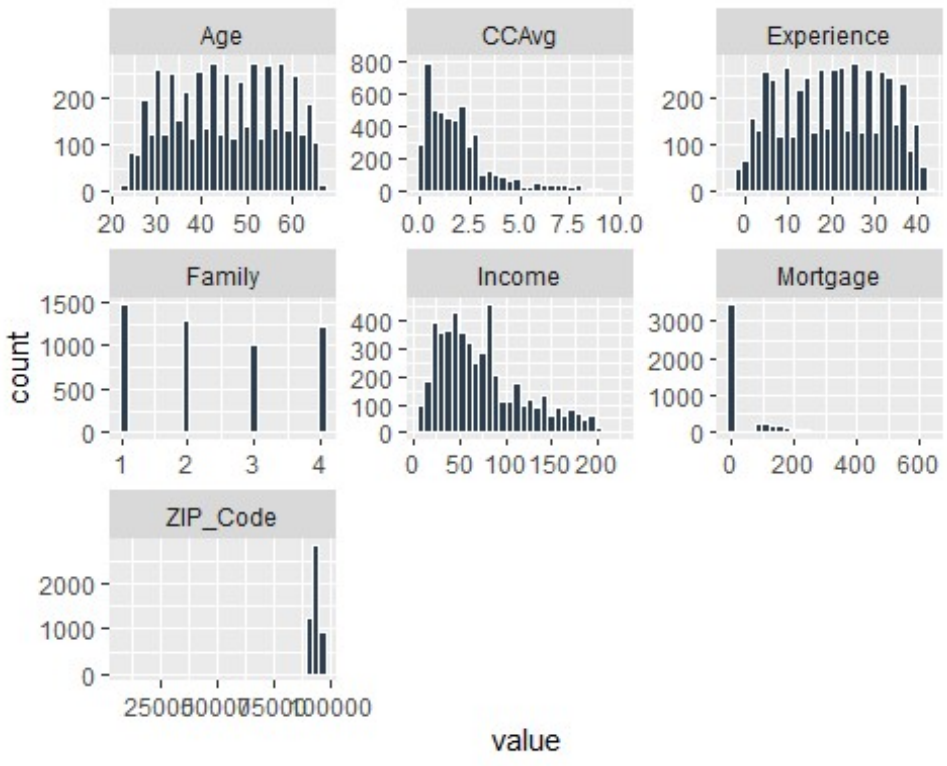
Name	datos_sub
Number of rows	5000
Number of columns	7
_____	
Column type frequency:	
numeric	7
_____	
Group variables	None

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
Age	0	1	45.34	11.46	23	35.0	45.0	55.0	67	
Experience	0	1	20.10	11.47	-3	10.0	20.0	30.0	43	
Income	0	1	73.77	46.03	8	39.0	64.0	98.0	224	
Family	0	1	2.40	1.15	1	1.0	2.0	3.0	4	
CCAvg	0	1	1.94	1.75	0	0.7	1.5	2.5	10	
Mortgage	0	1	56.50	101.71	0	0.0	0.0	101.0	635	
ZIP_Code	0	1	93152.50	2121.85	9307	91911.0	93437.0	94608.0	96651	

Distribuciones y valores atípicos

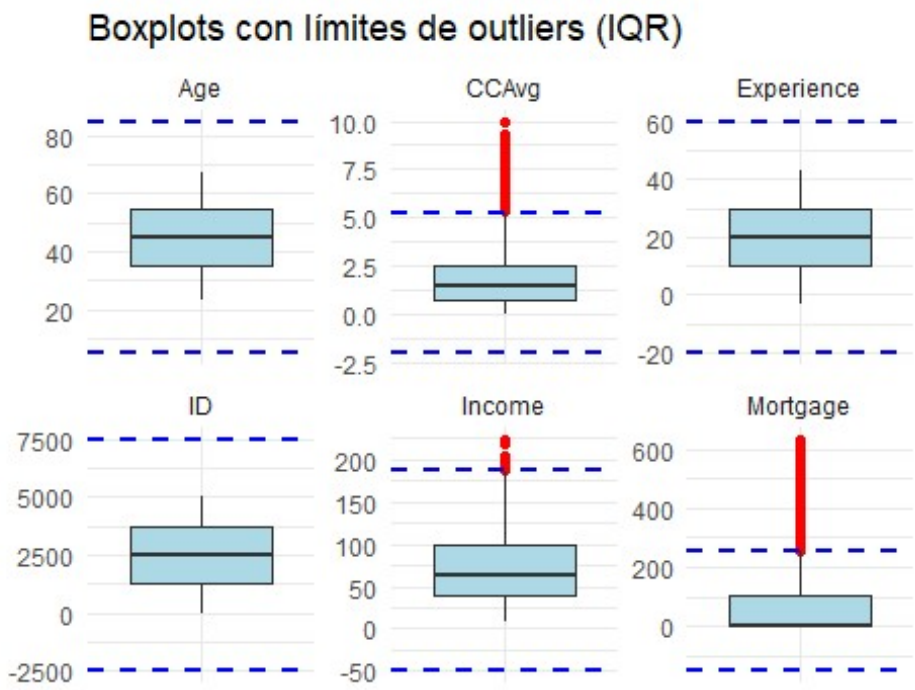
```
datos %>%
  select(Age,Experience,Income,Family,CCAvg,Mortgage,ZIP_Code) %>%
  gather(key, value) %>%
  ggplot(aes(value)) +
    facet_wrap(~key, scales="free") +
    geom_histogram(bins=30, fill="#2c3e50", color="white")
```



```r
# Seleccionar las columnas categóricas que quieres analizar

datos %>%
  select(Education, Securities_Account, CD_Account, CreditCard, Online,Personal_Loan,Online
) %>%
  gather(key, value) %>%
  ggplot(aes(factor(value))) +
    facet_wrap(~key, scales="free_x") +
    geom_bar(fill="lightblue") +
    labs(x="", y="Conteo")
```


```r
# Seleccionar únicamente las variables deseadas
datos_sub <- datos %>%
  select(ID, Age, Experience, Income, CCAvg, Mortgage)

# Calcular límites por variable para outliers (IQR)
limites <- datos_sub %>%
  gather(variable, valor) %>%
  group_by(variable) %>%
  summarise(
    q1 = quantile(valor, 0.25, na.rm = TRUE),
    q3 = quantile(valor, 0.75, na.rm = TRUE)
  ) %>%
  mutate(
    iqr = q3 - q1,
    lim_inf = q1 - 1.5 * iqr,
    lim_sup = q3 + 1.5 * iqr
  )

# Preparar datos largos
datos_long <- datos_sub %>%
  gather(variable, valor)
```

```r
# Unir límites a los datos
datos_plot <- left_join(datos_long, limites, by = "variable")

# Crear gráfico
ggplot(datos_plot, aes(x = "", y = valor)) +
  geom_boxplot(fill = "lightblue", outlier.colour = "red") +
  geom_hline(aes(yintercept = lim_sup), linetype = "dashed", color = "blue", size = 0.8) +
  geom_hline(aes(yintercept = lim_inf), linetype = "dashed", color = "blue", size = 0.8) +
  facet_wrap(~variable, scales = "free_y") +
  labs(title = "Boxplots con límites de outliers (IQR)",
       x = "", y = "") +
  theme_minimal()
```

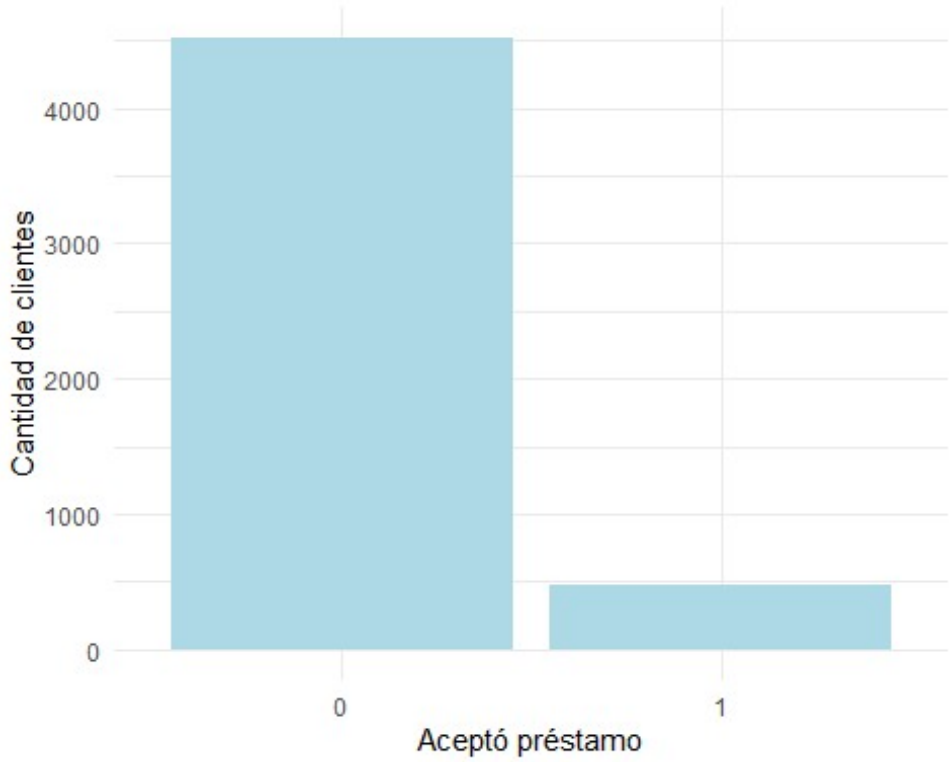



```r
# Seleccionar variables numéricas específicas
variables <- c("Income", "CCAvg", "Mortgage")
datos_sub <- datos %>% select(all_of(variables))

# Convertir a formato largo
datos_long <- datos_sub %>%
  pivot_longer(cols = everything(), names_to = "variable", values_to = "valor")

# Calcular límites para cada variable
limites <- datos_long %>%
  group_by(variable) %>%
  summarise(
    q1 = quantile(valor, 0.25, na.rm = TRUE),
    q3 = quantile(valor, 0.75, na.rm = TRUE),
    iqr = q3 - q1,
    lim_inf = q1 - 1.5 * iqr,
    lim_sup = q3 + 1.5 * iqr,
```

```
    .groups = "drop"
  )

# Unir límites a los datos largos
datos_plot <- left_join(datos_long, limites, by = "variable")

# Crear boxplots facetados
ggplot(datos_plot, aes(x = "", y = valor)) +
  geom_boxplot(fill = "lightblue", outlier.colour = "red") +
  geom_hline(aes(yintercept = lim_sup), linetype = "dashed", color = "blue") +
  geom_hline(aes(yintercept = lim_inf), linetype = "dashed", color = "blue") +
  facet_wrap(~ variable, scales = "free_y") +
  labs(title = "Boxplots con límites de outliers (IQR)", x = "", y = "") +
  theme_minimal()
```

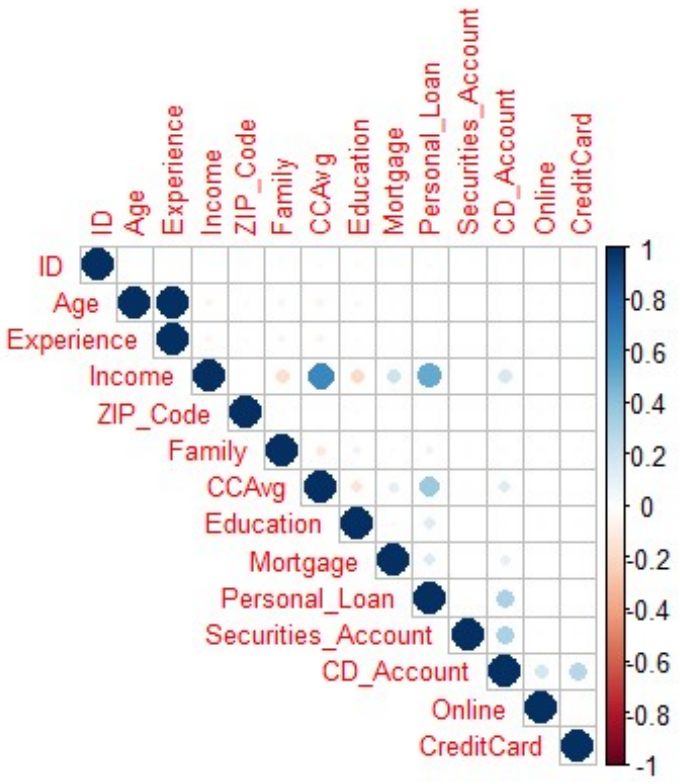
Distribución de la variable objetivo

```
# Proporción de personas que aceptaron o no el préstamo
datos %>%
  count(Personal_Loan) %>%
  mutate(pct = n / sum(n))

## # A tibble: 2 × 3
##   Personal_Loan     n   pct
##           <dbl> <int> <dbl>
## 1             0  4520 0.904
## 2             1   480 0.096

# Gráfico de barras
ggplot(datos, aes(x = factor(Personal_Loan))) +
  geom_bar(fill = "lightblue") +
  labs(x = "Aceptó préstamo", y = "Cantidad de clientes") +
  theme_minimal()
```



Relaciones entre variables

```
# Matriz de correlación
corr <- cor(datos)
corrplot(corr, method = "circle", type = "upper", tl.cex = 0.8)
```



```r
# Asegurarse de que Personal_Loan sea numérica
datos$Personal_Loan <- as.numeric(datos$Personal_Loan)

# Seleccionar solo variables numéricas
datos_numericos <- datos[sapply(datos, is.numeric)]

# Calcular correlación con la variable objetivo
correlaciones_objetivo <- cor(datos_numericos)["Personal_Loan", ]

# Ordenar por fuerza de correlación (absoluta)
correlaciones_ordenadas <- sort(abs(correlaciones_objetivo), decreasing = TRUE)

# Excluir la correlación consigo misma
correlaciones_ordenadas <- correlaciones_ordenadas[names(correlaciones_ordenadas) != "Perso
nal_Loan"]

# Mostrar las variables más correlacionadas
print(correlaciones_ordenadas)
```

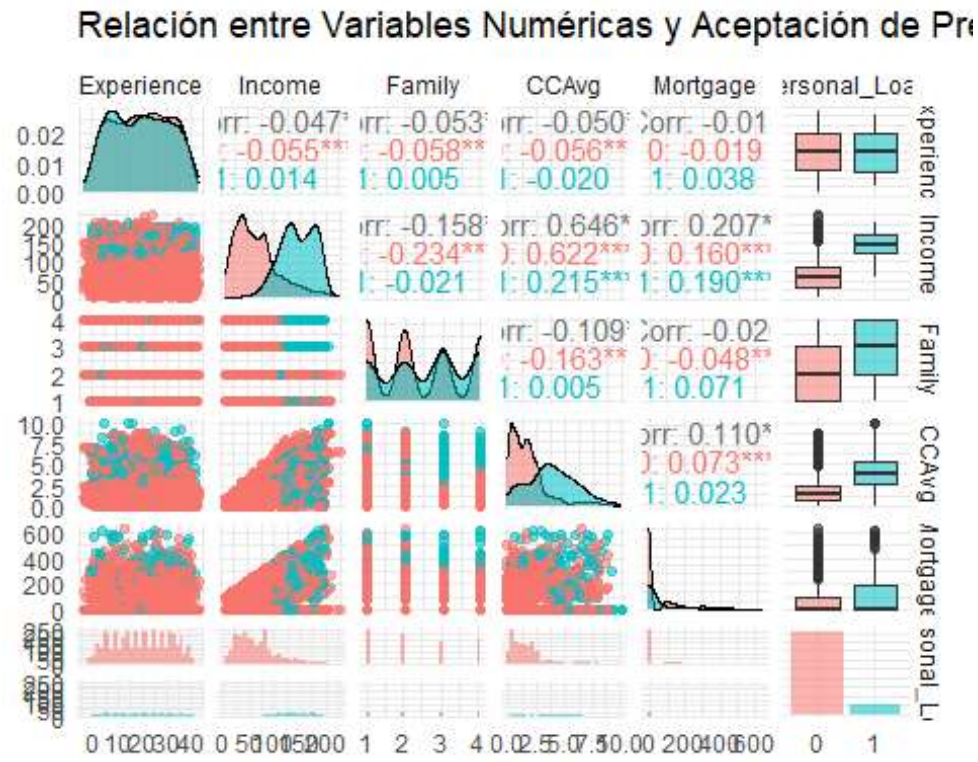
```
##             Income              CCAvg         CD_Account           Mortgage
##       0.5024622925       0.3668905334       0.3163548294       0.1420952363
##          Education             Family                 ID Securities_Account
##       0.1367215500       0.0613670440       0.0248011655       0.0219538822
##                Age         Experience             Online         CreditCard
##       0.0077256172       0.0074130981       0.0062778154       0.0028015088
##           ZIP_Code
##       0.0001073764
```

```r
# Tomar las 5 más correlacionadas
top_vars <- names(head(correlaciones_ordenadas, 5))

# Calcular rango
rangos_top <- data.frame(
  Variable = top_vars,
  Min = sapply(datos[top_vars], min),
  Max = sapply(datos[top_vars], max)
)
print(rangos_top)
```

```
##              Variable Min Max
## Income         Income   8 224
## CCAvg           CCAvg   0  10
## CD_Account CD_Account   0   1
## Mortgage     Mortgage   0 635
## Education   Education   1   3
```

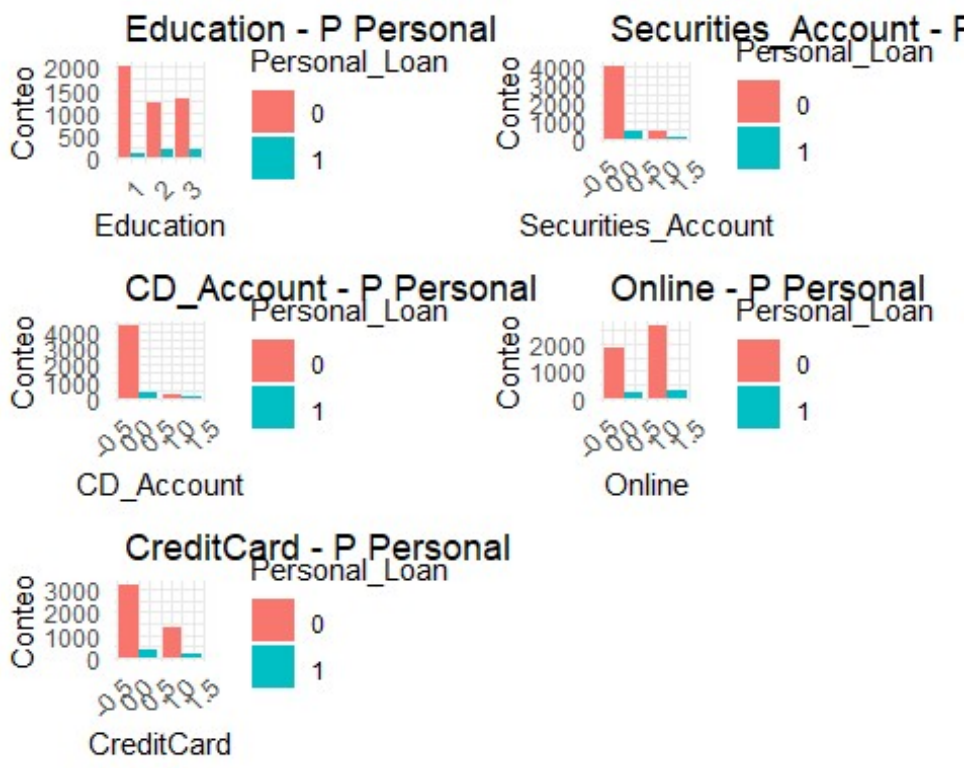
Relación entre Variables Numéricas y Aceptación de Préstamo Personal

```r
# Seleccionar las columnas numéricas que quieres analizar
num_cols <- c( 'Experience', 'Income', 'Family', 'CCAvg', 'Mortgage')

# Convertir la columna 'Personal_Loan' a factor para la visualización
datos$Personal_Loan <- as.factor(datos$Personal_Loan)

# Crear el gráfico de pares (pair plot)
ggpairs(datos[, c(num_cols, 'Personal_Loan')],
        aes(color = Personal_Loan, alpha = 0.5)) +
  theme_minimal() +
  ggtitle("Relación entre Variables Numéricas y Aceptación de Préstamo Personal")
```



Relación entre Variables Categóricas y Aceptación de Préstamo Personal

```
# Seleccionar las columnas categóricas que quieres analizar
cat_cols <- c('Education', 'Securities_Account', 'CD_Account', 'Online', 'CreditCard')

# Crear una lista para almacenar los gráficos
plots <- list()

# Generar gráficos de barras para cada columna categórica
for (col in cat_cols) {
  p <- ggplot(datos, aes_string(x = col, fill = 'Personal_Loan')) +
    geom_bar(position = "dodge") +
    labs(title = paste("", col, "- P Personal"),
         x = col,
         y = "Conteo") +
    theme_minimal() +
    theme(axis.text.x = element_text(angle = 45, hjust = 1))
  plots[[col]] <- p
}

# Organizar los gráficos en una cuadrícula 2x3
grid.arrange(grobs = plots, ncol = 2)
```



Pre-Procesamiento ### Eliminar una de las mas correlacionadas y las variable no predictivas

```
# Eliminar columnas no numéricas como ID o ZIP
DataSet <- datos %>% select(-Age, -ID, -ZIP_Code)

DataSet
```

```
## # A tibble: 5,000 × 11
##    Experience Income Family CCAvg Education Mortgage Personal_Loan
##         <dbl>  <dbl>  <dbl> <dbl>     <dbl>    <dbl> <fct>
##  1          1     49      4   1.6         1        0 0
##  2         19     34      3   1.5         1        0 0
##  3         15     11      1   1           1        0 0
##  4          9    100      1   2.7         2        0 0
##  5          8     45      4   1           2        0 0
##  6         13     29      4   0.4         2      155 0
##  7         27     72      2   1.5         2        0 0
##  8         24     22      1   0.3         3        0 0
##  9         10     81      3   0.6         2      104 0
## 10          9    180      1   8.9         3        0 1
## # ℹ 4,990 more rows
## # ℹ 4 more variables: Securities_Account <dbl>, CD_Account <dbl>, Online <dbl>,
## #   CreditCard <dbl>
```

Winsorizing: No elimina datos, solo los limita

```
winsorize <- function(x, low = 0.01, high = 0.99) {
  q_low <- quantile(x, low)
  q_high <- quantile(x, high)
  x[x < q_low] <- q_low
  x[x > q_high] <- q_high
  return(x)
}
```

```r
# Aplicar a columnas
DataSet$Income <- winsorize(datos$Income)
DataSet$CCAvg <- winsorize(datos$CCAvg)
DataSet$Experience <- winsorize(datos$Experience)
DataSet$Mortgage <- winsorize(datos$Mortgage)
```

Analisis de balanceo de Clases

```r
DataSet %>%
  count(Personal_Loan) %>%
  mutate(porcentaje = round(100 * n / sum(n), 2))
```

```
## # A tibble: 2 × 3
##   Personal_Loan     n porcentaje
##   <fct>         <int>      <dbl>
## 1 0              4520       90.4
## 2 1               480        9.6
```

Dividir los datos (train / test)

```r
library(caTools)
set.seed(123)
split <- sample.split(DataSet$Personal_Loan, SplitRatio = 0.7)
train <- subset(DataSet, split == TRUE)
test <- subset(DataSet, split == FALSE)

skim(train)
```

Data summary

Name	train
Number of rows	3500
Number of columns	11
_____	
Column type frequency:	
factor	1
numeric	10
_____	
Group variables	None

Variable type: factor

skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
Personal_Loan	0	1	FALSE	2	0: 3164, 1: 336

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
Experience	0	1	20.21	11.49	-1	10.0	20.0	30.0	41.00	▆▆▆▆▇

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
Income	0	1	74.16	45.99	10	39.0	64.0	99.0	193.00	
Family	0	1	2.38	1.15	1	1.0	2.0	3.0	4.00	
CCAvg	0	1	1.95	1.74	0	0.7	1.6	2.6	8.00	
Education	0	1	1.87	0.84	1	1.0	2.0	3.0	3.00	
Mortgage	0	1	54.36	96.31	0	0.0	0.0	99.0	431.01	
Securities_Account	0	1	0.11	0.31	0	0.0	0.0	0.0	1.00	
CD_Account	0	1	0.06	0.24	0	0.0	0.0	0.0	1.00	
Online	0	1	0.59	0.49	0	0.0	1.0	1.0	1.00	
CreditCard	0	1	0.29	0.45	0	0.0	0.0	1.0	1.00	

```
skim(test)
```

Data summary

Name	test
Number of rows	1500
Number of columns	11
_____	
Column type frequency:	
factor	1
numeric	10

Group variables	None

Variable type: factor

skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
Personal_Loan	0	1	FALSE	2	0: 1356, 1: 144

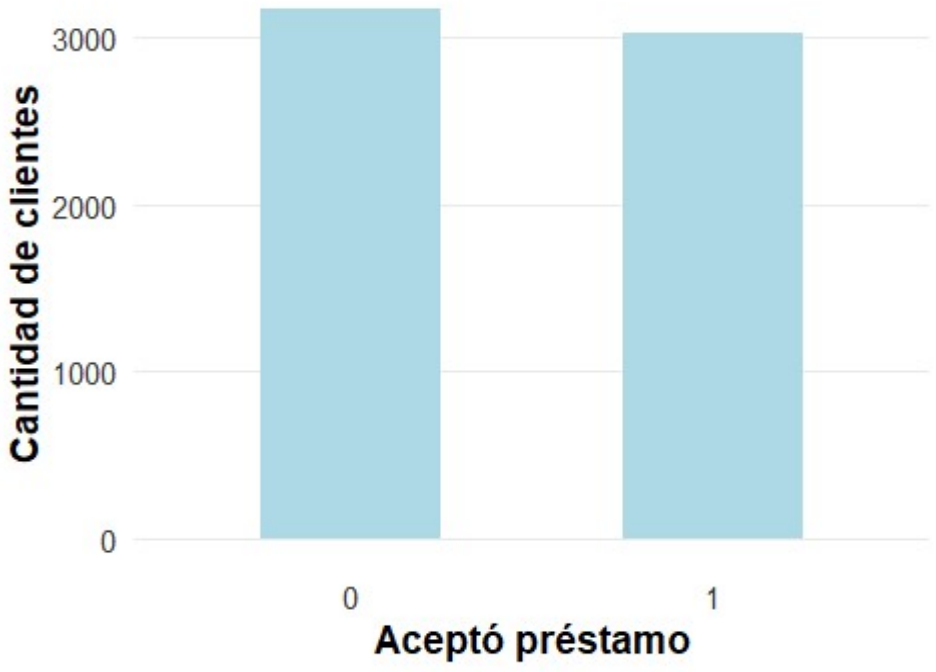
Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
Experience	0	1	19.85	11.37	-1	10.00	20.00	29.0	41.00	▇▇▆
Income	0	1	72.74	45.56	10	38.00	63.00	94.0	193.00	▇▃▂
Family	0	1	2.44	1.15	1	1.00	2.0	4.0	4.00	▇▁▆▆
CCAvg	0	1	1.89	1.71	0	0.67	1.5	2.5	8.00	▇▆▁▁
Education	0	1	1.91	0.83	1	1.00	2.0	3.0	3.00	▇▁▆▁▆
Mortgage	0	1	58.78	102.32	0	0.00	0.0	104.0	431.01	▇▁▁▁
Securities_Account	0	1	0.10	0.30	0	0.00	0.0	0.0	1.00	▇▁▁▁
CD_Account	0	1	0.06	0.24	0	0.00	0.0	0.0	1.00	▇▁▁▁
Online	0	1	0.61	0.49	0	0.00	1.0	1.0	1.00	▅▁▁▆
CreditCard	0	1	0.30	0.46	0	0.00	0.0	1.0	1.00	▇▁▁▃

Balanceo de Clase

```r
library(smotefamily)

# Eliminar variables no predictivas
train_smote <- train

# Convertir variable objetivo a numérico 0/1 si es factor
train_smote$Personal_Loan <- as.numeric(as.character(train_smote$Personal_Loan))

# Aplicar SMOTE
smote_data <- SMOTE(X = train_smote[,-which(names(train_smote) == "Personal_Loan")],
                    target = train_smote$Personal_Loan,
                    K = 5)

# Recuperar datos balanceados
datos_balanceados <- smote_data$data
names(datos_balanceados)[names(datos_balanceados) == "class"] <- "Personal_Loan"

datos_balanceados$Personal_Loan <- as.factor(datos_balanceados$Personal_Loan)

# Verificar proporciones
table(datos_balanceados$Personal_Loan)

##
##    0    1
## 3164 3024

datos_balanceados

## # A tibble: 6,188 × 11
##    Experience Income Family CCAvg Education Mortgage Securities_Account
##         <dbl>  <dbl>  <dbl> <dbl>     <dbl>    <dbl>              <dbl>
##  1         13    129      3   4.1         3        0                  0
##  2         21    134      4   5.5         2        0                  0
##  3         37    124      3   5           2      170                  0
##  4         10    172      4   1           2      295                  0
##  5          8    145      1   2.7         3        0                  0
##  6         13    168      2   1.3         3        0                  0
##  7         22     95      2   3.9         2        0                  0
##  8         38    134      3   4           2        0                  0
##  9         22    102      3   4.5         3        0                  0
## 10          2    182      2   3.2         2        0                  0
## # ℹ 6,178 more rows
## # ℹ 4 more variables: CD_Account <dbl>, Online <dbl>, CreditCard <dbl>,
## #   Personal_Loan <fct>

library(ggplot2)

ggplot(datos_balanceados, aes(x = factor(Personal_Loan))) +
  geom_bar(
    fill = "lightblue",          # color más elegante
    width = 0.5                # controla el ancho de las barras
  ) +
  labs(
    x = "Aceptó préstamo",
    y = "Cantidad de clientes",
    title = "Distribución de aceptación de préstamos balanceados"
  ) +
  theme_minimal(base_size = 14) +   # tamaño base de fuente más grande
  theme(
    plot.title = element_text(hjust = 0.5, face = "bold"), # título centrado y en negrita
    axis.text = element_text(color = "gray20"),            # texto de ejes más estilizado
```

```
    axis.title = element_text(face = "bold"),              # títulos de ejes en negrita
    panel.grid.major.x = element_blank(),                  # quita líneas verticales
    panel.grid.minor = element_blank()
  )
```



Modelado

```
# ============================================================
# Función para calcular la métrica KS
# ============================================================

# actual: variable real (0/1)
# predicted_prob: probabilidades predichas para la clase positiva (1)

calcular_ks <- function(actual, predicted_prob) {

  # Crear un data frame con valores reales y probabilidades predichas
  df_ks <- data.frame(
    actual = as.numeric(as.character(actual)),
    predicted_prob = predicted_prob
  )

  # Ordenar de mayor a menor probabilidad predicha
  df_ks <- df_ks[order(-df_ks$predicted_prob), ]

  # Calcular acumulado de eventos (clase 1)
  df_ks$cum_eventos <- cumsum(df_ks$actual) / sum(df_ks$actual)

  # Calcular acumulado de no eventos (clase 0)
  df_ks$cum_no_eventos <- cumsum(1 - df_ks$actual) / sum(1 - df_ks$actual)

  # Calcular diferencia absoluta entre ambas distribuciones acumuladas
  df_ks$diferencia <- abs(df_ks$cum_eventos - df_ks$cum_no_eventos)

  # Obtener el valor máximo de la diferencia
```

```
  ks <- max(df_ks$diferencia)

  # Devolver el estadístico KS
  return(ks)
}
```

Regresion Logistica

Entrenar modelo de regresión logística

```
modelo_glm <- glm(Personal_Loan ~ ., data = datos_balanceados, family = "binomial")
summary(modelo_glm)

##
## Call:
## glm(formula = Personal_Loan ~ ., family = "binomial", data = datos_balanceados)
##
## Coefficients:
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)        -1.284e+01  4.225e-01 -30.402  < 2e-16 ***
## Experience          2.911e-03  4.402e-03   0.661   0.5084
## Income              5.948e-02  1.976e-03  30.100  < 2e-16 ***
## Family              6.752e-01  4.786e-02  14.108  < 2e-16 ***
## CCAvg               3.152e-01  3.307e-02   9.531  < 2e-16 ***
## Education           1.928e+00  8.411e-02  22.917  < 2e-16 ***
## Mortgage            7.955e-04  4.205e-04   1.892   0.0585 .
## Securities_Account -1.378e+00  2.334e-01  -5.903 3.56e-09 ***
## CD_Account          5.030e+00  2.708e-01  18.575  < 2e-16 ***
## Online             -6.525e-01  1.134e-01  -5.756 8.62e-09 ***
## CreditCard         -1.129e+00  1.369e-01  -8.249  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 8575.2  on 6187  degrees of freedom
## Residual deviance: 2735.2  on 6177  degrees of freedom
## AIC: 2757.2
##
## Number of Fisher Scoring iterations: 7
```

Predicciones sobre test

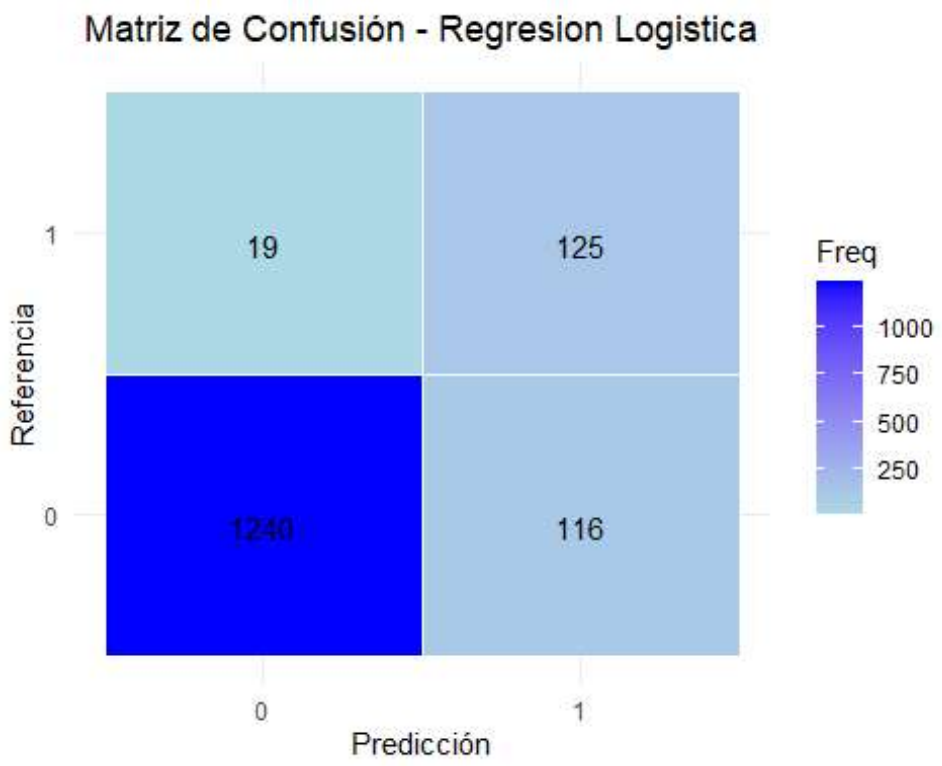
```
# Predicciones en probabilidades
prob_pred <- predict(modelo_glm, newdata = test, type = "response")

# Convertir a clases con umbral 0.5
pred_clase <- ifelse(prob_pred > 0.5, 1, 0)
```

Metricas

```
# Convertir a factores para matriz
pred_clase <- factor(pred_clase, levels = c(0,1))
real <- factor(test$Personal_Loan, levels = c(0,1))

# Matriz de confusión
conf <- confusionMatrix(pred_clase, real, positive = "1")
conf

## Confusion Matrix and Statistics
##
```

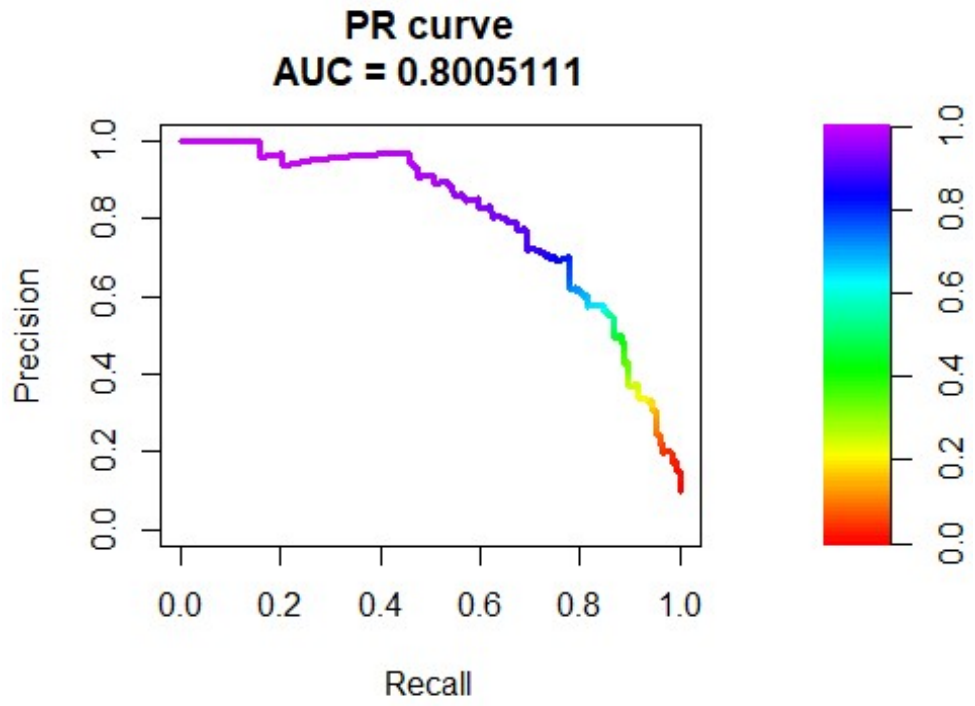
```
##          Reference
## Prediction    0    1
##          0 1240   19
##          1  116  125
##
##                Accuracy : 0.91
##                  95% CI : (0.8944, 0.924)
##     No Information Rate : 0.904
##     P-Value [Acc > NIR] : 0.2297
##
##                   Kappa : 0.6015
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.86806
##             Specificity : 0.91445
##          Pos Pred Value : 0.51867
##          Neg Pred Value : 0.98491
##              Prevalence : 0.09600
##          Detection Rate : 0.08333
##    Detection Prevalence : 0.16067
##       Balanced Accuracy : 0.89125
##
##        'Positive' Class : 1
##
```

```r
# Convertir la matriz de confusión en un dataframe
cm_df <- as.data.frame(conf$table)

# Crear el gráfico con `ggplot2`
ggplot(data = cm_df, aes(x = Prediction, y = Reference)) +
  geom_tile(aes(fill = Freq), color = "white") +
  scale_fill_gradient(low = "lightblue", high = "blue") +
  geom_text(aes(label = Freq), vjust = 1) +
  theme_minimal() +
  labs(title = "Matriz de Confusión - Regresion Logistica",
       x = "Predicción",
       y = "Referencia") +
  theme(plot.title = element_text(hjust = 0.5))
```



Curvas

```r
library(PRROC)

# Convertir las etiquetas reales a formato binario (1/0)
labels <- as.numeric(as.character(test$Personal_Loan))

# Crear la curva Precision-Recall y calcular AUC-PR
pr <- pr.curve(scores.class0 = prob_pred, weights.class0 = labels, curve = TRUE)

# Mostrar el área bajo la curva (AUC-PR)
cat("Área Bajo la Curva de Precision-Recall (AUC-PR):", pr$auc.integral, "\n")

## Área Bajo la Curva de Precision-Recall (AUC-PR): 0.8005111

# Graficar la curva Precision-Recall
plot(pr)
```



Árbol de decisión

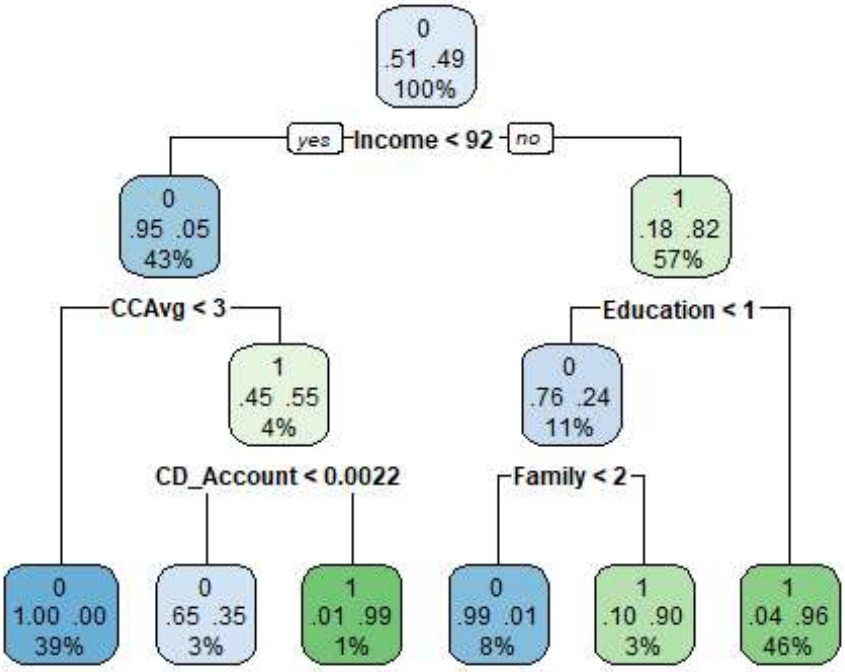
Usamos el dataset ya balanceado que hicimos antes ### Entrenar modelo de Arbol de decisión

```
library(rpart)
library(rpart.plot)

modelo_arbol <- rpart(Personal_Loan ~ ., data = datos_balanceados, method = "class", parms
= list(split = "gini"))
summary(modelo_arbol)

## Call:
## rpart(formula = Personal_Loan ~ ., data = datos_balanceados,
##     method = "class", parms = list(split = "gini"))
##   n= 6188
##
##           CP nsplit  rel error     xerror        xstd
## 1 0.74173280      0 1.00000000 1.00000000 0.013003258
## 2 0.11970899      1 0.25826720 0.26058201 0.008671676
## 3 0.04662698      2 0.13855820 0.13988095 0.006564670
## 4 0.01223545      3 0.09193122 0.09358466 0.005434333
## 5 0.01000000      5 0.06746032 0.07308201 0.004827444
##
## Variable importance
##     Income      CCAvg  Education     Family CD_Account CreditCard
##         39         24         16         11          6          3
##
## Node number 1: 6188 observations,    complexity param=0.7417328
##   predicted class=0  expected loss=0.4886878  P(node) =1
```

```
##     class counts:  3164  3024
##    probabilities: 0.511 0.489
##   left son=2 (2651 obs) right son=3 (3537 obs)
##   Primary splits:
##       Income     < 92.01075     to the left,  improve=1780.6660, (0 missing)
##       CCAvg      < 2.800709     to the left,  improve=1222.8670, (0 missing)
##       CD_Account < 0.0007947645 to the left,  improve= 763.3755, (0 missing)
##       Education  < 1.000438     to the left,  improve= 632.9414, (0 missing)
##       Family     < 1.001474     to the left,  improve= 294.8599, (0 missing)
##   Surrogate splits:
##       CCAvg      < 2.503794     to the left,  agree=0.793, adj=0.516, (0 split)
##       Family     < 3.999716     to the right, agree=0.627, adj=0.129, (0 split)
##       CD_Account < 0.0007947645 to the left,  agree=0.619, adj=0.112, (0 split)
##       Education  < 2.99801      to the right, agree=0.609, adj=0.086, (0 split)
##       CreditCard < 0.9995869    to the right, agree=0.608, adj=0.084, (0 split)
##
## Node number 2: 2651 observations,    complexity param=0.01223545
##   predicted class=0  expected loss=0.05054696  P(node) =0.4284098
##     class counts:  2517   134
##    probabilities: 0.949 0.051
##   left son=4 (2406 obs) right son=5 (245 obs)
##   Primary splits:
##       CCAvg      < 2.95         to the left,  improve=133.033000, (0 missing)
##       CD_Account < 0.002160646  to the left,  improve= 57.850330, (0 missing)
##       Income     < 82.11848     to the left,  improve= 44.663630, (0 missing)
##       Education  < 1.036352     to the left,  improve=  5.386130, (0 missing)
##       Mortgage   < 213.0222     to the left,  improve=  3.595167, (0 missing)
##   Surrogate splits:
##       Income < 91.0255      to the left,  agree=0.908, adj=0.004, (0 split)
##
## Node number 3: 3537 observations,    complexity param=0.119709
##   predicted class=1  expected loss=0.1829234  P(node) =0.5715902
##     class counts:   647  2890
##    probabilities: 0.183 0.817
##   left son=6 (692 obs) right son=7 (2845 obs)
##   Primary splits:
##       Education  < 1.000438     to the left,  improve=576.10510, (0 missing)
##       Family     < 2.000156     to the left,  improve=222.95970, (0 missing)
##       CD_Account < 0.0007947645 to the left,  improve=119.51540, (0 missing)
##       Income     < 114.0025     to the left,  improve= 59.55168, (0 missing)
##       CCAvg      < 0.6037505    to the left,  improve= 58.34493, (0 missing)
##   Surrogate splits:
##       Family             < 1.001474     to the left,  agree=0.820, adj=0.078, (0 split)
##       CCAvg              < 0.4011619    to the left,  agree=0.815, adj=0.052, (0 split)
##       Securities_Account < 0.9995616    to the right, agree=0.806, adj=0.007, (0 split)
##       Experience         < 0.0055887    to the left,  agree=0.805, adj=0.006, (0 split)
##
## Node number 4: 2406 observations
##   predicted class=0  expected loss=0  P(node) =0.3888171
##     class counts:  2406     0
##    probabilities: 1.000 0.000
##
## Node number 5: 245 observations,    complexity param=0.01223545
##   predicted class=1  expected loss=0.4530612  P(node) =0.03959276
##     class counts:   111   134
##    probabilities: 0.453 0.547
##   left son=10 (169 obs) right son=11 (76 obs)
##   Primary splits:
##       CD_Account < 0.002160646  to the left,  improve=42.64199, (0 missing)
##       Education  < 1.036352     to the left,  improve=25.44557, (0 missing)
##       Income     < 82.11848     to the left,  improve=21.77606, (0 missing)
##       CCAvg      < 4.357911     to the right, improve=14.46077, (0 missing)
##       Online     < 0.002160646  to the left,  improve=13.47308, (0 missing)
```

```
##   Surrogate splits:
##       Securities_Account < 0.07300817   to the left,  agree=0.759, adj=0.224, (0 split)
##       Mortgage           < 140.4963     to the left,  agree=0.722, adj=0.105, (0 split)
##       Income             < 85.09119     to the left,  agree=0.718, adj=0.092, (0 split)
##       Experience         < 34.05793     to the left,  agree=0.710, adj=0.066, (0 split)
##       CreditCard         < 0.002160646  to the left,  agree=0.694, adj=0.013, (0 split)
##
## Node number 6: 692 observations,    complexity param=0.04662698
##   predicted class=0  expected loss=0.2384393  P(node) =0.1118293
##     class counts:   527   165
##    probabilities: 0.762 0.238
##   left son=12 (515 obs) right son=13 (177 obs)
##   Primary splits:
##       Family     < 2.02645      to the left,  improve=207.115900, (0 missing)
##       CD_Account < 0.002934876  to the left,  improve= 60.187310, (0 missing)
##       Mortgage   < 183          to the left,  improve= 26.452330, (0 missing)
##       Income     < 129.3419     to the left,  improve=  7.618980, (0 missing)
##       CCAvg      < 0.65         to the left,  improve=  7.234977, (0 missing)
##   Surrogate splits:
##       CD_Account < 0.002934876  to the left,  agree=0.802, adj=0.226, (0 split)
##
## Node number 7: 2845 observations
##   predicted class=1  expected loss=0.04217926  P(node) =0.4597608
##     class counts:   120  2725
##    probabilities: 0.042 0.958
##
## Node number 10: 169 observations
##   predicted class=0  expected loss=0.3491124  P(node) =0.02731092
##     class counts:   110    59
##    probabilities: 0.651 0.349
##
## Node number 11: 76 observations
##   predicted class=1  expected loss=0.01315789  P(node) =0.01228184
##     class counts:     1    75
##    probabilities: 0.013 0.987
##
## Node number 12: 515 observations
##   predicted class=0  expected loss=0.01165049  P(node) =0.0832256
##     class counts:   509     6
##    probabilities: 0.988 0.012
##
## Node number 13: 177 observations
##   predicted class=1  expected loss=0.1016949  P(node) =0.02860375
##     class counts:    18   159
##    probabilities: 0.102 0.898
```

```r
# Visualizar árbol
rpart.plot(modelo_arbol, type = 2, extra = 104, fallen.leaves = TRUE)
```



Predicciones sobre test

```
# Predicción de probabilidades para clase 1
prob_pred_arbol <- predict(modelo_arbol, newdata = test, type = "prob")[,2]

# Predicción de clases con umbral 0.5
pred_clase_arbol <- ifelse(prob_pred_arbol > 0.5, 1, 0)

# Convertir a factores
pred_clase_arbol <- factor(pred_clase_arbol, levels = c(0,1))
real <- factor(test$Personal_Loan, levels = c(0,1))
```

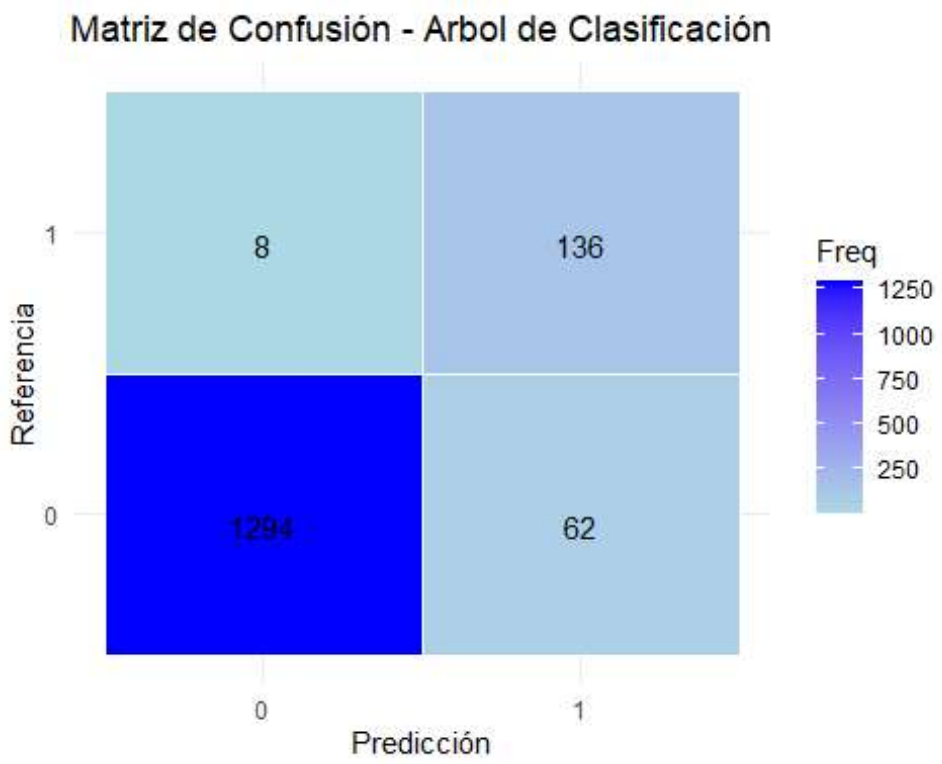
Metricas

```
conf_arbol <- confusionMatrix(pred_clase_arbol, real, positive = "1")
conf_arbol
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 1294    8
##          1   62  136
##
##                Accuracy : 0.9533
##                  95% CI : (0.9414, 0.9634)
##     No Information Rate : 0.904
##     P-Value [Acc > NIR] : 7.606e-13
##
##                   Kappa : 0.7697
##
##  Mcnemar's Test P-Value : 2.378e-10
##
##             Sensitivity : 0.94444
```

```
##             Specificity : 0.95428
##          Pos Pred Value : 0.68687
##          Neg Pred Value : 0.99386
##              Prevalence : 0.09600
##          Detection Rate : 0.09067
##    Detection Prevalence : 0.13200
##       Balanced Accuracy : 0.94936
##
##        'Positive' Class : 1
##
```

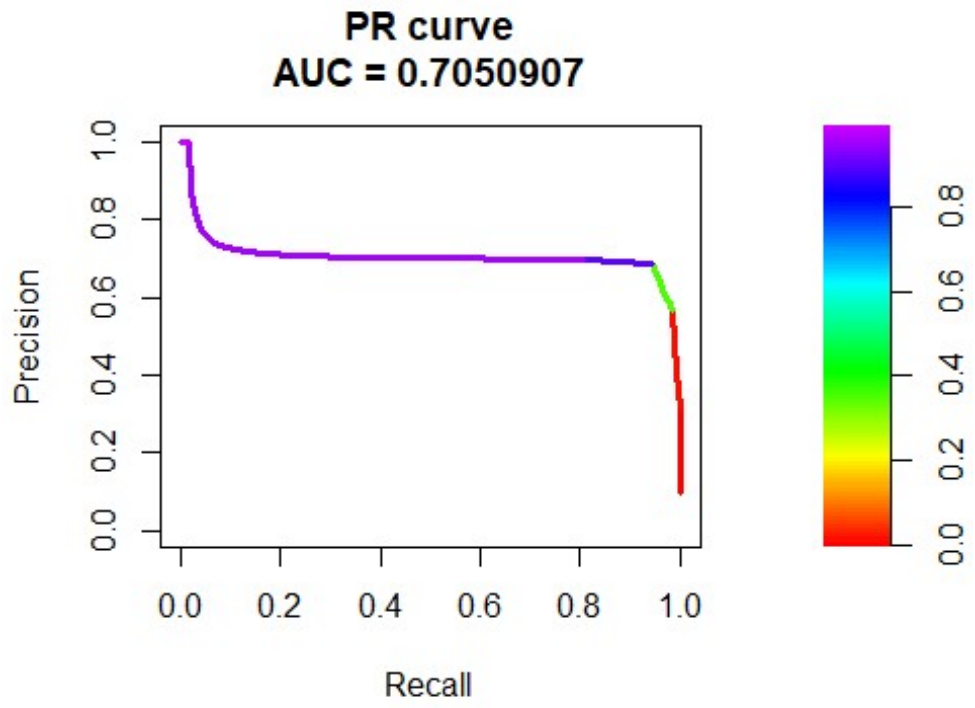
```r
# Convertir la matriz de confusión en un dataframe
cm_df <- as.data.frame(conf_arbol$table)

# Crear el gráfico con `ggplot2`
ggplot(data = cm_df, aes(x = Prediction, y = Reference)) +
  geom_tile(aes(fill = Freq), color = "white") +
  scale_fill_gradient(low = "lightblue", high = "blue") +
  geom_text(aes(label = Freq), vjust = 1) +
  theme_minimal() +
  labs(title = "Matriz de Confusión - Arbol de Clasificación",
       x = "Predicción",
       y = "Referencia") +
  theme(plot.title = element_text(hjust = 0.5))
```



Curvas

```r
library(PRROC)

# Convertir las etiquetas reales a formato binario (1/0)
labels <- as.numeric(as.character(test$Personal_Loan))

# Crear la curva Precision-Recall y calcular AUC-PR
pr <- pr.curve(scores.class0 = prob_pred_arbol, weights.class0 = labels, curve = TRUE)
```

```r
# Mostrar el área bajo la curva (AUC-PR)
cat("Área Bajo la Curva de Precision-Recall (AUC-PR):", pr$auc.integral, "\n")

## Área Bajo la Curva de Precision-Recall (AUC-PR): 0.7050907

# Graficar la curva Precision-Recall
plot(pr)
```



Red Neuronal

Las redes neuronales requieren que los datos estén normalizados (generalmente entre 0 y 1). ### Normalizacion de Datos

```r
# Seleccionar solo variables numéricas (sin ID ni ZIP Code)
cols_a_normalizar <- setdiff(names(datos_balanceados), c("Personal_Loan"))

# Normalización Min-Max
normalizar <- function(x) {
  return((x - min(x)) / (max(x) - min(x)))
}

# Aplicar al train y test
train_nn <- train_smote
train_nn[cols_a_normalizar] <- lapply(train_smote[cols_a_normalizar], normalizar)

test_nn <- test
test_nn[cols_a_normalizar] <- lapply(test[cols_a_normalizar], normalizar)
```

Entrenar la red neuronal

```r
library(neuralnet)
library(caret)
library(pROC)

set.seed(123)
modelo_nn <- neuralnet(Personal_Loan ~ .,
                       data = train_nn,
                        hidden = c(10,5),            # Primera capa con 10 neuronas, segu
nda con 5
                        linear.output = FALSE,       # Indica que la función de activació
n de salida no es lineal (sigmoidal por defecto)
                        act.fct = "logistic",       # Especifica la función de activación
para las capas ocultas
                        stepmax = 1e6                # Aumentamos las iteraciones permiti
das para mejor convergencia
                      )  # salida binaria



summary(modelo_nn)
```

```
##                     Length Class      Mode
## call                     7 -none-     call
## response              3500 -none-     numeric
## covariate            35000 -none-     numeric
## model.list               2 -none-     list
## err.fct                  1 -none-     function
## act.fct                  1 -none-     function
## linear.output            1 -none-     logical
## data                    11 data.frame list
## exclude                  0 -none-     NULL
## net.result               1 -none-     list
## weights                  1 -none-     list
## generalized.weights      1 -none-     list
## startweights             1 -none-     list
## result.matrix          174 -none-     numeric
```

```r
# Visualizar red
plot(modelo_nn)
```

Predicciones en el test

```r
# Predecir sobre test
pred_nn <- compute(modelo_nn, test_nn[, -which(names(test_nn) == "Personal_Loan")])
prob_pred_nn <- pred_nn$net.result

# Convertir a clases con umbral 0.5
pred_clase_nn <- ifelse(prob_pred_nn > 0.5, 1, 0)

# Evaluar
pred_clase_nn <- factor(pred_clase_nn, levels = c(0,1))
real <- factor(test_nn$Personal_Loan, levels = c(0,1))
```

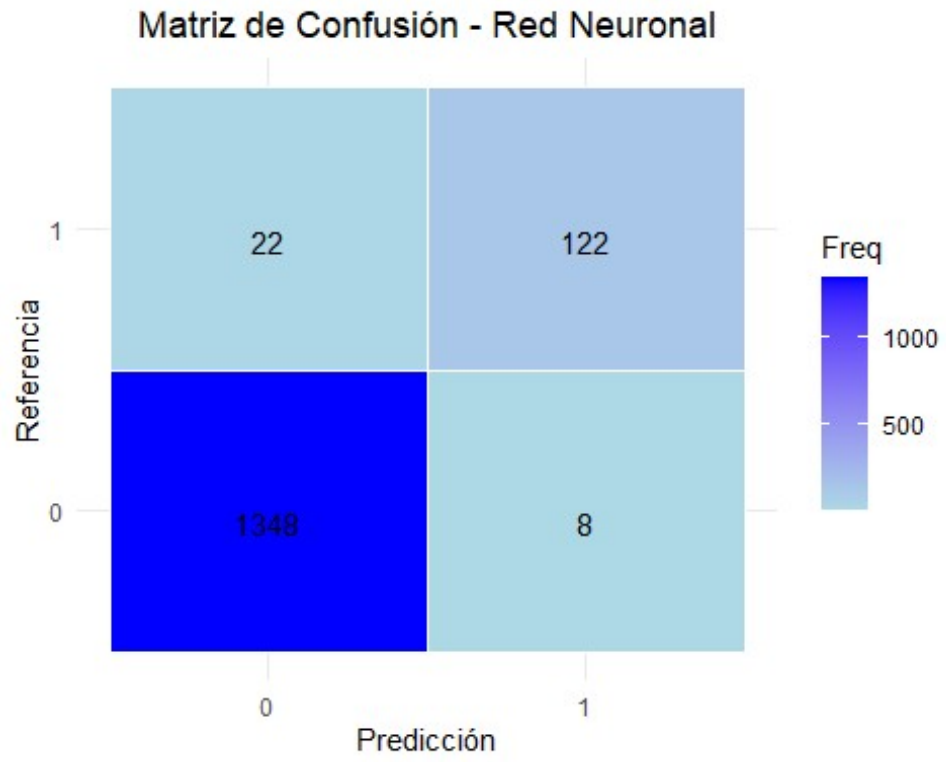
Metricas

```r
conf_nn <- confusionMatrix(pred_clase_nn, real, positive = "1")
conf_nn
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
```

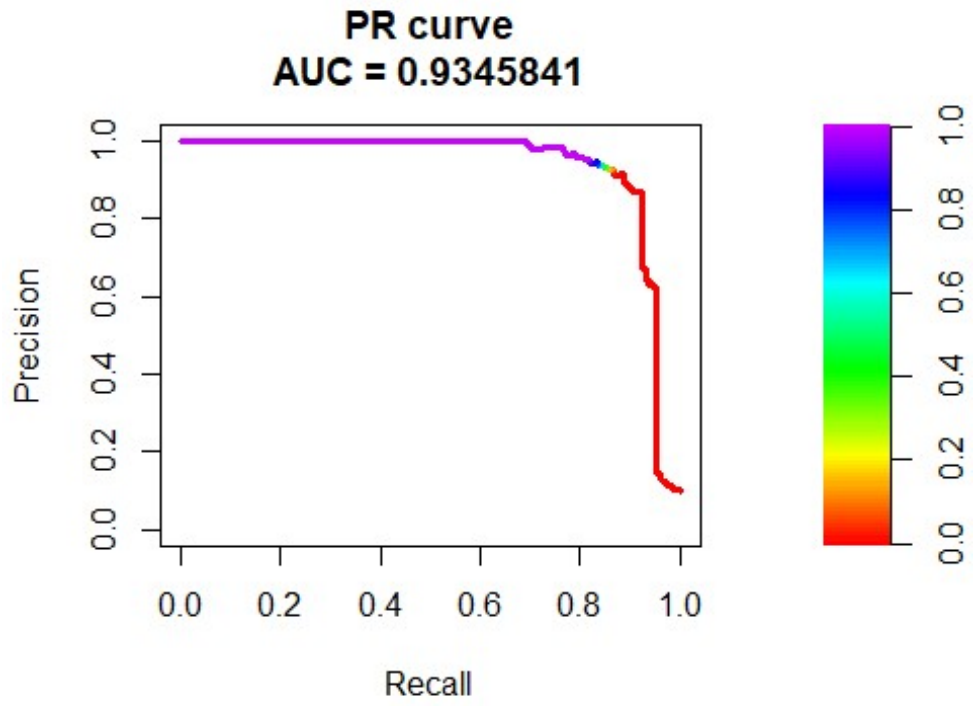
```
##          0 1348   22
##          1    8  122
##
##                Accuracy : 0.98
##                  95% CI : (0.9716, 0.9865)
##     No Information Rate : 0.904
##     P-Value [Acc > NIR] : < 2e-16
##
##                   Kappa : 0.8795
##
##  Mcnemar's Test P-Value : 0.01762
##
##             Sensitivity : 0.84722
##             Specificity : 0.99410
##          Pos Pred Value : 0.93846
##          Neg Pred Value : 0.98394
##              Prevalence : 0.09600
##          Detection Rate : 0.08133
##    Detection Prevalence : 0.08667
##       Balanced Accuracy : 0.92066
##
##        'Positive' Class : 1
##
```

```r
# Convertir la matriz de confusión en un dataframe
cm_df <- as.data.frame(conf_nn$table)

# Crear el gráfico con `ggplot2`
ggplot(data = cm_df, aes(x = Prediction, y = Reference)) +
  geom_tile(aes(fill = Freq), color = "white") +
  scale_fill_gradient(low = "lightblue", high = "blue") +
  geom_text(aes(label = Freq), vjust = 1) +
  theme_minimal() +
  labs(title = "Matriz de Confusión - Red Neuronal",
       x = "Predicción",
       y = "Referencia") +
  theme(plot.title = element_text(hjust = 0.5))
```



Curvas

```
library(PRROC)

# Convertir las etiquetas reales a formato binario (1/0)
labels <- as.numeric(as.character(test$Personal_Loan))

# Crear la curva Precision-Recall y calcular AUC-PR
pr <- pr.curve(scores.class0 = prob_pred_nn, weights.class0 = labels, curve = TRUE)

# Mostrar el área bajo la curva (AUC-PR)
cat("Área Bajo la Curva de Precision-Recall (AUC-PR):", pr$auc.integral, "\n")

## Área Bajo la Curva de Precision-Recall (AUC-PR): 0.9345841

# Graficar la curva Precision-Recall
plot(pr)
```




```
cat("Anexo: \n")

## Anexo:

cat("\n")

cat("Kolmogorov-Smirnov (KS): Esta métrica mide la máxima distancia entre las distribucione
s acumuladas de las probabilidades predichas para las clases positiva y negativa. \n")

## Kolmogorov-Smirnov (KS): Esta métrica mide la máxima distancia entre las distribuciones
acumuladas de las probabilidades predichas para las clases positiva y negativa.

cat("\n")

# Calcular KS usando las probabilidades predichas
ks_glm <- calcular_ks(test$Personal_Loan, prob_pred)

# Mostrar KS
cat("Estadístico KS - Regresión Logística:", round(ks_glm, 4), "\n")

## Estadístico KS - Regresión Logística: 0.7914

# Calcular KS usando las probabilidades predichas del árbol
ks_arbol <- calcular_ks(test$Personal_Loan, prob_pred_arbol)

# Mostrar KS
cat("Estadístico KS - Árbol de Decisión:", round(ks_arbol, 4), "\n")

## Estadístico KS - Árbol de Decisión: 0.9114

# Calcular KS usando las probabilidades predichas de la red neuronal
ks_nn <- calcular_ks(test_nn$Personal_Loan, prob_pred_nn)
```

```r
# Mostrar KS
cat("Estadístico KS - Red Neuronal:", round(ks_nn, 4), "\n")

## Estadístico KS - Red Neuronal: 0.9089
```

8. **Bibliografía**

Amat Rodrigo, J. (2020). *Regresión logística con Python*. https://www.cienciadedatos.net/documentos/py17-regresion-logistica-python.html

Amat Rodrigo, J. (2023, October). *Árboles de decisión con Python: regresión y clasificación*. Zenodo. https://doi.org/10.5281/zenodo.10006330

Anderson, R. (2007). *The Credit Scoring Toolkit: Theory and Practice for Retail Credit Risk Management and Decision Automation*. OUP Oxford. https://books.google.es/books?id=mp0SDAAAQBAJ

García, R. (2021). EL PERCEPTRÓN: UNA RED NEURONAL ARTIFICIAL PARA CLASIFICAR DATOS. *REVISTA DE INVESTIGACIÓN EN MODELOS MATEMATICOS APLICADOS A LA GESTION Y LA ECONOMIA*, *8*(I).

Hand, D. J., & Henley, W. E. (1997). Statistical Classification Methods in Consumer Credit Scoring: A Review. *Journal of the Royal Statistical Society. Series A (Statistics in Society)*, *160*(3), 523–541. http://www.jstor.org/stable/2983268

IBM. (2023). *¿Qué es un árbol de decisión?* Https://Www.Ibm.Com/Es-Es/Topics/Decision-Trees#:~:Text=Un%20%C3%A1rbol%20de%20decisi%C3%B3n%20es%20un%20algoritmo%20de,Nodo%20ra%C3%ADz%2C%20ramas%2C%20nodos%20internos%20y%20nodos%20hoja.

Murrell, J., & Kavlakoglu, E. (2024, January 19). *¿Qué es una matriz de confusión?* IBM. https://www.ibm.com/mx-es/topics/confusion-matrix

Montes de Oca, J. (2015, July 20). *¿Qué es y cómo funciona?* Economipedia.Com

Montgomery, D. C., Peck, E. A., & Vining, G. G. (2021). *Introducción to linear regression analysis*. John Wiley & Sons.

Palacios Burgos, F. J. (2024). *Redes Neuronales con GNU/Linux: Capítulo 2. Conceptos Básicos sobre RNA*. https://www.ibiblio.org/pub/linux/docs/LuCaS/Presentaciones/200304curso-glisa/redes_neuronales/curso-glisa-redes_neuronales-html/x38.html

Rico, C., Paredes, M., Fernández, N., & Marino, B. (2009). MODELACIÓN DE LA ESTRUCTURA JERÁRQUICA DE MACROINVERTEBRADOS BENTÓNICOS A TRAVÉS DE REDES NEURONALES ARTIFICIALES Modeling of the Hierarchical Structure of Freshwater Macroinvertebrates Using Artificial Neural Networks. In *Acta biol. Colomb* (Vol. 14). http://hidroinformatica.unipamplona.edu.co

Tan, P.-N., Karpatne, A., Steinbach, M., & Kumar, V. (2019). Classification: Basic Concepts and Techniques. In *Introduction to Data Mining* (2nd ed., pp. 133–213). John Wiley & Sons.

Walke, K. (2019). *Bank Personal Loan Modelling*. https://www.kaggle.com/datasets/krantiswalke/bank-personal-loan-modelling