PONTIFICIA UNIVERSIDAD CATÓLICA DE VALPARAÍSO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA INFORMÁTICA

PREDICCIÓN DE FUGAS DE CLIENTES EN UNA COMPAÑÍA DE SEGUROS UTILIZANDO REDES NEURONALES ARTIFICIALES EVOLUTIVAS

CLAUDIO IGNACIO MARTÍNEZ ROMERO

TESIS DE GRADO
MAGÍSTER EN INGENIERÍA INFORMÁTICA

Diciembre de 2010

Pontificia Universidad Católica de Valparaíso
Facultad de Ingeniería
Escuela de Ingeniería Informática

PREDICCIÓN DE FUGAS DE CLIENTES EN UNA COMPAÑÍA DE SEGUROS UTILIZANDO REDES NEURONALES ARTIFICIALES EVOLUTIVAS

CLAUDIO IGNACIO MARTÍNEZ ROMERO

Profesor Guía: **José Luis Martí Lara**

Programa: **Magíster en Ingeniería Informática**

Diciembre de 2010

Resumen

Resumen

En este trabajo se presenta un modelo que permite predecir la fuga de clientes de una compañía de seguros bajo el contexto de inteligencia de negocios. Este modelo predictivo es obtenido mediante la aplicación de distintas técnicas de minería de datos usadas en un problema de clasificación binaria, donde el modelo deba explicar el comportamiento de los clientes (fuga o no fuga). Los resultados obtenidos, mediante el análisis de datos, son discutidos por el experto de negocio y ayudan a explicar el fenómeno de fuga, y tomar las medidas necesarias para aumentar la tasa de retención de clientes.

Palabras claves: inteligencia de negocios, fuga de clientes, minería de datos, clasificación binaria, modelo predictivo.

Abstract

This thesis presents a predictive model solution to forecast the insurance company churn, under context business intelligence. The predictive model is obtained using different techniques of data mining, applied to a binary classification problem, where the model must to explain the costumers behavior (churn or not). The results obtained by data analysis are discussed with the business expert and they are useful to explain the churn phenomena and took the actions for increase customer retention.

Key words: business intelligence, customer churn, data mining, binary classification, predictive model.

Índice de Contenido

Índice de Figuras

Índice de Tablas

Introducción

Las organizaciones modernas disponen cada vez de más datos sobre sus negocios; de éstos pueden obtener información relevante para mejorar el desempeño e innovar en procesos para transformarse en entidades más competitivas y exitosas. En este punto es donde la inteligencia de negocios provee de distintas herramientas y técnicas, que permiten ser implementadas en las diferentes áreas del negocio. Estos recursos podrían definirse como las tecnologías, aplicaciones y prácticas para la recolección, integración, análisis y presentación de información que permite apoyar y mejorar la toma de decisiones.

La predicción de la fuga de clientes ha sido una arista de intensivo estudio por parte de las empresas prestadoras de bienes y servicios. Esto se centra en que la cartera de clientes es uno de los mayores activos para cualquier institución, lo que genera incentivos para mantener y aumentar el número de clientes que la componen; mantenerlos genera ventajas competitivas sustentables en el tiempo, puesto que entregan utilidades en el largo plazo. Considerando el desafío de obtener una alta tasa de retención de clientes, en los últimos años se han utilizado diferentes métodos para encontrar patrones de comportamiento para predecir la fuga de clientes. Dentro de estos métodos se encuentran algoritmos de clasificación basados en redes neuronales, lógica difusa, redes bayesianas, árboles de decisión y regresión logística, entre otros.

La organización elegida para este trabajo corresponde a una compañía de seguros de vida, cuyo objetivo principal es obtener un modelo que permita identificar patrones de comportamiento de los clientes con mayor probabilidad de fuga, y así realizar una acción temprana con el fin de enfocar nuevas estrategias de retención, orientados a los segmentos indicados, además de analizar las posibles causas que motivan este comportamiento.

La estructura del presente documento se encuentra estructurado como sigue: en el capítulo 1 se plantea la descripción del problema, partiendo con una breve introducción al mercado de seguros y en especial el mercado de seguros de vida. En el capítulo 2 se presenta el estado del arte de la inteligencia de negocios y minería de datos, con la descripción de las metodologías y técnicas utilizadas para resolver el problema de clasificación binaria. En el capítulo 3 se expone el diseño de la solución detallando cada etapa del trabajo realizado, mientras que el capítulo 4 contiene el análisis y preparación de los datos del estudio. En el capítulo 5 se realiza la aplicación de la minería de datos. Finalizando se presentan las conclusiones y el trabajo futuro propuesto.

CAPÍTULO 1

Descripción del Problema

1.1 Descripción del Mercado de Seguros

La venta de seguros en Chile puede ser realizada por compañías de seguros generales, denominadas de primer grupo, o por compañías de seguros de vida, llamadas de segundo grupo. Las primeras cubren el riesgo de pérdida o deterioro de los bienes o el patrimonio, mientras que las compañías de seguros de vida cubren los riesgos de las personas o bien garantizan a ésta, dentro o al término de un plazo, un capital, una póliza saldada o una renta para el asegurado y/o sus beneficiarios. En forma excepcional, los riesgos de accidentes personales y los de salud pueden ser cubiertos por ambos tipos de compañías.

Los riesgos de crédito sólo pueden asegurarse en compañías de seguros generales que tengan por objeto exclusivo precisamente cubrir este tipo de riesgo pudiendo, además, cubrir los de garantía y fidelidad.

Las compañías de seguros sólo pueden ser sociedades anónimas constituidas en Chile con dicho objeto exclusivo. En consecuencia, las entidades aseguradoras extranjeras no pueden ofrecer ni contratar seguros en Chile, directamente o a través de intermediarios. La infracción de esta prohibición es constitutiva de delito.

La contratación de un seguro se formaliza mediante la emisión de una póliza de seguro, la cual es el documento justificativo del contrato que establece los derechos y obligaciones del asegurado y del asegurador. Mediante este contrato el asegurador se obliga, en el caso que se produzca un siniestro cubierto por la póliza, a indemnizar al asegurado o a sus beneficiarios de acuerdo a las condiciones del seguro; por su parte, el asegurado se obliga al pago de una prima estipulada en la póliza. Como excepción, podrá contratarse con modelos no registrados tratándose de seguros generales en que tanto asegurado como beneficiario sean personas jurídicas, y cuando la prima anual sea superior a UF 200.

Las primas de seguros en Chile son fijadas libremente por los aseguradores. Asimismo, las comisiones por intermediación también son libremente convenidas entre asegurador y corredor de seguros, dejándose constancia de ella en la respectiva póliza.

Las compañías aseguradoras deben contratar sus seguros utilizando los modelos de pólizas y cláusulas que se encuentran registrados en el Registro de Pólizas de la Superintendencia de Valores y Seguros (SVS) [1]. Ésta corresponde al organismo gubernamental encargado de la supervisión y regularización de las compañías de seguros, cuyo objetivo es velar el cumplimiento de las leyes, reglamentos, estatutos y otras disposiciones que rijan el funcionamiento de estos mercados.

En la figura 1 se puede apreciar el crecimiento del mercado desde 1985 hasta el 2008. Se evidencia la participación de las empresas a las cuales corresponde la venta efectiva de seguros, tanto de seguros de vida como para seguros generales.

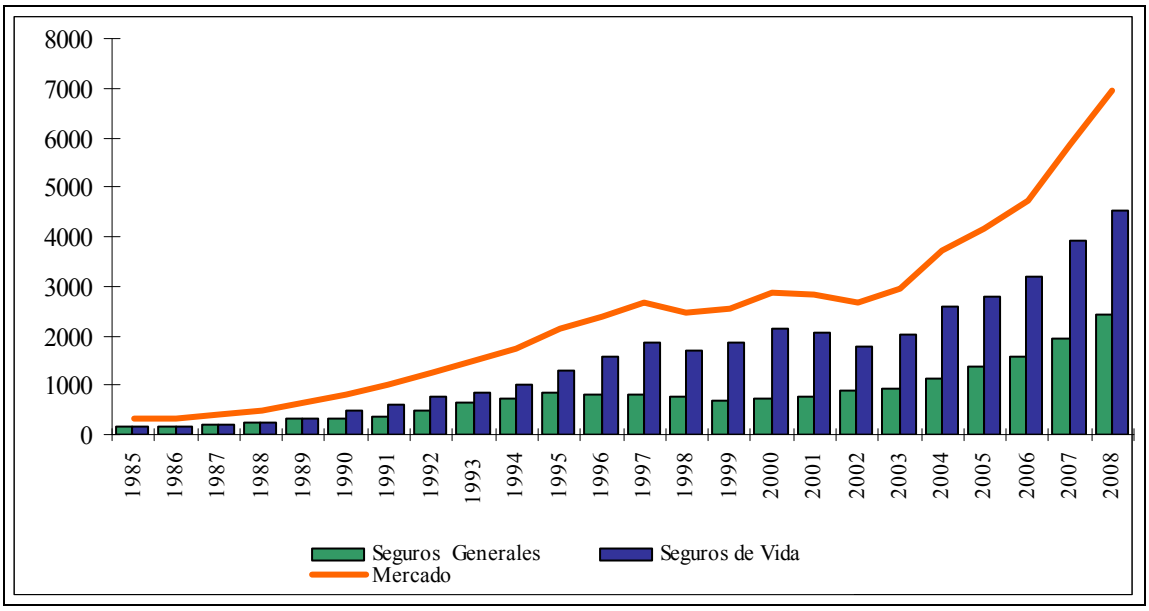


Figura 1: Número de Participantes en el Mercado Asegurador

Otro fenómeno que se puede observar en la figura 2, es el aumento de las inversiones de las compañías aseguradoras, siendo el mercado de seguros de vida el que más inversiones tiene. Esta estadística indica que este mercado es altamente competitivo por lo cual, para tener mejores ingresos, debe existir una buena política para captar nuevos clientes y fidelizar al los existentes.

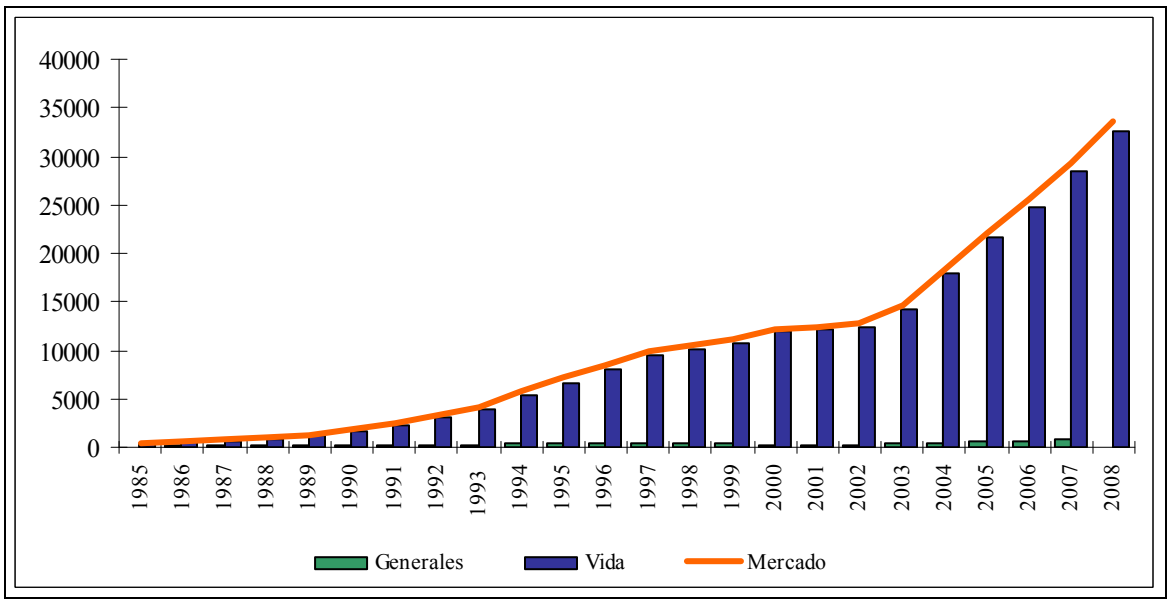


Figura 2: Comportamiento de Inversiones en el Mercado Asegurador

1.2 Descripción de la Empresa

La empresa dónde se realizó el estudio es una compañía de seguros del mercado chileno, perteneciente a un grupo internacional ligado a los seguros con más de ochenta años en el mercado.

En los años ochenta, ubicó su primera oficina en el centro de Santiago y tiempo después comenzó la expansión a regiones, comenzando en Viña del Mar, Chillán, Concepción, San Felipe y Temuco. En 1987, como respuesta a la necesidad de un mayor respaldo tecnológico, se creó una empresa de soporte computacional especial para seguros, pensiones y servicios financieros. Poco tiempo después, se fortalece el negocio con la creación de rentas vitalicias como apoyo a los clientes que inician su jubilación.

El año 1998 fue nombrado **Asegurador Oficial** de la segunda Cumbre de las Américas que se realizó en Chile logrando la clasificación AAA por la prestigiosa clasificadora de riesgo internacional Duff & Phelps [1].

1.3 Situación Actual

Las estrategias que toma cada compañía con el fin de retener a los clientes, son vitales para mantenerlos cautivos. La empresa de estudio no posee un modelo concreto para la detección de fugas de sus clientes, por lo tanto, las decisiones se basan principalmente sobre un análisis multidimensional de los datos, y en el propio conocimiento de los analistas de negocio.

La consecuencia de lo anterior es que poseen un elevado número de clientes que son "falsos positivos", es decir, que son incluidos dentro del segmento objetivo al que se le aplicarán acciones de retención, siendo que este incremento genera una carga de trabajo y de recursos excesiva para el Departamento de Ventas, ya que se le aplican ofertas y se les planifica realizar visitas. Esta carga origina, por otro lado, que a muchos de los potenciales clientes en fuga no se les aplique las acciones de retención.

1.4 Justificación de resolver el problema.

La identificación del cliente de forma temprana permite la aplicación de estrategias de fidelización y retención de manera eficaz y eficiente. Existen estudios en el que demuestran beneficio tangible para las empresas con las políticas de retención [2]; un ejemplo claro es el aumento de los ingresos generados por el incremento en promedio del número de pólizas contratadas. Por otra parte, al tener una pronta identificación de potenciales clientes a fuga, es posible mejorar la focalización de los recursos utilizados por las estrategias de retención. Esta focalización permite optimizar los esfuerzos sobre los clientes más propensos y que realmente necesitan acciones de retención.

Otro punto importante a considerar, es que al perder un cliente inmediatamente se gatilla la acción de atraer uno nuevo, el cual, al desconocer su comportamiento a priori, podría ser potencialmente más "peor" cliente que el perdido.

Como se aprecia en la figura 3, los beneficios generados por un cliente en lo largo del tiempo, se comportan de la siguiente manera: durante el año 0, se realiza la incorporación del cliente, aquí sólo se genera costos de incorporación del cliente, producto de los costos operativos de gestión y venta de pólizas de seguro. Cabe destacar que en el mercado asegurador, esto representa los costos más altos en el ejercicio. Desde el año 1 en delante, se observa un crecimiento de los beneficios generados por el cliente, en base a la reducción de costos operativos y a las referencias que éste entrega a sus cercanos, lo que indirectamente disminuye el costo de buscar un cliente [3].

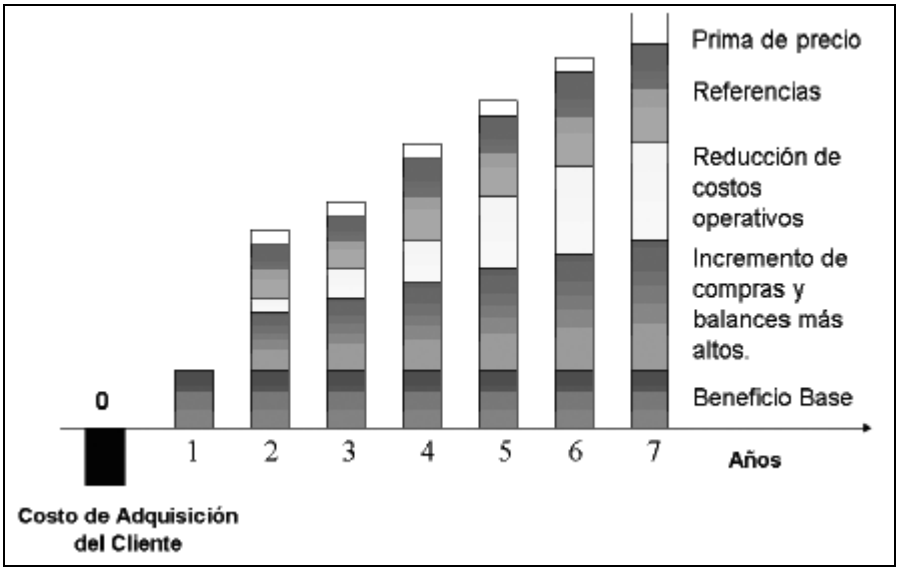


Figura 3: Beneficios generados por un cliente fidelizado

Como conclusión y justificación, resulta muy importante contar con un buen modelo de predicción de fugas que complemente y ayude a la institución, ya que los beneficios esperados de retener al cliente, en vez de remediar la fuga con la captación de nuevos, son mayores y menos costos para la institución en cuestión.

1.5 Objetivos de la investigación

1.5.1 Objetivo General

Obtener un modelo de predicción de fugas enfocado a los clientes de seguros de vida, mediante técnicas de minería de datos, que permita mejorar las acciones de retención de clientes.

1.5.2 Objetivos Específicos

- Investigar la aplicación de modelos predictivos en el fenómeno de fugas de clientes en instituciones prestadoras de servicios.

- Comprender el comportamiento de los clientes a partir de sus datos históricos.

- Generar nuevo conocimiento para la organización respecto a su cartera de clientes.

- Entregar y validar la información clave al analista para crear nuevas estrategias y políticas comerciales para la retención de clientes.

1.6 Alcances y/o limitaciones del trabajo

El presente trabajo se enmarca en el estudio y obtención de un modelo predictivo para detectar clientes que cierren sus pólizas de vida o de accidentes personales de manera voluntaria, quedando fuera del estudio, las pólizas de seguros generales, o que hayan sido cerradas por otros motivos adicionales a los mencionados. Queda fuera de alcance del presente trabajo, la generación de las políticas de retenciones en base a los resultados obtenidos, o apoyo en decisiones comerciales asociadas a la empresa en la cual se desarrolló esta investigación.

CAPÍTULO 2

Estado del Arte

En este capítulo se presentan las bases teóricas de la investigación y los conocimientos que se deben tener presentes en la realización de la misma. El capítulo comienza explicando los distintos procesos existentes para el descubrimiento del conocimiento en bases de datos, realizando una comparación entre ellas, posteriormente se presenta breve introducción de algoritmos de predicción y técnicas utilizadas para resolver problemas de clasificación, más las consideraciones para construir un modelo predictivo. Seguidamente se profundiza en las redes neuronales, sus características, además de mencionar algoritmos evolutivos y de la generación de una técnica híbrida de redes neuronales evolutivas, además se incorpora un resumen de las técnicas utilizadas para la extracción de reglas a partir de una red neuronal entrenada. Finalmente se presentan lo métodos de evaluación de un algoritmo de clasificación.

2.1 Proceso de Descubrimiento del Conocimiento

El proceso de descubrimiento del conocimiento (KDD) es en el cual se realza selección, pre-procesamiento, sub-muestreo y transformaciones de la data contenida en una base de datos; para luego aplicar métodos o algoritmos de minería de datos e identifican un subconjunto de patrones enumerados que llegarán a ser el "conocimiento". En otras palabras una definición más apropiada sería el siguiente: "*Consiste en un proceso no trivial, de identificar patrones previamente desconocidos, válidos, nuevos, potencialmente útiles y comprensibles dentro de los datos*" [4]. Cabe destacar que este proceso es iterativo e interactivo entre cada una de las distintas etapas, ya que pequeños cambios en cualquiera de éstas puede afectar de gran manera el resultado final. En la figura 4 se observan las distintas etapas dentro de un proceso KDD.

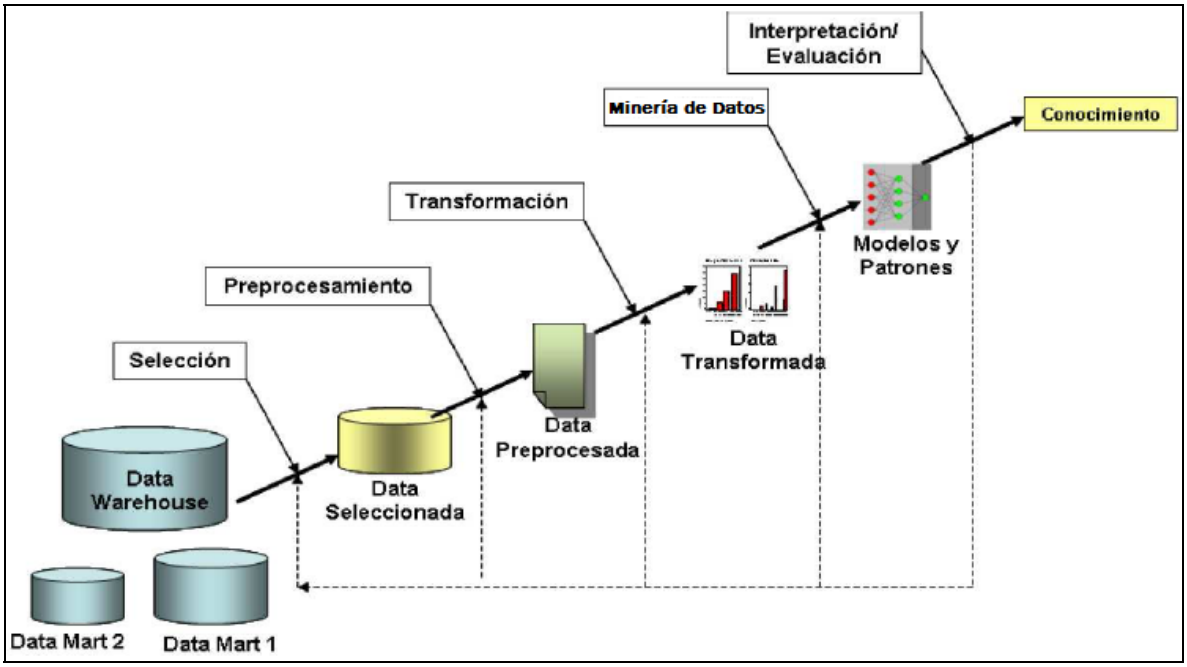


Figura 4: Etapas del KDD

2.1.1 Entendimiento del Problema

En esta etapa se debe desarrollar un entendimiento del dominio de la aplicación y el conocimiento previo relevante; una vez realizado se debe identificar el objetivo del proceso KDD desde la perspectiva del cliente. El desarrollo de esta etapa establece los lineamientos para continuar las siguientes fases del proceso, es decir, que el éxito depende de los objetivos y decisiones que se tomen aquí. Un 80% de la importancia para llegar al éxito proviene en la forma de abordar el problema, definir cuáles pueden ser las pautas para llegar a la solución y la forma de implementarlas para solucionar el problema con éxito [5]. La tabla 1 describe la importancia de las tareas.

Tarea	Porcentaje del Tiempo dedicado	Porcentaje de importancia para llegar al éxito final
Definir el Problema	10 %	15 %
Explorar la Solución	9 %	14 %
Implementación de los resultados	1 %	51 %
Preparación de los datos	60 %	15 %
Procesamiento de los datos	15 %	3 %
Modelado y testeo de los datos.	5 %	2 %

Tabla 1: Tiempo e importancia de las fases

2.1.2 Selección

En esta fase los datos deben de ser seleccionados de forma coordinada por el analista de negocio. Se debe crear un conjunto de datos objetivo o enfocarse en subconjuntos de variables y/o muestras de datos dónde se realizará el descubrimiento del conocimiento. El analista debe indicar cuales son datos más relevantes, como también asegurar la disponibilidad de éstos.

La obtención de datos puede realizarse directamente desde los sistemas transaccionales, archivos planos, archivos semiestructurados o desde un *data warehouse*. Esta decisión depende totalmente de la disponibilidad de los datos; por ejemplo, se optará por la obtención desde los sistemas transaccionales, si no se dispone de un *data warehouse*. Sin embargo, la situación ideal es contar con un repositorio centralizado, independiente de los sistemas transaccionales, ya que al obtener los datos directamente de éstos últimos se corre el riesgo de trabajar con datos con algún grado de ruido o baja calidad.

2.1.3 Preprocesamiento

En esta etapa se realiza la limpieza de datos, la cual considera operaciones básicas como la eliminación del ruido, recolección de la información necesaria para modelar o contabilizar el ruido, decisión sobre estrategias para manejar campos de datos perdidos, y contabilización de la información en las secuencias de tiempo y cambios. Todo esto genera un nuevo conjunto de datos más significativo para trabajar.

Esta tarea toma aproximadamente un 60% del tiempo [5].

2.1.4 Transformación

La fase de transformación busca preparar la información que se tiene para que pueda ser procesada por los algoritmos de minería de datos y además, reducir la cantidad de información redundante para simplificar las tareas posteriores; esto es disminuir los datos y proyectarlos, identificando características útiles para representarlos dependiendo de los objetivos del trabajo.

Las principales tareas son: utilizar métodos de transformación o de reducción de la multidimensionalidad para detectar, analizar y eliminar *outliers*; disminuir el número efectivo de variables bajo consideración; completar datos ausentes, o bien, encontrar representaciones de datos más intuitiva y manejable [6].

2.1.5 Minería de datos

Una vez que se tenga los datos transformados se debe comparar los objetivos de la primera etapa del proceso KDD con los de un método particular de minería de datos. Luego se procede a elegir el algoritmo de minería de datos más apropiado, seleccionando el método para ubicar los patrones en los datos.

Los algoritmos de minería de datos usados en el proceso de extracción de conocimiento se pueden clasificar en [6]:

- **Para análisis exploratorio de datos:** se utilizan técnicas iterativas y visuales; permiten revisar y obtener una idea previa de la estructura de los datos, agrupamientos, puntos de operación, tendencias, etc. Estas técnicas se utilizan preferentemente de manera previa a la realización de cualquier técnica estadística o, en este caso, de minería de datos. De esta forma el analista consigue un entendimiento básico de sus datos y de las relaciones existentes entre las variables analizadas. No obstante, también muchas de las técnicas que se incluyen en este grupo son utilizadas en fases posteriores para el análisis de los resultados, su validación, extracción de nuevas conclusiones, detección de patrones anómalos, etc.
- **Dirigidos por la verificación**: se limita a comprobar hipótesis que plantea el analista. En este grupo se encuentran las técnicas basadas en SQL y las herramientas llamadas *generadoras de SQL* como herramientas de consultas y los cubos OLAP.
- **Dirigidos por el descubrimiento**: en este grupo se encuentran todas las técnicas cuyos resultados pueden ser descriptivos o predictivos. Los algoritmos predictivos toman el comportamiento pasado para prever un comportamiento futuro, mientras los algoritmos descriptivos ayudan a la compresión de este comportamiento. Dentro de las técnicas descriptivas se tiene:
  - » Visualización: generalmente se usan para identificar las causas de problemas o incidencias y buscar las posibles soluciones, siempre y cuando se disponga de la base de información necesaria en la que buscar.
  - » Segmentación: se busca agrupar los datos en grupos de acuerdo a la relación que se encuentre ellos; también se puede generar una jerarquía entre grupos.
  - » Asociación: consiste en establecer las posibles relaciones entre acciones o sucesos aparentemente independientes. Así, se puede reconocer cómo la ocurrencia de un determinado suceso puede inducir la aparición de otros.
  - » Asociación secuencial: incluye el factor tiempo, por lo cual permite reconocer el tiempo que transcurre o suele transcurrir entre el suceso inductor y los sucesos inducidos.

  Dentro de las técnicas predictivas se consideran:

  - » Clasificación: permiten asignar a un elemento la pertenencia a un determinado grupo o clase. Esto se lleva a cabo a través de la pertenencia a cada clase en los valores de una sede de atributos o variables. Se establece una característica a cada clase y su expresión, en función de las distintas variables y además se establece el grado de discriminación o influencia de estas últimas. Con ello, es posible clasificar un nuevo elemento una vez conocidos los valores de las variables presentes en él.

» Regresión: busca establecer el comportamiento futuro más probable de una variable o una serie de variables a partir del comportamiento pasado y presente de las mismas o de otras de las cuales dependan.

» Series temporales: se define como una secuencia de puntos de datos medida a lo largo del tiempo, distanciada en un intervalo, frecuentemente de igual tamaño [8]. El análisis de estas series permite predecir eventos futuros basados en experiencias pasadas, es decir, predecir puntos de datos antes de poder realizar su medición.

2.1.6 Interpretación de los resultados

La interpretación de los patrones obtenidos es un proceso complejo, posiblemente se deba retomar pasos anteriores si los resultados no son lo suficientemente claros; en primera instancia se debe modificar algunas de las decisiones tomadas durante esas etapas, haciendo para ello uso de la nueva información obtenida. Además, también puede implicar la visualización de los modelos/patrones obtenidos, o la visualización de los datos entregados por los mismos.

La verificación de resultados incluye determinar el grado de cumplimiento de los objetivos establecidos durante la primera fase del proceso KDD, así como la validación del conocimiento extraído. Durante esta fase se debe verificar la coherencia de la información obtenida con otros tipos de conocimiento ya previamente extraído o aceptado en la organización, resolviendo las posibles inconsistencias existentes. Si los objetivos finales han sido alcanzados, se debe consolidar el conocimiento descubierto incorporándolo en otro sistema para acciones adicionales, o simplemente documentarlo y reportarlo a las partes interesadas.

2.2 Proceso para minería de datos

Un proyecto de minería de datos requiere de la aplicación de un cierto proceso estructurado que permita aplicar las técnicas explicadas anteriormente y obtener los resultados esperados. La utilización de una facilita la planificación y dirección del proyecto, permitiendo tener un mejor control del mismo. A continuación se presentan los más utilizados.

2.2.1 CRISP-DM

CRISP-DM (*Cross-Industry Standard Process for Data Mining*) [9] es un proceso para el desarrollo de proyectos de minería de datos que se ha convertido en un estándar *de facto*. Contiene un conjunto de actividades seleccionadas en base a la experiencia de "ensayo y error" recogida a través de numerosos proyectos del consorcio CRISP-DM, el cual está compuesto de numerosas empresas que poseen amplia experiencia en el análisis de datos.

Se describe como un proceso jerárquico, que consiste en un conjunto de tareas descritas en cuatro niveles de abstracción, desde el general hasta el específico: fase, tareas generales, tareas específicas e

instancias de proceso. Dichas tareas están ordenadas en primer lugar, horizontalmente, en seis fases sucesivas que recorren toda la vida del proyecto de minería de datos, desde la definición de los objetivos del negocio que se pretende obtener hasta la vigilancia y el mantenimiento del modelo que se proponga e implemente. Cada una de esas fases se ha subdividido, a su vez, en tareas ordenadas en un esquema jerárquico, desde un mayor a un menor nivel de detalle.

CRISP-DM distingue entre el modelo de referencia y la guía del usuario. El modelo de referencia presenta una vista rápida de las fases, tareas y sus salidas, y describe lo que hay que hacer en un proyecto de minería de datos. La guía del usuario da consejos y trucos mucho más detallados para cada fase, y para cada tarea dentro de una fase, y describe cómo desarrollar un proyecto de análisis de datos [9].

Contexto

El contexto del proyecto dirige el cambio entre el nivel general y el especializado. Para cada uno de los contextos se distinguen cuatro dimensiones [9]:

- El dominio de aplicación, la cual es el área en donde se realizará el proyecto.
- El tipo de problema de minería de datos se va a abordar.
- El aspecto técnico que permite considerar posibles problemas que se puedan presentar durante la minería de datos.
- El tipo de herramientas y técnicas que se usarán durante el proyecto.

Un contexto específico de minería de datos es un valor para una o más de estas dimensiones. Por ejemplo, un proyecto que abarca el problema de clasificación para la predicción de fuga de clientes, constituye un contexto específico. Cuantos más valores de dimensiones de diferentes contextos se cubran, más concreto es el contexto del proyecto.

Proyección en el contexto

Se distinguen dos tipos de proyecciones entre los niveles genérico y especializado [9]:

- Proyección para el presente: Este tipo de proyección se aplica sólo si se está aplicando el modelo genérico para llevar a cabo un solo proyecto y proyectar las tareas generales y sus descripciones. Por lo tanto se tiene una proyección sencilla para un solo uso.
- Proyección para el futuro: Este tipo de proyección se aplica si se especializa el modelo genérico de acuerdo a un contexto predefinido, encaminándolo a un modelo de proceso especializado para usar en el futuro en contextos similares.

El tipo de proyección apropiado depende del contexto específico y de las necesidades de cada organización.

Cómo se realiza la proyección

La estrategia básica para proyectar el modelo genérico de proceso al nivel especializado es la misma para todos los tipos de proyecciones:

- Analizar el contexto específico.
- Eliminar cualquier detalle que no sea aplicable en dicho contexto.
- Añadir detalles específicos al contexto.
- Especializar contenidos genéricos de acuerdo a características concretas del contexto.
- Posiblemente, renombrar los contenidos genéricos.

La metodología CRISP-DM define las diferentes fases de las que consta un proyecto, las tareas correspondientes y las relaciones entre ellas; las primeras se muestran en la figura 5. El orden de ejecución de estas fases no es estrictamente secuencial, ya que frecuentemente a lo largo del desarrollo del proyecto, es necesario volver a ejecutar alguna de las fases, dependiendo de los resultados obtenidos en las fases previas. Las etapas son las siguientes:

- **Análisis del Problema de Negocio**: corresponde a la fase inicial que incluye la comprensión de los objetivos y requerimientos del proyecto desde una perspectiva de negocio, con el fin de convertirlos en objetivos y en una planificación.
- **Análisis de los Datos**: recolección inicial de datos para identificar su calidad y descubrir las relaciones entre los más evidentes para las primeras hipótesis de relaciones ocultas entre ellos.
- **Preparación de los Datos**: construcción de la base de datos a partir de los datos primarios. Esta tarea se desarrolla en numerosas ocasiones y no de una forma demasiado estructurada. Incluye la selección de tablas, registros y atributos, así como su transformación y preparación para las herramientas de modelado.
- **Modelamiento**: se seleccionan y aplican varias técnicas de modelado. Generalmente existen varias técnicas para el mismo problema y cada una exige una entrada de datos particular, por ello, es necesario interactuar con la fase anterior para adecuar la base de datos de trabajo. Los parámetros son calibrados.
- **Evaluación**: una vez creado el modelo, se debe evaluar el rendimiento del mismo y la integridad de todos los datos, teniendo en cuenta que se han introducido todos los criterios de negocio. Se debe dar la aprobación final al uso del modelo de minería de datos.
- **Desarrollo**: normalmente los proyectos de minería de datos no terminan en la implantación del modelo, sino en el incremento de conocimiento obtenido a partir de los datos. Para ello es imprescindible documentar y presentar los resultados de manera comprensible.

Las flechas indicadas en la figura 5 indican las relaciones más habituales entre las fases, mientras que el círculo exterior simboliza la naturaleza cíclica de la minería de datos, ya que la solución final obtenida puede conducir al planteamiento de nuevas hipótesis que den origen a otros proyectos.

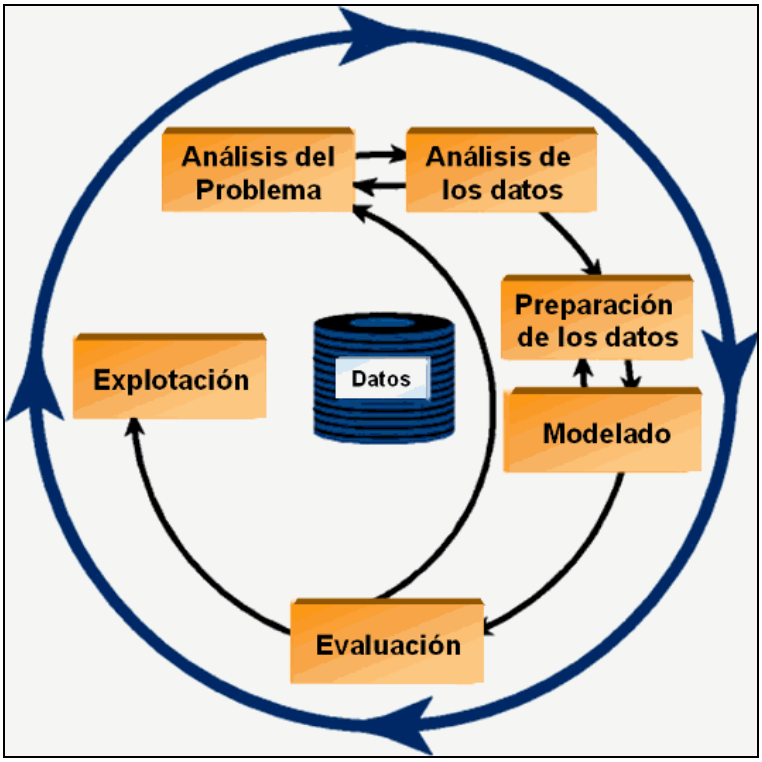


Figura 5: Etapas del modelo de referencia CRISP-DM

2.2.2 SEMMA

SEMMA (*Sample, Explore, Modify, Model and Assess*) fue desarrollada por SAS Institute, quien la define como el proceso de selección, exploración y modelamiento de grandes cantidades de datos para descubrir patrones de negocio desconocidos [10]. Las fases que define se presentan a continuación.

Muestreo

El muestreo corresponde al proceso de extracción de datos desde la población sobre la cual se aplicará el análisis. Este muestreo puede obtenerse de forma aleatoria o también puede ser un subconjunto de datos de un *data warehouse* que cumplan ciertas condiciones específicas para poder trabajar con ellas. La razón de trabajar con una muestra de la población es la simplificación del estudio y la disminución de la carga y tiempo de proceso.

La mejor muestra será aquella que, teniendo un error asumible, contenga el número mínimo de observaciones. En el caso de que se utilice un muestreo aleatorio, se debe optar por lo siguiente:

- El nivel de confianza de la muestra (usualmente el 95% o el 99%).
- El tamaño máximo de la muestra (número máximo de registros), en cuyo caso el sistema deberá informar del error cometido y la representatividad de la muestra sobre la población original.
- El error muestral que está dispuesto a cometer, en cuyo caso el sistema informará del número de observaciones que debe contener la muestra y su representatividad sobre la población original.

Para facilitar esta tarea, es necesario contar con herramientas de extracción dinámica. En el caso del muestreo, las herramientas deben tener la opción de, dado un nivel de confianza, fijar el tamaño de la muestra y obtener el error, o bien fijar el error y obtener el tamaño mínimo de la muestra que proporcione este grado de error.

Exploración

Una vez definida la población, se deberá determinar cuáles son las variables explicativas relevantes que servirán como entradas al modelo. Para ello es importante hacer una exploración de la información disponible que permita eliminar variables que no influyen y considerablemente.

El objetivo de este paso es simplificar lo más posible el problema, con el fin de optimizar la eficiencia del modelo. En este paso se pueden emplear herramientas que permitan visualizar de forma gráfica la información, utilizando las variables explicativas como dimensiones.

También se pueden emplear técnicas estadísticas que ayuden a poner de manifiesto relaciones entre variables. A este respecto resultará imprescindible una herramienta que permita la visualización y el análisis estadístico integrado.

Manipulación

Tratamiento realizado sobre los datos de forma previa al modelado, en base a la exploración realizada en el paso anterior. En este paso se define de forma clara las entradas del modelo a realizar.

Modelado

Este paso permite establecer una relación entre las variables explicativas y las variables objeto del estudio, que posibilitan inferir el valor de las mismas con un nivel de confianza determinado.

Evaluación de resultados

En la última etapa se evalúa la validez de los resultados obtenidos en la fase anterior. Para ello se utilizan pruebas de bondad de ajuste, al igual que otros métodos estadísticos que contrastan los resultados obtenidos en la muestra usada en las fases anteriores, con los resultados que se obtienen luego con otras muestras distintas.

2.2.3 Comparación entre KDD, CRISP-DM y SEMMA

Según un estudio comparativo entre estos 3 métodos, CRISP-DM y SEMMA pueden ser vistos como una implementación del método KDD descrito por Fayadd [11]. En una primera instancia, CRISP-DM parece ser más completo que SEMMA ya que abarca, casi uno a uno, las tareas del KDD; sin embargo, para desarrollar un proyecto de minería de datos, se debe tener un real entendimiento de todos los aspectos del negocio y de las metas del usuario final, como por ejemplo un analista de negocio. En este sentido ambos métodos permiten guiar el trabajo de cómo realizar un proyecto de obtención de conocimiento por medio de datos. Para el caso de esta tesis, se eligió la metodología CRISP-DM por disponer de pasos definidos y claros para el buen desarrollo de un proyecto de estas características, dónde existe un trabajo extenso en la transformación y limpieza de los datos a utilizar.

2.3 Técnicas y algoritmos para el modelado predictivo.

Para un proyecto de minería de datos se usa el modelado predictivo para analizar una base de datos y determinar algunas de las características de su contenido. Los datos deben incluir observaciones completas de tal manera de poder generar un modelo que pueda realizar predicciones lo más acertadas posibles.

Los algoritmos clasificadores tratan de encontrar, a partir de la información disponible, un modelo, patrón o estructura que explique, con un cierto grado de confiabilidad, la variable dependiente. En términos estadísticos se tiene lo siguiente [12]:

"Se define como un modelo predictivo o de regresión, aquel que permite establecer una relación matemática entre un conjunto de variables explicativas x_i *y una variable dependiente de las mismas* y_i*, para todo (i=1,..., M) dónde M es la cantidad de observaciones".*

Dichos algoritmos caen dentro de las tareas de análisis de dependencia y que pueden definirse como "*aquellos en los que una variable o conjunto de variables se identifican como variables dependientes que van a ser explicadas por otras variables conocidas denominadas variables independientes*". Como contraste, un análisis interdependiente es aquel en el que ninguna variable o grupo de variables es definido como dependiente o independientes, y dónde el análisis del conjunto se realiza simultáneamente. Los clasificadores pueden ser utilizados para:

- Determinar la relación causa-efecto de unas variables con respecto a otra, analizando.
- Analizar cuáles son las variables que más influyen en la variable a explicar para seleccionarlas como parámetros de entrada.
- Determinar agrupamiento de variables de entrada.

En la figura 6 se despliega una taxonomía de los principales tipos de algoritmos de clasificación.

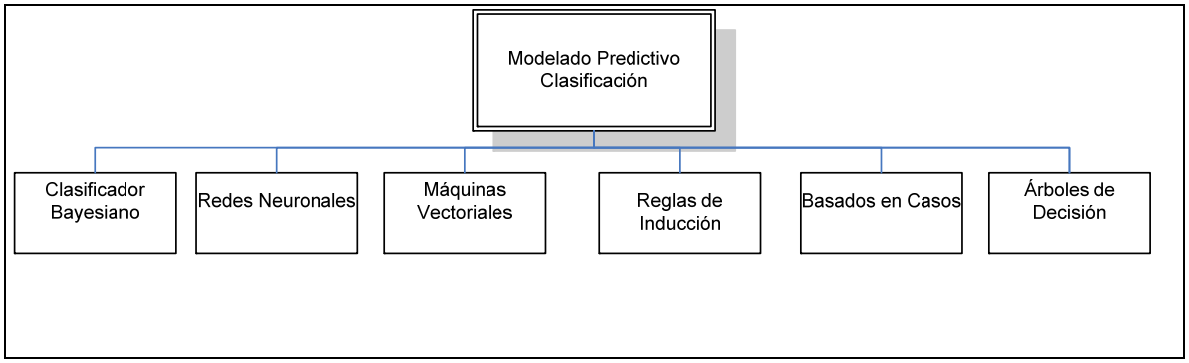


Figura 6: Taxonomía de algoritmos de clasificación

2.4 Creación de los Modelos Predictivos.

La correcta selección de variables es muy importante en los problemas de modelado, ya que una mala elección puede dificultar e incluso impedir alcanzar resultados satisfactorios.

Tras la selección de las variables que van a intervenir en el modelado, es necesario un conjunto de datos que van a ser utilizados por la herramienta de modelado para obtener los nuevos modelos. También es preciso otro conjunto de datos para comprobar la validez de esos resultados.

Para obtener dichos conjuntos es necesario un método de selección de muestras representativas. Para la validación de modelos no sólo es necesario un conjunto de datos sino también una estrategia que evite, en la medida de lo posible, los efectos derivados de modelar basándose en un conjunto finito de datos.

2.4.1 Representación de un Modelo

Un ejemplo de representación de un modelo sobre un algoritmo en particular sería la de un árbol de decisión mediante la división de nodos por un solo campo y particiones del espacio de entrada en hiperplanos paralelos a los ejes de los atributos no se podría descubrir la fórmula $x=y$ en los datos. La figura 7 explicita la representación gráfica de un modelo.

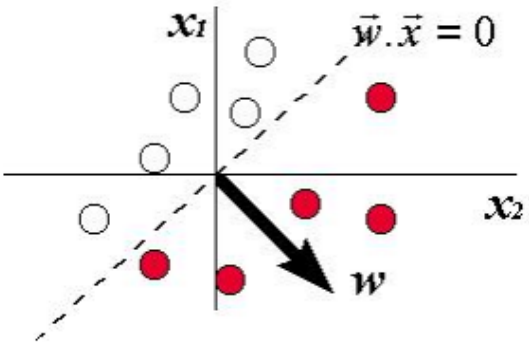


Figura 7: Representación de modelo por hiperplanos

Uno de los principales problemas que se tiene al momento de realizar una representación de un modelo predictivo es el sobreentrenamiento o el sobreajuste del mismo. Se dice que un modelo ***m1*** está sobreentrenado o sobreajustado si, dado un espacio de modelos ***M***, existe otro modelo ***m2*** que obtiene mejores resultados que ***m1*** sobre el conjunto completo donde va a ser implementado (población), obteniendo ***m1*** mejores resultados sobre el conjunto de entrenamiento. Existen técnicas para evitar el sobreajuste de los datos que serán tratadas en el punto evaluación del error.

- Representaciones muy generales del modelo son poco eficientes.
- Representaciones excesivamente adaptadas al problema implican riesgo de sobreentrenamiento.
- Las representaciones complejas mejoran el factor predictivo del modelo, pero aumentan la dificultad en la búsqueda y empeoran la interpretabilidad del modelo.

2.4.2 Métodos de búsqueda y optimización

Una vez fijada la representación del modelo (o familia de representaciones) y el criterio de evaluación del modelo, el problema puede ser reducido a un problema de optimización del tipo: "*Encontrar los parámetros/modelos de la familia seleccionada que optimicen el criterio de evaluación del modelo*".

De modo general se pueden establecer cinco criterios de clasificación para los procedimientos de optimización basados en: la solución, el grado de aleatoriedad del proceso de búsqueda, la dirección preferente de búsqueda, el número de candidatos a solución simultáneamente y en la función a optimizar. A continuación se explicará en detalle cada criterio de clasificación.

a) Basado en la solución

- Numérico: si la solución queda completamente especificada en términos de un conjunto de ***m*** parámetros o atributos.
- Combinatorial: si para especificar la solución no sólo hay que definir un conjunto de ***m*** parámetros, sino también el orden (total o parcial) con que éstos se combinan para dar dicha solución. En último término puede ocurrir que sea irrelevante la naturaleza de los atributos y sólo importe su orden; este caso se habla de problemas basados en el orden.

b) Basado en el grado de aleatoriedad del proceso de búsqueda

- Determinista o dirigido: el procedimiento de búsqueda es completamente determinista, es decir, en las mismas condiciones de partida proporciona idénticos resultados. Las técnicas deterministas, por lo común son las más eficientes, pero son muy específicas y precisan mucho conocimiento adicional sobre la función objetivo así como el uso de hipótesis suplementarias de buen comportamiento, entre

otros inconvenientes. Las técnicas clásicas de optimización, ya sean analíticas o enumerativas, son todas deterministas.

- Aleatorio o al azar: el procedimiento de búsqueda es completamente aleatorio. Habitualmente, se delimita una región de búsqueda y se toman puntos al azar dentro de ella. Después, mediante argumentos estadísticos, se puede dar una estimación de máxima verosimilitud para el valor del óptimo. Estas técnicas no requieren ninguna información adicional y se pueden aplicar a cualquier tipo de problemas, pero son poco eficientes.
- Estocástico u orientado: se combinan en proporción variable la búsqueda determinista con la búsqueda aleatoria. La componente determinista orienta la dirección de búsqueda y la aleatoria se encarga de la búsqueda local, buscando un punto intermedio entre la máxima eficiencia de las técnicas deterministas y la máxima eficacia de las aleatorias.

c) Basado en la dirección preferente de búsqueda

- En Profundidad o explotador: la búsqueda da prioridad a la explotación de las soluciones disponibles antes que a la exploración de nuevas soluciones.
- En anchura o explorador: la búsqueda da prioridad a la exploración de nuevas soluciones antes que a la explotación de las disponibles.

Muchos métodos de búsqueda no son explotadores o exploradores puros, sino que combinan ambos procedimientos; de hecho es deseable tener control sobre la relación explotación-exploración. Idealmente, esta relación debería poderse expresar como un cuociente entre dos parámetros al que se llama grado de penetración.

d) Basado en el número de candidatos a solución simultáneamente

- Simple: se mantiene un único candidato a la solución que se va actualizando sucesivamente para proporcionar, presumiblemente, soluciones cada vez más exactas del problema.
- Múltiple: se mantienen simultáneamente varios candidatos a solución con los cuales se va acotando cada vez con más precisión la región (o regiones) donde se encuentran los óptimos. Son las más apropiadas para implantaciones en paralelo. Aunque son computacionalmente más costosas, las búsquedas múltiples presentan tal cantidad de ventajas sobre las búsquedas simples que se deberían usar siempre que fuera posible, aunque no se disponga de procesadores en paralelo.

e) Basado en la función a optimizar

- Ciego: el proceso a optimizar funciona como una caja negra que ante ciertos valores de los parámetros devuelve un valor del objetivo, es decir, no se dispone de ninguna información explícita sobre la aplicación. En la bibliografía inglesa, a estos métodos de optimización se les designa como "*blackbox optimization"*, que se puede traducir como optimización de cajas negras. La ventaja de estos métodos es que proporcionan algoritmos de búsqueda de propósito general, los cuales son muy fáciles de implantar para un problema específico [13].
- Heurístico: se dispone de cierta información explícita sobre el proceso a optimizar, pudiéndose aprovechar para guiar la búsqueda. A dicha información útil para la búsqueda se le llama conocimiento específico. Las técnicas heurísticas proporcionan algoritmos dedicados de búsqueda, esto es, específicos para un problema concreto y difícilmente adaptable para cualquier otro. Tradicionalmente se ha tratado de añadir la máxima cantidad de conocimiento específico en los problemas de optimización, dado que es lo más eficaz para búsqueda. Sin embargo, en problemas reales de mediana complejidad resulta muy difícil, cuando no imposible, encontrar tal información y la que se encuentra no suele ser de muy buena calidad. A efectos prácticos, el conocimiento específico sólo sirve verdaderamente cuando es de buena calidad, y aun así se debe tener cuidado porque acentúa la tendencia a estancar la búsqueda en óptimos locales.

2.5 Redes Neuronales Artificiales.

El interés por comprender los procesos cognitivos del cerebro humano fue, en definitiva lo que permitió el surgimiento de las Redes Neuronales Artificiales (RNA), debido a que el cerebro es un sistema de procesamiento de la información extremadamente complejo, cuyo modo de funcionamiento es eminentemente paralelo y cuyo comportamiento no puede describirse por medio de modelos sencillos como lo son los lineales. Se buscó modelarlo con la esperanza de que se pudieran crear sistemas pensantes que tuvieran mejores resultados en tareas como clasificación, problemas de decisión, predicciones y sistemas de control adaptables que un sistema computacional convencional.

Las RNA están inspiradas de la estructura del cerebro, y fueron concebidas para resolver cierto tipo de problemas especialmente mal resueltos por las técnicas de programación tradicionales. Formalmente, una red neuronal es un modelo computacional con un conjunto de propiedades específicas, como son la habilidad de adaptarse o aprender, generalizar u organizar la información, todo esto basado en un procesamiento eminentemente paralelo" [14].

2.5.1 Historia de las Redes Neuronales

Los primeros teóricos que concibieron los fundamentos de la computación neuronal fueron Warren McCulloch, un neurofisiólogo, y Walter Pitts, un matemático, quienes en 1943 lanzaron una teoría acerca de la forma de trabajar de las neuronas [15]. Ellos modelaron una red neuronal simple mediante circuitos eléctricos.

En 1949 Donald Hebb fue el primero en explicar los procesos del aprendizaje (que es el elemento básico de la inteligencia humana) desde un punto de vista psicológico, desarrollando una regla de como el aprendizaje ocurría. Aún hoy, este es el fundamento de la mayoría de las funciones de aprendizaje que pueden hallarse en una red neuronal. Su idea fue que el aprendizaje ocurría cuando ciertos cambios en una neurona eran activados. También intentó encontrar semejanzas entre el aprendizaje y la actividad nerviosa. Los trabajos de Hebb formaron las bases de la Teoría de las Redes Neuronales [16].

En 1959, Widrow publica una teoría sobre la adaptación neuronal y unos modelos inspirados en ella [17]: el Adaline (*Adaptative Linear Neuron*) y el Madaline (*Multiple Adaline*). Estos modelos fueron usados en numerosas aplicaciones y permitieron usar, por primera vez, una red neuronal en un problema importante del mundo real: filtros adaptativos para eliminar ecos en las líneas telefónicas.

En 1962, Rosemblatt publica los resultados de un ambicioso proyecto de investigación, el desarrollo del Perceptrón [18], un identificador de patrones ópticos binarios y salida binaria. Las capacidades del Perceptrón se extendieron al desarrollar la regla de aprendizaje delta, que permitía emplear señales continuas de entrada y salida.

En 1969, Minsky y Papert realizan una seria crítica del Perceptrón [19], revelando serias limitaciones, como su incapacidad para representar la función XOR debido a su naturaleza lineal. Este trabajo creó serias dudas sobre las capacidades de los modelos conexionistas y provocó una caída en picada de las investigaciones.

En los años 80 se produce el renacimiento del interés por el campo gracias sobre todo al trabajo del grupo PDP (*Parallel Distributed Processing*) creado por Rumelhart, McClelland & Hinton. Como resultado de los trabajos de este grupo salieron las publicaciones con más influencia desde la crítica de Minsky y Papert. Destaca el capítulo dedicado al algoritmo de retropropagación [20], que soluciona los problemas planteados por Minsky y Papert y extiende enormemente el campo de aplicación de los modelos de computación conexionistas. Hopfield elabora un modelo de red consistente en unidades de proceso interconectadas que alcanzan mínimos energéticos, aplicando los principios de estabilidad desarrollados por Grossberg. El modelo de Hopfield resultó muy ilustrativo sobre los mecanismos de almacenamiento y recuperación de la memoria. Su entusiasmo y claridad de presentación dieron un nuevo impulso al campo y provocaron el incremento de las investigaciones.

En 1987 se desarrolla la IEEE International Conference on Neural Networks con 1700 participantes en San Diego. Se crea la International Neural Networks Society (INNS), en 1988 se publica la revista Neural

Networks por el INNS, en 1989 se publica la revista Neural Computation y en 1990 se publica la revista Transactions on Neural Networks por el IEEE.

2.5.2 Modelo de Redes Neuronales

Los elementos básicos que posee un modelo de red neuronal son los siguientes:

- Un conjunto de conexiones, cada una de ellas caracterizada por su entrada X_i, ésta se habrá convertido en una señal $X_i * W_{ji}$, donde W_{ji} es el peso o fuerza de la conexión con la entrada i -ésima de la neurona j. De acuerdo con el signo del peso W_{ji} se tienen conexiones excitadoras cuando es positivo, y conexiones inhibidoras cuando es negativo.
- Una función de propagación correspondiente a $\sum X_i * W_{ji}$, que produce la suma ponderada de las entradas de acuerdo a los correspondientes pesos de las conexiones.
- Una función de activación $\phi(*)$, que representa el proceso algorítmico que transforma el resultado de la función de propagación en la salida real de la neurona. En la mayoría de los casos la función de activación y la de propagación son idénticas; la función de activación determina el nivel de activación de la neurona en términos de la actividad existente en sus entradas. Hay una infinidad de funciones para ser utilizadas como función de activación en una red neuronal artificial, pero se pueden distinguir tres grandes clases: tipo escalón, lineal y sigmoide. Es más frecuente utilizar funciones sigmoidales (1), puesto que éstas y sus derivadas son continuas.

$$(1)\ Y_k = F_k(S_k) = \frac{1}{1+e^{-S_k}}$$

- En la función de activación el valor de la salida puede ser comparada con algún valor umbral para determinar la salida de la neurona. Si la suma es mayor que el valor umbral, neurona generará una señal. Si la suma es menor que el valor umbral, ninguna señal será generada.

De acuerdo con este modelo se puede describir el comportamiento de la neurona (o nodo) con las expresiones utilizadas anteriormente:

- X_i :entrada a la neurona i
- $X_i * W_{ji}$: entrada neta o función de propagación de la neurona i-ésima
- W_{ji} : peso de la conexión de la neurona i-ésima a la neurona j-ésima

- $\sum X_i * W_{ji}$: Función de activación de la neurona i-ésima

2.5.3 Arquitectura de Redes Neuronales Artificiales

Este concepto se refiere básicamente a la manera en que se interconectan los distintos nodos que forman la red. Normalmente los nodos se organizan como una secuencia de capas con un determinado patrón de interconexión entre las diferentes neuronas que las forman, y con un patrón de conexión entre las neuronas de las distintas capas. Uno de los rasgos que puede ayudar a definir una capa es el hecho de que todas las neuronas que la forman usan la misma función de propagación. En la figura 9 se muestra un ejemplo de arquitectura.

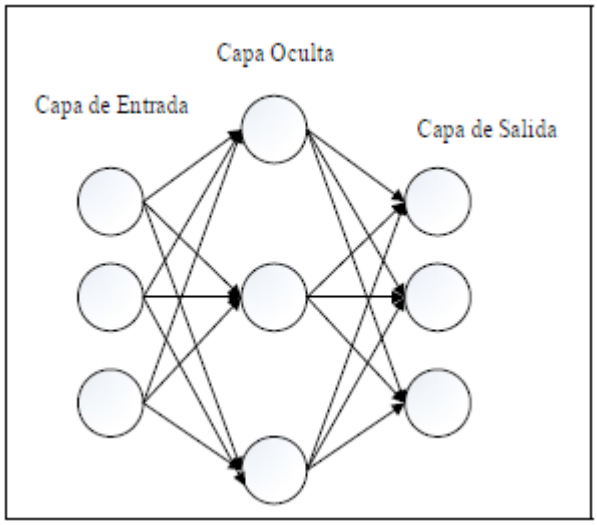


Figura 8: Red Neuronal Artificial

A continuación se explicarán los distintos tipos de arquitecturas para redes neuronales.

a) Arquitectura según distinción de las capas:

- **Capa de entrada**: está compuesta por neuronas que reciben entradas procedentes del entorno. sus pesos se mantienen constantes, y su misión simplemente es la de distribuir dicha entrada al resto de los elementos de proceso que constituyen la red.
- **Capa oculta**: es aquella que no tiene conexión directa con el contorno, es decir, que no es de entrada ni de salida. Crean una representación interna de los patrones de entrada. Puede haber más de una capa oculta.
- **Capa de salida**: es aquella cuyas neuronas proporcionan la respuesta de la red neuronal.

b) Arquitectura según flujo de señales:

- **Redes *Feedforward***: la información circula en un único sentido desde las neuronas de entrada a las de salida. Representan sistemas lineales dado que no existen nodos de realimentación tal como nos muestra la figura 10.

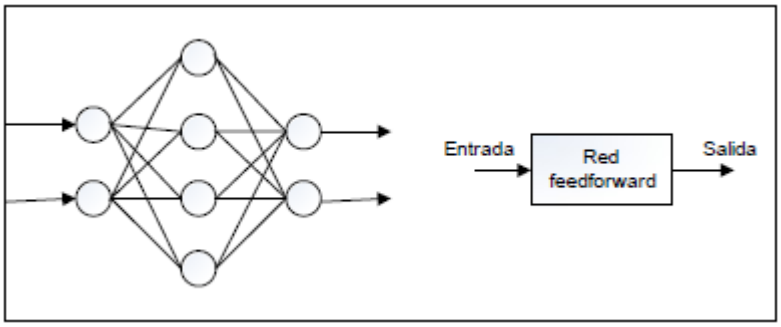


Figura 9: Red Feedforward

- **Redes *Feedback* o Recurrentes**: como muestra la figura 11, la información puede circular entre las capas en cualquier sentido. Representan sistemas no lineales mediante unidades de realimentación.

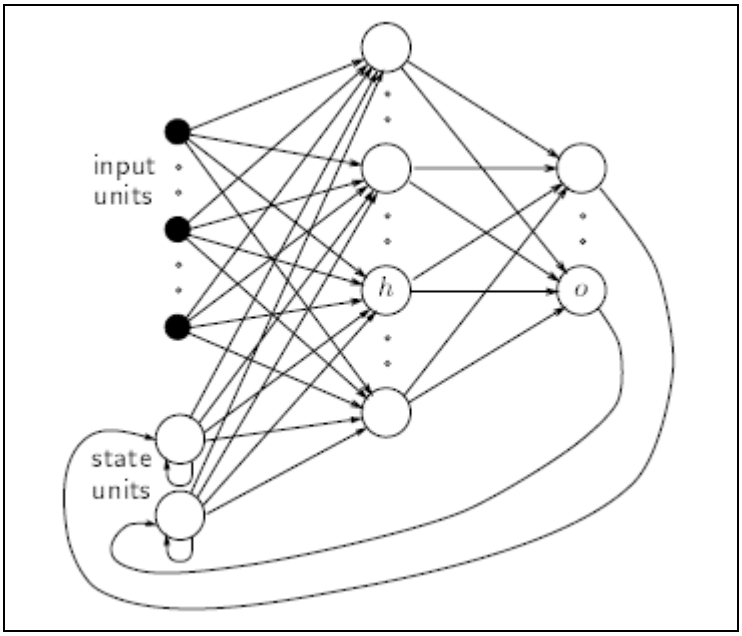


Figura 10: Red Recurrente

2.5.4 Aprendizaje de una Red Neuronal Artificial

Se denomina aprendizaje al modelo del entorno en el que la red neuronal trabaja. Una de las principales ideas sobre las que se basan las redes neuronales artificiales es la de responder a los estímulos del entorno mediante un proceso de aprendizaje por el cual va adaptando los pesos de las conexiones de sus nodos, de tal forma que memoriza los ejemplos de entrenamiento que se le presentan; la forma de aprendizaje indica cómo el entorno influye en ese proceso de aprendizaje. Dentro de los tipos de aprendizaje se tienen:

- **Supervisado**: se presentan los conocimientos en forma de pares de [entrada, salida deseada]. La salida real de la red es comparada con el valor deseado de salida, Los pesos, que normalmente han sido establecidos de manera aleatoria en un principio, son ajustados por la red de manera que en la siguiente iteración, producirá un resultado más cercano entre el valor esperado y la salida real, hasta ajustar la salida a un margen de error aceptable.

- **No Supervisado**: durante este proceso de aprendizaje a la red no se la presenta la salida deseada. Sus principales utilidades son, entre otras, descubrir las regularidades presentes en los datos, extraer rasgos o patrones.

- **Reforzado**: Se sitúa entre los dos anteriores, de forma que, por una parte se emplea la información del error cometido, pero se sigue sin poseer la salida deseada. Este aprendizaje descansa en la idea de premio-castigo, donde se refuerza toda aquella acción que permita una mejora del modelo.

- **Híbrido**: coexisten en la red los dos tipos básicos de aprendizaje, el supervisado y el no supervisado, normalmente en distintas capas de neuronas.

2.5.5 Algoritmos de Aprendizaje

En este contexto, el aprendizaje puede ser visto como el proceso de ajuste de los parámetros libres de la red neuronal artificial. Por lo tanto, partiendo de un conjunto de pesos sinápticos aleatorios, el proceso de aprendizaje busca optimizar ese conjunto de pesos que permitan que la red pueda generalizar de forma adecuada. El proceso de aprendizaje consiste en encontrar el conjunto de pesos sinápticos que minimizan (o maximizan) la función. El método de optimización proporciona una regla de actualización de los pesos que en función de los patrones de entrada y modifica de manera iterativa los pesos hasta alcanzar el punto óptimo de la red neuronal.

Método por Descenso del Gradiente

Los principales algoritmos de aprendizaje de una red neuronal se basan en el cálculo del gradiente de la función de error, esto es, de la derivada de la función de error con respecto a los distintos parámetros ajustables de la red (en general, los pesos de las conexiones). Se trata de intentar encontrar el mínimo de la función de error mediante la búsqueda de un punto donde el gradiente se anule.

Una de las variantes más utilizadas que se basan en la derivada es el descenso del gradiente [21]. En este método se realizan sucesivos ajustes a los pesos se hacen de forma individual para cada W_i en sentido opuesto al vector de gradiente $\partial E[n]/\partial W_i[n]$:

$$W_i[n+1] = W_i - \alpha * \partial E[n]/\partial W_i[n]$$

donde α es un parámetro conocido como tasa de aprendizaje, que ha de tomar un valor convenientemente pequeño. Al pasar de la iteración *n* a la *n + 1*, el algoritmo aplica la corrección:

$$\Delta W_i[n] = W_i[n+1] - W_i[n] = -\alpha * \partial E[n]/\partial W_i[n]$$

Para valores positivos muy pequeños de la tasa de aprendizaje y funciones de error globales, la formulación del algoritmo de descenso por el gradiente permite que la función de error decrezca en cada iteración. La tasa de aprendizaje α tiene, por tanto, una enorme influencia en la convergencia del método de descenso por el gradiente. Si α es pequeña, el proceso de aprendizaje se desarrolla suavemente, pero la convergencia del sistema a una solución estable puede llevar un tiempo excesivo. Si α es grande, la velocidad de aprendizaje aumenta, pero existe el riesgo de que el proceso de aprendizaje diverja y el sistema se vuelva inestable. Es habitual añadir un término de momento que en ocasiones puede acelerar el aprendizaje y reducir el riesgo de que el algoritmo se vuelva inestable. La nueva ecuación de actualización del parámetro ajustable W_i tiene la forma:

$$\Delta W_i[n] = W_i[n+1] - W_i[n] = -\alpha * \partial E[n]/\partial W_i[n] - \Delta\gamma * W_i[n+1]$$

donde α es la tasa de aprendizaje y γ es la constante de momento.

Existen otros algoritmos de aprendizaje más sofisticados (por ejemplo, aquellos que consideran la información suministrada por las derivadas de segundo orden), que en general, proporcionan mejores resultados que el descenso por el gradiente, a veces simplemente con una leve modificación, pero estos no serán tratados en este capítulo, dado que no son relevantes como objeto de estudio de la tesis.

Algoritmo de Retropropagación o *Backpropagation*

La regla de aprendizaje del Perceptrón de Rosenblatt [22] y el algoritmo de los mínimos cuadrados de Widrow y Hoff [21] fueron diseñados para entrenar redes de una sola capa. El primer algoritmo de entrenamiento para redes multicapa fue desarrollado por Paul Werbos en 1974 [23] en un contexto general

para cualquier tipo de redes, siendo las redes neuronales una aplicación especial, razón por la cual el algoritmo no fue aceptado inicialmente dentro de la comunidad de desarrolladores de redes neuronales. El algoritmo se popularizó cuando fue incluido en el libro *Parallel Distributed Processing Group* [19] por los psicólogos David Rumelhart y James McClelland, lo que trajo consigo un auge en las investigaciones con redes neuronales, siendo la retropropagación una de las redes más ampliamente empleadas, aun actualmente.

La retropropagación es un tipo de red de aprendizaje supervisado, que emplea un ciclo propagación – adaptación de dos fases. Una vez que se ha aplicado un patrón a la entrada de la red como estímulo, éste se propaga desde la primera capa a través de las capas superiores de la red, hasta generar una salida. La señal de salida se compara con la salida deseada y se calcula una señal de error para cada una de las salidas.

Las salidas de error se propagan hacia atrás, partiendo de la capa de salida, hacia todas las neuronas de la capa oculta que contribuyen directamente a la salida. Sin embargo las neuronas de la capa oculta sólo reciben una fracción de la señal total del error, basándose aproximadamente en la contribución relativa que haya aportado cada neurona a la salida original. Este proceso se repite, capa por capa, hasta que todas las neuronas de la red hayan recibido una señal de error que describa su contribución relativa al error total. Basándose en la señal de error percibida, se actualizan los pesos de conexión de cada neurona, para hacer que la red converja hacia un estado que permita clasificar correctamente todos los patrones de entrenamiento.

La importancia de este proceso consiste en que, a medida que se entrena la red, las neuronas de las capas intermedias se organizan a sí mismas de tal modo que las distintas neuronas aprenden a reconocer diferentes características del espacio total de entrada. Después del entrenamiento, cuando se les presente un patrón arbitrario de entrada que contenga ruido o que esté incompleto, las neuronas de la capa oculta de la red responderán con una salida activa si la nueva entrada contiene un patrón que se asemeje a aquella característica que las neuronas individuales hayan aprendido a reconocer durante su entrenamiento. Y a la inversa, las unidades de las capas ocultas tienen una tendencia a inhibir su salida si el patrón de entrada no contiene la característica para reconocer, para la cual han sido entrenadas.

Durante el proceso de entrenamiento, la red retropropagación tiende a desarrollar relaciones internas entre neuronas con el fin de organizar los datos de entrenamiento en clases. Esta tendencia se puede extrapolar, para llegar a la hipótesis consistente en que todas las unidades de la capa oculta de la red son asociadas de alguna manera a características específicas del patrón de entrada como consecuencia del entrenamiento. Lo que sea o no exactamente la asociación puede no resultar evidente para el observador humano, lo importante es que la red ha encontrado una representación interna que le permite generar las salidas deseadas cuando se le dan las entradas, en el proceso de entrenamiento. Esta misma representación interna se puede aplicar a entradas que la red no haya visto antes, y la red clasificará estas entradas según las características que compartan con los ejemplos de entrenamiento.

2.5.6 Redes Neuronales Artificiales Evolutivas

Durante esta última década, se ha observado un interés creciente por el desarrollo de métodos para solucionar problemas complejos de optimización uniobjetivo y multiobjetivos, incluyendo su aplicación a problemas combinatorios no lineales [24]. Los algoritmos metaheurísticos constituyen uno de los campos de investigación más activos en optimización; su estudio ha ido progresando, surgiendo continuamente nuevas ideas en un intento de alcanzar una mayor eficiencia en el proceso de solución.

Actualmente los métodos de optimización heurísticos basados en poblaciones han despertado el interés de la comunidad científica debido a su capacidad para explorar espacios de soluciones multimodales y multidimensionales, de manera rápida y eficiente.

Una alternativa a los métodos de optimización planteados anteriormente para el aprendizaje de redes neuronales artificiales corresponde a la utilización de estos algoritmos metaheurísticos, específicamente, los basados en poblaciones del tipo evolutivos. Estos algoritmos son un tipo de búsqueda estocástica que permite encontrar la solución en espacios complejos, los cuales se han utilizado con éxito en el campo de las redes neuronales para encontrar la estructura adecuada de la red, y para calcular el valor de los pesos de las conexiones evitando quedar atrapado en mínimos locales, generalmente derivados de un sobreentrenamiento.

En el proceso de evolución son esenciales los operadores de selección para introducir, por una parte, presión selectiva, y por otra diversidad, de forma tal que el algoritmo de búsqueda obtenga la mejor red logrando un compromiso entre su capacidad de explotar la localización de las mejores soluciones y de explorar otras localizaciones del espacio, con el fin de obviar el problema de la obtención de óptimos locales. En la figura 12 se presenta una taxonomía de algoritmos evolutivos

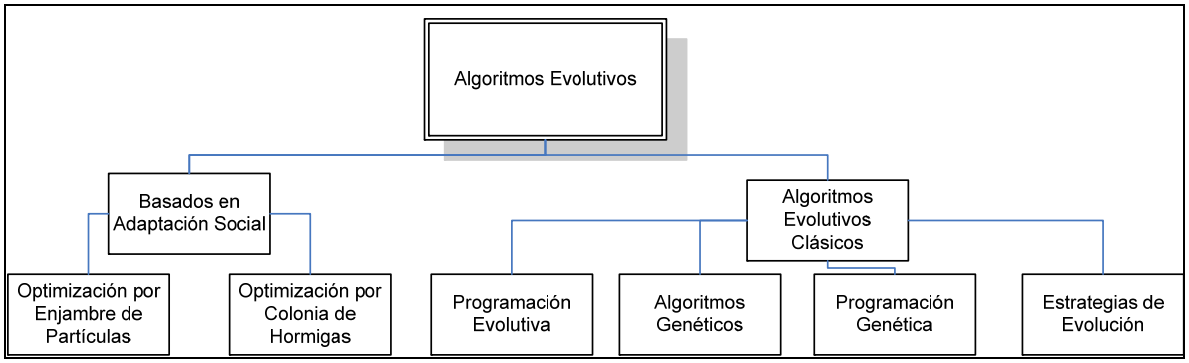


Figura 11: Taxonomía de Algoritmos Evolutivos

Una red neuronal evolutiva correspondería a tener un método híbrido, en el cual se posee un algoritmo evolutivo como método de optimización para el ajuste de parámetros .Según lo expuesto en [25], se ha demostrado que la optimización por enjambre de partículas tiene mejor rendimiento en comparación a los algoritmos genéticos, por lo cual el estudio a realizar en esta tesis se basará en un esquema de enjambre de partículas, dejando de lado las otras técnicas de optimización.

Optimización por Enjambre de Partículas

La optimización por enjambre de partículas (PSO) pertenece a las técnicas estocásticas de cálculo evolutivo, considerada unas de las más representativas de la rama de la inteligencia de enjambre (*Swarm Intelligence*), al igual que la técnica de optimización por colonia de hormigas (ACO). Ambos tratan de imitar los comportamientos sociales de un colectivo a partir de la interacción de los individuos entre sí y con el entorno [26].

Los inicios del PSO como método estocástico de optimización global se remontan a los estudios realizados por Kennedy y Eberhart [27], quienes intentaron simular gráficamente el movimiento sincronizado e impredecible de grupos tales como los bancos de peces o las bandadas de aves, motivados por la capacidad que poseen estos grupos para separarse, reagruparse o encontrar alimento. Paralelamente con trabajos previos en el ámbito de la biología y de la sociología, concluyen que el comportamiento, inteligencia y movimiento de estas agrupaciones (entre las cuales podríamos incluir a los seres humanos con un cierto grado de abstracción) está relacionado directamente con la capacidad para compartir información y para aprovechar la experiencia acumulada por cada uno de sus congéneres.

En la tecnología utilizada en [28], Kennedy y Eberhart introducen el término general de partícula o agente para representar a los peces, pájaros, abejas, hormigas o cualquier otro tipo de individuos que exhiban un comportamiento social como grupo, en forma de una colección de agentes que interactúan entre sí. De acuerdo con los fundamentos teóricos del método, el movimiento de cada una de estas partículas hacia un objetivo común en dos dimensiones está condicionado por dos factores básicos: la memoria autobiográfica de la partícula o nostalgia, y la influencia social de todo el enjambre. A nivel computacional, como método de optimización, esta filosofía puede extenderse a un espacio D-dimensional de acuerdo con el problema bajo análisis. La posición instantánea de cada una de las partículas de la población en el espacio representa una solución potencial, siendo del número de incógnitas del problema original. Básicamente, el proceso evolutivo se reduce a mover cada partícula dentro del espacio de soluciones con una velocidad que variará de acuerdo a su velocidad actual, a la memoria de la partícula y a la información global que comparte el resto del enjambre, utilizando una función de bondad/ajuste (*fitness*) para cuantificar la calidad de cada partícula en función de la posición que esta ocupe.

Más allá de la propia naturaleza del método, los esquemas existentes para su implementación son muy diversos. Dependiendo de cómo se actualicen las posiciones de las partículas surgen las versiones síncrona y asíncrona del algoritmo. Adicionalmente dependiendo de cómo se haga fluir la experiencia acumulada por el enjambre sobre el movimiento de cada una de las partículas que lo integran, se puede distinguir entre PSO local y global. La combinación de estas variantes (síncrona vs., asíncrona, local vs. global) resume los esquemas desarrollados e investigados comúnmente, aunque en la literatura existen otras variantes a los esquemas convencionales, en la mayoría de los casos fruto de modificaciones introducidas por los propios autores para mejorar el rendimiento del algoritmo original en aplicaciones concretas [29].

Desde el punto de vista de su algoritmo, la ventaja principal del PSO es su rápida convergencia, en comparación con otros algoritmos de optimización global como los genéticos (GA), colonia de hormigas, enfriamiento simulado y otros de optimización global. Para aplicar PSO con éxito, uno de los problemas claves se encuentra en cómo representar la solución del problema en las partículas del PSO. Esta representación de la solución del problema afectará directamente la facilidad del uso del algoritmo, su funcionamiento y comprensión [30]. En esta sección se realiza un análisis en profundidad de los principios del PSO como método de optimización, comenzando con los fundamentos y analizando los esquemas que se han implementado, con el objetivo de estudiar la influencia de los principales parámetros del algoritmo sobre la convergencia de la optimización, buscando un compromiso entre precisión y costo computacional.

Algoritmo PSO

Una partícula está definida como un vector, el cual está compuesto por la velocidad que posee la partícula, su posición, y su memoria [31]. El algoritmo PSO se inicializa con partículas (soluciones) aleatorias, y a medida que se realicen las iteraciones, buscará soluciones más optimas actualizando cada partícula. Una partícula se considera como un punto dentro del espacio de búsqueda D-dimensional, donde cada dimensión representa una de las incógnitas del problema. El enjambre es representado por $X = (x_{i1}, x_{i2}, x_{i3}, ..., x_{id})$, donde i = 1,2,…,n corresponde la i-ésima partícula, y d = 1,2,…,D corresponde sus dimensiones.

En cada generación (iteración), cada una de las partículas es actualizada de acuerdo a dos valores importantes. El primero corresponde a la mejor posición encontrada por una partícula, conocido como "mejor personal" (p_{best}), el cual es representado como $P_i = (p_{i1}, p_{i2}, p_{i3}, ..., p_{id})$ para la i-ésima partícula. El segundo valor corresponde a la mejor posición que ha encontrado el enjambre, conocido como el "mejor global" (p_{gbest}). Ambos valores se actualizan en cada iteración, y se utilizan para ajustar la velocidad con que se mueve una determinada partícula en cada una de las dimensiones. A su vez, la velocidad es utilizada para determinar la nueva posición que alcanzará una partícula en la siguiente iteración del algoritmo. La influencia que tiene la mejor posición personal sobre la velocidad de una partícula se conoce como el factor o componente de cognición, y la influencia de la mejor posición del enjambre se conoce como el componente social [31].

En [30] Eberhart incorporó un factor de inercia w en la ecuación que determina la velocidad (2) de la partícula para balancear la búsqueda global y local. La nueva posición de una partícula está definida en (3).

$$(2)\ v_{id}(t+1) = w * v_{id}(t) + c_1 r_1 (p_{id}(t) - x_{id}(t)) + c_2 r_2 (p_{gd}(t) - x_{id}(t))$$

$$(3)\ x_{id}(t+1) = x_{id}(t) + v_{id}(t+1)$$

donde r_1 y r_2 corresponden a valores aleatorios independientes en el intervalo [0,1], y los parámetros y

c_2 corresponden a las variables que controlan la influencia de las componentes cognitiva y social. Para prevenir que las velocidades de las partículas incrementen infinitamente, se incorpora un parámetro $V_{\max}$ que delimita el rango de velocidad $[-V_{\max}, V_{\max}]$ que puede tomar una partícula:

$$(4)\ if(v_{ij} > V_{\max})\ or\ (v_{ij} < -V_{\max})$$

$$v_{ij} = sign(v_{ij}) * V_{\max}$$

El pseudocódigo de la implementación de PSO es como sigue:

Inicialización de enjambre con posiciones y velocidades aleatorias
Repetir hasta iteración máxima
Para cada partícula
Evaluar fitness de partícula (función costo).
Si la actual fitness es mejor que P_{id} *entonces* P_{id} *el actual valor.*
Si P_{id} *es mejor que la mejor global asignar* P_{gd} *a actual valor fitness de la partícula.*
Cambiar la velocidad y posición de las partículas.
Fin
Fin

El algoritmo de PSO posee un reducido número de parámetros; el mal ajuste de éstos puede provocar que el PSO converja a un óptimo en muchas iteraciones. A menudo, el PSO podría encontrar un mal óptimo en pocas iteraciones (fenómeno conocido como convergencia prematura), o a un mal óptimo en muchas iteraciones. A continuación se estudiará cada parámetro del algoritmo PSO y la influencia que tiene sobre este.

Los autores de [30] proponen un modelo quántico de PSO, en donde el concepto de velocidad ya no existe, sino que el enjambre se moverá por una función de movimiento dada en (5). Esta función de movimiento compara la posición actual de una partícula con el promedio de las mejores posiciones del enjambre, a diferencia del PSO tradicional que compara la posición de una partícula con la mejor posición del enjambre.

$$(5)\ x(t+1) = p \pm \left| m_{best} - x(t) \right| \bullet \ln(1/u)$$

donde m_{best} representa el vector con la media de todas las mejores posiciones encontradas por las partículas a nivel de dimensión. Calcular la media de las mejores posiciones en cada iteración del algoritmo tiene un alto costo computacional, por lo que se debe considerar un número reducido de partículas:

$$(6)\ m_{best} = \frac{1}{M}\sum_{i=1}^{M} p_i = \left(\frac{1}{M}\sum_{i=1}^{M} p_{i1}, \frac{1}{M}\sum_{i=1}^{M} p_{i2}, \ldots, \frac{1}{M}\sum_{i=1}^{M} p_{id} \right)$$

El parámetro *p* en (5) depende de la mejor posición de la partícula p_{id} y de la mejor posición global según:

$$(7)\ p = \varphi \cdot p_{id} + (1-\varphi)\cdot p_{gd}$$

donde φ y u son valores aleatorios entre [0,1], β es un coeficiente de expansión-contracción que toma un valor entre un rango de [0,1.7]. Esta propuesta fue probada en tres implementaciones (Rosenbrok, Rastring, Griewank) [30], obteniendo resultados superiores a la versión tradicional de PSO en términos de velocidad de convergencia.

2.6 Extracción de Reglas desde una Red Neuronal.

Dentro de los principales algoritmos desarrollados para la extracción de reglas de una red neuronal entregada se tienen los siguientes [38]:

- Algoritmo "*M of N* ": Realiza búsqueda de reglas de la forma: SI (*M* de los siguiente *N* antecedentes son verdaderos) ENTONCES. Los pasos del algoritmo son:
    a) Para cada nodo, hacer un agrupamiento de los pesos de entrada en grupos similares.
    b) Encontrar el promedio de cada grupo a partir de los pesos que forman el grupo.
    c) Remover grupos que tengan poco efecto en los nodos de salida.
    d) Depurar pesos y actualizar el bias de la red.
    e) Formar reglas de cada nodo
    f) Simplificar las reglas obtenidas a la forma "*M of N*"
- Algoritmo VIA: Se basa en el manejo de intervalos de validación, donde cada uno de éstos limita la activación de patrones en la red. Inicialmente, el usuario puede asignar intervalos arbitrarios a todos (o a un subconjunto de) los nodos y el algoritmo refina estos intervalos. Este proceso se realiza en forma iterativa, detectando y excluyendo los valores de activación que son lógicamente incoherentes con los pesos y los sesgos de la red. Este mecanismo, garantiza que cada patrón de activación es consistente con el conjunto inicial de intervalos. Existen dos posibles salidas del proceso de análisis "VIA":
    a) La rutina converge: los intervalos resultantes constituyen un sub-espacio que incluye todos los patrones de activación en concordancia con la periodicidad inicial.
    b) Contradicción: si se genera un intervalo vacío significa que el límite inferior de un intervalo excede a su límite superior, por ende no existirá patrón de activación alguno que pueda

satisfacer las limitaciones impuestas por el intervalo inicial de validez. Por consiguiente, los intervalos iniciales son incompatibles con los pesos y los sesgos de la red.

- Algoritmo "SUBSET": Es una de las primeras aproximaciones a un algoritmo descomposicional. Este algoritmo define una búsqueda basada en la extracción de reglas de la forma SI - ENTONCES desde una red *Multi-Layer Perceptron* (MLP) con unidades binarias de salida; adicionalmente la red tiene únicamente valores binarios de entrada. El algoritmo encuentra las combinaciones de pesos positivos conectados a un nodo que se active. Estos conjuntos de conexiones se combinan con pesos negativos para formar reglas que activen el nodo. El algoritmo realiza una búsqueda exhaustiva sobre todas las conexiones entrantes de un nodo y, como tal, es necesario restringir el espacio de búsqueda.
- Algoritmo "RULENET": Es un algoritmo que emplea una técnica descomposicional para generar reglas de la forma SI – ENTONCES – SINO. Fue propuesto por McMillan, Mozer y Smolensky y se basa en los pasos del método científico, el cual induce una hipótesis e iterativamente la refina hasta explicar la observación. El algoritmo se basa en los siguientes pasos:
    a) Entrenamiento de la red neuronal.
    b) Extracción simbólica de reglas (usando las conexiones más fuertes de la red)
    c) Inserción de reglas extraídas en la red (prueba de hipótesis)

El proceso termina cuando la base de las reglas extraías caracteriza el dominio del problema. La red a utilizar tiene 3 capas definidas: capa de entrada, capa nodo de condición y capa de salida.

El vector de pesos que conecta los nodos de entrada a los nodos de condición es usado para detectar la condicionalidad de la regla para ser aprendida. Después del entrenamiento, cada nodo de condición representa una regla. La matriz de pesos que conecta los nodos de condición a los nodos de salida es un conjunto que asegura que hay un único mapeo entre las entradas y las salidas. La extracción de reglas es alcanzada por la descomposición del vector de pesos para los componentes de condición de la regla y la descomposición de la matriz para los componentes de acción de la regla.

- Algoritmo "REANN": Consiste en un algoritmo basado en una red MLP que consta de las siguientes fases:
    a) El número de nodos ocultos de la red se determina automáticamente de una manera constructiva por adición de nodos, uno tras otro basado en el desempeño de de la red sobre los datos de entrenamiento.
    b) La red neuronal es podada de tal manera que las conexiones pertinentes y nodos de entrada son eliminados, mientras que su exactitud de predicción se mantiene.
    c) Los valores de activación de los nodos ocultos son “discretizados” mediante una agrupación heurística eficiente.
    d) Finalmente, se extraen las reglas mediante el examen de los valores de activación discretos de los nodos ocultos utilizando un algoritmo de extracción de reglas.

2.7 Criterios de Validación de un Modelo.

Para cada algoritmo de minería de datos es necesario establecer criterios de evaluación que proporcionen medidas de la diferencia entre el resultado del modelo y el proceso real. Para ello se distinguirá entre el error real y el error aparente:

- Si se supone que existe una población de la cual han sido extraídos los datos de entrenamiento de forma aleatoria, se define el error real o poblacional como el error que se comete al ser evaluado el modelo sobre los datos de la población.
- Se define error aparente o error muestral como el error del modelo sobre la muestra de casos utilizados en el diseño o construcción del mismo.

El objetivo del modelo es minimizar el error real cometido que al ser desconocido debe de ser estimado. Todos los métodos de estimación se basan en la suposición de la existencia de un conjunto de entrenamiento, compuesto por varios patrones. Este conjunto se utilizará para construir y evaluar el modelo, por lo que la existencia de un buen conjunto de aprendizaje resulta fundamental.

Para la estimación de los errores se utilizarán diferentes procedimientos en función de los objetivos del método. En particular, si el objetivo del método es realizar una regresión (variable dependiente continua) es necesario el uso de una medida de distancia para realizar la evaluación del error: **chi-cuadrado**, **correlación,** etc. Una vez seleccionada la medida de distancia a utilizar, se puede realizar una estimación del error del modelo mediante el cálculo de la distancia entre la salida predicha por el modelo y la salida real.

Evaluación de modelos de clasificación.

En el caso de algoritmos de clasificación, los estimadores de error calcular la proporción de prototipos incorrectamente etiquetados por el clasificador. Por lo tanto para analizar los errores generados a partir de un modelo de un modelo de clasificación se emplean las siguientes herramientas:

- Matriz de Confusión: es una herramienta de visualización que se emplea en aprendizaje supervisado. Cada columna de la matriz representa el número de predicciones de cada clase, mientras que cada fina representa a las instancias en la clase real, como se muestra en la tabla 2. Una ventaja de su utilización es que permiten ver con facilidad si el sistema está realizando una mala clasificación entre dos clases.

Matriz de Confusión		
Clase real	Clase a predecir	
	SI	NO
SI	TP: Verdaderos Positivos	FN: Falsos Negativos
NO	FP: Falsos Positivos	TN: Verdaderos Negativos

Tabla 2: Ejemplo de matriz de confusión

- **Tasa de Error**: Corresponde a número de errores dividido por el número de muestras.

$$Error = \frac{|FN| + |FP|}{N}, \text{ donde } N = |FN| + |FP| + |TN| + |TP|$$

- **Exactitud (Accuracy)**: Es la proporción del número total de predicciones que son correctas.

$$Accuracy = \frac{|TP| + |TN|}{|FP| + |FN| + |TP| + |TN|}$$

- **Recall**: Es la proporción de casos positivos que fueron identificados correctamente. En clasificación binaria es llamado sensibilidad.

$$\text{Re}call = \frac{|TP|}{|TP| + |FN|}$$

- **Precisión**: Es la predicción de casos positivos que fueron clasificados correctamente. Se define como:

$$\text{Pr}ecisión = \frac{|TP|}{|TP| + |FP|}$$

- **Tasa de Falsos Negativos**: Es la proporción de casos positivos que son incorrectamente clasificados como negativos.

$$FNRate = \frac{|FN|}{|TP| + |FN|}$$

- **Tasa de Falsos Positivos**: Es la proporción de casos negativos que son incorrectamente clasificados como positivos.

$$FPRate = \frac{|FP|}{|TN| + |FP|}$$

- **Curvas ROC (*Receiver operating characteristic)***: Es una representación gráfica de la sensibilidad de los verdaderos positivos versus los falsos positivos. En la figura12 se puede apreciar que el punto (0,1) corresponde a una clasificación perfecta. La línea diagonal que divide el espacio de la ROC en áreas de la clasificación buena o mala. Los puntos por encima de la línea diagonal indican buenos resultados de clasificación, mientras que los puntos por debajo de la línea indican resultados erróneos.

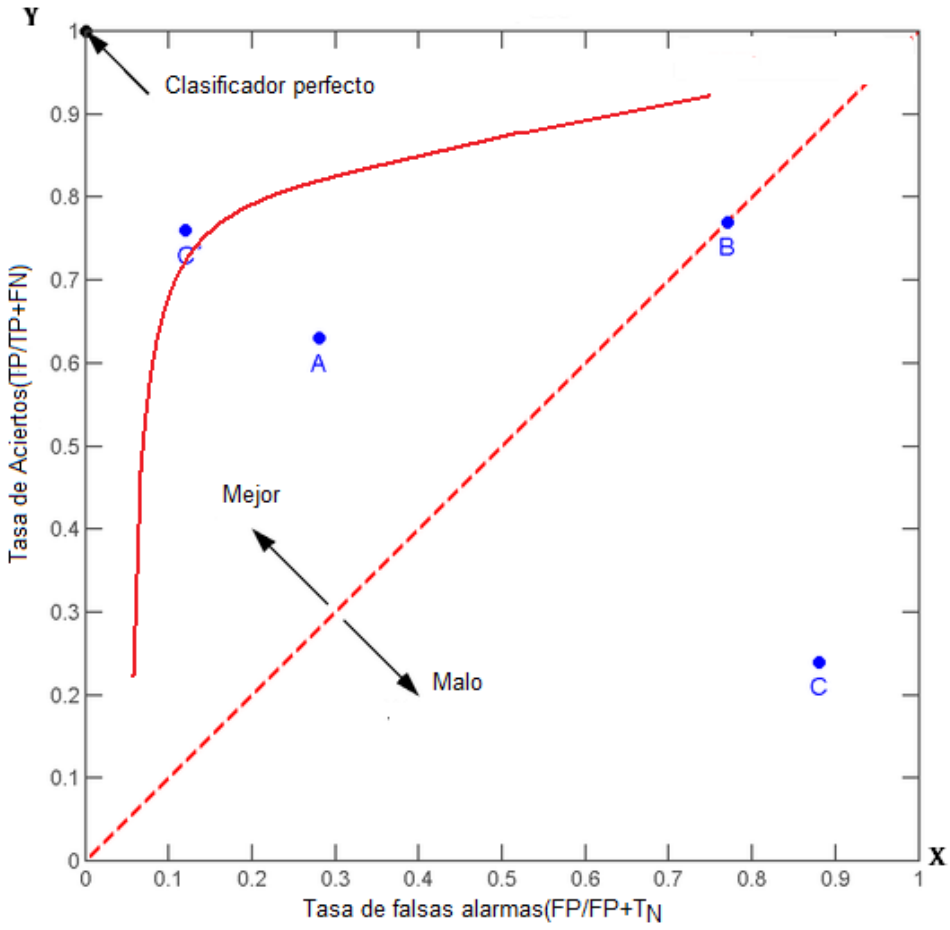


Figura 12: Representación de curva ROC

CAPÍTULO 3

Diseño de la Solución

En este capítulo se detallará el diseño de la investigación utilizando el proceso CRISP-DM descrito en el estado del arte, para obtener una solución al problema planteado para la fuga de clientes. La primera fase (comprensión del negocio) se encuentra abordada en el capítulo 1, por lo cual se definirá el diseño de las siguientes etapas propuestas.

3.1 Selección y descripción de variables

El primer paso consiste en la obtención e integración de los datos necesarios de todas las fuentes de que se disponen. Esta tarea consiste en la realización de una recopilación de los datos para la minería de datos, utilizando para ello la lista de recursos disponibles elaborada en la fase de comprensión del negocio. Dicha lista se incorpora un *data warehouse* corporativo en SQL Server, un sistema AS400 Maestro de Póliza y la base de datos de clientes en SQL Server.

Por motivos de confidencialidad los datos personales de clientes de la compañía no son los originales, sin embargo, se asegura que existe absoluta concordancia con el conjunto de datos entregado y representan completamente la realidad.

La selección del conjunto inicial de variables fue realizada en conjunto con los expertos del negocio, quienes transmitieron su experiencia y conocimientos sobre el cierre anticipado de las pólizas de seguro. El personal del área de Informática realizó la selección de las variables solicitadas inicialmente por los expertos desde los sistemas *legacy* de la compañía.

La muestra utilizada para este trabajo es de 17.832 registros asociados a clientes que poseen pólizas de vida y de accidentes personales correspondiente al periodo 2008-2010. Esta muestra incluye sólo las pólizas cerradas de manera voluntaria por el cliente.

En base a la experiencia y al conocimiento del experto del negocio es posible identificar las variables que son relevantes para construir modelos el modelo predictivo. En primera instancia se logró identificar un conjunto de 25 variables que, a juicio del experto, pueden explicar el comportamiento del cliente.

En las tablas 3, 4, 5, 6 y 7 se despliega el conjunto de variables candidatas a integrar el modelo, clasificadas en distintos grupos de acuerdo al tipo de característica del cliente al que se refieren.

Variables de Póliza		
Nombre	**Tipo**	**Descripción**
Número de pólizas contratadas	Discreta	Representa el número de pólizas contratadas por el cliente en la compañía.
Monto total de primas	Continua	Suma del monto total de primas a pagar por el cliente según su frecuencia de pago.
Meses pagados por póliza	Discreta	Cantidad de meses pagados de la póliza.
Línea de póliza	Categórica	Corresponde a la línea del producto, si corresponde a pólizas de vida o de accidentes personales.
Prima mensual	Continua	Monto de prima a cancelar mensualmente.
Código de frecuencia de pago	Categórica	Frecuencia de pago del monto mensual.
Prima anualizada	Continua	Monto de prima a pagar de forma anual.
Modo de pago	Categórica	Modo de pago de la prima

Tabla 3: Variables asociadas a la póliza de seguro

Variables comportamiento de cliente		
Nombre	**Tipo**	**Descripción**
Compró producto VP	Fecha	Fecha de compra de producto Vida Protegida
Compró producto HO	Fecha	Fecha de compra de producto Hogar
Compró producto SG	Fecha	Fecha de compra de producto Seguro General
Compró upgrade de póliza	Fecha	Fecha de compra de upgrade de póliza

Tabla 4: Variables de comportamiento de cliente

Variables del Asesor de Póliza		
Nombre	**Tipo**	**Descripción**
Grupo socioeconómico del asesor	Categórica	Categorización del asesor según su nivel socioeconómico
Agencia actual de asesor	Categórica	Agencia asignada al asesor al momento de la venta de póliza
Edad de asesor al momento de la venta	Discreta	Edad del asesor al momento de la venta de póliza
Sexo de asesor	Categórica	Género de asesor al momento de la venta
Estado civil asesor	Categórica	Estado civil del asesor al momento de la venta
Fecha contrato de asesor	Fecha	Fecha de inicio de contrato del asesor de la venta
Fecha finiquito de asesor	Fecha	Fecha de finiquito del asesor de la venta
Fecha finiquito de asesor actual	Fecha	Fecha de finiquito del asesor asignado actualmente al cliente.

Tabla 5: Variables asociadas al asesor de póliza

Variables Demográficas		
Nombre	**Tipo**	**Descripción**
Grupo Socioeconómico	Categórica	Categorización del cliente según su nivel socioeconómico
Ocupación	Categórica	Ocupación laboral del cliente al momento de contrato de póliza
Sexo	Categórica	Género del cliente
Edad	Discreta	Edad en años del cliente.
Estado Civil	Categórica	Estado civil del cliente al momento de contrato de póliza
Indicador de lealtad marca vehículo	Categórica	Indicador de lealtad basado en permanencia de marca de un vehículo determinado

Tabla 6: Variables asociadas al cliente (demográficas)

Variable dependiente		
Nombre	**Tipo**	**Descripción**
Estado de póliza	Categórica	Estado de la póliza al momento del estudio

Tabla 7: Variable objetivo

3.2 Preprocesamiento de datos

En esta etapa se definen los criterios para solucionar las inconsistencias que se encuentren. Dentro del conjunto de datos a utilizar existen dos tipos de inconsistencias: valores faltantes y valores fuera de rango. Los valores faltantes corresponden a la inexistencia del valor de un registro en cierta variable y los fuera de rango, a valores que se escapen de los rangos normales de las variables, como por ejemplo, edades con valores en negativo.

Las inconsistencias producen distorsiones sobre las distribuciones de las variables. Para demostrar el efecto que produce este fenómeno, considerar que para una variable con valores permitidos en el rango de 20 a 60, se ingresa un registro con valor 500. Al transformar los valores de la variable dentro de un intervalo [0, 1], el mínimo (20) tomaría como nuevo valor 0 y el máximo (500) tomaría como nuevo valor el 1. Ahora bien, si se considera eliminar el valor fuera de rango, es posible obtener la distribución correcta de la variable. En la figura 13 se representa la comparación de ambas variables, antes y después de la eliminación del valor fuera de rango.

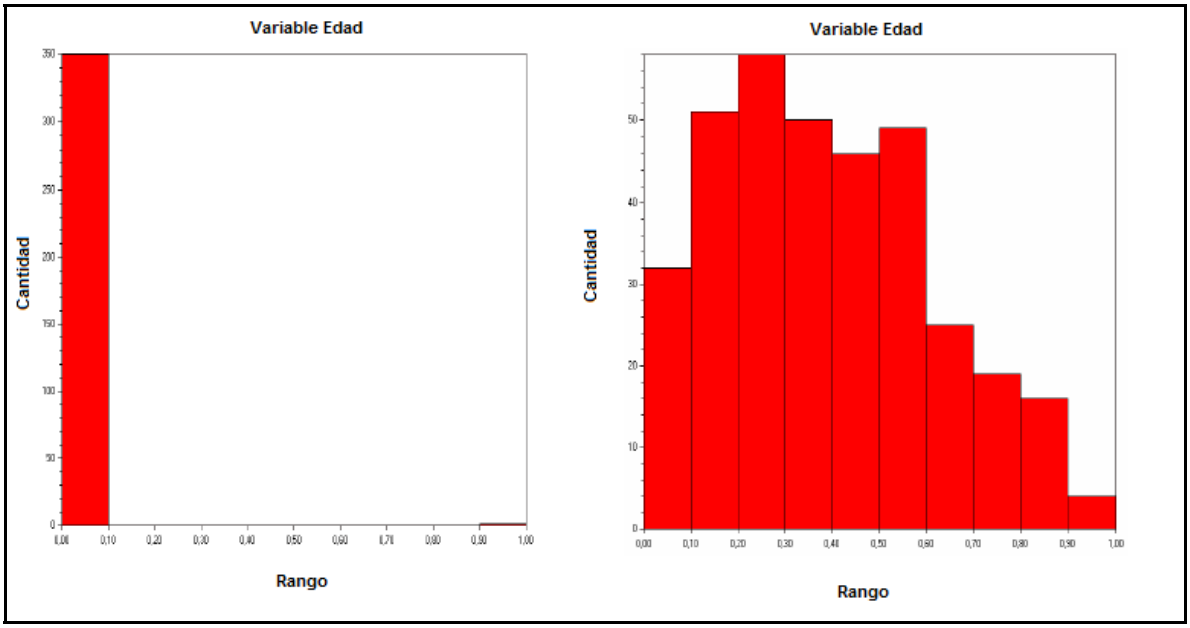


Figura 13: Ejemplo de valor fuera de rango

La identificación de un valor fuera de rango es bastante más complejo que la identificación de un valor faltante, dado que el registro existe en la base de datos y no es nulo (o blanco). Al ser un valor erróneo o inconsistente, una de las formas en que se identifican estos casos es por medio de la delimitación de rangos permitidos, definiendo una cota inferior y una cota superior para los valores de cada variable. De esta forma, si el valor de un registro se encuentra fuera de las cotas permitidas para la variable, se declara como valor fuera de rango. El resultado final de esta etapa, es tener una base de datos sin inconsistencias para el uso posterior durante el experimento.

3.3 Transformación de datos

El objetivo central de esta etapa es obtener una base de datos de trabajo para la aplicación del análisis estadístico, como para la aplicación de la técnica de minería de datos. Las transformaciones a aplicar son:

- Conversión de variables texto a variables categóricas numéricas.
- Escalamiento y estandarización de valor sobre un rango.
- Creación de nuevas variables a partir de las variables originales.

3.4 Selección de la técnica de minería de datos

Para el desarrollo del modelo predictivo, es posible utilizar distintas técnicas de minería de datos que permiten modelar un problema de predicción. Algunas de estas técnicas pueden ser los árboles de decisión, máquinas de soporte vectorial, árboles de decisión, redes bayesianas o regresión logística, entre otras [31]. Sin embargo las redes neuronales artificiales han sido aplicadas, con notable éxito, en problemas de clasificación,

simulación, procesado de señales y predicción de series temporales, en campos tan dispersos como la ingeniería, la economía o las telecomunicaciones. La base matemática sobre la que descansa esta técnica, su relativa sencillez y, especialmente, su extendida popularidad, se justifican para muchos investigadores para su uso. Un punto importante sobre las técnicas estadísticas habitualmente empleadas es que son lineales en los parámetros, es decir, se presupone que las relaciones entre las covariables del problema no revisten excesiva complejidad. En cambio, en otras disciplinas se realizan estudios en los que intervienen multitud de factores para los cuales, en muchas ocasiones, se obvian las interacciones o, simplemente, no se evalúan con suficiente rigor. Existen infinidad de variables para describir la situación y el contexto de un problema específico. La simplificación del problema, efectivamente, facilita el análisis pero, por contrapartida, conlleva una pérdida de generalización y de precisión en el estudio.

Las redes neuronales son capaces de captar los matices que se escapan a los métodos estadísticos más simples. En efecto, hay evidencias de los buenos resultados proporcionados en tareas bien definidas donde las interacciones entre las covariables del problema son significativas. Se ha establecido que las RNA son equivalentes a técnicas estadísticas paramétricas y no paramétricas [31], sin embargo, realmente ocupan un lugar intermedio y privilegiado; actúan como técnicas semi-paramétricas, es decir, más flexibles que los métodos paramétricos pero requieren menos parámetros que los métodos no paramétricos [31]. Otras ventajas son inherentes a su constitución y fundamentos neurobiológicos:

- Tratamiento no lineal de la información proporcionado por la interconexión de elementos simples de procesado no lineal (neurona).
- Capacidad de establecer relaciones entrada-salida a través de un proceso de aprendizaje.
- Aprendizaje adaptativo que les permite a llevar a cabo ciertas tareas mediante un entrenamiento con ejemplos ilustrativos. No es necesario tener modelos a priori ni se necesita especificar funciones de distribución de probabilidad.
- Robustez o tolerancia a fallos dado que almacenan la información aprendida de forma distribuida en las conexiones entre neuronas.
- Uniformidad de análisis y diseño proporcionado por teorías conjuntas que describen los diferentes algoritmos y aplicaciones.

En general, el uso de las redes neuronales artificiales está justificado en problemas donde se pueda aprovechar su potencial para el análisis no lineal de la información, aprovechando su función como memoria asociativa distribuida que evita en gran medida las dificultades en la adquisición de conocimiento experto, la tolerancia al ruido proporcionada por su arquitectura inherentemente paralela y su adaptabilidad. Estas características hacen a las redes neuronales preferibles a otros métodos matemáticos en problemas para los cuales:

- No es posible encontrar un conjunto de reglas sistemáticas que describan completamente el problema.
- Se dispone de una cantidad razonable de ejemplos representativos del problema.

- Hay que trabajar con datos imprecisos o incoherentes.
- Se tiene un gran número de variables que definen el problema (alta dimensionalidad del problema).
- Las condiciones del problema son cambiantes.

Como técnica de minería de datos se utiliza una técnica híbrida, la cual está compuesta de una red neuronal artificial del tipo *feedforward*, que adicionalmente se le incorpora un algoritmo evolutivo como método de aprendizaje y ajuste de la red. Se menciona como limitante, que sólo se utiliza este híbrido para probar el rendimiento y la capacidad clasificadora utilizando una técnica como son las redes neuronales, las cuales tienen cierto éxito comprobado en tareas de este tipo, y una técnica basada en nuevas investigaciones sobre aplicaciones de algoritmos evolutivos para resolución de problemas de optimización.

3.5 Evaluación y discusión de resultados

En esta etapa se evalúan los resultados del modelo según el nivel de acierto en la predicción de la clase para cada cliente. El resultado de esta etapa es la identificación de aquellas características que determinan el patrón de fuga, y poder ayudar al analista de negocio entregando la información necesaria para determinar las causas y motivos que gatilla esta conducta y determinar acciones que permitan:

- **Aumentar la rentabilidad**: si se identifica y retiene a un cliente, y se logra que éste mantenga su póliza vigente por más tiempo dentro de la compañía, es posible ir aumentando las utilidades y rentabilidad del negocio ya que a ese cliente se le pueden ofrecer otro tipo de productos y servicios.

- **Focalizar los esfuerzos de retención:** al identificar a los clientes potenciales a fuga, es posible focalizar de mejor manera los recursos para provocar que los esfuerzos para la retención sean más efectivas.

- **Mejora del servicio entregado**: al identificar los motivos o características que determinan que un cliente cierre sus pólizas, se permite la mejora continua de los productos y servicios entregados. Esto se logró revisando la importancia de cada variable incluida en el modelo

- **Fidelizar a los clientes**: al mejorar la calidad del servicio y productos entregados, aumenta la fidelidad del cliente; esto conlleva a que los clientes sean menos sensibles a la captación generada por la competencia, lo que genera que se deba invertir en la captación de nuevos clientes, incurriendo en un costo alto para la empresa de venta de seguros.

En el siguiente capítulo se centra en la etapa de análisis y preparación de los datos, la cual es de gran importancia para el inicio de la aplicación de la minería de datos y la extracción del conocimiento enfocado a

CAPÍTULO 4

Análisis y Preparación de los Datos

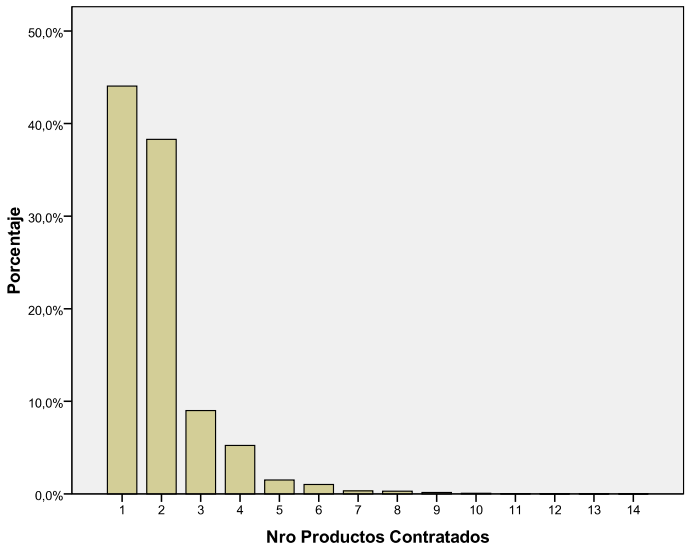
El preprocesado de los datos es una de las tareas más importantes dentro de todo proceso de extracción de conocimiento según lo explicado en el capítulo 2 y descrito en el diseño de la solución. A continuación se describe en detalle las tareas de cada etapa.

4.1 Descripción, exploración y análisis de calidad de los datos

Luego de definir las variables a utilizar en el experimento, se procedió a realizar un análisis descriptivo y exploratorio de estas variables para determinar cuáles son sus características, relaciones entre ellas y, además, poder verificar la calidad de los datos entregados pudiendo identificar valores faltantes y fuera de rango.

V1: Variables de Póliza

Número de pólizas contratadas: Corresponde a la cantidad de productos que posee el cliente dentro de la compañía. Se incluye esta variable para el estudio, ya que existe una relación inversamente proporcional a la cantidad de productos que un cliente tiene en la compañía con la tendencia a la fuga tal como se presente en la figura 14.



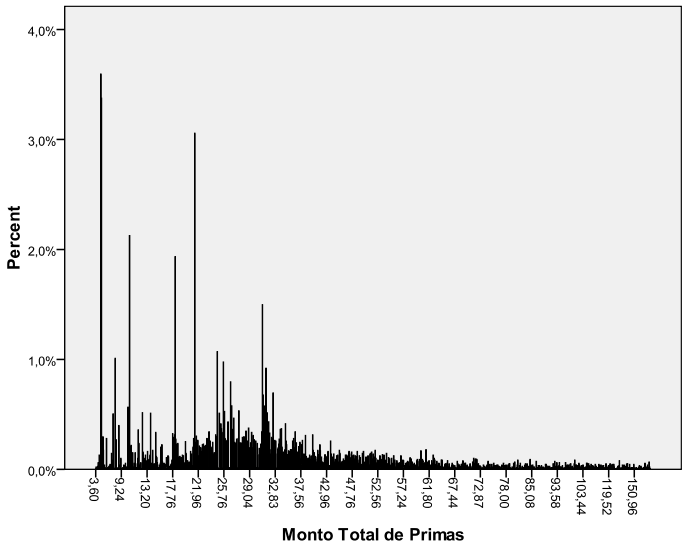
Estadísticos	
Nro Productos Contratados	
Media	1,90
Mediana	2,00
Moda	1
Desv. est.	1,179
Varianza	1,390
Mínimo	1
Máximo	14

Figura 14: Distribución de "Número de pólizas contratadas"

Monto total de primas: Corresponde al monto total de las primas a pagar de la póliza contratada. Se incluye esta variable ya que a juicio del experto del negocio, una razón de fuga sería que a mayor el monto de prima a pagar, es más posible que el cliente deje de pagar y cierre sus pólizas. En la Figura 15 se aprecia la distribución de la variable. En la tabla 8 se puede apreciar que un 81% de los datos corresponde a primas cuyos valores van desde las 3 a 50 UF.

Monto Total de Primas (UF)			
	Frecuencia	Porcentaje	Porcentaje Acumulado
100 y más	576	3,2	3,2
3 a 50	14448	81,0	84,3
50 a 100	2807	15,7	100,0
Total	17831	100,0	

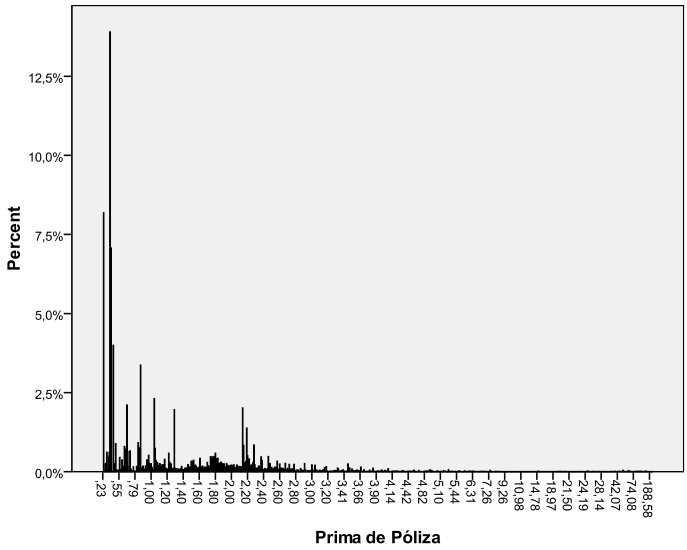
Tabla 8: Porcentaje y Frecuencia de "Monto Total de Primas"



Estadísticos	
Media	34,9674
Mediana	29,0400
Moda	5,28
Desv. est.	27,68328
Varianza	766,364
Mínimo	3,60
Máximo	392,17

Figura 15: Distribución "Monto Total de Primas"

Prima mensual (UF): Monto a pagar mensualmente de la póliza. Al consultar con el experto del negocio, éste nos indica que el monto total de primas se obtiene a partir de la prima anualizada (prima mensual * frecuencia de pago) por la cantidad de productos contratados de Vida o AP tal como se presenta en la figura 16.



Estadísticos	
Prima mensual de Póliza	
Media	1,8728
Mediana	,9700
Moda	,44
Desv. est.	7,51055
Varianza	56,408
Mínimo	,23
Máximo	374,50

Figura 16: Prima Mensual (UF)

Frecuencia de Pago: Esta variable corresponde a la frecuencia de pago de la póliza, según la figura 17 y la tabla 8, el 97,2% de las pólizas tiene frecuencia de pago mensual.

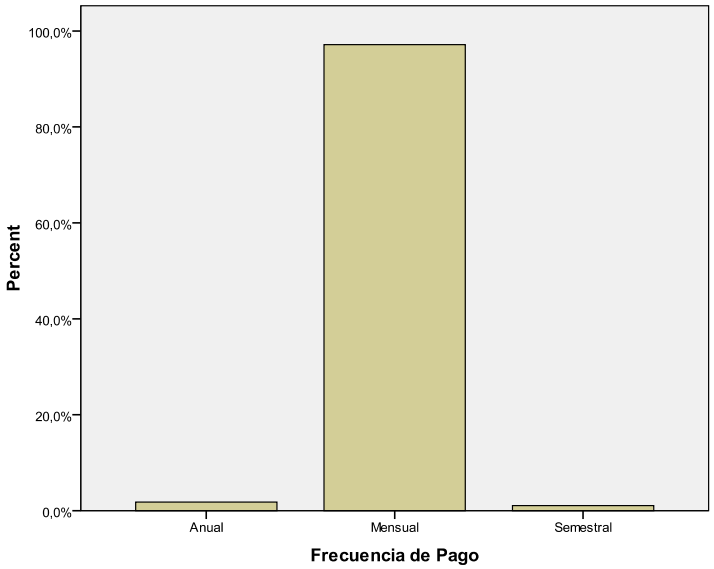
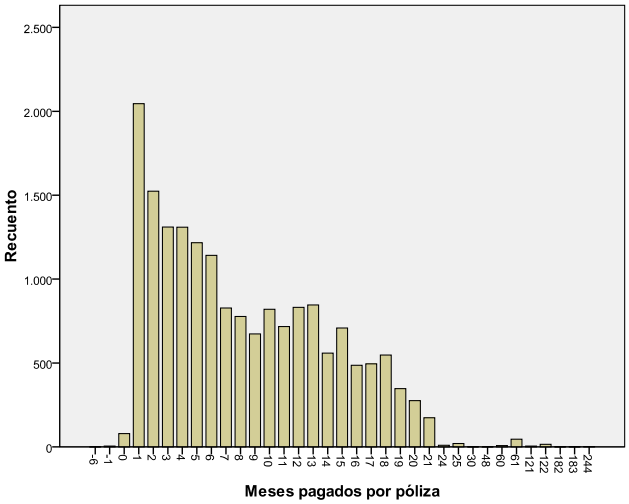


Figura 17: Distribución de variable "Frecuencia de Pago"

Frecuencia de Pago			
	Frecuencia	Porcentaje	Porcentaje acumulado
Anual	319	1,8	1,8
Mensual	17325	97,2	99,0
Semestral	187	1,0	100,0
Total	17831	100,0	

Tabla 9: Frecuencia y Porcentaje de variable "Frecuencia de Pago"

Meses pagados por póliza: Esta variable se incluye en primera instancia, ya que muestra la cantidad de meses pagados de la póliza; este dato es de relevancia ya que está estrictamente relacionada con la frecuencia de pago y si existe morosidad en la vida de la póliza. En la figura 18 se observa claramente que existen valores fuera de rango, por lo que no es un dato confiable desde el origen.



Estadísticos	
Meses pagados por póliza	
Media	8,42
Mediana	7,00
Moda	1
Desv. típ.	8,100
Varianza	65,614
Mínimo	-6
Máximo	244

Figura 18: Distribución "Meses pagados por póliza"

Línea de póliza: Corresponde a la línea de producto de la póliza; en la figura 19 se muestran los dos grandes productos que maneja la compañía que son: póliza de vida y póliza de accidentes personales**.** Para este estudio no se consideraron otros tipos de productos.

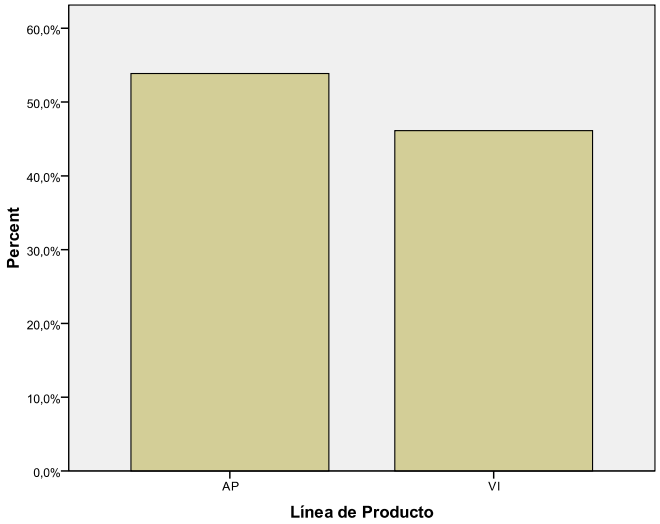


Figura 19: Líneas de Producto

Modo de pago: Modo de pago en el cual realiza el cliente su pago de póliza. En la Figura 20 se observa que el método de pago más utilizado por los clientes, es el automático. Cabe destacar que esta clase fue agrupada por el analista de negocio, la cual contenía las subclases PAC y PAT.

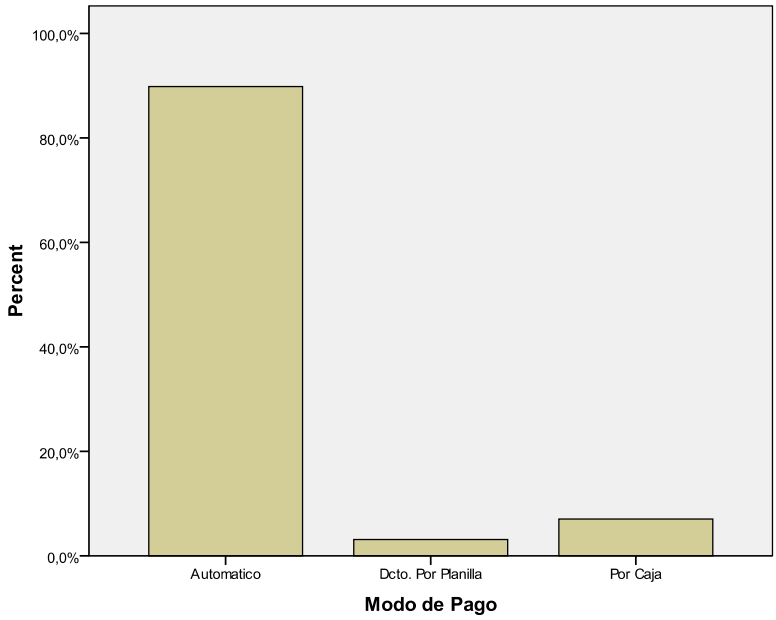


Figura 20: Modo de Pago

V2: Variables Demográficas.

Grupo Socioeconómico: Esta variable representa el sector de la población en la que se encuentra el cliente. En la Figura 21 se observa que para esta variable hay un gran porcentaje de valores faltantes (16,1%), lo que se verifica en la tabla 10.

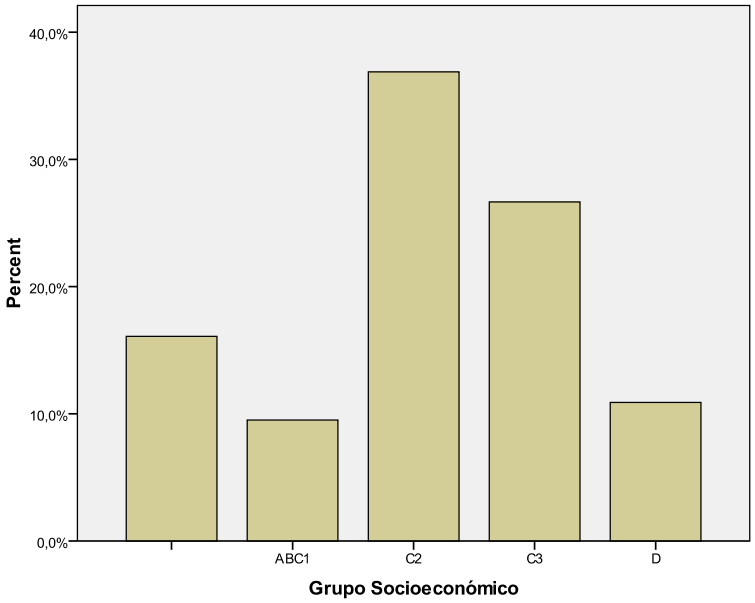


Figura 21: Grupo Socioeconómico

Grupo Socioeconómico			
	Frecuencia	Porcentaje	Porcentaje Acumulado
	2869	16,1	16,1
ABC1	1694	9,5	25,6
C2	6575	36,9	62,5
C3	4752	26,7	89,1
D	1941	10,9	100,0
Total	17831	100,0	

Tabla 10: Frecuencia y Porcentaje de variable "Grupo Socioeconómico"

Ocupación: Esta variable tiene relación con la actividad que ejerce el cliente dueño de la póliza. En la Figura 22 se muestra que existen muchas clases para esta variable, por lo que en el paso siguiente se realizará una transformación a esta variable agrupando características similares. El analista de negocio nos indica la importancia de esta variable, ya que es un claro indicador del nivel educacional del cliente.

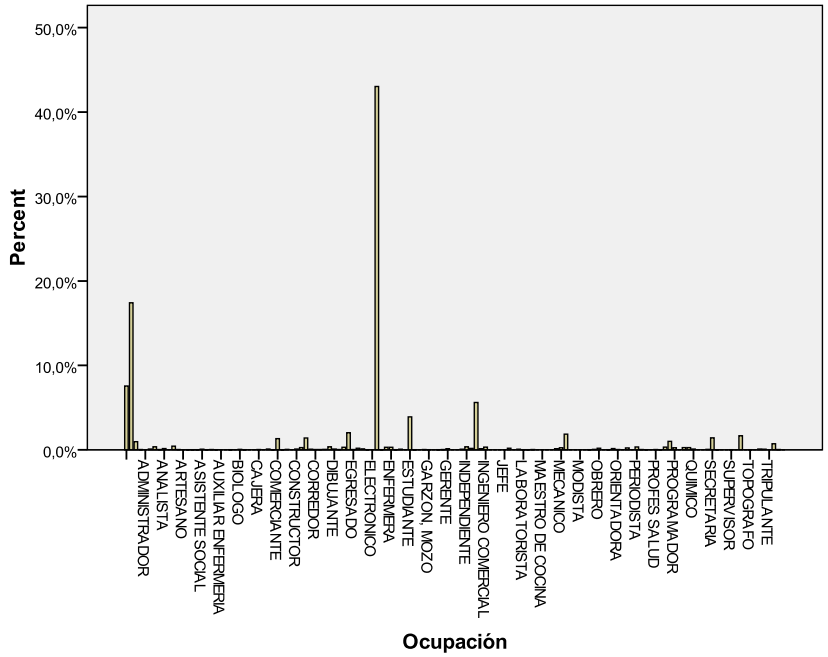


Figura 22: Distribución de variable "Ocupación"

Sexo: La relación entre esta variable y el fenómeno de fuga ha sido constantemente discutida en variados estudios. En [32] concluyen que existen diferencias notorias entre ambos sexos, en donde las mujeres son consideradas más estables en sus preferencias y, por ende, tienen una menor probabilidad de fugarse de la institución, a diferencia de los hombres que se consideran más inestables en sus preferencias y, por ende, con mayores tendencias a la fuga. Sin embargo, en [33] se obtienen resultados totalmente distintos y contrapuestos respecto al estudio anterior. En la figura 23 se observa que la variable Sexo (cliente) tiene una distribución en dónde más del 60% de las muestras corresponden a clientes del sexo masculino.

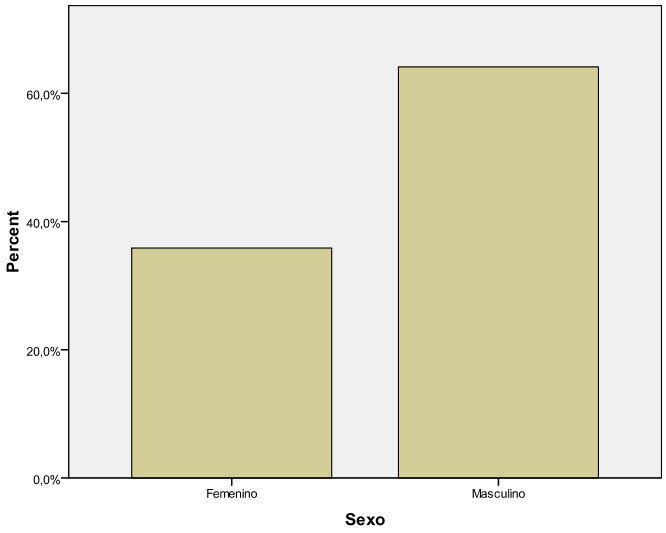
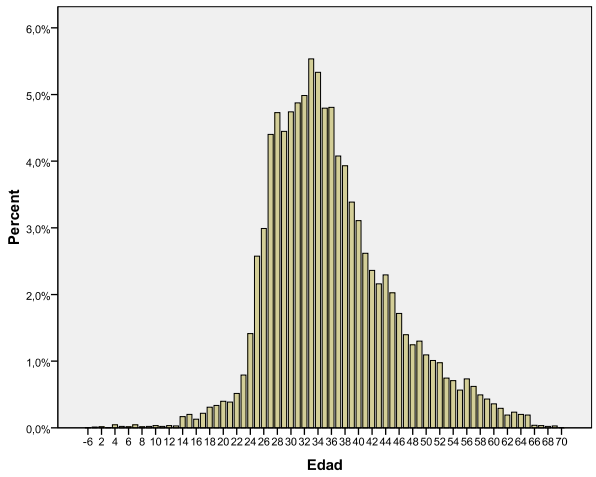


Figura 23: Distribución variable "Sexo"

Edad: En [32] argumentan que la edad es una variable discriminadora respecto a comportamientos asociados a la fuga de clientes. En dicho estudio, se concluye que los clientes de mayor edad son más estables en sus preferencias y, por ende, tienen menores tendencias a cambiarse a otra institución. En caso contrario, para un cliente joven se concluyó que son más inestables en sus preferencias, lo que aumenta su tendencia a cambiarse de institución. Al extrapolar esto a las pólizas de seguro, los clientes de mayor edad son más estables, ya que para ellos es más difícil poder abrir una nueva póliza de seguro por las enfermedades preexistentes y por el encarecimiento de la mantención de las pólizas. En cambio los clientes jóvenes tienden a cerrar sus productos por no sentir la necesidad de tener una póliza de vida.

En la figura 24 se observa que existen valores fuera de rango para las edades de los clientes. Haciendo un segundo análisis de esta variable, contrastándola con la variable objetivo, indica que 19 registros pertenecientes a la clase RENUNCIADA presentan valores fuera del rango de estudio.



Estadísticos	
Edad	
Media	35,98
Mediana	35,00
Moda	33
Desv. típ.	9,084
Varianza	82,511
Mínimo	-6
Máximo	70

Figura 24: Distribución de variable "Edad"

Indicador de lealtad marca vehículo: En la figura 25 se muestra la distribución de la variable Indicador de Lealtad de marca de vehículo. Esta variable se incorporó por ser un buen referente del comportamiento de un cliente respecto a su fidelidad ante a una marca determinada. Esta variable tiene 3 clases: LEAL, PARCIALMENTE LEAL y NO LEAL. Según la tabla 11 se observa que la suma de las dos últimas clases supera el número de la primera clase.

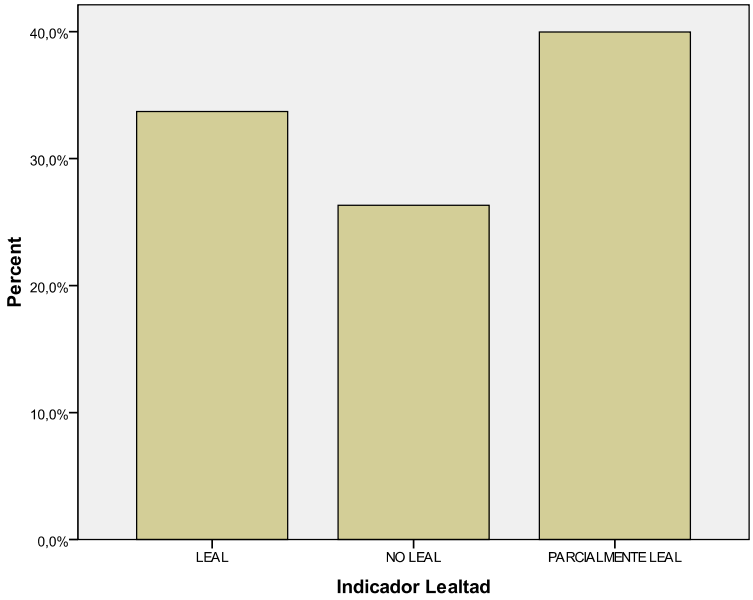


Figura 25: Indicador de Lealtad

Indicador Lealtad * Estado de Póliza				
		Estado de Póliza		Total
		RENUNCIADA	VIGENTE	
Indicador Lealtad	LEAL	1504	4506	6010
	NO LEAL	890	3803	4693
	PARCIALMENTE LEAL	1361	5767	7128
Total		3755	14076	17831

Tabla 11: Indicador de Lealtad/Estado de Póliza

V3: Variables del Asesor de Venta de la Póliza

Grupo socioeconómico del asesor: En la figura 26 se observa que para la variable Grupo Socioeconómico del Agente de Emisión posee alrededor del 18% con valores faltantes.

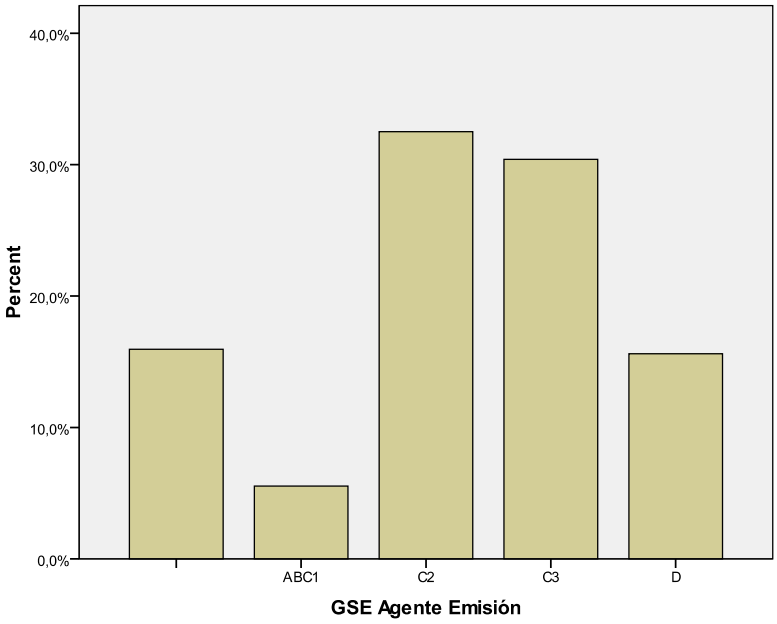


Figura 26: Grupo Socioeconómico del Agente Emisor

Sucursal de Agente de Emisión: Para esta variable, en la figura 27 se despliega la distribución, la cual nos demuestra que tiene varias clases dependiendo de la oficina. En la próxima etapa se agruparán estas clases de variables dependiendo de la zona geográfica de la sucursal.

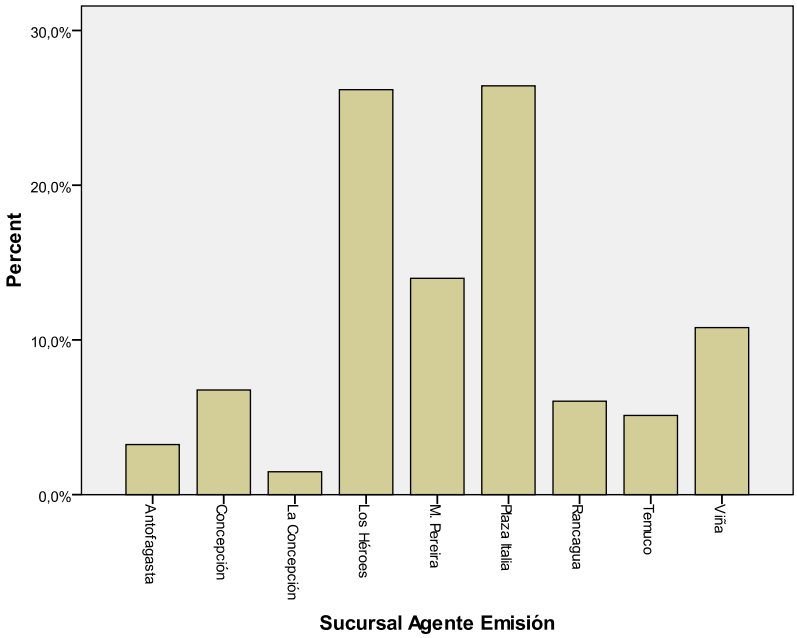
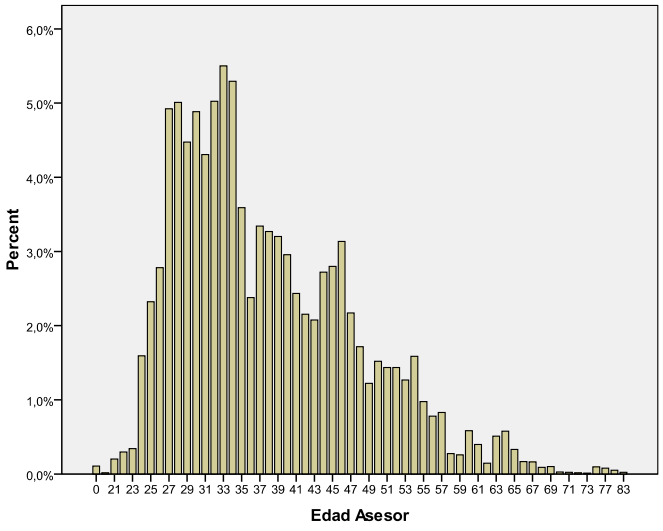


Figura 27: Sucursal Agente Emisión

Sucursal Agente Emisión * Estado de Póliza				
		Estado de Póliza		Total
		RENUNCIADA	VIGENTE	
Sucursal Agente Emisión	Antofagasta	95	482	577
	Concepción	263	943	1206
	La Concepción	42	221	263
	Los Héroes	1120	3547	4667
	M. Pereira	357	2136	2493
	Plaza Italia	898	3813	4711
	Rancagua	229	847	1076
	Temuco	393	520	913
	Viña	358	1567	1925
Total		3755	14076	17831

Tabla 12: GSE Sucursal Agente Emisión/Estado de Póliza

Edad de asesor al momento de la venta: En la Figura 28 se encuentra la distribución de la edad del Asesor de la póliza; se puede apreciar que existen valores fuera de rango, considerando que la edad laboral de un Asesor es desde los 18 años hasta los 65. Dada esta información, la tabla 18 indica que 18 registros presentan valores fuera de rango de la edad (menores de 18 y mayores de 65), dentro de la clase de interés que es la póliza RENUNCIADA.



Estadísticos	
Edad Asesor	
Media	37,80
Mediana	35,00
Moda	33
Desv. típ.	9,955
Varianza	99,108
Mínimo	0
Máximo	83

Figura 28: Distribución "Edad Asesor"

Estado civil asesor: En la Figura 29 para la variable Estado Civil de Asesor, se presentan 3 registros que no poseen información, y contrastando con la tabla 13, estos registros pertenecen a la clase "RENUNCIADO" de la variable objetivo.

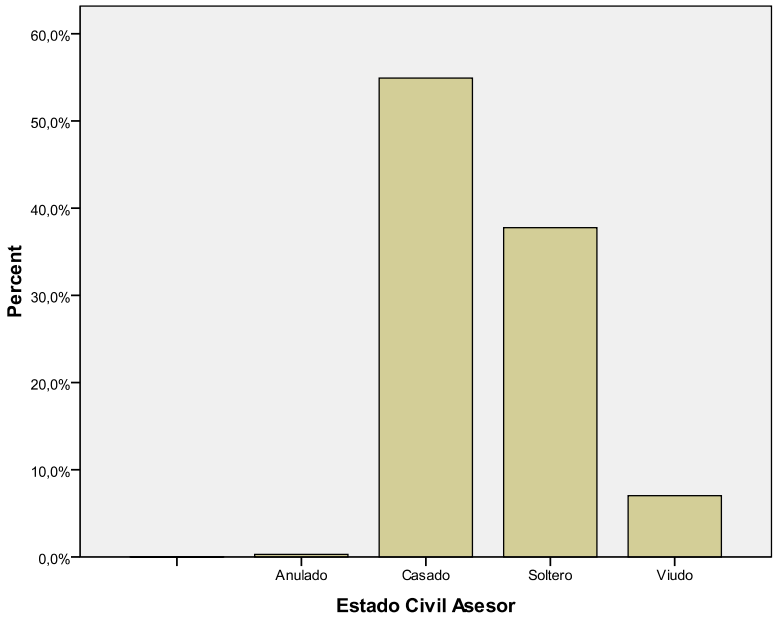


Figura 29: Estado Civil Asesor

Estado Civil Asesor * Estado de Póliza				
		Estado de Póliza		Total
		RENUNCIADA	VIGENTE	
Estado Civil Asesor		3	0	3
	Anulado	15	36	51
	Casado	1906	7886	9792
	Soltero	1616	5117	6733
	Viudo	215	1037	1252
Total		3755	14076	17831

Tabla 13: Estado Civil Asesor/Estado de Póliza

Variable Dependiente

La variable dependiente, en la cual se basa la fuga o no de un cliente, se define por el estado de la póliza. Esta variable fue medida dentro del periodo de extracción de los datos, que corresponde a 24 meses.

En la figura 30, se puede apreciar que la clase de mayor interés (RENUNCIADA) corresponde a un 21,1% del total de la muestra de clientes, es decir que la tasa de retención de la institución estudiada para un periodo de 24 meses, corresponde a un 78,9%. Considerando este dato de interés para el negocio, se justifica aún más poder encontrar un modelo adecuado que permita determinar de manera temprana el patrón de comportamiento de los clientes propensos a cerrar sus pólizas.

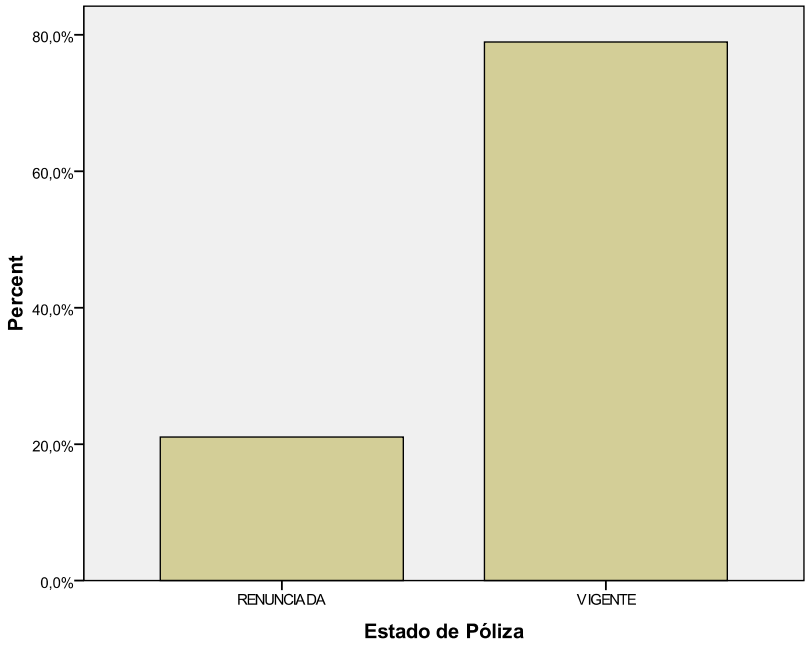


Figura 30: Estado de Póliza

Conclusiones de la etapa.

Del análisis inicial de los datos, se obtuvieron las siguientes conclusiones:

- Para la variable "grupo socioeconómico de asesor" se tiene un alto porcentaje de valores faltantes . Dada la dificultad de poblar estos valores en base a otros datos del cliente, se opta por no considerarla en el estudio.
- Se observa que para varias variables con valores fuera de rango que exceden de los valores límites.
- Las variables asociadas al asesor, como el estado civil y el sexo, se eliminan ya que a juicio del analista de negocio, esta información no influye en el cierre de las pólizas.

4.2 Preparación de los datos

4.2.1 Selección de variables.

Según el resultado de la etapa anterior, la tabla 14 lista las variables a utilizar para la preparación de los datos.

Nombre	Tipo	Valores
Grupo Socioeconómico	Categórica	ABC1, C2, C3, D
Número de pólizas contratadas	Discreta	1 a 14 pólizas
Monto total de primas	Continua	3 a 400 UF
Meses pagados por póliza	Discreta	-6 a 244 meses
Línea de póliza	Binaria	VI (Vida), AP (Accidente Personal)
Modo de pago	Categórica	Automático, Por Caja, Descuento Planilla
Ocupación	Categórica	Multiclase
Sexo	Binaria	Masculino, Femenino
Edad	Discreta	-6 a 70 años
Estado Civil	Categórica	Soltero, Casado, Anulado, Viudo
Indicador de lealtad marca vehículo	Categórica	Leal, Parcialmente Leal, No Leal
Edad de asesor al momento de la venta	Discreta	0 a 83 años
Antigüedad de asesor.	Discreta	
Compró producto VP	Fecha	Fecha de compra
Compró producto HO	Fecha	Fecha de compra
Compró producto SG	Fecha	Fecha de compra
Compró upgrade de póliza	Fecha	Fecha de compra
Estado de póliza	Binaria	Vigente, Renunciada

Tabla 14: Selección de Variables de estudio

4.2.2 Limpieza de datos.

En esta etapa se aplicaron los criterios y soluciones para las inconsistencias encontradas en la base de datos, datos duplicados, valores faltantes y valores fuera de rango.

Para la identificación de los datos duplicados, se estimó que eran aquellos registros correspondientes al mismo cliente que repetían los mismos datos, excepto el tipo de póliza y el monto de la prima a pagar. Se

optó por dejar sólo un registro correspondiente a su póliza de vida. La tabla 15 muestra la cantidad y porcentaje de casos duplicados encontrados en la base de datos entregada.

Identificación de casos duplicados			
	Frecuencia	Porcentaje	Porcentaje acumulado
Casos duplicados	6999	39,3	39,3
Casos no duplicados	10832	60,7	100,0
Total	17831	100,0	

Tabla 15: Resumen de casos duplicados

La identificación de un valor faltante se realizó en la base a la ausencia del valor de un registro. La tabla 16 muestra el porcentaje de valores faltantes que tiene cada variable de entrada respecto al total de registros.

Nombre Variable	% completados	% faltantes	% total
Grupo Socioeconómico	83,8%	16,2%	100%
Número de pólizas contratadas	100%	0%	100%
Monto total de primas	100%	0%	100%
Meses pagados por póliza	100%	0%	100%
Línea de póliza	100%	0%	100%
Modo de pago	100%	0%	100%
Ocupación	75,3%	24,7%	100%
Sexo	100%	0%	100%
Edad	100%	0%	100%
Estado Civil	99,99%	0.01%	100%
Indicador de lealtad marca vehículo	100%	0%	100%
Edad de asesor al momento de la venta	100%	0%	100%
Antigüedad de asesor.	100%	0%	100%
Compró producto VP	100%	0%	100%
Compró producto HO	100%	0%	100%
Compró producto SG	100%	0%	100%
Compró upgrade de póliza	100%	0%	100%
Estado de póliza	100%	0%	100%

Tabla 16: Resumen de % de datos faltantes

La identificación para un valor fuera de rango u *outliers*, se definió intervalos para los valores permitidos para las variables de entrada, cada uno con un valor mínimo y un valor máximo para las variables numéricas. En la tabla 17 se presentan las variables y sus rangos permitidos.

Nombre Variable	Valor mín	Valor max	% fuera rango
Número de pólizas contratadas	1	14	0%
Monto total de primas	3	400	0%
Meses pagados por póliza	0	60	1%
Edad	14	80	0,7%
Edad de asesor al momento de la venta	18	65	1%
Antigüedad de asesor.	0	400	0%

Tabla 17: Cotas superior e inferior para variables

Para resolver las inconsistencias se emplearon las siguientes soluciones:

- **Eliminación de registros**: Esta solución se basa en eliminación de todos aquellos objetos (clientes en nuestro caso) que tengan uno o más valores faltantes o fuera de rango dentro de sus registros de sus variables. Para el caso de los valores fuera de rango, se optó por eliminar los registros que tenían menos de un 1%. Se verificó que con esta eliminación es poco significativa la pérdida respecto al total de clientes.
- **Creación de nueva categoría**: Esta solución se implementó para los datos discretos faltantes, dado a que existían datos para ciertas variable, que se perderían al eliminarlos si sólo poseían una variable categórica con valor faltante.

4.2.3 Cambio de Formato de Variables

Las variables con formato texto son variables categóricas. Para realizar el experimento, cada una de ellas se codifica con valor 1 en cada neurona por separada. La tabla 18 muestra las variables sufrieron esta transformación:

Nombre de Categoría	Valor
Grupo Socioeconómico	
ABC1	1
C2	2
C3	3
D	4
No Informado	5
Indicador de Lealtad	
Leal	1
Parcialmente Leal	2
No Leal	3
Sexo	
Masculino	0
Femenino	1
Estado Civil	
Soltero	1
Casado	2
Viudo	3
Anulado	4
Línea de Producto	
Vida	0
Accidente Personal	1
Modo de Pago	
Automático	1
Por Caja	2
Descto. Planilla	3
Frecuencia de Pago	
Anual	1
Semestral	2
Mensual	3

Tabla 18: Cambio de formato de variables categóricas

Para las variables desplegadas en la tabla 19, se consideró la fecha de contrato como indicador de cambio, por lo que se reemplazaron dichas fechas con un valor 1, y la ausencia de registros con valor 0.

Nombre de Categoría	Valor
Compró producto VP	
Adquirió producto	1
No adquirió producto	0
Compró producto HO	
Adquirió producto	1
No adquirió producto	0
Compró producto SG	
Adquirió producto	1
No adquirió producto	0
Compró upgrade de póliza	
Adquirió producto	1
No adquirió producto	0

Tabla 19: Cambio de formato de variables de fecha

4.2.4 Generación de nuevas variables

Dada la complejidad con algunas variables categóricas se generaron nuevas variables para facilitar el estudio. La tabla 20 muestra las nuevas variables que generaron nuevas:

Variable anterior	Nueva Variable	Descripción	Valor
Ocupación	Tipo Profesión	Nivel educacional alcanzado por el cliente a partir de su ocupación laboral.	1=Profesional 0=No Profesional
Fecha Finiquito Asesor Actual	Cliente Huérfano	Indica si un cliente no tiene asesor asignado a su póliza.	1=Huérfano, 0=con asesor.
Fecha contrato Asesor Venta	Antigüedad Asesor	Cantidad de meses de permanencia del asesor en la compañía.	0-400 meses
Fecha Finiquito Asesor Venta	Cambio Asesor	Indica si un cliente tuvo un cambio de asesor durante su permanencia	0=Sin cambio, 1= Con cambio
Sucursal de Emisión	Zona Cliente	Indica la zona geográfica del cliente.	1=Norte, 2=Centro, 3=Sur

Tabla 20: Resumen de generación de nuevas variables

4.2.5 Escalamiento de Variables

Para asegurar una mejor convergencia de los algoritmos es recomendable escalar las variables dentro de un mismo intervalo, siendo la idea central dejar las variables en una escala común y comparable [27], por lo tanto todas las variables fueron escaladas dentro de un intervalo [0,1].

4.3 Resumen del capítulo

Durante la etapa de preparación de los datos, se han realizado distintas tareas que permiten dejar una base de datos de trabajo consistente. Dentro de estas tareas cabe destacar el escalamiento de las variables y las transformaciones de aquellas del tipo categóricas, dejándolas como nodos separados con valor uno. Estos cambios son necesarios por el tipo de algoritmo a aplicar que tiene como restricción la representación de los datos de entrada deben ser numéricos, como es el caso de la red neuronal evolutiva. En el siguiente capítulo se realiza la ejecución de los experimentos y el despliegue de los resultados obtenidos en cada uno de ellos.

CAPÍTULO 5

Aplicación de Minería de Datos

Todos los pasos realizados en la preparación de los datos, ha permitido obtener una consistente base de datos de trabajo. De la observación de los datos se han obtenido algunas conclusiones, aunque no es sino en este momento cuando es posible intentar extraer información más útil que permita comprender mejor la interrelación entre las variables a analizar, y conclusiones que sean de real apoyo a evaluar el proceso.

En este capítulo se analizan los resultados obtenidos por distintos modelos implementados. Se utilizaron dos conjuntos para la definición de los modelos: entrenamiento y prueba; con el conjunto de entrenamiento se construye el modelo y con el conjunto de prueba se evalúa la capacidad de clasificación. Para hacer comparables estos resultados se utilizaron los mismos conjuntos para todos los modelos implementados.

El objetivo central de esta etapa corresponde a la búsqueda de la mejor topología de red, la que será llamada *configuración óptima de parámetros*; esta configuración se caracteriza por obtener los mejores resultados en términos de predicción sobre el conjunto de prueba. En la primera sección se discute la mejor forma para obtener la configuración óptima de parámetros para cada modelo y se describe la forma en que se construyeron los conjuntos de entrenamiento y prueba. La segunda sección compara los resultados obtenidos por cada modelo realizado, detallando las configuraciones óptimas de parámetros.

5.1 Generación de conjunto de datos para la generación de modelos

La forma en se generaron los conjuntos de entrenamiento y de prueba fue mediante la aplicación del método de validación cruzada *10-fold* [34]. Este método divide la base total de datos en diez subconjuntos con igual número de objetos; luego se asignan nueve de estos subconjuntos para conformar el conjunto de entrenamiento y un conjunto para la validación. El conjunto de entrenamiento se utiliza para la construcción del modelo y para la selección de los parámetros y el conjunto de test para evaluar la efectividad. Dicha partición y posterior asignación se realiza diez veces, teniendo como objetivo central que cada subconjunto constituya una vez al conjunto de prueba. Esto último minimiza el sesgo ocasionado por la construcción, evaluación y posterior selección de un modelo basado en un conjunto particular de objetos.

La ventaja de utilizar este método permite la calibración de los distintos parámetros bajo distintos grupos de datos para obtener el mejor resultado posible

5.2 Ajuste de parámetros

Se comienza el experimento con el ajuste de los parámetros del modelo, los cuales se dividirán en dos tipos: parámetros fijos, los cuales se mantienen sin alteración durante todo el experimento; y parámetros configurables, los cuales se van ajustando por cada experimento.

5.2.1 Parámetros fijos

- **Nodos de entrada y salida de la red:** corresponden a las variables seleccionadas y adecuadas (transformación) en la etapa anterior, las cuales permitirán realizar una aproximación funcional óptima. Se inicializan las variables independientes, el número de neuronas de la capa de entrada está determinado por el número variables. La capa de salida está determinada por la variable dependiente, puesto a que se necesita discriminar entre dos categorías, bastará con utilizar una única neurona (por ejemplo, salida 1 para la categoría A, salida 0 para la categoría B).

- **Número de partículas de la población (enjambre):** este parámetro corresponde a la cantidad de individuos del enjambre que componen cada generación del algoritmo evolutivo. Se toma un valor de fijo de treinta partículas, puesto a que la literatura recomienda dicho tamaño para la mayoría de los problemas a optimizar..

- **Número de ciclos o épocas:** cantidad máxima de iteraciones del algoritmo evolutivo para que pueda converger a una solución.

5.2.2 Parámetros configurables

- **Inicialización del enjambre:** corresponde al rango de inicialización óptimo de los pesos de la red neuronal.

- **Velocidad máxima de la partícula $V_{\max}$:** permite la movilidad de la partícula sobre el universo de soluciones para encontrar una matriz de pesos sináptica óptima.

- **Parámetros c_1, c_2, r_1, r_2 :** aceleran la convergencia del algoritmo y evitan la caída en mínimos locales.

- **Peso de inercia w :** controla el impacto de la historia previa de las velocidades actuales, compensando las capacidades de exploración global (amplias) y local (cercanas).

5.2.3 Experimentos de Ajustes de Parámetros

El objetivo de estos experimentos es determinar cuál es la configuración óptima de parámetros que permitan entrenar la red neuronal y así obtener el mejor resultado para la resolución del problema de clasificación.

A menudo, la naturaleza estocástica de los algoritmos evolutivos les imposibilita obtener una misma solución cuando se realiza un nuevo proceso de optimización, es decir, las nuevas soluciones obtenidas son próximas (parecidas) a las encontradas en ensayos previos, al punto que es posible hablar de un rango de variabilidad de soluciones factibles. Este fenómeno se conoce como "ruido" [35] y está relacionado al comportamiento aleatorio de algunos de los parámetros de los propios algoritmos. Pese a ello, se ha determinado que una sintonización apropiada de parámetros disminuiría considerablemente el margen de esta variabilidad.

Para encontrar la mejor configuración se realizará un análisis que se basa en evaluar diferentes escenarios del algoritmo mediante ensayos, calculando la media y desviación estándar de las soluciones encontradas por cada configuración. Una vez realizados los ensayos, la configuración y los resultados se tabulan en una tabla para decidir cuál es la configuración que entrega el mejor error.

Inicialización de pesos iniciales (Inicialización del enjambre)

Se hace una asignación de pesos pequeños generados de forma aleatoria, en un rango de valores entre [−0.5, 0.5] para iniciar el ajuste del resto de parámetros del algoritmo evolutivo.

Parámetros *c1, c2* y V_{max}

Para el ajuste de estos parámetros se definieron los siguientes valores para la realización de los experimentos, tal como se presenta en la tabla 21.

Parámetro	1	2	3	4
c1 y c2	[2, 2]	[2, 0.75]	[0.75, 2]	[1.75, 0.75]
Vmax	0.1	0.05	0.15	0.2

Tabla 21: Configuraciones para V_{max} , *c1 y c2*

Los resultados obtenidos modificando los parámetros indicados se muestran en las tablas 22, 23, 24, 25 y en las figuras 31, 32, 33 y 34.

V_{max}	*c1 y c2*	μ MSE	σ MSE
0.1	[2, 2]	**2,20E-04**	**2,11E-04**
	[2, 0.75]	3,70E-04	4,60E-04
	[0.75, 2]	6,56E-03	5,16E-03
	[1.75, 0.75]	7,42E-03	5,36E-03

Tabla 22: Resultados para Vmax=0.1

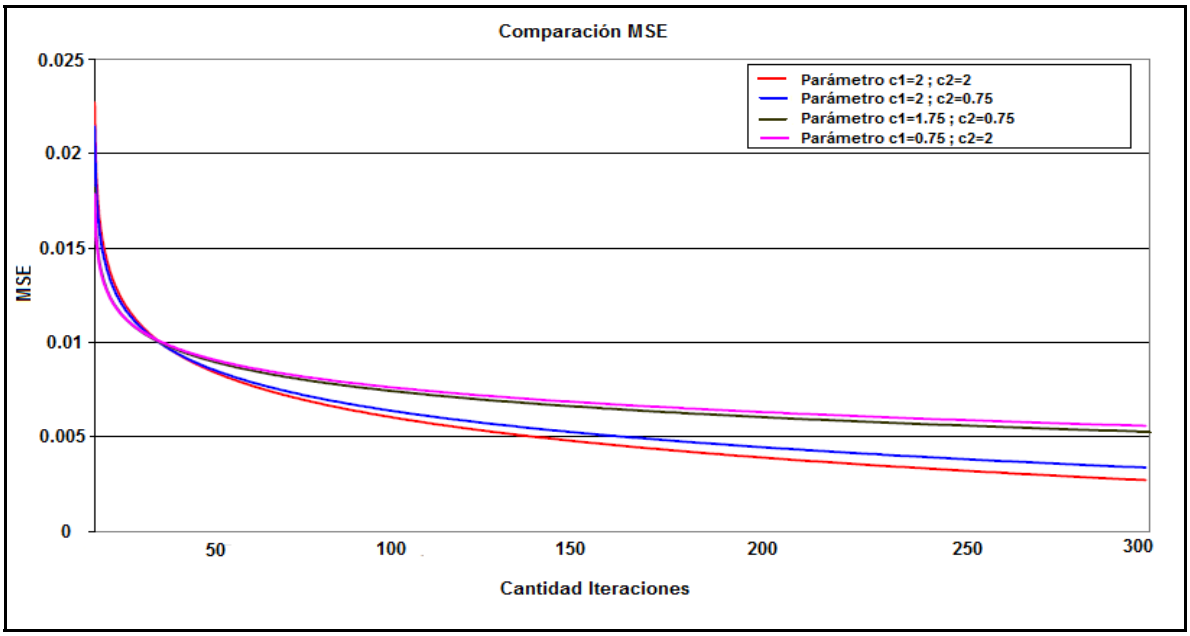


Figura 31: Comparación del error para Vmax=0.1

V_{max}	*c1 y c2*	μ MSE	σ MSE
0.05	[2, 2]	**2,18E-04**	**2,12E-04**
	[2, 0.75]	3,62E-03	3,40E-03
	[0.75, 2]	4,67E-03	2,87E-03
	[1.75, 0.75]	4,81E-03	5,36E-03

Tabla 23: Resultados para Vmax=0.05

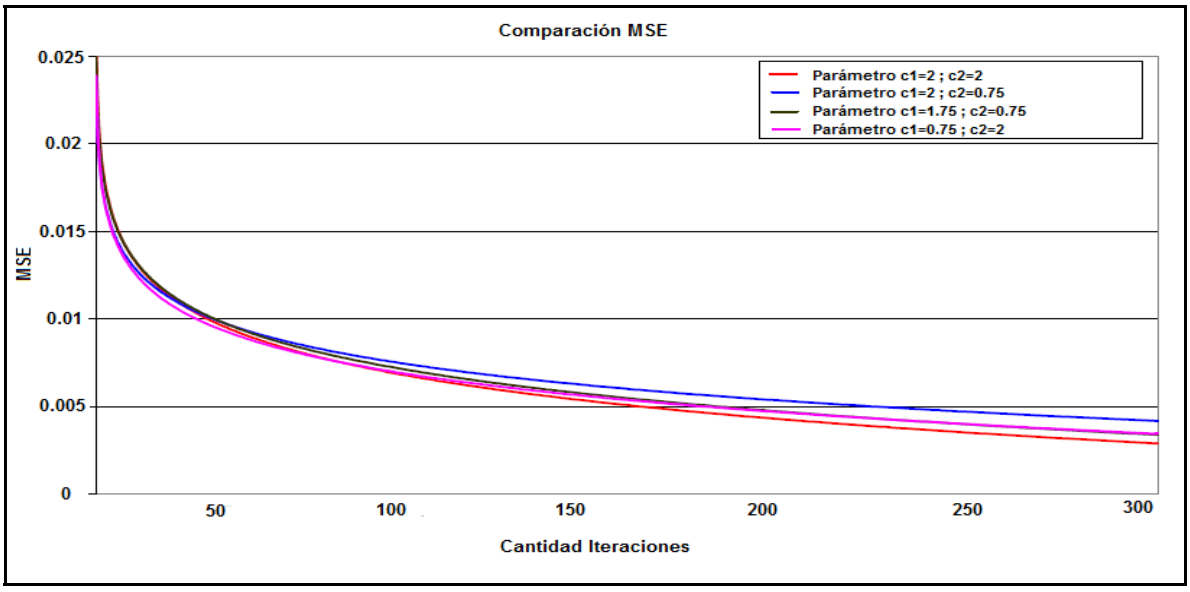


Figura 32: Comparación del error para Vmax=0.05

V_{max}	*c1 y c2*	μ MSE	σ MSE
0.15	[2, 2]	4,77E-04	3,40E-04
	[2, 0.75]	**6,12E-06**	**5,66E-06**
	[0.75, 2]	4,96E-03	4,50E-03
	[1.75, 0.75]	4,10E-03	5,56E-03

Tabla 24: Resultados para Vmax=0.15

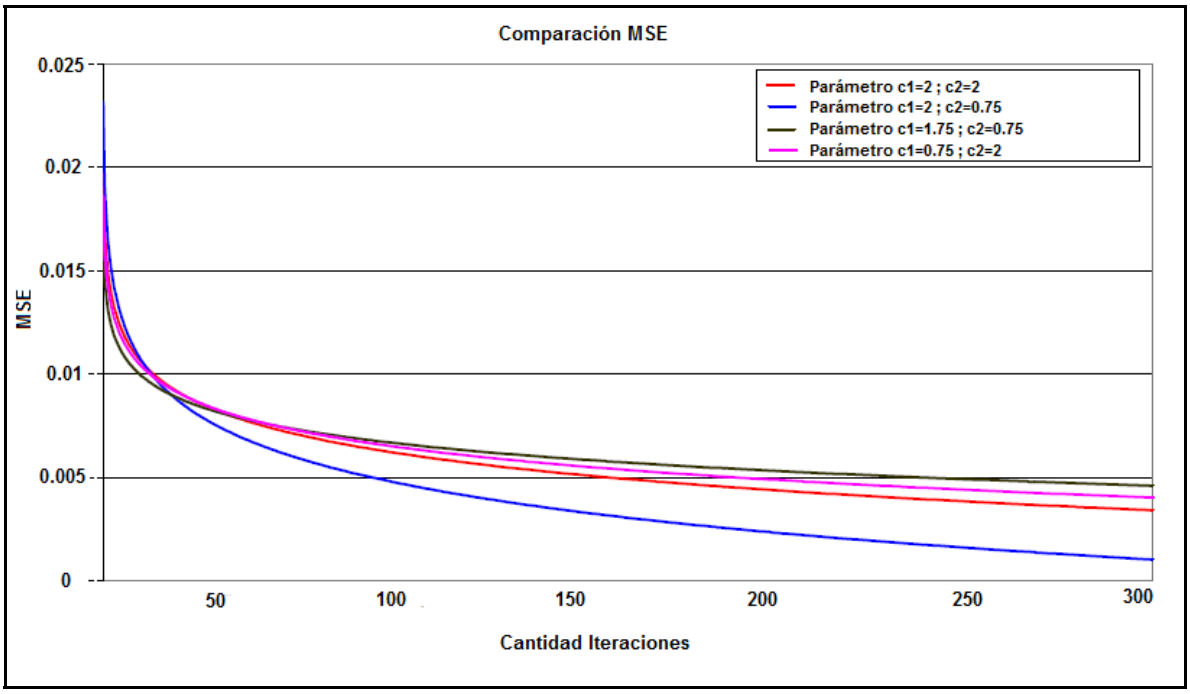


Figura 33: Comparación del error para Vmax=0.15

V_{max}	*c1 y c2*	μ MSE	σ MSE
0.2	[2, 2]	**5,20E-05**	**3,88E-05**
	[2, 0.75]	7,26E-04	5,48E-04
	[0.75, 2]	9,78E-03	5,30E-03
	[1.75, 0.75]	1,11E-02	1,18E-02

Tabla 25: Resultados para Vmax=0.2

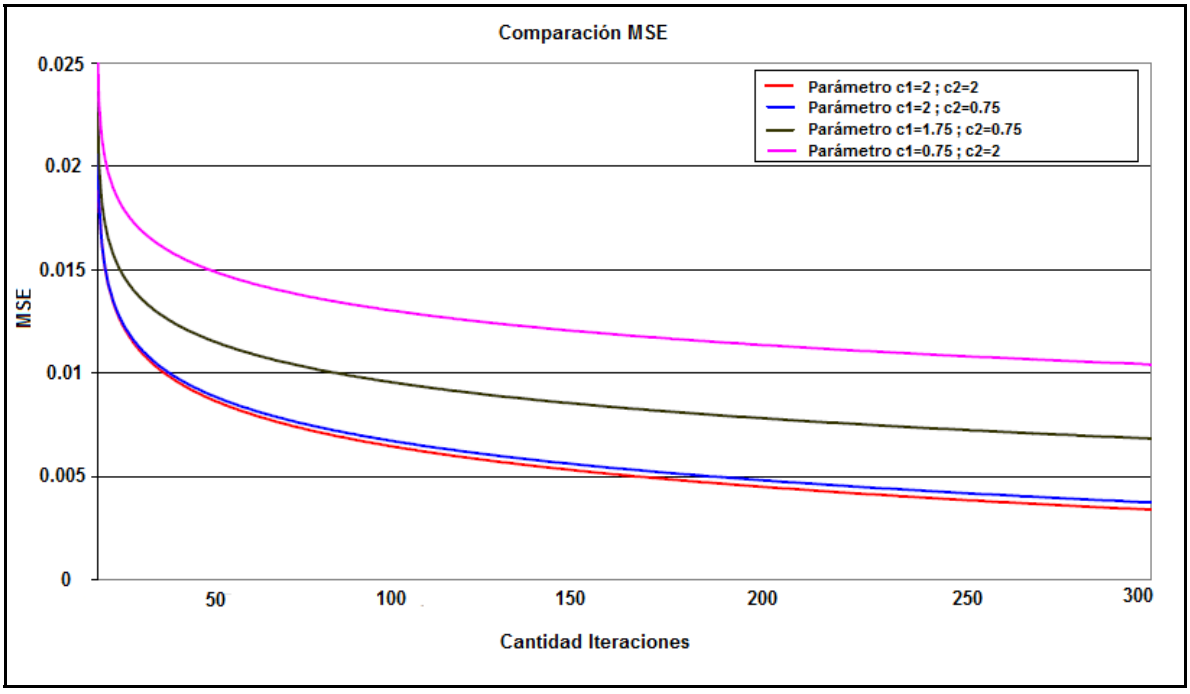


Figura 34: Comparación del error para Vmax=0.2

Parámetro *w*

Pare el ajuste este parámetro se definió los siguientes valores indicados en la tabla 26:

Parámetro	1	2	3	4
W	0.3	0.4	0.5	0.6

Tabla 26: Configuraciones para parámetro *w*

Los resultados obtenidos modificando los parámetros indicados son entregados en la tabla 27:

w	μ MSE	σ MSE
0.3	2,26E-02	2,40E-02
0.4	4,62E-03	7,62E-03
0.5	5,43E-04	5,38E-04
0.6	7,27E-05	6,32E-05

Tabla 27: Resultados para parámetro *w*

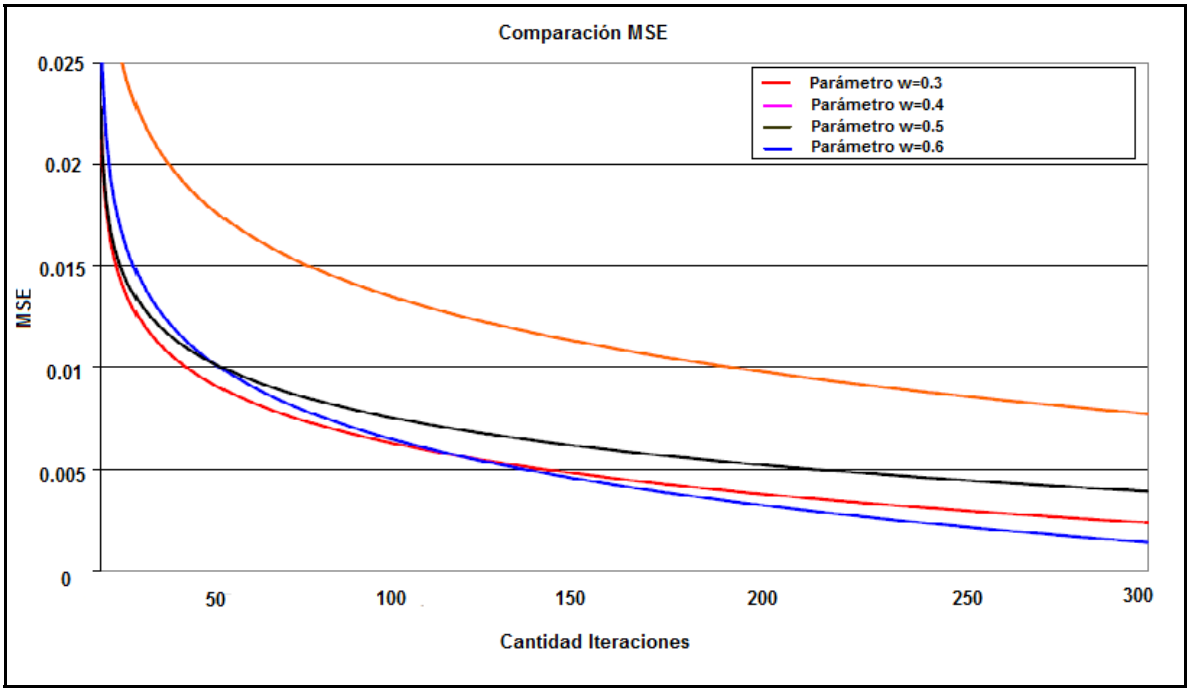


Figura 35: Comparación del error para parámetro *w*

Parámetros *r1 y r2*

Para el ajuste de estos parámetros se definieron los valores indicados en la tabla 28:

Parámetro	1	2	3	4
r1 y r2	[1, 1]	[2, 0.75]	[0.75, 2]	[1.75, 0.75]

Tabla 28: Configuraciones para parámetros *r1 y r2*

Los resultados obtenidos modificando los parámetros indicados en la tabla 29.

r1 y r2	μ MSE	σ MSE
[1, 1]	4,86E-03	4,50E-03
[2, 0.75]	1,16E-05	1,21E-05
[0.75, 2]	4,90E-06	5,56E-06
[1.75, 0.75]	5,01E-03	6,26E-03

Tabla 29: Resultados para parámetro *r1 y r2*

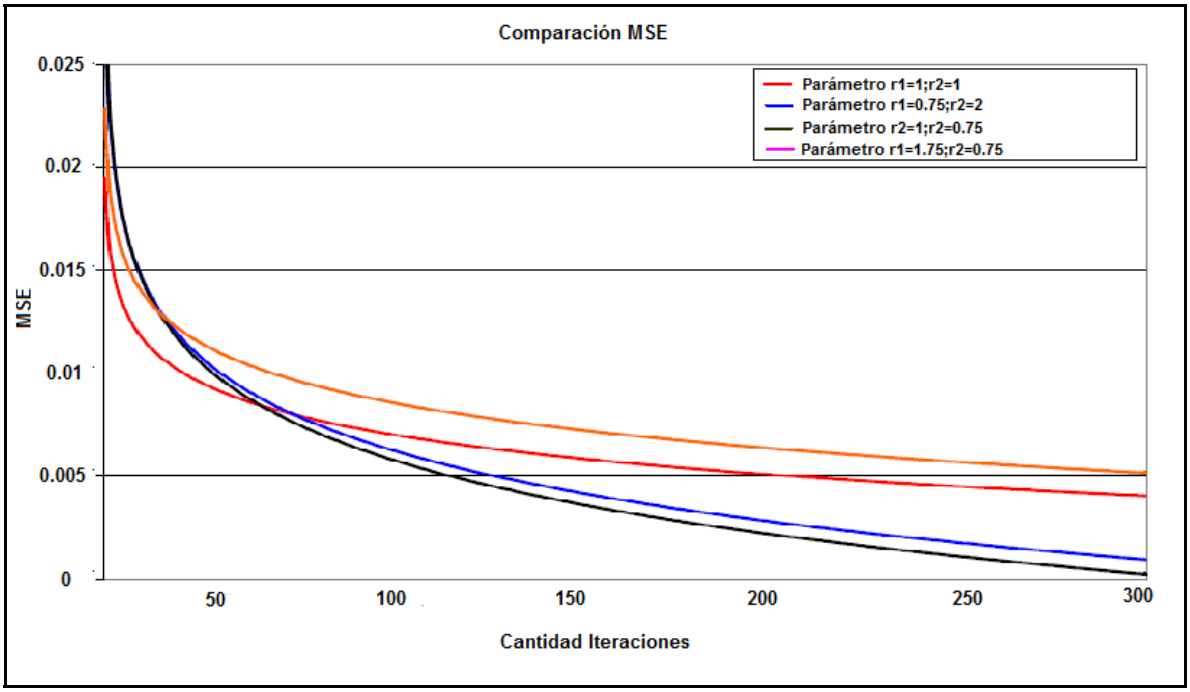


Figura 36: Comparación del error para parámetros *r1 y r2*

5.2.4 Función de Activación

Para aprovechar la capacidad de las redes neuronales de aprender relaciones complejas o no lineales entre variables, se recomienda la utilización de funciones no lineales al menos en las neuronas de la capa oculta. Si bien existen varias funciones disponibles (ej. funciones de base radial), se utilizará una función sigmoidal del tipo tangente hiperbólica como función de activación en las neuronas de la capa oculta, si se utiliza otro tipo de función, por la naturaleza de los datos, la red se muestra incapaz de aproximar la función y diverge.

La elección de la función de activación en las neuronas de la capa de salida dependerá del tipo de tarea impuesto. Dado que el problema es de clasificación, las neuronas normalmente toman la función de activación sigmoidal, al contrario de las tareas de predicción o aproximación de una función, en las cuales generalmente las neuronas toman la función de activación lineal.

5.2.5 Experimentos de Ajustes de Nodos de Capa Oculta

En la sección anterior se realizó un análisis de los parámetros del algoritmo de aprendizaje; sin embargo, en todos los ensayos se utilizó un número de neuronas de capa oculta igual a dos como estructura de ajuste. Por lo tanto, es necesario sensibilizar la estructura con el fin de encontrar una cantidad de nodos ocultos ideal para mejorar el nivel de clasificación. Para esto, se realizó un proceso de análisis para determinar la cantidad de nodos, utilizando el porcentaje de acierto en la clasificación.

En la tabla 30 se resumen los resultados obtenidos al aplicar la metodología de ajuste de parámetros, donde se muestra la evolución del porcentaje de acierto, tanto para el conjunto de entrenamiento como para el conjunto de test, respecto al mejor valor obtenido para el número de nodos de la capa oculta.

		Entrenamiento	Prueba
Nodos Capa Oculta	Partículas	% Acierto	% Acierto
2	30	85,8	84,5
3	**30**	**86,2**	**86,1**
4	30	85,8	85,9
5	30	84,8	84,6
6	30	83,5	82,8
7	30	86,8	83,5
8	30	85,5	82,4
9	30	84,4	82,1
10	30	83,5	81,6

Tabla 30: Resultados configuración de nodos capa oculta

Conclusiones

El mayor porcentaje de aciertos sobre el conjunto de entrenamiento se obtuvo con el modelo que utiliza tres nodos para la capa oculta, configuración que obtiene como resultado un 86,2% de acierto en la predicción. Se puede observar que al aumentar el valor del número de nodos en la capa oculta se genera una disminución en el porcentaje de aciertos sobre el conjunto de prueba. Esto se debe ya que al penalizar en forma excesiva los errores del conjunto de entrenamiento, el modelo se sobreajusta produciendo una disminución en la generalización del modelo.

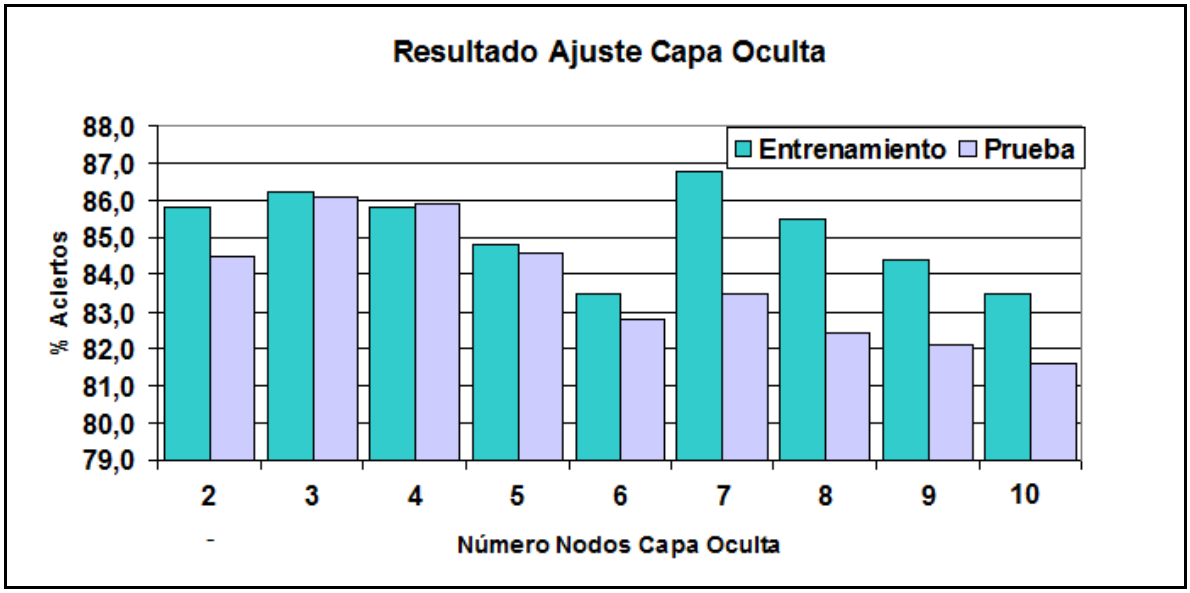


Figura 37: Comparación de cantidad de aciertos

CAPÍTULO 7

Evaluación y discusión de resultados

Luego de haber presentado la técnica de minería de datos a utilizar y el diseño e implementación de esta solución, es hora de verificar si los resultados son los esperados, es decir, si se ha logrado realizar una buena clasificación de los clientes en nivel aceptable. Para ello se realiza la simulación de la mejor red neuronal, real y complejo, una vez optimizados y fijos todos los parámetros del mismo. En la tabla 30 se muestra la mejor configuración de la red neuronal a utilizar.

Parámetros	Valor
Inicialización (Pesos randómicos)	[-0,5 0,5]
c1 y c2	[2, 0.75]
w	0.6
Vmax	0.15
r1 y r2	[0.75, 2]
Número de Iteraciones	300
Número capas capas ocultas	1
Número nodo ocultos	3
Función activación capa oculta	$\tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$
Función activación capa salida	$sig(x) = \frac{1}{1 + e^{-x}}$

Tabla 31: Mejor Configuración de Red Neuronal Evolutiva

Los resultados de las simulaciones se presentan en forma de extraer el mayor conocimiento posible acerca del comportamiento de las redes neuronales y especialmente del algoritmo de aprendizaje utilizado en su entrenamiento, su eficiencia, eficacia y rendimiento para este tipo de problema.

Para el análisis de los resultados arrojados por la simulación se examina los índices de sensibilidad, especificidad y eficacia, y del análisis de curvas ROC.

7.1 Rendimiento de Red Neuronal

El modelo de red neuronal finalmente seleccionado obtuvo buenos resultados a partir del conjunto de pruebas. Así, estableciendo la función umbral con un punto de corte igual a 0,5 en la salida de la red, las salidas bajo este valor eran consideradas como “pólizas renunciadas” y las mayores “pólizas vigentes”. En

términos de porcentaje, la sensibilidad, especificidad y eficacia de la red fueron todos del 86.1%. Por su parte, el área total bajo la curva ROC dio como resultado 0.8345

Como el proceso de aprendizaje neuronal usado es la minimización del error cuadrático medio total, el entrenamiento en el que existen muchas muestras de la clase "póliza vigente"' y muy pocas de la clase "renunciada" no es aconsejable, ya que la clase "póliza vigente" sería predominante en este proceso de aprendizaje y, lógicamente, la red se entrenará más en el sentido de minimizar el error de la "póliza vigente" que del "póliza renunciada". Como se puede observar en la tabla 31, solamente se alcanza un 50,1% de porcentaje de éxito en la clasificación de la clase "póliza renunciada", lo cual claramente demuestra que el entrenamiento y aprendizaje estuvo cargado hacia la clase que predomina.

Matriz de Confusión

Estado de Pólizas	Pronosticado		
	Renunciada	Vigente	% acierto
Renunciada	210	209	50,1%
Vigente	56	1431	96,2%
Porcentaje total	14,0%	86,0%	86,1%

Tabla 32: Matriz de Confusión

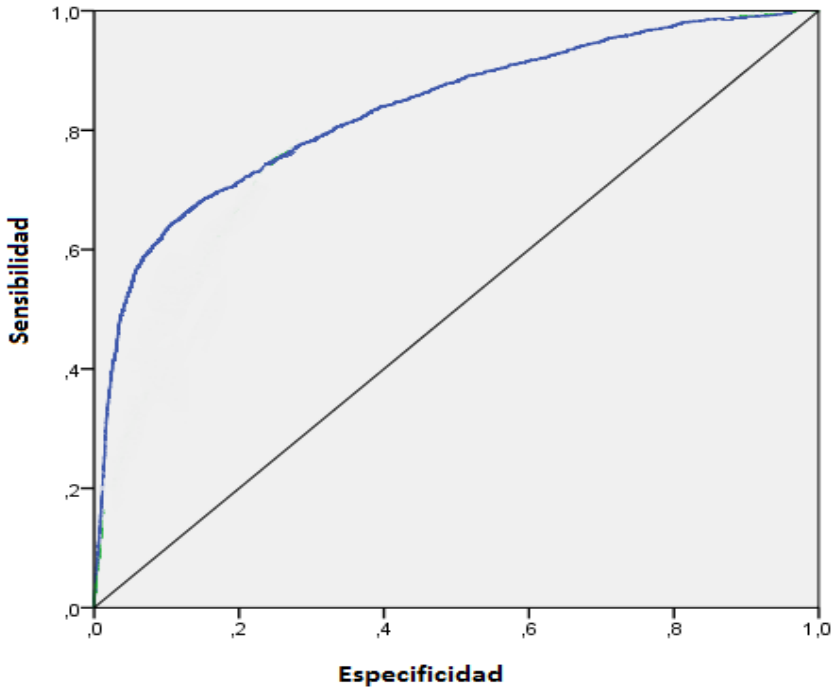


Figura 38: Curva ROC

Para el problema descrito anteriormente existen dos soluciones para solucionar este problema: se puede reducir la cantidad de muestras de la clase más grande (perdiendo la capacidad de generalización de la red), o bien se duplica las muestras de estructuras regulares defectos la clase más pequeña disminuyendo así la tendencia de predominancia de la clase más grande [37].

Dado los resultados entregados, existe un buen nivel en la clasificación de los clientes por parte del modelo de red neuronal; sin embargo existe una diferencia en los clientes marcados con su póliza renunciada. Al modificar los datos de trabajo, según la recomendación de duplicar la cantidad de datos de la clase "póliza renunciada" y disminuir los datos de la clase "póliza vigente" se obtienen los resultados de la tabla 32. Al analizar se ve que mejoró considerablemente el porcentaje de acierto en la clasificación de la clase "póliza renunciada" teniendo un área total bajo la curva ROC de **0.8810**; sin embargo, el porcentaje global de aciertos disminuyó al perder generalización de la red, quedando en un **80,8%.**

Matriz de Confusión Training			
	Pronosticado		
Clases	0	1	% acierto
0	2436	1040	70,1%
1	402	4032	90,9%
Porcentaje total	35,9%	64,1%	81,8%

Matriz de Confusión Testing			
	Pronosticado		
Clases	0	1	% acierto
0	600	264	69,4%
1	126	1036	89,2%
Porcentaje total	35,8%	64,2%	80,8%

Tabla 33: Matriz de Confusión 2

Analizando los resultados obtenidos, se revisa un mal rendimiento en la clasificación de la clase "póliza renunciada" (cercano a un 70% de acierto). De todas maneras, al revisar las curvas ROC y el área bajo la curva, se tiene un buen desempeño (área de 0.8810), que si bien no es cercana a 1, se considera aceptable dentro del marco del problema a estudiar. Como se mencionó anteriormente los resultados se explican por la baja cantidad de muestras de la clase "póliza renunciada", por lo que el entrenamiento y validación se "cargan" a la clase con más datos representativos.

7.2 Extracción de reglas del modelo de red neuronal

El algoritmo para la extracción de reglas desde el modelo de red neuronal elegido es una adaptación del *M of N*, por lo tanto, para la aplicación se tiene como entrada la red neuronal entrenada explicada en el punto anterior y la matriz de pesos sinápticos de la capa oculta.

Nodos	**1**	**2**	**3**	**Nodos**	**1**	**2**	**3**
1	0,25795	0,58304	-0,36156	**20**	-0,10236	-0,34412	-0,01535
2	0,85947	0,02515	0,63462	**21**	-0,28207	-0,14767	-0,81634
3	0,35821	0,15299	0,54478	**22**	-0,40446	0,12785	0,64394
4	0,66038	-0,04528	0,64528	**23**	-0,07014	-0,34265	0,02942

5	0,66364	-0,05909	0,48182	**24**	0,86635	0,05269	0,62195
6	-0,21313	-0,05882	0,56471	**25**	0,99061	-0,34412	0,96484
7	-0,03636	0,44318	-0,36174	**26**	0,5473	-0,14767	0,31081
8	0,85086	0,05901	-0,2681	**27**	0,02312	0,12785	0,91911
9	0,80648	0,04576	-0,43107	**28**	0,77273	-0,34265	0,91036
10	0,65217	-0,34412	0,36217	**29**	0,54559	0,05269	0,36705
11	0,31889	-0,14767	-0,19277	**30**	0,89263	0,98019	-0,06256
12	0,52632	0,12785	-0,1209	**31**	0,4766	0,84222	0,27639
13	0,39468	-0,34265	0,08764	**32**	0,05405	0,2973	-0,18599
14	0,59784	0,05269	0,44303	**33**	0,2381	0,09524	0,40476
15	0,47968	-0,34412	0,31409	**34**	0,65574	0,01639	0,85246
16	0,60586	-0,14767	-0,51171	**35**	0,12067	-0,35924	0,69091
17	0,6623	0,12785	-0,26569	**36**	0,66667	0,0119	0,82143
18	0,80882	-0,34265	-0,4022	**37**	0,87778	0,00222	0,83556
19	-0,17036	0,05269	-0,65158	**38**	0,34545	-0,03636	0,29091

Tabla 34: Matriz de pesos sinápticos

Aplicación de algoritmo de agrupamiento

Al aplicar este algoritmo se combina valores cercanos de los pesos sinápticos y se actualiza con sus promedios. Se utiliza el algoritmo *K-means* con 3 *clusters* definidos para este propósito para cada una de los pesos de las conexiones. El resultado de este proceso es el siguiente:

Centros de Cluster			
	Cluster		
	1	2	3
Nodo 1	0,01329	0,49066	0,65420
Nodo 2	0,06645	0,06494	0,25284
Nodo 3	0,26334	0,60117	0,23860

Tabla 35: Centros obtenidos de algoritmo *K-means*

Nodo de Entrada	Cluster	Distancia	Nodo de Entrada	Cluster	Distancia
7	1	0,521	**26**	2	0,307
11	1	0,324	**27**	2	0,597
13	1	0,587	**28**	2	0,502
19	1	0,446	**29**	2	0,268
20	1	0,39	**33**	2	0,358
21	1	0,632	**34**	2	0,311
23	1	0,411	**35**	2	0,481
32	1	0,374	**36**	2	0,292
2	2	0,381	**37**	2	0,458
3	2	0,261	**38**	2	0,344

4	2	0,176	**1**	3	0,53
5	2	0,21	**8**	3	0,278
6	2	0,705	**9**	3	0,321
10	2	0,401	**12**	3	0,214
14	2	0,224	**16**	3	0,487
15	2	0,401	**17**	3	0,128
22	2	0,917	**18**	3	0,637
24	2	0,394	**30**	3	0,785
25	2	0,678	**31**	3	0,803

Tabla 36: Cluster definidos para los pesos

Extracción de reglas.

En esta etapa se identifica qué grupos de pesos no tienen efecto sobre la entrada total de activación de cada neurona de la red. Los grupos que no exceden el *umbral* definido (ejemplo: el *bias* o el promedio de pesos en el caso de la red evolutiva) de la neurona son eliminados. Se asume que la neurona tiene una máxima activación con un valor cercano a 1 o una inactivación con un valor cercano a 0; por lo tanto, cada entrada a la neurona puede interpretarse como una función booleana. De ahí que la deducción se reduce a determinar la situación en la cual la regla es verdadera.

De los grupos obtenidos, se debe dividir en dos subconjuntos correspondientes a uno con los pesos positivos y otros con los pesos negativos. Finalmente, dado α como la entrada de activación al conjunto de grupos enviados a la neurona, se definen el valor máximo y mínimo de activación por las expresiones:

$$a_{\min} = \sum_{j=1}^{n} \min\{w_j, 0\} \; ; \; a_{\max} = \sum_{j=1}^{n} \max\{w_j, 0\}$$

donde w_j corresponde a los pesos de entrada de la neurona.

Para cada neurona de la capa oculta se realizan los siguientes pasos:

- Si $a_{\max}$ es menor que el promedio, entonces eliminar todos los grupos.
- Extraer un conjunto de grupos positivos *P* con la condición que la suma de los pesos de la entrada w_p sea mayor al promedio.
- Para cada conjunto de datos encontrado:
  - Calcular la entrada de activación $a = a_{\max} + w_p$
  - Si a es mayor al promedio, entonces construir reglas para el grupo y continuar con el siguiente conjunto.
- Extraer un conjunto de grupos negativos *N* con la condición que w_n sea menor que a menos el promedio, donde w_n es la suma de los pesos de N.
- Con cada grupo *N* crear una regla para el grupo positivo y negativo.
- Al final se debe obtener reglas de la forma: SI *umbral* < promedio_pesos * numero_aciertos (Entradas) ENTONCES Z

Para la extracción de las reglas, se codifican los nodos de entrada de la siguiente forma:

$$X_i = \{X_1, X_2, X_3 ... X_n\}$$; donde *n*=38 nodos de entrada.

y se verifica qué entrada a la neurona de la capa oculta cumple las condiciones del algoritmo planteado. Luego de realizar los cálculos, al comparar los valores de los nodos agrupados se obtiene la siguiente regla:

- SI $0{,}2 < 0{,}42 * 3(X_5, X_{20}, X_{32}) + 0{,}28 * 2 (X_{10}, X_{25})$ ENTONCES Y

Generalización de la regla.

De la regla obtenida con el procedimiento anterior es posible realiza la generalización de la regla en base a los parámetros de entrada en la forma "*M de N"*:

SI (3 de (Cambio de Asesor, Cliente Huérfano, Pago por Caja) ENTONCES Y

SI (2 de (Meses Pagados, Prima Mensual) ENTONCES Y

Conclusiones.

Realizando el algoritmo de extracción de reglas de la red neuronal, se pudo llegar a la obtención de dos patrones de características de un cliente, del cual se debe tener en consideración para analizar su comportamiento. Al revisar las reglas obtenidas al realizar la generalización, se infiere lo siguiente:

- Si un cliente posee un cambio de asesor, además de ser cliente huérfano y posee modo de pago por caja, entonces ese cliente cierra su póliza.
- Si un cliente tiene meses pagados atrasados y tiene frecuencia de pago mensual, entonces ese cliente cierra su póliza.

7.3 Análisis de sensibilidad de variables

El método más común para realizar un análisis de sensibilidad consiste en fijar el valor de todas las variables de entrada a su valor medio e ir variando el valor de una de ellas a lo largo de todo su rango, con el objeto de observar el efecto que tiene sobre la salida de la red. Siguiendo este método, se fue registrando los cambios que se producían en la salida de la red, cada vez que se aplicaba un pequeño incremento en una variable de entrada. Se propone como objetivo cuantificar la influencia que tiene cada variable de entrada. Según [36] la suma de los cambios producidos proporciona una medida intuitiva de sensibilidad. Esta medida representaría el efecto relativo que tiene una variable de entrada sobre la salida de la red. Así, un valor cercano a 0 indicaría poco efecto o sensibilidad; a medida que se fuese alejando de 0, indicaría que el efecto va aumentando. Esta medida de sensibilidad se obtuvo mediante la siguiente expresión:

$$S_{ik} = \sum_{n}^{N} | X_{kn} - X_{k\min} |$$

donde

S_{ik} = medida de sensibilidad de la variable de entrada *i* sobre la salida *k*

X_{kn} = valor de la salida k obtenido con el incremento *n* en la variable de entrada *i*

$X_{k\min}$ = valor de la salida *k* obtenido con el valor mínimo posible de la variable de entrada *i*

En la tabla 33 se presentan dichos valores ordenados de mayor a menor. Así, los primeros valores de la tabla corresponden a las variables de entrada con más influencia o relación con la salida de la red. Así, se puede observar que las variables que tienen mayor influencia son: la cantidad de meses pagados clientes ABC1, pago por caja, clientes huérfanos, cliente viudo y la prima mensual.

Variable Predictora	Sensibilidad
Meses Pagados	0,095
Cliente ABC1	0,071
Pago por Caja	0,057
Cliente huérfano	0,048
Cliente Viudo	0,045
Cliente Anulado	0,044
Prima Mensual	0,044
Frecuencia de pago anual	0,04
Cambio de Asesor	0,038

Tabla 37: Variables más influyentes

CAPÍTULO 8

Conclusiones y Trabajo Futuro

En este capítulo se resumen las principales conclusiones derivadas del trabajo realizado, en el cual se verifica el cumplimiento de los objetivos propuestos y revisando los principales resultados obtenidos el modelo propuesto. Además, propone el trabajo futuro que se genera a partir de este trabajo.

8.1 Conclusiones del uso de redes neuronales

En este informe se entregó un resumen del marco teórico de la búsqueda del conocimiento a través de distintas técnicas de minería de datos, específicamente, utilizando modelos de clasificación en base a las redes neuronales artificiales, específicamente de las redes MLP, siendo en este caso un algoritmo híbrido ya que su algoritmo de aprendizaje se basó en uno evolutivo conocido como Optimización por Enjambre de Partículas (PSO). De este marco se han obtenido conclusiones que han sido tenidas en cuenta en el diseño de la solución.

De la implementación del algoritmo se pueden sacar conclusiones que ayudan a comprender la forma en que se comporta la red neuronal con aprendizaje basado en PSO:

- Las funciones de activación deben ser acotadas en su dominio, por eso, se utilizan las funciones de tipo sigmoidal; de otro modo, la red se muestra incapaz de aproximar la función y diverge.
- La curva de aprendizaje de la red oscila fuertemente en las primeras iteraciones, producto del cambio del vector de pesos (partícula) en el entrenamiento de la red.
- La red neuronal realiza una aproximación de tipo global de la transformación entrada/salida, lo que favorece su capacidad de generalización y permite una mayor eficacia en la capacidad de clasificación. Esto es producto a que el algoritmo de aprendizaje es evolutivo e independiente de la derivada. En cambio, si se utiliza una la red neuronal con aprendizaje con retropropagación, se debe utilizar funciones de activación dependientes de la derivada, lo que realiza una aproximación de tipo local y esto limita su capacidad de generalización.
- Las redes neuronales tienen la desventaja sobre a otros métodos de clasificación en el ajuste complejo de los parámetros para encontrar la mejor configuración de la red y la forma de extraer las reglas de clasificación desde la red entrenada.
- La utilización de una adaptación del algoritmo *M de N* para realizar la extracción de reglas, permitió conocer un patrón de características de cliente que es importante tomar en cuenta a la hora de realizar el análisis.

8.2 Conclusiones del negocio

El modelo propuesto es una buena herramienta en el apoyo de la toma de decisiones para la identificación de los posibles clientes que cerrarán sus pólizas, por ende, serán clientes fugados. Con la detección de este tipo de cliente se pueden hacer más efectivas las políticas de retención. El analista de negocio puede realizar las siguientes tareas utilizando el modelo entregado:

- Utilizar modelos de predicción de fuga de clientes de manera regular, de forma de tener un comportamiento proactivo al problema de cierre de pólizas.
- Entregar a los asesores de pólizas un listado de sus clientes con mayores características de cierre de póliza para aplicar las políticas de retención apropiadas a ese cliente.
- Complementar la información entregada por los modelos de predicción de fuga con un análisis multidimensional en los *data marts* de la compañía con la información histórica de los clientes.
- Eliminar del modelo las variables con menos implicancia en el comportamiento de cliente en el cierre de sus pólizas.

8.3 Trabajo futuro

- Entregar al analista de negocio una interfaz de usuario amigable, en la cual pueda cargar rápidamente los datos y obtener los datos de las variables más significativas para su uso.
- Mejorar aspectos claves tales como velocidad, carga computacional y precisión de la convergencia en la red neuronal; asimismo minimizar el número de parámetros de la red, de manera de que su implementación no resulte desfavorablemente costosa.
- Cambiar el algoritmo de aprendizaje a retropropagación para ver el comportamiento de la red y comparar los resultados a nivel de velocidad de convergencia y resultados obtenidos.
- Utilizar otra topología de red neuronal, como por ejemplo, red recurrente del tipo Elman o Jordan, para comparar con los resultados obtenidos con la red feedforward en temas de precisión de los resultados.
- Comparar los resultados obtenidos con otro tipo de clasificadores, como por ejemplo máquinas de soporte vectorial, clasificadores neurodifusos u otros clásicos como regresión logística o árboles de decisión.

8.4 Conclusiones Personales

La enseñanza impartida por el programa de Magíster, enfocada al ámbito profesional, permitió dirigirse al problema, indicado como génesis de este trabajo, desde una perspectiva analítica y llevarlo de forma estructurada y seria a la vista del usuario de negocio. El curso de “Inteligencia de Negocios” permitió elegir una línea de estudio y profundizar más en los temas relacionados en descubrimiento del conocimiento basado en el análisis de datos.

Como el trabajo tiene una fuerte base matemática con usos de métodos estocásticos, se destaca la importancia del ramo de “Redes Neuronales Evolutivas” que contribuyó a encontrar una buena herramienta para desarrollar la investigación. También el programa contaba con asignaturas enfocadas al trabajo tales como “Sistemas de Apoyo a la Planificación Estratégica”, que permitió conocer la importancia de la información en la realidad empresarial del siglo XXI y en la toma de decisiones que justifica, desde el punto de vista de negocio, la necesidad de tener datos reales para realizar acciones en la empresa. Asimismo se destaca la enseñanza de las series de tiempo enfocadas al ámbito profesional aplicadas al pronóstico en los negocios.

A nivel personal, el trabajo realizado permitió exponer en la empresa abrir una nueva línea de negocio dirigida a la realización de proyectos de inteligencia de negocios, aplicando los conocimientos adquiridos por el programa y poder gestionarlos de mejor manera.

Referencias Bibliográficas

[1] Superintendencia de Valores y Seguros: *Estructura del Mercado Asegurador* [En línea]. Disponible en http://www.svs.cl, 2011

[2] A. Athanassopoulos: *Customer satisfaction cues to support market segmentation and explain switching behavior*. Journal of Business Research, 47:197, 2000.

[3] F. Reichheld and E. Sasser. *Zero defections: Quality comes to services*. Harvard Business Review, 105:111, 2000.

[4] U. Fayyad. *Data mining and knowledge discovery: Making sense out of data*. IEEE Expert-Intelligent Systems & Their Applications, 11:20 25, 1996

[5] Dorian Pyle*, Business Modeling and Data Mining*, Ed. Morgan Kaufmann, 2003

[6] Molinero Luis. *Construcción de modelos de regresión multivariantes*. Alce Ingeniería Abril 2002.

[7] Paralic, J. Andrássyová, E *Intelligent Knowledge Discovery* Dept. of Cybernetics and Artificial Intelligence.

[8] Pérez López, César, *Econometría de las series temporales*, Pearson Educación S. A., Prentice Hall, Madrid, 2006.

[9] SPSS, *CRISP-DM 1.0 Step-by-step data mining guide*, 2000.

[10] SAS, *Data Mining and the Case for Sampling Solving Business Problems Using SAS® Enterprise Miner™ Software*, 1998.

[11] Azevedo Ana, Santos Manuel Filipe, *KDD, SEMMA AND CRISP-DM: a Parallel Overview*. IADIS, ISBN: 978-972-8924-63-8, 2008

[12] Yu Zhao , Bing Li, Xiu Li, Wenhuang Liu and Shouju Ren, *Customer Churn Prediction Using Improved One-Class Support Vector Machine*. Cims Research Center, Automation Department, Tsinghua University, Beijing 100084, China.

[13] Gang Wang, Erik Goodman, William Punch, *On the Optimization of a Class of Blackbox Optimization Algorithms.* GARAGe/Case Center Technical Report, 1997.

[14] McCulloch, W.S., Pitts, W. *A logical calculus of the ideas immanent in nervous activity*, Bulletin of Mathematical iophysics, Vol. 5 pag. 115-133, 1943.

[15] Hebb, D.O. *Organization of behavior*. New York, Editorial Science Editions, 1949.

[16] Widrow, B. Adaptative *sampled-data systems, a statistical theory of adaptation*. IRE WESCON Convention Record, parte 4. New York, Institute of Radio Engineers, 1959.

[17] Rosemblatt, R. *Principles of neurodynamics*. New York, Editorial Spartan Books,
1959.

[18] Minsky, M.L., Papert, S. *Perceptrons*. Cambridge MA, Editorial MIT Press, 1969.

[19] Rumelhart D.E., McClelland, J.L. & Group, *Parallel Distributed Processing. Explorations in the Microstructure of Cognition*. Cambridge, MA: MIT Press, 1986.

[20] A. Abraham, H. Guo, H. Liu, *Swarm Intelligence: Foundations, Perspectives and Applications*, Studies in Computational Intelligence (SCI), Springer-Verlag, vol. 26, pages 3-25, November 2006.

[21] Widrow B., Hoff M.E., *Adaptive switching circuits. En 1960 IRE WESCON Convention Record*, pag. 96-104, New York, 1960.

[22] Rosenblatt, F., *The Perceptron: A probabilistic Model ,for information storage and organization in the brain*, Psycological Review, 1958.

[23] Werbos Paul. *Beyond Regresion: New tools for prediction and Anayisis in the behavioral Sciences*. Harvard, Cambridge, 1974.

[24] A. Khosla, S. Kumar, K.K. Aggarwal, J. Singh, *Particle Swarm for Fuzzy Models Identification*, Studies in Computational Intelligence (SCI), Springer-Verlag, vol 26, pages 149-173, November 2006.

[25] Álvarez Nelson, Crawford Broderick, *Optimización de funciones a través de Optimización por Enjambre de Partículas y Algoritmos Genéticos.* Pontificia Universidad Católica de Valparaíso, Escuela de Ingeniería Informática, Valparaíso, Chile, 237-1099.

[26] J. Kennedy, R.C. Eberhart, *Particle Swarm Optimization*, Proceedings of the IEEE International Conference on Neural Networks, vol. 4, pages 1942-1948, December 1995.

[27] J. Kennedy, R.C. Eberhart, Y. Shi, *Swarm Intelligence*, The Morgan Kaufmann Series in Artificial Intelligence, Morgan Kaufmann, San Francisco - California, pages 3-80, April 2001.

[28] J.R. Pérez, *Contribución a los Métodos de Optimización Basados en Procesos Naturales y su Aplicación a la Medida de Antenas de Campo Próximo*, a Thesis presented to the University of Cantabria, October 2005.

[29] J. Liu, J. Sun, W. Xu, *Quantum-Behaved Particle Swarm Optimization with Adaptive Mutation Operator*, Lecture Notes in Computer Science (LNCS), Springer-Verlag, vol. 4221, pages 959-967, September 2006.

[30] A. Abraham, H. Guo, H. Liu, *Swarm Intelligence: Foundations, Perspectives and Applications*, Studies in Computational Intelligence (SCI), Springer-Verlag, vol. 26, pages 3-25, November 2006

[31] Serrano Agustín, *Aplicación de las redes neuronales artificiales a la predicción del resultado del trasplante renal pediátrico*, Universisd de Valencia, 2004

[32] B. Mittal and W.A. Kamakura. *Satisfaction, repurchase intent, and repurchase beha- vior: investigating the moderating effect of customer characteristics*. Journal of Marketing Research, 1(131{142), 2000.

[33] M.G. Dekimpe and Z. Degraeve. *The attrition of volunteers*. European Journal of Operational Research, 98(1):37{51, 1997.

[34] C.I.Mosier. *Problems and designs of cross-validation*. Educational and Psychological Measurement, 11:6 11, 1951.

[35] T. Bartz-Beielstein, D. Blum, J. Branke, *Particle Swarm Optimization and Sequential Sampling in Noisy Environments*, Operations Research/Computer Science Interfaces, Springer US, pages 261-273, August 2007

[36] Palmer Pol, A. Montaño Moreno, *Predicción del consumo de éxtasis a partir de redes neuronales artificiales*, Facultad de Psicología. Universidad de las Islas Baleares

[37] Haykin *S, Neural networks: a comprehensive foundation* Englewood Cliffs, New Jersey: IEEE Press 1994

[38] R. Andrews, J. Diederich, and A. B. Tickle*, Survey and critique of techniques for extracting rules from trained artifcial neural networks,* Knowledge-Based Systems, vol. 8, no. 6, pp. 373:389, 1995