Universidad de Lima

Facultad de Ingeniería y Arquitectura

Carrera de Ingeniería de Sistemas

DATA MINING PARA MODELO PREDICTIVO DE VENTAS Y SERVICIOS DE MANTENIMIENTO EN UN CONCESIONARIO AUTOMOTRIZ LIGERO

Tesis para optar el Título Profesional de Ingeniero de Sistemas

Jocelyn Pamela Becerra Rojas

Código 20131602

Enrique Martin Villarreal Roca

Código 20132359

Asesor

Juan Manuel Gutiérrez Cárdenas

Lima – Perú

Octubre de 2021

UNIVERSIDAD DE LIMA
MCMLXV
SCIENTIA ET PRAXIS

APPLYING DATA MINING IN PREDICTIVE MODEL FOR LIGHT CAR SALES AND MAINTENANCE SERVICES IN A DEALERSHIP BUSINESS

TABLA DE CONTENIDO

RESUMEN ..... **x**

ABSTRACT ..... **xi**

INTRODUCCIÓN **1**

CAPÍTULO I: PLANTEAMIENTO DEL PROBLEMA **3**

1.1. Formulación del problema 3

1.2. Objetivo de la investigación 6

1.3. Justificación 7

CAPÍTULO II: ESTADO DEL ARTE **9**

CAPÍTULO III: MARCO TEÓRICO **23**

3.1. Análisis Predictivo 23

3.2. Modelo Predictivo 23

3.3. Data Mining 24

3.4. Árboles de Decisión 25

3.5. Bosque Aleatorio (Random Forest) 26

3.6. Árbol de clasificación y regresión (CART) 27

3.7. Análisis de Componente Principal (PCA) 28

3.8. Redes Neuronales 29

3.9. Multilayer Perceptron 29

3.10. Validación Cruzada (Cross Validation) 30

3.11. Hyperparameter Tuning 31

3.12. Algoritmo de Retropropagación (Back Propagation) 32

3.13. Método de optimización Grid Search 33

CAPÍTULO IV: DESARROLLO DE LA SOLUCIÓN PROPUESTA **34**

4.2. Alcance 48

CAPÍTULO V: PRUEBAS Y RESULTADOS **49**

5.1. Predicción de Ventas 49

5.2. Asistencia al servicio de Mantenimiento 64

CONCLUSIONES **72**

RECOMENDACIONES **74**

REFERENCIAS **75**

ANEXOS 79

ÍNDICE DE TABLAS

Tabla 4.1 Variables Propuestas para la investigación....36
Tabla 4.2 Interpretación de resultados....37
Tabla 4.3 Matriz de correlación obtenida....37
Tabla 4.4 Variables resultantes obtenidas....38
Tabla 4.5 Coeficientes de cada dimensión....39
Tabla 4.6 Parámetros Adicionales....49
Tabla 5.7 Resultados del primer enfoque....50
Tabla 5.8 Resultados: Entrenamiento enfoque 2....51
Tabla 5.9 Resultados: Pruebas enfoque 2....52
Tabla 5.10 Mejores resultados después de afinamiento....57
Tabla 5.11 Análisis de regresión lineal múltiple - Correlación....62
Tabla 5.12 Resultados sin afinamiento....63
Tabla 5.13 Resultados con afinamiento....64
Tabla 5.14 OBB Error con 4 variables....64
Tabla 5.15 Matriz de confusión - Entrenamiento (Random Forest)....65
Tabla 5.16 Matriz de confusión - Pruebas (Random Forest)....65
Tabla 5.17 Grid search basado en Out-of-bag-error....67
Tabla 5.18 Grid search basado en validación cruzada....67
Tabla 5.19 Matriz de confusión....67
Tabla 5.20 Matriz de confusión - Pruebas (Árbol de clasificación)....69
Tabla 5.21 Matriz de confusión - Pruebas (Árbol de clasificación con podado)....70
Tabla 5.22 Resultado de la precisión de los modelos desarrollados....71

ÍNDICE DE FIGURAS

Figura 1.1 Histórico de Cantidad de Ventas a nivel nacional mensualizado (2013-2017), (Proporcionado por la Organización) 4
Figura 1.2 Cantidad de Atenciones del Servicio Taller por Servicio en km (2016) (Proporcionado por la Organización) 5
Figura 2.3 Modelo de Red Neuronal Deep Learning 13
Figura 2.4 Modelo PCA - BP Neuronal Network 15
Figura 2.5 Metodología Revisada 19
Figura 3.6 Técnicas de Data Mining 25
Figura 3.7 Modelo de entrenamiento basado en árboles de decisión 27
Figura 3.8 Modelo de una capa oculta 30
Figura 3.9 Modelo de automatización de afinamiento 31
Figura 3.10 Interpretación geométrica del rol de la unidad oculta 32
Figura 3.11 Distribución de resultados del Método de GridSearch 33
Figura 4.12 Diagrama de estructura de la investigación 34
Figura 4.13 Diagramas de estructuras de los modelos de predicción de ventas 35
Figura 4.14 Histograma resultante 38
Figura 4.15 Funciones de Activación 43
Figura 4.16 Diagramas de estructuras de los modelos de asistencias de servicios de mantenimiento 44
Figura 4.17 Data set centralizado de los servicios de mantenimientos 46
Figura 5.18 Entrenamiento Enfoque 1 50
Figura 5.19 Pruebas enfoque 1 51
Figura 5.20 Validación cruzada 53
Figura 5.21 Entrenamiento Corr + BP 54
Figura 5.22 Entrenamiento PCA + BP 54
Figura 5.23 Pruebas Corr + BP 54
Figura 5.24 Pruebas PCA + BP 55
Figura 5.25 Pseudocódigo planteado para el modelo inicial 56
Figura 5.26 Pseudocódigo Planteado para el afinamiento de hiperparámetros 56
Figura 5.27 Entrenamiento BP sin afinamiento – 1er Enfoque 57

Figura 5.28 Entrenamiento BP con afinamiento – 1er Enfoque 58
Figura 5.29 Pruebas BP sin afinamiento – 1er Enfoque 58
Figura 5.30 Pruebas BP con afinamiento – 1er Enfoque 59
Figura 5.31 Entrenamiento BP sin afinamiento – 2do Enfoque 59
Figura 5.32 Entrenamiento BP con afinamiento – 2do Enfoque 60
Figura 5.33 Pruebas BP con afinamiento – 2do Enfoque 60
Figura 5.34 Pruebas BP con afinamiento – 2do Enfoque 61
Figura 5.35 Resultados Real Vs Predicción 62
Figura 5.36 Resultados de Durbin Watson 63
Figura 5.37 OOB Error vs el número de árboles 65
Figura 5.38 Curva ROC - Random Forest sin afinamiento 66
Figura 5.39 Importancia de variables predictores 68
Figura 5.40 Árbol de clasificación 68
Figura 5.41 Curva ROC - Árbol de Clasificación sin Poda 69
Figura 5.42 Árbol de clasificación con podado 70
Figura 5.43 Curva ROC - Árbol de clasificación con Poda 70

ÍNDICE DE ANEXOS

ANEXO 1: Diagrama de Pareto - Marcas 80
ANEXO 2: Unidades Vendidas por marca VS Ingresos (USD) 80
ANEXO 3: Servicio de mantenimiento por Marca 81
ANEXO 4: Nro de Servicios de mantenimiento por tipo de KM 81
ANEXO 5: Histórico de servicios de mantenimientos 82

RESUMEN

Últimamente el nivel de competencia entre las empresas del rubro automotriz ligero suele ser muy alto, debido a las diversas estrategias desarrolladas por los competidores. Nuestro estudio busca fortalecer la evaluación de pronósticos que permita mejorar la capacidad de la organización para anticiparse a eventos futuros en los procesos importantes del negocio, tales como las ventas y los servicios de mantenimiento. Para lograr dicho objetivo se consultaron investigaciones relacionadas a técnicas de *Data Mining*, las cuales realizan un análisis de información bajo un enfoque predictivo. El desarrollo de la investigación involucra diseñar diferentes modelos aplicando métodos como regresiones, redes neuronales y árbol de decisión, a una base de datos histórica de una organización automotriz, realizando previamente una selección de datos mediante técnicas como la matriz de correlación y PCA (*Principal Component Analysis*). Finalmente, se realiza una evaluación sobre los resultados obtenidos luego de comparar los modelos planteados, donde encontramos para los pronósticos de ventas, el modelo de redes neuronales implementado con PCA obtiene mejores resultados; mientras que, para los pronósticos de servicios de mantenimiento, el modelo predominante es el implementado con *Random Forest.*

Palabras Clave: Modelo predictivo, ventas, servicios de taller, minería de datos, automotriz, aprendizaje profundo, árbol de decisión

ABSTRACT

Lately the level of competition between companies in the light automotive industry is reaching a very high level, due to the various strategies developed by many competitors. Our study seeks to strengthen the evaluation of forecasts to improve the organization's capability to anticipate future events in important business processes, such as sales and maintenance services. To achieve this objective, investigations related to Data Mining techniques were consulted, in order to perform an information analysis with a predictive approach. Our research involves designing different models applying methods such as regressions, neural networks and decision trees, to a historical database of an automotive organization, previously selecting data using techniques such as the correlation matrix and PCA (*Principal Component Analysis*). Finally, an evaluation is carried out on the results obtained after comparing the proposed models, where we find out that for sales forecasts, the neural network model implemented with PCA obtains better results; whereas, for maintenance services forecasts, the predominant model is the one implemented with Random Forest.

Keywords: Predictive model, sales, workshop services, Data Mining, automobile, deep learning, decision tree

INTRODUCCIÓN

En el mundo de las empresas del rubro automotriz encontramos diversas estrategias de negocios que solventan la toma de decisiones de la alta gerencia, aquellas que finalmente marcarán el rumbo que tomará la organización. En función de ello, el uso de las tecnologías de información ha ido transformando por completo este escenario en los últimos años, logrando obtener inferencias más acertadas a partir de un activo presente en todas las organizaciones el cual gana cada vez mayor importancia, estamos hablando de los datos históricos. Esta valiosa información que se obtiene a partir de la operativa diaria del negocio muchas veces no suele aprovecharse por completo, ya que únicamente la utilizan para realizar análisis descriptivos, los cuales sólo resultan útiles para mostrar una visión actual de la organización. Sin embargo, no pueden brindar un enfoque predictivo de cómo podría irle a la organización a corto y mediano plazo en el futuro, algo muy necesario para definir una estrategia de posicionamiento en el mercado.

Siendo las ventas el foco principal de todas las organizaciones con fines de lucro, se vuelve fundamental tener una solución que permita realizar predicciones acerca del nivel de venta de sus productos, ya que de esta manera se podría gestionar con mayor eficiencia aspectos relevantes al negocio, tales como el aprovisionamiento mensual de autos, contratación de personal especializado y elaboración de presupuestos. Asimismo, resulta importante identificar aquellos clientes con alta probabilidad de asistir a su próximo servicio de mantenimiento, puesto que diversos estudios han demostrado que brindar un adecuado servicio de posventa, genera lealtad y fidelidad en los clientes hacia la organización.

Considerando los planteamientos que el negocio ha establecido, este trabajo de investigación basará sus fundamentos en trabajos de autores que hayan realizado investigaciones aproximadas a la problemática descrita. El cual será evaluado en una organización que es parte uno de los grupos empresariales más importantes y con mayor potencial de desarrollo del país en el rubro automotriz. Se realizará un análisis de la información histórica para determinar aquellas variables clave del negocio, las cuales servirán como entrada para diversos modelos de análisis predictivo. Para el diseño y

construcción de dichos modelos se emplearán técnicas de *Data Mining*, tales como Redes neuronales y Árboles de decisión, utilizando herramientas de programación como Python, R y otros. Finalmente, se validará cuál de las metodologías investigadas logró un mejor resultado para nuestro caso de estudio.

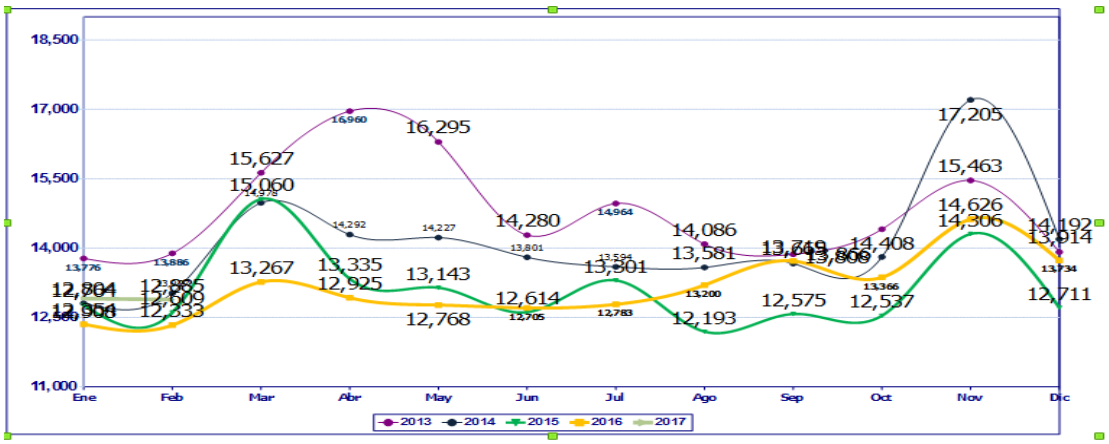
CAPÍTULO I: PLANTEAMIENTO DEL PROBLEMA

1.1. Formulación del problema

En la actualidad existe un elevado nivel de competitividad entre las empresas del rubro automotriz ligero en Lima, en las grandes ciudades, dado que existen diferentes estrategias adoptadas por los competidores, tales como ajustes de precios por segmentación, campañas de ventas dirigidas y facilidades para la adquisición inmediata de un auto. A las empresas que pertenecen al sector, se les han presentado sendas oportunidades de mercado como lo mencionado por la (INEI, 2012) el cual dice, "el parque automotor de Lima - Perú ha venido incrementándose año tras año desde el 2008". Asimismo, revisando artículos y revistas se encontró que, (Gestión, 2017): "Para el año 2017 la venta de vehículos nuevos estaría alrededor de las 180,000 unidades, cerca de 5% más respecto al año 2016, proyectó el banco Scotiabank". Tomando como premisa los datos investigados, todas las empresas que compiten en el mercado están en la necesidad de destacarse para tener un posicionamiento ventajoso en el mercado en el que se desenvuelven. Dentro de la organización en la cual se apoyará nuestra investigación, perteneciente al rubro mencionado, se identifica un problema relacionado con la disminución del nivel de ventas respecto a otros años, en los cuales se realizó una mejor performance en el mercado. Si nos guiamos de la información consultada del incremento del parque automotor de Lima, las posibles ventas de autos que se pronosticaron debieron haber incrementado el nivel de ingresos de la organización proporcionalmente a dicho pronóstico. Al parecer las estrategias comerciales aplicadas no fueron las adecuadas y eso se ha reflejado en la disminución del nivel de ventas de la organización en años anteriores. En la Figura 1.1, se puede apreciar que las ventas han ido descendiendo respecto a años anteriores.

Figura 1.1

Histórico de Cantidad de Ventas a nivel nacional mensualizado (2013-2017), (Proporcionado por la Organización)

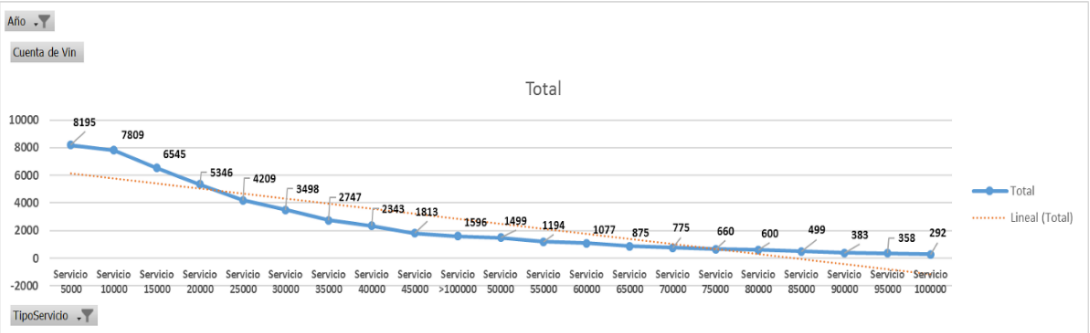


Si bien dentro de la organización cuentan con modelos descriptivos de los cuales se generan indicadores con el propósito de brindar una idea de la situación actual de la empresa, al no tener un modelo predictivo están dejando de lado muchas oportunidades ya que no se pueden anticipar a situaciones futuras que estén relacionadas al proceso de venta.

Si nos adentramos más en la organización, encontramos otro problema. Este se relaciona con el servicio de posventa. A partir de la revisión de la data histórica de los clientes que asisten al servicio de taller, se identificó una disminución notoria en el nivel de clientes mensuales. La clasificación de este servicio se da a través de la cantidad de kilometraje que un auto acumula luego de su venta. La figura 1.2 muestra que la mayoría de los clientes asisten a su servicio de mantenimiento dentro de los primeros kilómetros de recorrido de su vehículo, mientras más kilómetros se acumule suele ser menor el interés por parte del cliente de llevar su auto a mantenimiento. Bajo este enfoque, la organización busca constantemente poder identificar a aquellos clientes que se encuentren dispuestos a llevar su auto al servicio de mantenimiento para ofrecer promociones personalizadas y fidelizarlos. Sin embargo, la organización hasta principios del 2017 aún no contaba con ninguna técnica ni método para lograr lo comentado.

Figura 1.2

Cantidad de Atenciones del Servicio Taller por Servicio en km (2016) (Proporcionado por la Organización)



Una de las soluciones que la organización ha encontrado es la de tercerizar la realización del análisis predictivo. Sin embargo, el proveedor que realiza este servicio no cubre las necesidades predictivas de la organización, por ejemplo, no brinda un pronóstico del nivel de ventas de un punto de venta, lo cual es algo que se hace necesario para poder saber a qué local brindarle mayor atención por sobre los demás.

Si bien el proveedor contempla el hallazgo de aquellos clientes clave para el servicio de mantenimiento, existen algunos otros inconvenientes con la tercerización del servicio de predicciones. Uno de ellos es el tiempo de respuesta ofrecido por el proveedor, este suele ser muy extenso y dado que un modelo predictivo necesita siempre estar alimentado por datos nuevos, suelen presentarse desfases entre la información enviada y los datos utilizados por el proveedor generando predicciones con un menor nivel de acierto. Asimismo, la confidencialidad de los datos, el cual como se mencionó es uno de los activos más importantes de la empresa, puede verse comprometido, ya que al proveedor encargado de realizar las predicciones se le brinda toda la información de las bases de datos que maneja la empresa. Finalmente, el aspecto que más impacta quizá al trabajar con una tercerización del servicio es que su costo resulta muy elevado, esto es justificado por la empresa encargada ya que se considera que la elaboración de predicciones de venta para la organización es muy complicada de realizar.

Se considera que la solución para poder resolver los problemas que se han indicado anteriormente debería ser la adopción de un sistema basado en técnicas de *Data Mining*, el cual sería gestionado por la propia organización con la capacidad de poder generar predicciones tanto para el del nivel de ventas, como para la problemática encontrada en el servicio de mantenimiento que ofrece la organización a sus clientes. Los beneficios obtenidos como resultado de esta investigación irían desde mejoras en la gestión operativa como el aprovisionamiento mensual de autos, contratación de personal especializado y elaboración de presupuestos; hasta la posibilidad de incrementar la cantidad clientes fidelizados mediante el servicio de posventa.

1.2. Objetivo de la investigación

1.2.1. Objetivo general

Elaboración de modelos predictivos a través de técnicas de *Data Mining* para el análisis de venta y posventa en una organización del rubro automotriz ligero, los cuales permitirán fortalecer la evaluación de pronósticos con el propósito de mejorar la capacidad de esta para anticiparse a diversos eventos futuros.

1.2.2. Objetivos específicos

a) Comparación de metodologías encontradas en investigaciones relacionadas con aplicación en *Data Mining*.

b) Implementar un modelo de red neuronal para la predicción de ventas de un periodo.

c) Implementar un modelo de árbol de decisión para la clasificación de clientes con predisposición a asistir al servicio posventa.

1.3. Justificación

El aporte sustancial de la investigación está en el ámbito empresarial, ya que les permitirá a los miembros de la organización obtener proyecciones de ventas, las cuales servirán para anticiparse a los eventos futuros y de esta manera lograr una ventaja significativa sobre los demás competidores del mercado automotriz.

Considerando el mercado peruano como enfoque para nuestra investigación, las empresas que se encuentran en el rubro automotriz deberían considerar un modelo predictivo en sus organizaciones, que contemple tanto variables directamente relacionadas al negocio como las que no, ya que estas últimas pueden ser determinantes de acuerdo con (Cruz et al., 2017):

> "Al analizar la situación competitiva del sector automotriz en el Perú, no se puede dejar de mencionar la actual situación económica, social y política del país, la misma que determina en gran medida la competitividad del sector.
> Al respecto, el Perú cuenta con una población joven y de crecimiento rápido, tiene apertura comercial hacia el exterior, un PBI y crecimiento económico relativamente estable con un sólido mercado financiero y ausencia de conflictos sociales o problemas militares, lo cual lo convierte en un país favorable para la inversión." (pp. 119)

En cuanto a los objetivos específicos, buscarán dar una posible solución a la problemática a dos de los procesos más relevantes de la organización. Dentro de los beneficios que la organización obtendría al contar con su propio modelo de predicciones, están la independencia con el proveedor que brinda actualmente el servicio, así como también el aseguramiento de la confidencialidad de sus datos trabajando todo al interno de la organización.

Actualmente, la organización cuenta con grandes bases de datos que son los repositorios finales de los sistemas de información que se utilizan. Éstas vienen siendo desaprovechadas ya sea por diversos factores de costos, limitantes de tiempo, capacitación de personal u otros factores. El uso del *Data Mining* se propone como

posible alternativa de solución al problema, ya que brinda diversas técnicas predictivas que involucran el procesamiento de grandes volúmenes de datos. Dentro de estas, redes neuronales y árboles de decisión se proponen como los principales pilares para nuestro estudio, los cuales se aplicarán para la detección de patrones de relación entre las variables, permitiendo generar pronósticos orientados a los objetivos específicos descritos anteriormente.

Estas oportunidades de mejora son los grandes motivantes de nuestra investigación, ya que nos permitirá explorar soluciones brindadas por otros autores y ver cómo podrían aplicar en el contexto local (Perú).

CAPÍTULO II: ESTADO DEL ARTE

Para empezar nuestra investigación, iniciamos con la búsqueda de una mejor comprensión de los tipos de análisis que se pueden realizar a partir de los datos históricos que ya nos había brindado la organización. Encontramos un estudio realizado por (Souza, 2014) cuyo objetivo fue identificar la metodología más adecuada para el respectivo caso planteado. Se evaluaron 3 tipos de técnicas analíticas clasificadas en descriptivo, predictivo y prescriptivo. De estos tres enfoques se puso especial énfasis en el análisis predictivo, donde de acuerdo con el autor debemos plantearnos responder a la pregunta "¿Qué va a suceder?" como referencia al momento de plantear los lineamientos del análisis. En este artículo se señala que usualmente la mayoría de las empresas con grandes cantidades de datos almacenadas procesan solo la información para verificar cómo se encuentran en la actualidad, más no se enfocan en pronósticos especializados a la predicción de eventos futuros que permitan anticipar situaciones adversas. El disponer de una gran cantidad de datos históricos brinda a las organizaciones el potencial de mejorar los métodos de pronóstico de venta, los cuales son explorados en el artículo con casos de éxito reales. Gracias al estudio realizado por este autor, se determinó el tipo de enfoque que se explorará en la presente investigación (predictivo), así como también las posibles técnicas de *Data Mining* que podrían calzar mejor con los objetivos planteados.

A partir de lo revisado en este primer artículo, se consultaron diversos estudios en los cuales se construyeron modelos estadísticos con la aplicación de métodos y técnicas predictivas. Como referencia base para lograr este propósito consultamos la investigación de, (Hashmi & Sheikh, 2012) los cuales mencionan que dentro del ciclo de la minería de datos (*Data Mining*) existen 6 fases principales, las cuales son: conocer el negocio, realizar la compresión de los datos, preparación de los datos, construcción del modelo, evaluación de resultados y despliegue de solución. A continuación, se presentará un resumen del artículo de investigación consultado detallando los aportes que estos brindaron para nuestra investigación por cada una de las fases del ciclo de *Data Mining*. La fase 6 del mencionado ciclo no será considerada dentro del alcance de esta investigación.

Fase 1

La primera fase consiste en responder la pregunta de por qué los clientes son más propensos a comprar el nuevo producto o en qué clientes deben enfocarse para realizar las campañas personalizadas con la finalidad de aumentar las ventas o la venta cruzada. Aplicando este concepto a nuestro caso de estudio, el enfoque del cliente se determinaría evaluando si este es un minorista o mayorista, la ubicación de residencia, marca o modelo de preferencia, así como otros factores del negocio de acuerdo con (Hashmi & Sheikh, 2012).

Fase 2

Para la segunda fase, nos apoyamos en la investigación realizada por (Alhilman et al., 2014) donde encontramos un caso evaluado en la empresa PT. X, en la cual se define la compresión de los datos como la determinación de los correspondientes parámetros, variables y atributos relacionados con el problema. Los pasos estipulados en la investigación para lograr este objetivo consisten en primero recoger los datos iniciales de la empresa de los diversos repositorios de información. Luego, para el correcto entendimiento del *dataset* obtenido, se asigna una definición funcional que explica mejor lo que representa el valor de la columna. Después de haber explorado los datos, el siguiente paso es la consolidación de estos de acuerdo con el periodo que se necesite trabajar en los procesos posteriores de modelización y/o predicción. Como último paso, se realiza una verificación de la calidad de los datos con el fin de evitar errores al momento de la extracción y procesamiento con la herramienta tecnológica utilizada. Una vez que se haya concluido con la comprensión de los datos, se deberá continuar con la determinación de cuáles serán los conjuntos de datos a utilizar.

Fase 3

En la tercera fase, indicada por (Hashmi & Sheikh, 2012) como la preparación de los datos, se define qué conjuntos de datos serán utilizados como entradas para el modelo en construcción, seleccionando las columnas y contenidos de estos. El autor menciona la importancia de incluir a los atributos sociales en los análisis predictivos dado que considera que estos atributos ayudan a obtener una mejor precisión y confiabilidad de los modelos de predicción. Cabe indicar que, si los atributos sociales no están disponibles pueden ser generados usando la teoría de grafos. Retomando la preparación de la data, se

sugiere la aplicación de técnicas como la reducción de atributos o la reducción de dimensiones, ya que muchos de los *dataset* propuestos como variables de entrada no aportan mayor significancia a la variable predictora.

Para explicar estos conceptos nos apoyamos en la investigación de (Zorman et al., 2002), donde se aborda el problema de encontrar nuevos conocimientos en forma de reglas en la base de datos, utilizando una mezcla de las dos técnicas: árbol de decisión y reglas de asociación. Inicialmente, se recolectó la base de datos del Hospital Colegio Médico de Osaka que consistía en 1251 casos compuestos por 60 atributos. Luego, se procedió a preprocesar los datos, teniendo como resultado 22 atributos; 4 continuos y 18 discretos. Se tuvieron que utilizar técnicas de discretización con el fin de aplicarlo en 4 atributos continuos. Estas técnicas son: equidistante, umbral y dinámico. El uso de diversas técnicas es un motivo por el que se tiene un número diferente de reglas en el conjunto de datos con la misma consecuencia. En las pruebas, se observó el número de reglas derivadas de diferentes enfoques. Las precisiones de los modelos de árboles de decisión variaron del 65,5% en el caso del objetivo de atributo de la nefropatía, y al 99,9% en el caso de atributo de destino Neuropatía (autonómica). Los árboles de decisiones finalmente fueron transformados en un conjunto de reglas, que son para un filtrado adicional y reducción, usando el mismo enfoque que para reglas de asociación. En comparación con el árbol de decisión, solo se seleccionó aquellas reglas, donde las variables resultantes fueron del conjunto de variables preseleccionados. Cabe indicar que el tamaño de la regla se limita sólo a normas con menos de cinco variables. Finalmente, los resultados obtenidos sobre la confianza mínima para ambos, se estableció en el 90%, y el apoyo mínimo para las reglas de asociación fue del 20%. De esta investigación se concluyó, que los conjuntos de reglas construidos por árboles de decisión son más pequeños que los de reglas de asociación. Adicionalmente a ello, se observa que el filtrado y la reducción no afectó las reglas derivadas del árbol de decisión en comparación con las reglas de asociación.

Otra técnica revisada en la literatura es la del análisis de componente principal (*Principal Componenet Analysis* - PCA), la cual tuvo aplicación en el estudio realizado por (Liu et al., 2009). Aquí se propuso como objetivo principal el pronosticar el nivel de inversión en capital humano en las distintas regiones de China. El método planteado en el artículo fue la combinación de la aplicación de un análisis de componente principal

(PCA) en conjunto con una red neuronal de regresión utilizando el famoso algoritmo de *Back Propagation* para la optimización de la función de error. El PCA se aplicó a un numeroso grupo de variables que constituyen el capital humano, las cuales según el autor Schultz estaban agrupadas en 4 *clusters* principales: formación educativa, nivel de salud, transferencia de fuerza de trabajo y habilidades adquiridas por experiencia. El análisis del componente principal se realizó para todas las dimensiones que conformaban los clústeres. Se llegó a obtener 4 dimensiones que fueron las más representativas. Cada dimensión tiene unos factores que multiplicados por los datos originales brindan un valor (*eigen value*) que junto con las otras seleccionadas serán datos de entrada para una red neuronal MLP. Luego de tener las variables de entrada definidas, se construyó la red neuronal en Matlab con los siguientes parámetros: 4 nodos de entrada, 7 en la capa oculta y 1 en la de salida. Se utilizó un step (ratio de aprendizaje) igual a 3000, funciones *tan-sigmoid* y *pure linear* para la transformación en la capa oculta y de salida respectivamente. De las 31 regiones a las que se les quiso realizar las predicciones se utilizó 26 como data de entrenamiento y 5 como data de prueba. El resultado obtenido fue muy preciso teniendo tan solo un 0.0001 de MSE (*Mean Square Error*).

Relacionando los estudios de los autores para la fase de preparación de datos, los aportes hacia nuestra investigación resultan muy provechosos dado que están muy relacionados al cumplimiento de los objetivos planteados. Tanto la obtención de pronósticos mensuales de venta como la asistencia de los clientes al servicio de taller de mantenimiento son procesos que almacenan gran cantidad de información en los repositorios de datos, motivo por el cual se debería seleccionar solo los atributos necesarios en base a la significancia que representen para el modelo planteado. De igual manera, el enfoque de contemplar variables sociales también aplicaría para nuestra investigación, ya que, de acuerdo con las personas del negocio variables como el PBI, el tipo de cambio y la participación de mercado de la marca juegan un rol importante en la decisión del cliente por la adquisición de un auto nuevo.

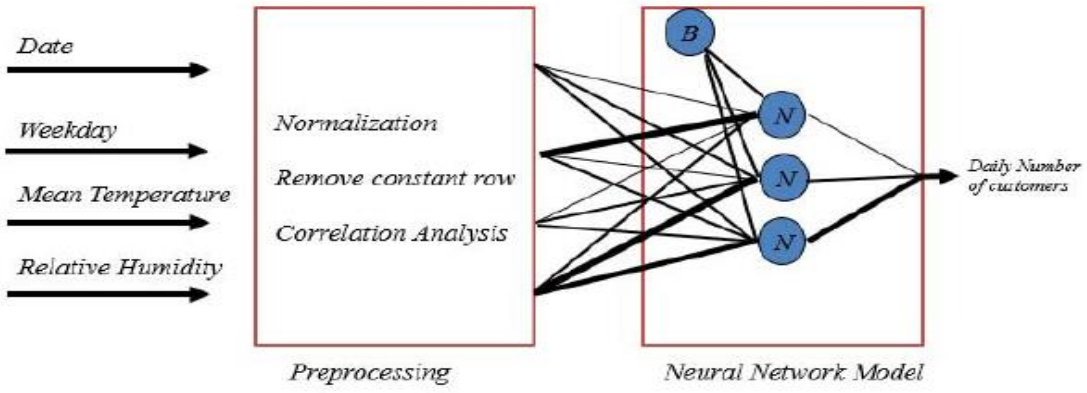
Fase 4 – Redes Neuronales

Continuando con la fase de 4 del ciclo del *Data Mining*, nos encontramos con la fase de construcción del modelo, para la cual se consultaron las investigaciones de (Lin & Tsai, 2016), (Han, 2008) y (Kaneko Yuta; Yada, 2016) con respecto al objetivo de obtener pronósticos mensuales de venta aplicando modelos de redes neuronales. En el primero,

para los autores, un modelo de predicción de ventas llamado *Deep Learning - Based Customer* es el más adecuado en lo que respecta a brindar soporte en la toma de decisiones sobre las ventas de una cadena de farmacias en Taiwán. Los autores trabajaron bajo el marco de trabajo conocido como UNISION, el cual considera resultados iterativos para la evaluación de problemas respecto a la toma de decisiones. Bajo este concepto se planteó realizar un cruce de información entre datos recogidos por el Instituto Climatológico de Taiwán y la data que se encontraba en cada farmacia. El proceso de la aplicación de *Deep Learning* se descompone en 3 secuencias. La primera es la colección de los datos (meteorológicos y los datos de los puntos de venta), seguido de una limpieza de datos realizada a través de una normalización junto con un análisis de correlación con el fin de hallar qué variables del clima se relacionan más fuerte para finalmente aplicar una red neuronal de capa oculta de 1 nodo escondido, donde cada neurona representa el nivel de ventas de cada farmacia (local). Esta red neuronal (ver figura 2.3) consideró los siguientes aspectos:

Figura 2.3

Modelo de Red Neuronal Deep Learning



Nota. De "A Deep Learning-Based Customer Forecasting Tool. In *2016* IEEE Second International Conference on Multimedia Big Data (BigMM) (pp. 198–205)" por (Lin & Tsai, 2016) (https://doi.org/10.1109/BigMM.2016.85)

- Algoritmo usado: La herramienta *MatLab* provee diversas funciones de entrenamiento para los modelos de redes neuronales. Se optó por la utilización de la función *Trainlm (Levenberg-Marquardt)*, ya que esta arrojaba los mejores resultados al momento de correr el modelo con dicha función.

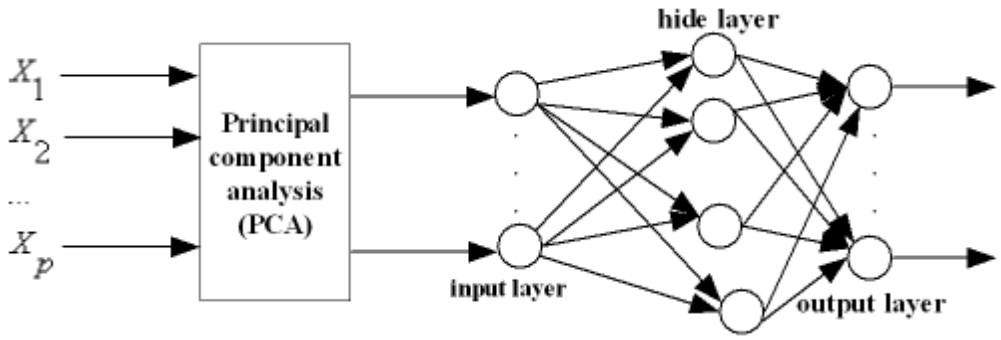
● Función de Transferencia: Para que el modelo pueda contemplar relaciones complejas entre las variables, se consideró que las funciones de transferencia se aplicarán en las capas escondidas y las de salida a través de *Tangent Sigmoid* y *Pure linear* respectivamente.

● Número de capas escondidas: Un modelo de redes neuronales que no considere capas escondidas no puede aprender relaciones no lineales entre las neuronas que lo componen. En este modelo se aplicó una capa escondida que tuvo como resultado un índice de correlación del 85%. Luego se procedió con el entrenamiento del modelo que más acertamiento tuvo con la data de verificación para poder corroborar con la data que no era de entrenamiento y se obtuvo un modelo con R = 84.2%. Finalmente se utilizó una interfaz gráfica para que el usuario pueda observar el flujo predictivo a partir de lo encontrado.

En su investigación, (Han, 2008) propone que el enfoque de segmentación de estos ya no sea respecto al producto, como ha venido siendo por mucho tiempo, sino que se realice una segmentación por el tipo de cliente buscando obtener una fidelización y compromiso del cliente con el negocio. El objetivo del autor fue combinar un método de análisis multivariante (PCA) junto con un modelo de redes neuronales BP (*Back Propagation*) que tome como valores de entrada lo obtenido por el algoritmo. Para ello realizó un estudio empírico con una muestra de 80 clientes vip de una empresa retail. Seguidamente se procedió a segmentar a los clientes por su comportamiento basado en 3 ámbitos que son características individuales de los clientes (edad, sexo, cantidad de hijos, etc), características del consumo (método de pago, frecuencia de consumo, etc) y actitud del consumidor (satisfacción del cliente, fidelidad). Luego con un experto del negocio se procedió a evaluar dándole puntajes a cada uno de los 80 clientes vip para poder aplicar el algoritmo de PCA, logrando remover la información superpuesta (dejar solo las variables más representativas). A través del MatLab se procesó y el resultado fueron 6 componentes principales que representaban a toda la muestra en un total de 85%. Finalmente, estas 6 variables obtenidas del PCA fueron utilizadas como variables de entrada para el procesamiento de las redes neuronales (ver figura 2.4).

Figura 2.4

Modelo PCA - BP Neuronal Network



Nota. De Customer Segmentation Model Based on Retail Consumer Behavior Analysis. In *2008 International Symposium on Intelligent Information Technology Application Workshops* (pp. 914–917), Por (Han, 2008) (https://doi.org/10.1109/IITA.Workshops.2008.225)

El procesamiento de la red neuronal fue un proceso complejo según lo comentado por el autor, así que se probó con cierta cantidad de nodos y a través de ensayo y error se llegó a una red donde el error fue solo de 10-3, obteniendo un resultado del 91,06% al momento de usar la data de verificación.

Por último, (Kaneko Yuta; Yada, 2016) nos presenta un enfoque *Deep Learning* y su aplicación para realizar la construcción de un modelo predictivo basado en redes neuronales, cuyo objetivo será poder determinar el cambio en el nivel de ventas en base a un día en particular. El estudio estuvo basado en el marco de trabajo H2O, el cual ha sido utilizado por diversos investigadores en el campo de *machine learning*. La aplicación del método consistió en la categorización de los productos de una cadena de supermercados por cantidad de atributos que poseen, obteniendo 3 grupos que iban de los más generales a los más específicos. La data por trabajar era información histórica de 3 años de las ventas en cada uno de los locales de la cadena de supermercado, se aplicó una binarización para clasificar el *dataset*, el cual arrojaba 1 cuando existía un incremento en el nivel de ventas con respecto al día anterior y 0 en el caso contrario. Seguidamente se procedió con la aplicación del enfoque *Deep Learning* a través de la aplicación H20, el cual permite la aplicación directa de la red neuronal para la obtención de rápidos resultados. El seteo de los parámetros consistió en indicar parámetros como la cantidad de capas ocultas (3 para todas las categorías y 100 unidades cada una), el número de veces que el aprendizaje será repetido (1000), el parámetro *dropout* el cual está relacionado a la regularización de los datos previniendo el ajuste excesivo del aprendizaje de la red y

por último la función de activación combinación de (*Rectifier and Dropout*). Los resultados obtenidos del entrenamiento sin usar la regla de regularización fueron un porcentaje de precisión del 84% para la categoría 1, 77% para la categoría 3 y 75% para la número 2; si se considera la regla mencionada anteriormente la precisión se incrementa en 1 o 3 % en todos los casos. De los resultados se puede observar que el modelo resulta eficaz (más de 75%) y útil para predecir si las ventas incrementarán o bajarán de un día para otro, lo cual permitirá tener una mejor visión comercial a los cargos gerenciales de la cadena de supermercados.

Para la implementación de los modelos de red neuronal se dispone de diversas herramientas que ya tienen incorporados las lógicas y funciones de las técnicas de *Data Mining*. En la investigación realizada por (Ozveren et al., 2014), nos encontramos un caso de estudio realizado en el campo de redes neuronales con técnicas que permitieron realizar predicciones sobre el problema de electricidad a corto plazo (*short term electricity forecast*) y aplicando un nuevo algoritmo llamado *Neuro-Evolution through Augmenting Topologies* (NEAT). Se inserta el concepto de STFL el cual permite conocer cuál es el nivel de consumo de los sistemas de energía electrónicos y así poder establecer un diferencial en el mercado del consumo eléctrico. Luego se menciona la posibilidad de implementar las NEAT a través de librerías que permiten construir código en algunos lenguajes de programación como Python, C, Pearl o Ruby; siendo Python el cual será utilizado en este estudio. El enfoque apunta a que, en lugar de usar una red neuronal artificial completamente conectada, se utilice sólo parcialmente conectada ya que con los algoritmos de NEAT se pueden obtener resultados aún más precisos y eficientes. Como parte de la metodología primero se enfocan en la generación de un script en Python describiendo sus principales ventajas como la posibilidad de elaborar código orientado a objetos con multiprocesamiento para la red neuronal, así como la gran variedad de librerías que permitan codificar operaciones matemáticas complejas. Se presenta luego la arquitectura de una ANN con NEAT la cual consiste en generar una población inicial de ANN’s considerando el tamaño de la muestra, el porcentaje de probabilidad de conexión de un nodo de entrada con el nodo de salida y la semilla para el generador de números aleatorios se introducen en este procedimiento, esto seguido de unos procesos iterativos con la finalidad de ajustar el modelo. De los resultados obtenidos se muestran un notable mejoramiento en el tiempo de procesamiento y aprendizaje del modelo, así como una reducción en el error porcentual absoluto cuando se combina la ANN con la técnica

NEAT.

Relacionando la literatura de los autores presentada hasta aquí, todos coinciden en la aplicación de un modelo de red neuronal del tipo *Back Propagation* utilizando el enfoque *Deep Learning* como propuesta de solución a los casos presentados en sus respectivos artículos. Todos los modelos trabajados toman un *dataset* de entrada ya previamente preparado (el cual saldría de la fase anterior de preparación de datos) para las etapas de entrenamiento y prueba. Se detalla los criterios que se tuvieron para la determinación de los componentes de una red neuronal, tales como la cantidad de neuronas para cada capa, las funciones de activación a seleccionar considerando el tipo de salida que se desea obtener, el número de repeticiones idóneo en relación con la cantidad de la muestra evaluada entre otros. Estos conceptos serán muy importantes a la hora del diseño de red neuronal que se planteará en nuestra investigación. Asimismo, para la implementación de los modelos de red neuronal, utilizaremos *Python* en su distribución especializada para modelo estadísticos la cual fue recomendada por (Ozveren et al., 2014) en su investigación.

Fase 4 – Árbol de Decisión

Se consultó la literatura relacionada al objetivo del pronóstico de asistencia de los clientes al servicio de taller de mantenimiento mediante árboles de decisión. Para abordar ello, nos basamos de los estudios realizados por (Xu et al., 2017), (Moreira et al., 2017) y (Xie et al., 2019) teniendo en el primero de ellos un escenario donde los autores realizan la construcción de un modelo predictivo que aplica el algoritmo *Random Forest*, el cual consiste en utilizar múltiples árboles de decisión para entrenar las muestras. Esta investigación tiene el objetivo de predecir la diabetes tipo II, que consiste en que el cuerpo no produce o utiliza la insulina correctamente. La insulina de tipo II, representa alrededor del 95% de los pacientes diabéticos. Inicialmente, se aplicó un preprocesamiento de datos para mejorar la calidad de los resultados. La información fue proporcionada por la Universidad de Virginia Escuela de Medicina, son 400 pruebas y cada prueba tiene 19 variables. Por ejemplo, edad, sexo, colesterol, hemoglobina, cintura, cadera, etc. Adicionalmente, se realiza la reducción de la dimensionalidad cuya finalidad es mejorar el rendimiento del algoritmo. Para mejorar la precisión del modelo construido, usualmente se realiza la discretización de variables continuas. La discretización es elegir el número de puntos de división de los datos, dividir cada característica en tres partes:

baja, media, alta y representar estos valores de características respectivamente, esta investigación utiliza una discretización de *k-means*. Después del preprocesamiento de los datos, el siguiente objetivo es encontrar relaciones entre las diversas características y encontrar patrones. Se empezó a construir un modelo que utiliza un algoritmo *Random Forest* para predecir si la persona padecerá diabetes. Se detallan los pasos que realiza el algoritmo.

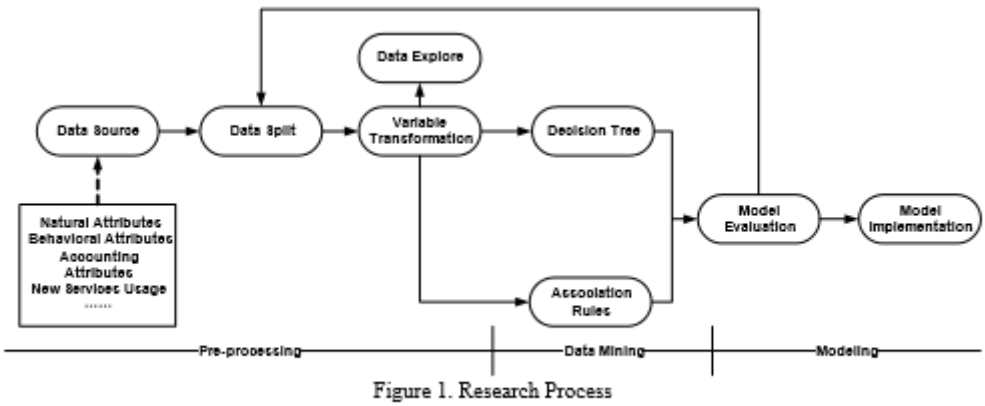
- Paso 1: Utiliza técnicas de remuestreo de *Bootstrap* para generar k muestras. Estas muestras cubren 2/3 del conjunto de datos, y el resto se llama *Out-of-bag* (OOB), estos datos pueden utilizarse para las pruebas.
- Paso 2: Utiliza las k muestras para formar k árboles de decisión. En cada nodo de cada árbol se selecciona aleatoriamente m características (m<m) en las M características, se sugiere empezar con m =√M y luego disminuir o aumentar m hasta que se obtenga error mínimo para el conjunto de datos. Por último, elige la mejor partición basándose en el criterio de G*ini*.
- Paso 3: Establece el mejor resultado de acuerdo con el mecanismo de votación por mayoría.

Las medidas utilizadas para realizar las métricas del rendimiento del modelo son la precisión, la sensibilidad y la especificidad. La precisión permite determinar la capacidad del modelo clasificador para producir un diagnóstico preciso. La sensibilidad permite medir la capacidad que tiene el modelo para identificar con precisión la ocurrencia de la variable resultante. La especificidad permite medir la capacidad que tiene el modelo para separar la variable resultante. Se concluyó que la precisión del modelo Random Forest se mejora aleatoriamente, y puede predecir con eficacia el riesgo de diabetes en caso se cuente con una cantidad suficiente de datos. Siguiendo con la idea del autor anterior, el siguiente artículo de los autores (Yang et al., 2008) se relaciona con el tema porque se centra en la predicción de las oportunidades de venta cruzada y presenta un enfoque innovador para pronosticar las oportunidades de venta más eficaces. Esta investigación combina dos técnicas: árbol de decisión y regla de asociación con el objetivo de descubrir oportunidades de venta cruzada. El resultado del análisis de los datos reales de los registros transaccionales muestra que este nuevo método puede mejorar en gran medida la exactitud de la predicción. Se observa en la Figura 2.5 el proceso aplicado en este estudio, que consiste en tres fases: datos de preprocesamiento, minería de datos y modelado. La primera parte permite limpiar la fuente de datos para

luego transferirla a la base de datos que se utilizará en la investigación. En paralelo, se identifican variables para el modelo del árbol de decisión y los servicios de asociación de candidatos de las reglas de asociación. Luego, los datos se dividen en dos grupos, entrenamiento y pruebas. Después, se realiza la transformación de las variables. Finalmente, se exploran los datos para identificar la distribución y características comunes.

Figura 2.5

Metodología Revisada



Nota. De "Using decision tree and association rules to predict cross selling opportunities". Por (Yang et al., 2008) (https://doi.org/10.1109/ICMLC.2008.4620698)

En la segunda parte, el autor propone utilizar el árbol de decisiones y la regla de asociación permitiendo averiguar los atributos comunes de los usuarios WAP (Este caso de estudio fue aplicado a una empresa de Telecomunicaciones y su servicio era el *Wireless Application Protocol* se seleccionó 7 variables de entrada del árbol de decisiones de la base de datos de clientes proporcionada por el proveedor de telecomunicaciones: nivel de gasto consumidor, edad, marca, antigüedad, tipo de cliente, motivos de reemplazo del teléfono móvil y características de itinerancia y las reglas de asociación entre servicio de asociación o mezcla de servicios y WAP). Con la finalidad de obtener los resultados del árbol de decisión y las reglas de asociación para luego realizar la pronosticación basándose en estas dos técnicas, que significa encontrar el conjunto de intersección y unión de sus resultados de pronóstico, y comparar su precisión. Finalmente, pusimos en práctica este método de pronóstico.

En la investigación de (Xie et al., 2019), proponen un método basado en árboles de decisión para la detención de la contracción ventricular prematura (PVC), PVC son un tipo de latido anormal (arritmia) del corazón. En primer lugar, se entrenó con 22 registros y se validó con otros 22 registros. En segundo lugar, el entrenamiento y las pruebas se realizaron con 90% y 10% de la data. Los resultados obtenidos muestran la fiabilidad del método basado en el algoritmo CART, el árbol de decisión que se poda para evitar el *overfitting*, permitiendo mejorar la clasificación. También se comparó con otros métodos para la misma problemática teniendo como mejor resultado el árbol de decisión.

Para la implementación de los modelos de árboles de decisión se dispone de varias herramientas que ya tienen incorporadas las lógicas y funciones de las técnicas *Data Mining*. En la investigación realizada por (Katamaneni et al., 2018), encontramos un caso de estudio donde realizan la implementación de árboles de decisión en *R Studio* y *Java* para poder encontrar la mejor herramienta para este algoritmo. La primera, es un entorno de desarrollo integrado que usualmente es utilizado para temas de estadística, computación y gráficos que incluye una consola, editor de sintaxis que ayuda para la ejecución de Código. Otra ventaja es que cuenta con una comunidad de desarrolladores y esta herramienta es aprovechada por grandes empresas como Google, Facebook, Microsoft, Bank of America. La segunda, es un lenguaje orientado a objetos, rápido, seguro y confiable, sin embargo, requiere la instalación de varios componentes adicionales. Varias ventajas y desventajas fueron revisadas por los investigadores para estas dos herramientas, como la interfaz, el tiempo en construir los modelos, requisitos necesarios para la instalación, la representación gráfica de los modelos y otros. De acuerdo con ello, se revisó que la representación gráfica es posible en ambos lenguajes, sin embargo, para Java requiere instalar componentes adicionales por lo que demanda más tiempo. Otro factor indicado es que R Studio es compatible para diferentes lenguajes mientras que Java no. A partir de ello, los investigadores indican que la herramienta R Studio es la mejor opción para implementar árboles de decisiones.

Relacionando la literatura de los autores presentada hasta el momento, los autores coinciden en la aplicación de árboles de decisión para realizar predicciones de clasificación como propuesta de solución presentados en sus respectivos artículos. Todos los modelos trabajados toman un *dataset* de entrada previamente preparado (el cual saldría de la fase anterior) preparación de los datos para la etapa de entrenamiento y prueba. Se detalla los criterios que tuvieron para la determinación de sus parámetros, tales

como cantidad de árboles, cantidad de características, profundidad del árbol, número de nodos y otros. Asimismo, para la implementación de los modelos de árboles de decisión, utilizaremos *R Studio* en su distribución especializada para modelo estadístico la cual fue recomendado por su investigación (Katamaneni et al., 2018).

Fase 5

Finalmente, la conclusión del ciclo se da con la validación de los resultados obtenidos en la construcción del modelo. Para el objetivo de pronósticos mensuales de venta se revisó el estudio de (Qin & Li, 2011) el cual nos muestra la obtención de predicciones de venta a través un algoritmo de redes neuronales llamado *Back Propagation*. Se definen las consideraciones que debe tener la red neuronal para que pueda realizar un adecuado aprendizaje, entre ellos está el número de capas ocultas que tendrá el modelo para no comprometer a la red con un *overfitting* o un excesivo tiempo de entrenamiento, así como también el número de neuronas (nodos) que se van a construir en cada una de las capas, el cual es determinado por una fórmula empleando los parámetros de la red (función de activación, número de entrenamientos, etc). Para el caso de estudio se utilizó la data de una empresa llamada *Country Kitchen Corporation* la cual se dedica a la venta de comida rápida y posee múltiples locales a lo largo de Estados Unidos. Se buscó construir un modelo de red neuronal que pueda realizar predicciones de las ventas de las tiendas en 15 regiones de esta cadena de comida, para ello se utilizó sólo una capa oculta y 3 variables de entrada (nodos) las cuales fueron publicidad, gastos de promoción y oponentes. A través de la fórmula descrita se calcula el número de nodos, 4 para este estudio, y se procede con la programación de la red en MatLab. Posteriormente se realiza la aplicación de *Cross Validation*, con la finalidad de evidenciar que el modelo de red neuronal no presente *overfitting* en base a la data utilizada para el entrenamiento. De los resultados obtenidos se realizó un comparativo del error relativo que se obtiene contra un análisis de correlación clásico, se puede comprobar que el modelo de redes neuronales obtiene un 3% mientras que el de correlación obtiene un 6%.

Para el objetivo de objetivo del pronóstico de asistencia de los clientes al servicio de taller de mantenimiento se trabajó con la investigación de (Moreira et al., 2017) dónde se utilizó una técnica de *Data Mining* (DM), llamada *Random Forest* (RF), aplicada a la atención médica para la identificación tempranas de trastornos de hipertensión en el embarazo para evitar las muertes maternas y fetal durante la gestación y el parto. El

modelo es útil porque las personas pueden tomar mejores decisiones en los momentos de incertidumbre. El criterio de partición utilizado en esta investigación para el método *Random Forest* es el de ganancia. También utilizan una validación cruzada para comprobar el nivel de precisión del clasificador. Luego, para medir la eficacia del modelo se usa la matriz de confusión que adicionalmente muestra el número de clasificaciones correctas e incorrectas por clases. Esta investigación utilizó los datos de 25 mujeres embarazadas que fueron obtenidos por los obstetras. El 20 % se encuentran en el rango de antes de las 20 semanas de gestación y el resto posterior de las 20 semanas. Los principales síntomas son presión arterial alta, pérdida de proteínas, dolor de cabeza, dolor epigástrico, vómito, visión borrosa, mareos, oliguria, también se consideró la edad gestacional. Los síntomas mencionados anteriormente se utilizaron para construir el modelo del árbol de decisión. Otra medida importante para los clasificadores es el área ROC (*Receiver Operating Characteristic*) es una herramienta para medir y especificar los problemas de diagnósticos en los desempeños de la atención médica. Cabe indicar que, el modelo propuesto tuvo un excelente desempeño para ROC. Esto se debe a que se tiene la mejor relación entre sensibilidad y especificidad. La hipertensión es un problema médico más frecuente durante la gestación y se presenta la evaluación del desempeño del clasificador Random Forest. También se comparó con otros tipos de clasificadores y presentó buenos desempeños en el área ROC.

De los estudios consultados para esta fase del ciclo se rescatan conceptos como la validación cruzada (*Cross Validation*) el cual permitirá evaluar un posible *overfitting* de los modelos a construir con las redes neuronales, así como también la generación de un modelo de regresión simple el cual permitirá validar la eficacia de ambas técnicas predictivas comparando sus resultados. En cuanto al objetivo relacionado a árboles de decisión, el aporte de la investigación consultada sería la aplicación de una matriz de confusión para validar la precisión y eficacia de los modelos que se elaborarán en nuestro estudio, así como también la aplicación del método curva *ROC* para tener visibilidad de la proporción de verdaderos positivos frente a los falsos positivos, los cual representaría la clasificación final pronosticada por el modelo.

CAPÍTULO III: MARCO TEÓRICO

3.1. Análisis Predictivo

Según (Hashmi & Sheikh, 2012) el análisis predictivo es una vertiente de la minería de datos, que combina el conocimiento del negocio y las técnicas de análisis estadístico, para extraer información predictiva oculta y para predecir tendencias futuras y patrones basados en el análisis de amplios volúmenes de datos históricos con variables diferentes. Variables que se pueden medir para predecir el comportamiento futuro de una persona o entidad. El análisis predictivo se aplica en las decisiones estratégicas relacionadas con nuevos mercados, adquisición de nuevas perspectivas y la retención de clientes, ventas.

Enfoque Análisis predictivo

El ciclo de minería de datos se ha dividido en seis fases: conocimiento del negocio, la interpretación de los datos, preparación del *dataset*, modelado, evaluación de pruebas y despliegue de acuerdo con (Hashmi & Sheikh, 2012). En su caso de estudio se utilizó un conjunto de datos de telecomunicaciones en el mundo real de un operador de telecomunicaciones. El conjunto de datos utilizado, compuesto por información prepago del abonado móvil. Esta investigación se evaluó con regresión y algoritmos de árboles de decisión, en esta se muestra la importancia de los atributos sociales en los análisis predictivos. Se observa la adición de atributos sociales como resultado de una mejora significativa en la precisión y la fiabilidad de los modelos de predicción.

3.2. Modelo Predictivo

Un modelo predictivo vendría ser una función matemática capaz de aprender la correlación entre un conjunto de variables de datos de entrada, normalmente empaquetadas en un uno o varios registros, y una variable de respuesta o de destino. Nos referimos a este aprendizaje como supervisado, debido a que durante la capacitación los

datos son presentados a un modelo predictivo con los datos de entrada y la salida o el resultado deseado. La capacitación o entrenamiento se repite hasta que el modelo aprende la función de correlación entre las variables de entrada y las salidas deseadas, esto de acuerdo con (Alex Guazzelli, 2012).

A través de diversas investigaciones que demuestran su eficiencia, se ha establecido que se puede llegar a generar una gran ventaja competitiva a partir de *Data Mining* en general y particularmente a partir de modelos predictivos. Para algunas aplicaciones de modelado predictivo, maximizar la precisión o una medida de utilidad es de suma importancia, incluso a expensas de capacidades explicativas más débiles (Hong & Weiss, 2001). Asimismo, se indica la importancia que están cobrando estos modelos en el ámbito corporativo, se menciona algunos casos como el de aplicaciones de estos modelos en distintos rubros de negocio.

3.3. Data Mining

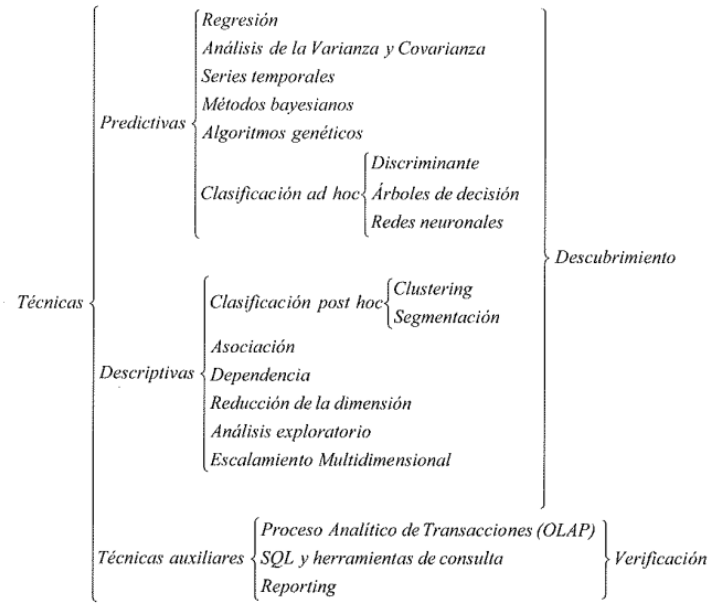
(Berlanga & Hurtado, 2013) nos brindan una definición de lo que es *Data Mining* y de lo que abarca como tal:

> Las técnicas de la minería de datos provienen de la Inteligencia Artificial y de la Estadística. Dichas técnicas no son más que algoritmos, más o menos sofisticados, que se aplican sobre un conjunto de datos para obtener unos resultados. Las técnicas más representativas son: redes neuronales, regresión lineal, árboles de decisión, modelos estadísticos, agrupamiento o clustering y reglas de asociación. (pp. 65)

Este concepto por lo tanto abarca una gran rama de posibles técnicas que pueden ser de utilidad para distintas investigaciones. Con el surgimiento de nuevas tecnologías capaces de procesar inmensas cantidades de datos, el *Data Mining* ha sido de los tópicos favoritos de los investigadores. Según el diagrama (ver figura 3.6) para lo que se busca en nuestra investigación necesitaríamos utilizar alguna técnica perteneciente a la categoría de Predictivas.

Figura 3.6

Técnicas de Data Mining



Nota. De "Cómo aplicar árboles de decisión en SPSS. *Reire, Revista d'Innovació i Recerca en Educació*, 65-79". Por (Berlanga & Hurtado, 2013).
https://revistes.ub.edu/index.php/REIRE/article/viewFile/5155/7229)

3.4. Árboles de Decisión

De acuerdo con (Lior & Oded, 2014) es un modelo predictivo, se define como un modelo jerárquico de decisiones y sus consecuencias. Cabe indicar que, cuando se utiliza un árbol de decisión para tareas de clasificación, se lo denomina árbol de clasificación y cuando se utiliza para tareas de regresión, se denomina árbol de regresión. Los árboles de clasificación se utilizan para clasificar un objeto o una en un conjunto predefinido de clases en función de sus valores atributos. Adicionalmente, los árboles usualmente son aplicados en áreas de negocio, medicina e ingeniería.

Un árbol de decisiones es un clasificador que tiene nodos formando una raíz que no posee bordes de entrada y el resto de los nodos poseen un borde entrante. Cabe indicar que, se le denomina interno aquellos nodos con bordes salientes y el resto de los nodos se les denomina hojas o terminales. En un árbol de decisión, los nodos internos se particionan en dos o más el espacio de acuerdo con la función discreta de los valores de

las variables de entrada. En el caso más sencillo y común, cada prueba considera una variable, que permite la partición del espacio conforme al valor de las variables, para las variables numéricas se utiliza rangos.

Según (Amaya, 2010), es una técnica que permite analizar decisiones secuenciales basadas en el uso de resultados y probabilidades asociadas. Se emplean en situaciones de toma de decisiones en las que permite mejorar una serie de ellas. Los árboles de decisión son similares en la estructura y componentes, cabe mencionar que se requiere de estos 4 componentes: alternativa de decisión, casos que pueden pasar, probabilidades que ocurra los casos y resultados de las posibles interacciones.

La importancia de los árboles de clasificación es que son excelentes predictores y también permite generar reglas de clasificación como "condición" → "resultado". Las técnicas de *Data Mining* basadas en árboles han tenido un desempeño satisfactorio en diversas áreas del conocimiento, las cuales quedan demostradas en la literatura revisada.

3.5. Bosque Aleatorio (*Random Forest*)

De acuerdo con lo especificado por (Moreira et al., 2017) *Random Forest* es un método que combina el resultado de varios clasificadores, donde cada uno de ellos corresponde a un árbol de decisión. Cada árbol construido tiene una profundidad máxima, es decir, sin poda. Este método tiene dos parámetros de entrada. El número T de árboles de decisión a construir y $M \leq m$, que es el número de características (de las m características), que se consideran para construir cada nodo del árbol de decisión.

Para la aplicación del algoritmo el conjunto de datos se divide aleatoriamente en varios subtipos y puede manejar mejores conjuntos de datos con un gran número de atributos. Cabe indicar que, el uso de subconjuntos de árboles convierte al algoritmo más fuerte que un sólo árbol, obteniendo una precisión significativa. Esta técnica también se puede considerar más precisa en comparación con otros métodos, tales como *artificial neuronal network* (ANN) y *support vector machine* (SVM).

Según (Xu et al., 2017) *Random Forest* fue propuesto por el Dr. Breiman en 2001, es un método exitoso de clasificación y regresión que consiste en combinar aleatoriamente varios árboles de decisión y agregar sus predicciones promediando.

Asimismo, *Random Forest* está basado en la teoría del aprendizaje estadístico, que consiste en utilizar *Bootstrap* de forma aleatoria de remuestreo para extraer múltiples versiones de los conjuntos de muestras de los conjuntos de datos.

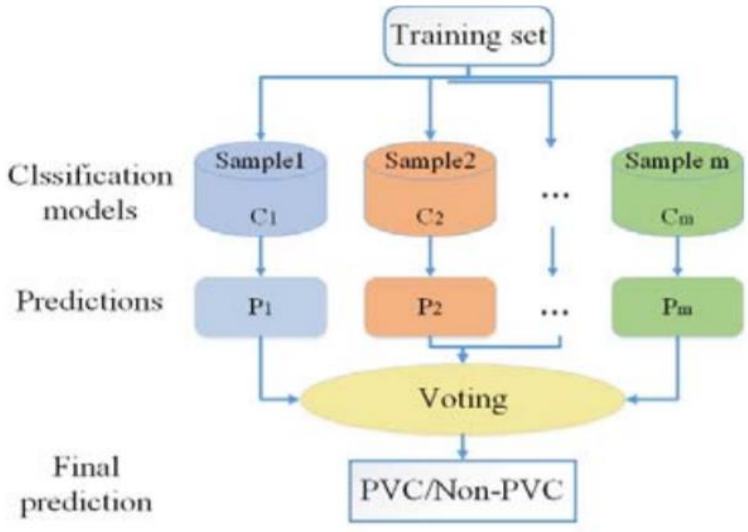
3.6. Árbol de clasificación y regresión (CART)

De acuerdo con (Xie et al., 2019) el algoritmo CART, es un árbol de decisión generado que se poda para evitar el *overfitting*, permitiendo mejorar el efecto de la clasificación. Se utiliza este algoritmo para construir árboles de clasificación por dos motivos: tiene una amplia gama de aplicaciones y el árbol que se genera puede ser podado para evitar el *overfitting*.

El algoritmo CART consiste en realizar la generación y poda del árbol de decisión. Para la construcción del árbol se seleccionan las mejores características de acuerdo con el índice Gini mínimo.

Figura 3.7

Modelo de entrenamiento basado en árboles de decisión



Nota. De "Research on Heartbeat Classification Algorithm Based on CART Decision Tree. *2019 8th International Symposium on Next Generation Electronics (ISNE)"*. Por (Xie et al., 2019) (https://ieeexplore.ieee.org/document/8896650)

3.7. Análisis de Componente Principal (PCA)

Según (Liu et al., 2009), PCA es una técnica de análisis multivariante que busca la comprensión y extracción de las dimensiones más importantes en un conjunto de datos. Bajo este enfoque, lo que se busca obtener es la reducción de dimensiones que componen un *dataset* extenso. El cálculo de las dimensiones se obtiene a partir de la teoría del problema de *Eigen*. Esta dice que todo *dataset* puede ser descompuesto en dimensiones cuya dirección representa lo que se conoce como *eigen vector*; y cuya magnitud escalar se conoce como *eigen value*. El término componente principal hace referencia a aquel eigen vector que represente la mayor varianza entre las variables analizadas, por lo tanto, este tendría el mayor *eigen value*. Aquellos *eigen values* que posean un valor muy bajo serían poco representativos para la distribución de las variables y por lo tanto podrían ser descartados, llegando de esta manera al objetivo de la reducción de las dimensiones originales. Para entender mejor esta técnica pensemos en un conjunto de datos representados de la siguiente forma:

$$x = (x_1, x_2, x_3, \ldots x_p)$$

Ecuación 1. Conjunto de Datos (Liu et al., 2009)

La teoría de componentes principales indica que el mismo conjunto de datos puede ser representado de la siguiente manera, donde cada F representa una dimensión (factor).

$$\begin{cases} F_1 = a_{11}x_1 + a_{21}x_2 + a_{31}x_3 \ldots + a_{p1}x_p \\ F_i = a_{1i}x_1 + a_{2i}x_2 + a_{3i}x_3 \ldots + a_{pi}x_p \\ F_p = a_{1p}x_1 + a_{2p}x_2 + a_{3p}x_3 \ldots + a_{pp}x_p \end{cases}$$

Ecuación 2. Representación de Dimensiones (Liu et al., 2009)

El valor de a_{ij} se determina en base a lo siguiente:

(1) $a_{1j}^2 + a_{2j}^2 + a_{3j}^2 + \ \ldots\ + a_{pj}^2 \ = \ 1, j = 1,2,3 \ \ldots p$

Ecuación 3. Sumatoria de coeficientes (Liu et al., 2009)

(2) F_1 vendría a ser el componente que representa la mayor varianza en toda la combinación lineal de las variables en estudio, F_2 sería el segundo y así sucesivamente. Finalmente, F_p sería una dimensión que no sería

representativa y por lo tanto podría ser descartada sin afectar la relación de las demás.

3.8. Redes Neuronales

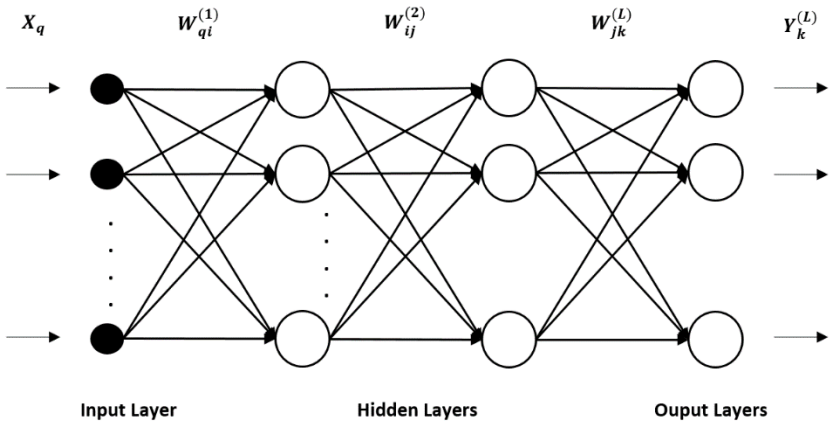
Es un concepto muy utilizado en diversas investigaciones. Según (Lin & Tsai, 2016) las redes neuronales son un conjunto de técnicas no paramétricas que son utilizadas en distintos campos de la ciencia e ingeniería, ya que brindan la posibilidad de resolver problemas que difícilmente podrán ser atendidos por otras técnicas, como regresiones lineales o polinomiales. Estas técnicas son utilizadas con la finalidad de obtener modelos no explícitos que relacionan variables de entrada y variables de salida, de tal manera que estas últimas puedan ser predichas a través de un proceso de entrenamiento (p. 198).

Las redes neuronales artificiales buscan imitar el comportamiento del proceso de aprendizaje del cerebro humano, tal como lo indican (Jain et al., 1996) en su artículo. Las ANNs (*Artificial Neuronal Network*) pueden clasificarse en dos grandes categorías las cuales son *Feed Foward Networks* y *Recurrent (feedback) Networks*, siendo la familia *Multilayer Perceptron* la más común de encontrar en las investigaciones. El paradigma del aprendizaje que tiene una red neuronal es el que más llama la atención, debido a que no depende de un conjunto de reglas brindadas por una persona, sino que esta aprende iterativamente por su cuenta, a través de las relaciones halladas entre las variables de entrada y salida (Jain et al., 1996).

3.9. Multilayer Perceptron

Según (Jain et al., 1996) es un tipo de red neuronal *Feedforward*, la cual consiste en un modelo multicapa donde hay una capa de entrada, una capa oculta y una capa de salida, todas conectadas de forma exitosa sin incluir conexiones *feedback* entre ellas, es decir sólo van en un sentido (ver Figura 3.8).

Figura 3.8
Modelo de una capa oculta



Nota. De "Artificial neural networks: a tutorial*", p*or (Jain et al., 1996). (https://ieeexplore.ieee.org/document/485891/authors)

Un perceptrón *multilayer* puede formar límites arbitrarios y representar una función booleana. Cada perceptrón dentro está conectado a otro de la capa subsecuente, a esta conexión se le asigna un peso (w) el cual junto con los parámetros determinados de un conjunto de patrones de entrenamiento (como $\{(x^{(1)},d^{(1)}), (x^{(1)},d^{(1)}), \ldots (x^{(p)},d^{(p)})\}$ donde $X^{(i)} \in R^n$ es el vector de entrada en el espacio patrón dimensional n, y $d^{(p)} \in [0,1]m$ el cual es un hipercubo dimensional-m) determinan las dimensiones de la red neuronal. Para la obtención de la variable w (peso) se utiliza un algoritmo de aprendizaje, el cual le permite a la red entender el comportamiento de las variables, siendo más utilizado para este tipo de predicciones el de *Back Propagation*.

3.10. Validación Cruzada (Cross Validation)

Según (Refaeilzadeh et al., 2009), la validación cruzada vendría a ser una técnica estadística para le evaluación y comparación de diversos algoritmos de aprendizaje, cuyo objetivo generalmente es realizar una división de los datos en n segmentos, utilizando el primero para aprender y entrenar el modelo, mientras que el restante para realizar la validación respectiva. En una típica validación cruzada, los *dataset* de entrenamiento y prueba son cruzados en rondas sucesivas, de modo que cada uno de los datos tiene la

oportunidad de ser validado con su contraparte, teniendo en cuenta que se debe determinar a través de la experimentación tanto el número de particiones a considerar como el de iteraciones. La forma básica de validación cruzada es la validación cruzada de k veces (*k-fold*), la cual consiste en particionar la data de entrenamiento en segmentos de cantidad iguales de datos (k), dejando sólo 1 (k-1) para la validación, repitiendo el proceso iterativamente hasta encontrar el número adecuado donde el valor de predicción obtenido por la validación cruzada llegue a la convergencia.

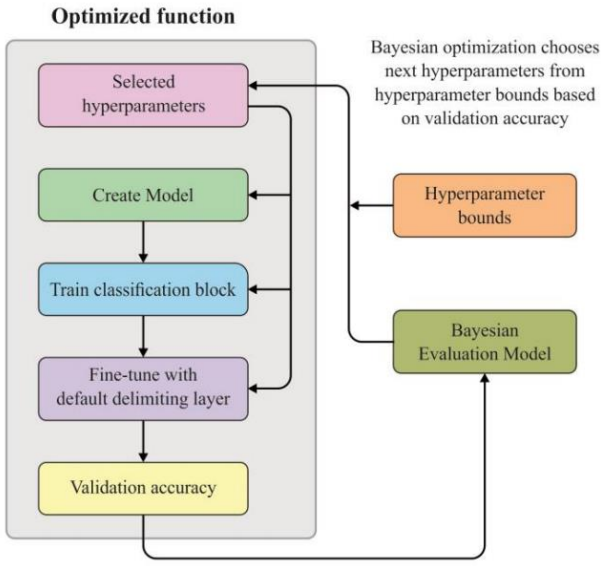
3.11. Hyperparameter Tuning

De acuerdo con (Shankar et al., 2020) es una técnica que ayuda a poder realizar una correcta selección de los parámetros de configuración para una implementación de Data Mining, logrando una mejor performance ya sea para un modelo clasificación o para uno de regresión.

En el caso de estudio de los autores se empleó una automatización para afinamiento de parámetros, haciendo que la selección de estos forme parte del proceso de creación del modelo (ver figura 3.9).

Figura 3.9

Modelo de automatización de afinamiento



Nota. De "Hyperparameter Tuning Deep Learning for Diabetic Retinopathy Fundus Image Classification*",* por (Shankar et al., 2020). (https://ieeexplore.ieee.org/document/9126771)

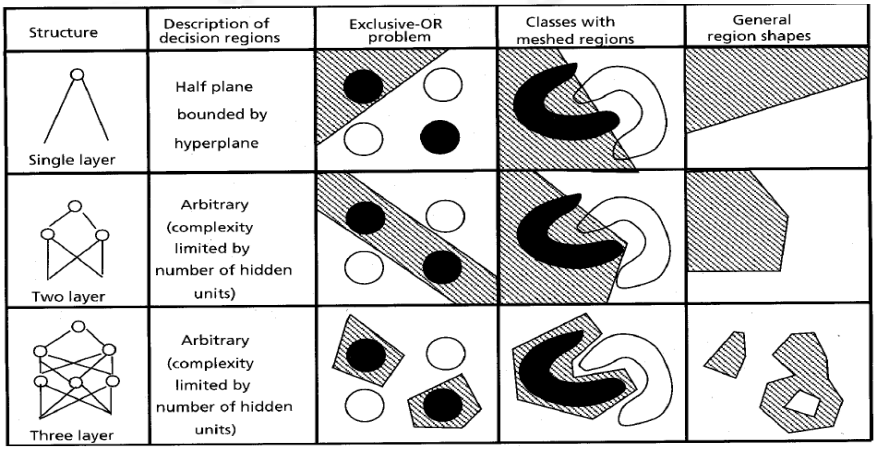
3.12. Algoritmo de Retropropagación (*Back Propagation*)

Como lo menciona (Jain et al., 1996) *Back Propagation* es un método de descenso gradiente para minimizar la función de coste de error cuadrático, el cual siempre está presente al momento de querer realizar regresiones.

Una interpretación geométrica mostrada en la Figura 3.10 puede ayudar a explicar el papel de las unidades ocultas (con función de activación). Cada neurona ubicada en la primera capa oculta forma un hiperplano en el espacio del patrón; los límites entre clases de patrones pueden ser aproximados por hiperplanos. Una neurona en la siguiente capa oculta forma una hiper región a partir de los outputs de las neuronas de la capa anterior, se obtiene una región de decisión a través de la operación AND en los hiperplanos. Las unidades de la capa de salida se combinan con las regiones de decisión encontradas por las unidades en la segunda capa oculta mediante operaciones lógicas XOR. Este escenario se representa sólo para explicar el papel de las unidades ocultas. Su comportamiento real, después de que la red esté entrenada, podría diferir. Una red neuronal de dos capas es capaz de formar fronteras de decisión más complejas que las mostradas en la Figura 3.10. Adicional a ello, los perceptrones multicapa con funciones de activación sigmoide pueden formar límites de decisión curvos en lugar de sólo límites lineales por piezas.

Figura 3.10

Interpretación geométrica del rol de la unidad oculta



Nota. De "Artificial neural networks: a tutorial", por Jain, A. K., Mao, J., & Mohiuddin, K. M. (1996). (https://ieeexplore.ieee.org/document/485891/authors)

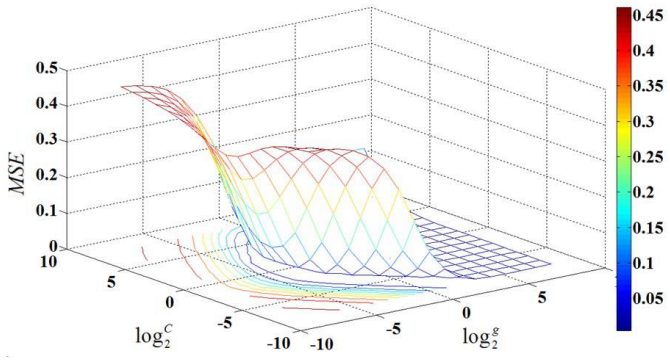
3.13. Método de optimización *Grid Search*

De acuerdo con lo revisado en el artículo de (Wang et al., 2012), el método de optimización "*Grid Search*" es conocido como el método del agotamiento, ya que se encarga de probar la mayor cantidad de combinaciones posibles de los valores parámetro de una función, en búsqueda de poder "afinar" la misma con los valores óptimos. Este método suele ser utilizado en diversas técnicas de *Data Mining* donde las funciones del modelo construido tengan una variedad de opciones para sus respectivos argumentos, tales como redes neuronales, modelos de SVM (*Support Vector Machine*), o árboles de decisión.

Se inicia con la definición del rango de valores para cada uno de los argumentos de la función que serán evaluados mediante múltiples iteraciones, así como también las métricas de evaluación para los resultados a obtener (como el MSE o el RMSE). Los valores obtenidos de cada combinación son almacenados para posteriormente compararlos e identificar el mejor resultado, dando como conclusión que aquel grupo de parámetros que arrojó este resultado serían los óptimos para la función del modelo planteado.

Figura 3.11

Distribución de resultados del Método de GridSearch



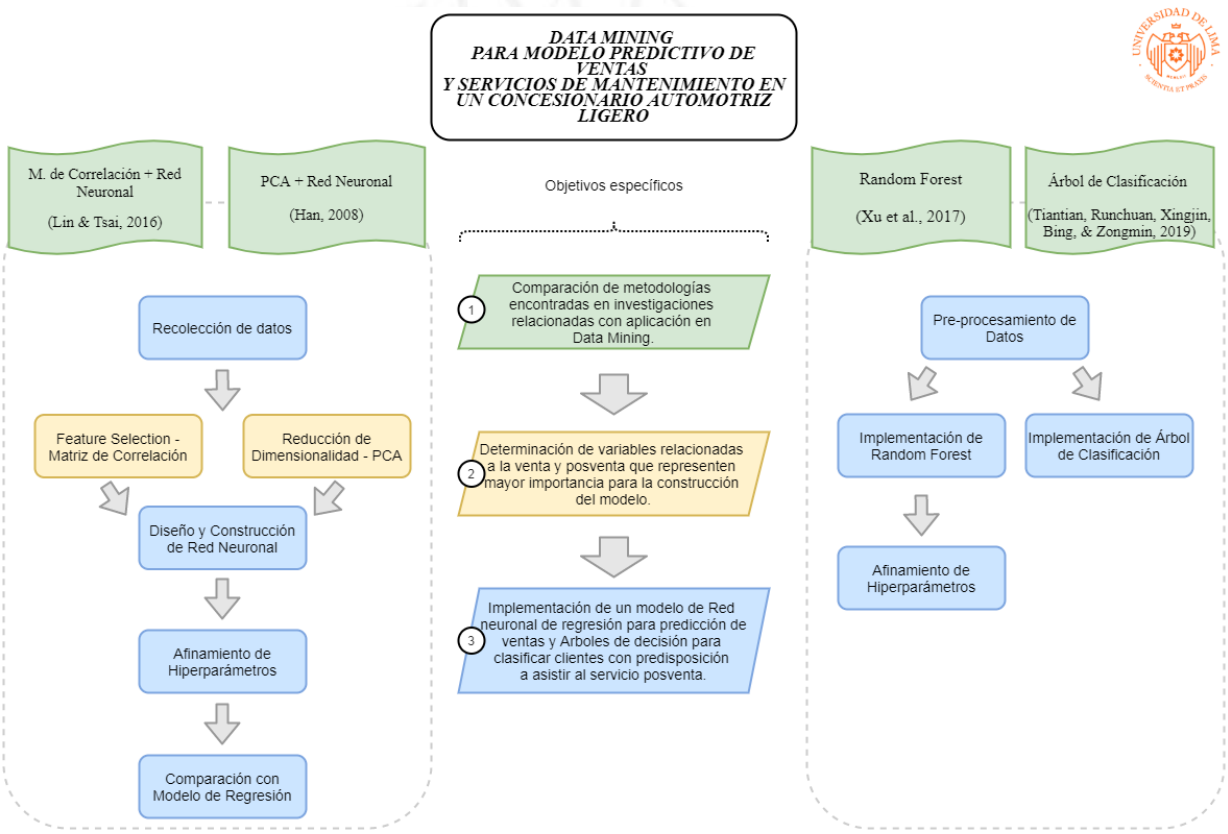
Nota. De "Grid Search Optimized SVM Method for Dish-like Underwater Robot Attitude Prediction" por (Wang et al., 2012). (https://ieeexplore.ieee.org/document/6274853)

CAPÍTULO IV: DESARROLLO DE LA SOLUCIÓN PROPUESTA

Nuestra investigación busca resolver la problemática descrita en los objetivos mencionados anteriormente, por ello se plantea utilizar dos técnicas de *Data Mining.*

Figura 4.12

Diagrama de estructura de la investigación



Métodos de Investigación

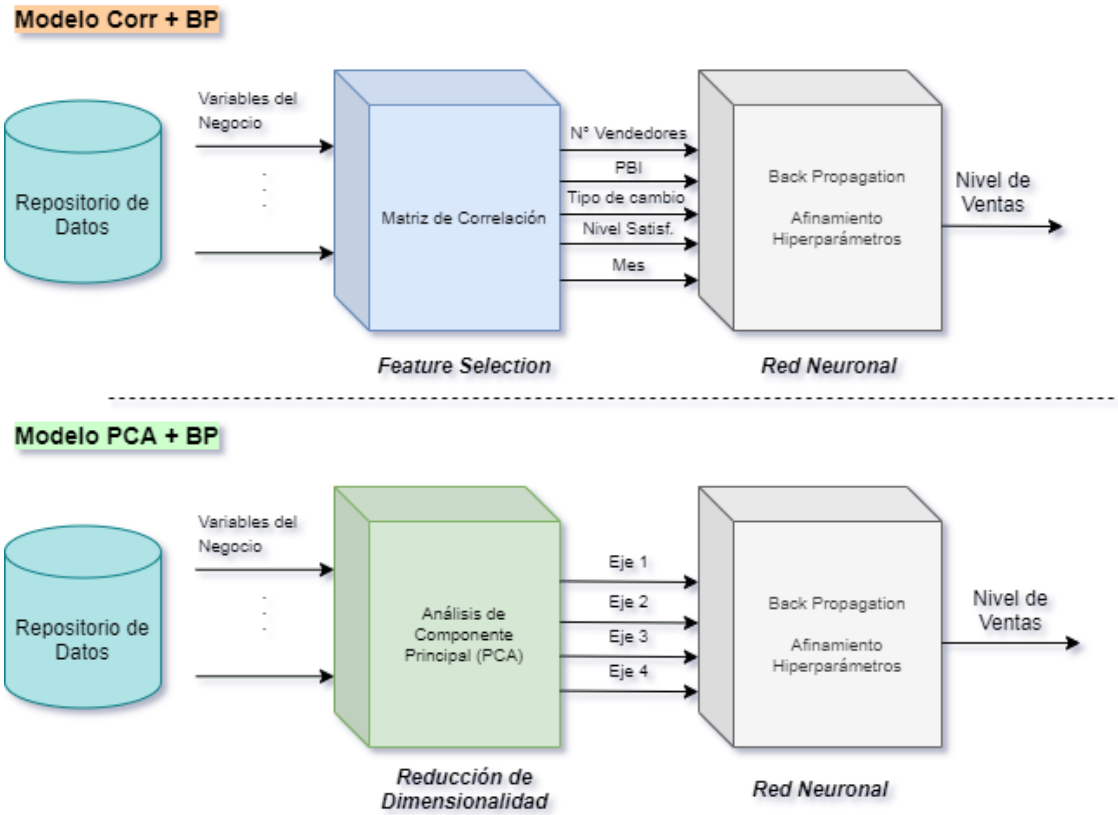
4.1.1 Predicción de Ventas Mensuales

Para el objetivo de conseguir el nivel de ventas para una marca de auto específica en la organización, se realizará una comparativa de la aplicación del método revisado en el estudio de los autores (Lin & Tsai, 2016). En su investigación se propone un enfoque *Deep Learning* para crear una herramienta capaz de predecir el nivel de ventas de cada

punto de venta en la organización. Dicho enfoque propone la utilización de variables que pueden afectar a la decisión del cliente al momento de la compra de un producto, la cuáles son seleccionadas a través de una matriz de correlación. Dicha metodología será contrastada con la de (Han, 2008), en la cual se buscó obtener un modelo clúster a partir de una red neuronal de *Back Propagation*. Seguidamente estas variables sirven como parámetros de entrada a un modelo de red neuronal el cual se encargará del procesamiento de la data para poder realizar las predicciones del nivel de ventas mensual por marca de auto que se busca encontrar. Cabe indicar que ambas investigaciones fueron aplicadas en diferentes tipos de organizaciones, donde el rubro del negocio es distinto con el que se trabajó en esta investigación. Lo que se busca con la comparativa es comprobar si las metodologías en estudio se adecuan al rubro de negocio permitiendo resolver la problemática de la organización.

Figura 4.13

Diagramas de estructuras de los modelos de predicción de ventas



A. Recolección de Datos

Dado que se busca estimar el nivel de las ventas de forma mensual por cada marca de la organización, se tomó en consideración aquellos datos relevantes que fueron indicados por los mismos especialistas del negocio. Las variables consideradas fueron las siguientes:

Tabla 4.1

Variables Propuestas para la investigación

N°	Variables Propuestas
1	Mes
2	Cantidad de Vendedores
3	Cantidad de Puntos de Venta
4	Cantidad de Prospectos
5	Nivel de Satisfacción del Cliente
6	Tipo de cambio (dólares)
7	PBI
8	Participación de Mercado

Se utilizó data histórica de ventas desde el año 2003 hasta 2018. Para completar el *dataset* se consultó información de diversas fuentes que maneja la organización, la mayoría de ellas de origen transaccional. Algunas de las variables presentadas son independientes a las acciones de la organización (PBI, Tipo de Cambio y Participación de mercado), sin embargo, se buscará corroborar si efectivamente están relacionadas al nivel de ventas mensual.

Tanto en la investigación de (Lin & Tsai, 2016) así como la de (Han, 2008) se sugiere realizar una estandarización de los datos con los que se van a trabajar. La razón para realizar dicha acción es que los datos se encuentran en diferentes magnitudes, por ejemplo, algunos están en medidas porcentuales y otros en medidas de unidad. La técnica propuesta para realizar dicha estandarización es la tipificación, la cual consiste en aplicar la siguiente fórmula a cada una de las columnas que conforman la data de entrada:

$$z = \frac{x-\mu}{\sigma}$$

Ecuación 4. Fórmula para normalización (Alhilman et al., 2014).

Donde z es el valor tipificado, μ la media aritmética, σ la desviación estándar y "x" el valor del dato original.

B. *Feature Selection* / Reducción de Dimensionalidad

1. Análisis de correlación

Luego de que la data ya se encuentra estandarizada, se aplican los métodos de selección de variables que se van a comparar. De acuerdo con la metodología de (Lin & Tsai, 2016) para una adecuada selección de variables es recomendable aplicar una matriz de correlación para hallar el valor "R" (coeficiente de Pearson). Esta medida permitirá conocer si existe algún tipo de dependencia lineal entre las variables planteadas. Los resultados obtenidos se interpretan con la Tabla 4.2.

Tabla 4.2

Interpretación de resultados

Valor R	Interpretación
1	Relación lineal perfecta
entre 0 y 1	Las variables tienden a tener una relación lineal positiva
0	No existe relación
entre -1 y 0	Las variables tienden a tener una relación lineal negativa
-1	Relación lineal inversa perfecta

Para realizar el análisis de correlación se utilizó el software Minitab 18 (Minitab, 2017) y se obtuvo la siguiente matriz (ver Tabla 4.3):

Tabla 4.3

Matriz de correlación obtenida

Variables	Mes	Cant. Vend.	Cant. Locales	Cant. Prospect.	Nivel Satisf.	Tipo Cambio	PBI	Part. Mercado.
Mes	X	0.005	-0.010	-0.033	0.044	0.013	0.003	-0.038
Cant.Vend.	0.005	X	0.935	0.822	-0.033	-0.139	-0.473	-0.704
Cant. Locales	-0.010	0.935	X	0.881	-0.081	-0.211	-0.509	-0.746
Cant. Prospect.	-0.033	0.822	0.881	x	-0.091	-0.424	-0.400	-0.640
Nivel Satisf.	0.044	-0.033	-0.081	-0.091	X	0.039	0.021	0.134
Tipo Cambio	0.013	-0.139	-0.211	-0.424	0.039	X	-0.187	0.243
PBI	0.003	-0.473	-0.509	-0.400	0.021	-0.187	x	0.495
Part. Mercado.	-0.038	-0.704	-0.746	-0.6040	0.134	0.243	0.495	x

De acuerdo con lo planteado por los autores (Lin & Tsai, 2016) se deben considerar como variables de entrada a la red neuronal a solo aquellas variables que presenten una independencia lineal entre las demás. Por ello se consideró solo al grupo de variables que posean entre sí valores r entre -0.25 y 0.25, dando como resultado lo siguiente (ver Tabla 4.4):

Tabla 4.4

Variables resultantes obtenidas

N°	Variables Obtenidas
1	Cantidad de Vendedores
2	Mes
3	Nivel de Satisfacción
4	Tipo de Cambio
5	PBI

2. Análisis de Componente Principal (PCA)

Según (Han, 2008) el método PCA debería poder seleccionar a las dimensiones más representativas de la data presentada, para luego tomar los valores de cada dimensión como entrada al modelo de red neuronal. Para la realización del análisis de componente principal se utilizó el software Tanagra (Tanagra, 2017). Este es un software libre cuyo propósito es apoyar investigaciones de *Data Mining*, tal como lo es el caso de este estudio. Los resultados arrojados fueron los siguientes (ver Figura 4.14).

Figura 4.14

Histograma resultante

Axis	Eigen value	Difference	Proportion (%)	Histogram	Cumulative (%)
1	3.773265	2.553417	47.17 %		47.17 %
2	1.219848	0.184973	15.25 %		62.41 %
3	1.034875	0.074077	12.94 %		75.35 %
4	0.960798	0.464025	12.01 %		87.36 %
5	0.496773	0.144788	6.21 %		93.57 %
6	0.351986	0.240657	4.40 %		97.97 %
7	0.111328	0.060201	1.39 %		99.36 %
8	0.051127	-	0.64 %		100.00 %
Tot.	8.000000	-	-	-	-

Se concluye que solo considerando hasta la dimensión 4 ya se contaría con la representatividad del 87% de la data original, por lo tanto, se podrían descartar las dimensiones restantes sin que afecte a la representatividad del modelo. La fórmula para cada dimensión (*eigen vector*) quedaría de la siguiente manera:

Tabla 4.5

Coeficientes de cada dimensión

Atributo	Media	Desv. Estándar	Eje_1	Eje_2	Eje_3	Eje_4
Mes	0.0000	0.9972	0.0026	0.1054	-0.7388	0.6604
Cant. Vend.	0.0000	0.9972	-0.4764	0.0727	-0.0473	-0.0614
Cant. Locales	0.0000	0.9972	-0.4954	0.0234	-0.0149	-0.0369
Cant. Prospectos	2.0828	0.9972	-0.4679	-0.1960	-0.0315	-0.0642
Nivel Satisf.	0.0121	0.9972	0.0610	0.1610	-0.6445	-0.7357
Tipo Cambio	0.0000	0.3269	0.1511	0.7883	0.1633	0.0536
PBI	0.0000	0.9972	0.3072	-0.0544	-0.0928	0.0043
Part. Mercado	0.0000	0.9972	0.4331	-0.0168	0.0009	-0.1012

Los valores de cada dimensión servirán como data de entrada para la red neuronal. Para hallar dichos valores cada coeficiente se multiplicará por el valor de la data original. Los factores quedarán dimensionados de acuerdo con la siguiente ecuación.

$$\begin{cases} F_1 = -0.002x_1 - 0.476x_2 - 0.495x_3 \ldots + 0.433x_p \\ F_2 = 0.105x_1 + 0.072x_2 + 0.0232x_3 \ldots - 0.016x_p \\ \ldots \end{cases}, \text{ Donde p = 8}$$

Ecuación 5. Representación de dimensiones obtenidas.

C. Diseño de la Red Neuronal

El tipo de red seleccionado para la presente metodología es el modelo *Multilayer Perceptron* (MLP). En las metodologías de ambos autores, se propone este tipo de red ya que lo que se busca obtener es un valor de salida numérico (regresión) por lo que quedarían descartados otros tipos de redes neuronales como las RNN (redes neuronales recurrentes). Se implementó la red neuronal en mención con la siguiente arquitectura:

1. Selección del Tipo de Red

Para la selección del tipo de red neuronal, según (Jain et al., 1996) se dispone de distintos tipos de redes neuronales. Según lo revisado hay dos clasificaciones las cuales están definidas de acuerdo con el objetivo que se busca cumplir. Tanto en la investigación realizada por (Lin & Tsai, 2016) como en la de (Han, 2008) se tomó el tipo de red neuronal llamado *Multilayer Perceptron Feed Forward.* Este es un modelo multicapa que es capaz de realizar iteraciones en una sola dirección.

2. Determinación del número de neuronas y número de capas ocultas

Ambas metodologías consideran una sola capa oculta para sus investigaciones. En el caso de las neuronas, para (Lin & Tsai, 2016) el número adecuado sería 13, mientras que para el caso (Han, 2008) se consideró 9.

Revisando otras investigaciones, se encontró que, para la determinación de este valor usualmente se entrenan los modelos hasta obtener el valor óptimo, probando diferentes cantidades de neuronas. Dentro de lo revisado encontramos un método utilizado por (Qin & Li, 2011) en su investigación. En ella se proponen 4 fórmulas para hallar la cantidad de neuronas que debe haber en una capa intermedia (oculta) como se ve en la imagen (ver Ecuación 6), donde m es la cantidad de variables en la capa de entrada, mientras que n es el número de variables en la capa de salida. Para nuestro estudio se aplicó la fórmula número 3.

$$q = \sqrt{mn} \quad (1)$$

$$q = \sqrt{mn} + a; a \in [1, \max(m, n)] \quad (2)$$

$$q = (m + n)/2 \quad (3)$$

$$q = m + 0.618(m - n) \quad (4)$$

Ecuación 6. Opciones de cálculo para número de neuronas (Qin & Li, 2011)

3. Algoritmo de Aprendizaje (Entrenamiento)

Como se había mencionado al principio, se utilizará el algoritmo de *Back Propagation* para el entrenamiento de la red. Según lo definido por (Simon, 2008) cuando se le presenta un patrón de entrenamiento a la red (p_1,t_1) (p_2,t_2), (p_3,t_3) … (p_q,t_q), este es propagado a través de las conexiones preexistentes produciendo una entrada n en cada uno de los nodos de la próxima capa. La entrada neta a la neurona j de la siguiente capa, debido a la presencia de un previo patrón de entrenamiento, está dada por la siguiente ecuación 7.

$$n_j^0 = \sum_{i=1}^{q} (\omega_{ji}^0 * p_i) + b^0$$

Ecuación 7. Cálculo de Pesos sinópticos (Simon, 2008)

Donde:

q: Número de componentes para el vector de entrada

m: Número de neuronas en la capa intermedia (oculta)

l: Número de neuronas de la capa de salida (*output*)

ω_{ji}^0: Peso que une la componente i en la capa de entrada con la neurona j de la capa intermedia.

p_i: Componente i del vector p que contiene el patrón de entrenamiento

b^0: Ganancia de la neurona en la capa oculta

$$a_j^0 = f^0(\sum_{i=1}^{q} (\omega_{ji}^0 * p_i) + b_j^0)$$

Ecuación 8. Cálculo de pesos sinópticos (Simon, 2008)

La salida de la neurona en la capa oculta está representada por a_j^0, las cuales finalmente serán las entradas a los pesos de conexión de la capa de salida. En la ecuación 8 se aplica la función de transferencia la cual procesa los pesos que conectan la capa oculta con la capa de salida. Finalmente, la capa de salida está descrita por la ecuación 9.

$$n_k^s = \sum_{j=1}^{m} (\omega_{ki}^s * a_j^0) + b_k^s$$

Ecuación 9. Cálculo del valor de salida de la red (Simon, 2008)

En donde,

ω_{ki}^s: Peso que une la neurona j de la capa intermedia con la neurona k de la capa de salida, que tiene “s” neuronas

a_j^0: Salida de la neurona j de la capa oculta, que cuenta con "m" neuronas

b_k^s: Ganancia de la neurona k en la capa de salida.

n_k^s: Entrada neta a la neurona k de la capa de salida

Finalmente, la red produce la salida: $a_k^s = f^s(n_s^k)$

Ecuación 10. Función de transferencia (Simon, 2008).

La segunda parte de *Back Propagation* indica que se debe calcular el error obtenido en cada iteración. Este se calcula restando lo obtenido por la red neuronal con la variable de salida original: $\delta_k = (t_k - a_k^s)$

Ecuación 11. Cálculo del error por iteración (Simon, 2008)

Se define la función de error para cada patrón de la siguiente forma:

$ep^2 = 1/2 \sum_{k=1}^{l} {\delta_k}^2$, donde ep = Error cuadrático medio

Ecuación 12. Fórmula del error cuadrático promedio (Simon, 2008)

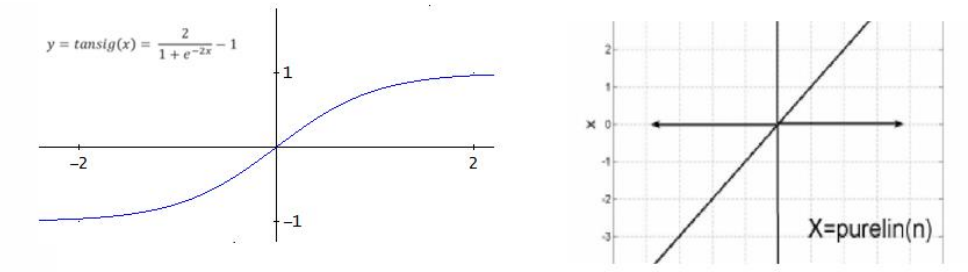
Esta función de error necesita ser optimizada para poder llegar a la convergencia, esto ocurre cuando el valor de la función de error toma el menor valor posible.

4. Funciones de Activación

Se definen dos funciones de activación para la red neuronal de BP propuesta. Ambas metodologías proponen la función de activación tangencial sigmoide en la capa oculta, mientras que una función lineal pura en la capa de salida. La primera según (Lin & Tsai, 2016), le permite a la red aprender las relaciones lineales y no lineales de las variables en estudio, permitiendo así una mejor convergencia. Finalmente, en la capa de salida ya no se utiliza otra transformación debido a que se busca obtener un resultado numérico (regresión) como output, por ello se utiliza la función lineal.

Figura 4.15

Funciones de Activación



Nota. De "A Deep Learning-Based Customer Forecasting Tool. In *2016 IEEE Second International Conference on Multimedia Big Data (BigMM)* (pp. 198–205)". Por Lin, K. Y., & Tsai, J. J. P. (2016). (https://doi.org/10.1109/BigMM.2016.85)

5. Validación Cruzada

Para verificar que los modelos de red neuronal han realizado un aprendizaje adecuado, se aplicará el método de validación cruzada. Este consiste en realizar una separación de la data en un determinado número de variables y buscar patrones de similitud entre estas separaciones, buscando que tengan un comportamiento uniforme. Una vez que se hayan obtenido estas separaciones, se procederá a calcular el error cuadrático y verificar el indicador. La variación de lo encontrado para cada separación no debería ser muy alta para que se pueda afirmar que la data de prueba es independiente de la data de entrenamiento lo cual le da mayor validez al modelo.

D. Afinamiento de hiperparámetros

La implementación de una red neuronal involucra ciertos parámetros a configurar para obtener la mejor predicción posible a través del modelo de red neuronal. A este proceso se le conoce como afinamiento de hiperparámetros (*HyperParameter Tuning*). Posterior a la aplicación de la metodología descrita hasta el punto anterior, se implementará un grupo de funciones que nos permitirán encontrar los mejores parámetros para nuestros modelos de red neuronal (corr. y pca). Entre ellos: *epochs*, batch_*size*, *learning rate*, *activation function*, *numbers of neurons* y *optimization algorithm*.

E. Comparación con un Modelo de Regresión

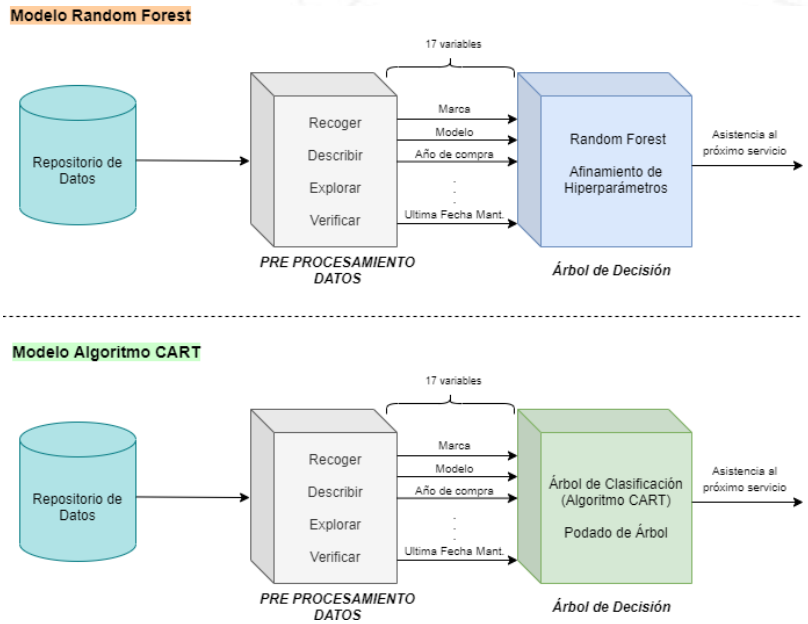
A fin de comparar la regresión obtenida por la red neuronal, se aplicará una comparativa a nivel de MSE con un modelo regresión lineal múltiple. Este tipo de modelo es el que más se ajusta a nuestro objeto de estudio revisado en la literatura de (Amral et al., 2007), ya que se busca obtener solo una variable de respuesta cuantitativa a partir de varias predictoras. Para este modelo se considerarán las mismas variables resultantes del método de selección que haya logrado mejor efectividad (Matriz de Corr. o PCA), y en caso el indicador de error salga muy elevado, se validará con todas las variables iniciales.

4.1.2 Predicción de Asistencias al servicio de Mantenimiento

Para poder lograr la clasificación de los clientes que estén predispuestos a asistir al servicio de mantenimiento se aplicará una combinación de metodologías encontradas en las investigaciones revisadas. Para el procesamiento de los datos nos basaremos en el estudio realizado por (Alhilman et al., 2014). Luego, se aplicará *Random Forest* para determinar las predicciones, esto según la implementación de (Xu et al., 2017). Asimismo, se realiza la comparativa con el árbol de clasificación de (Xie et al., 2019).

Figura 4.16

Diagramas de estructuras de los modelos de asistencias de servicios de mantenimiento



A. Preprocesamiento de Datos

De acuerdo con (Alhilman et al., 2014) esta es una actividad para determinar los correspondientes parámetros, variables y atributos relacionados con el problema. Consiste en la ejecución de los siguientes pasos:

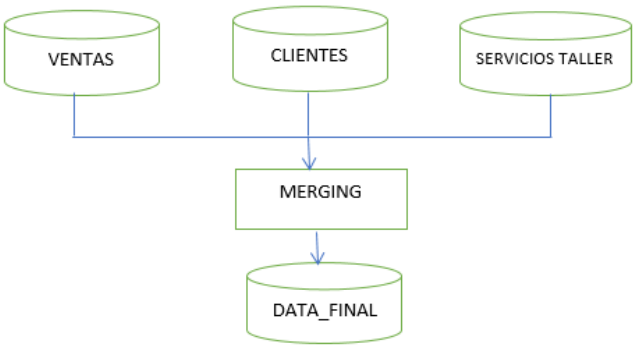
- Recoger datos: El *dataset* se obtiene de la información provista por la organización en estudio. Se recolectó los datos de ventas y clientes de un periodo que abarca abril 2013 hasta mayo 2017 así como también la información de los servicios de taller de mantenimiento de un periodo aproximado de 5 años.
- Describir datos: Los datos originalmente fueron exportados de los sistemas de información con nombres técnicos, los cuales serán reemplazados con una descripción acorde al negocio.
- Explorar datos: Consiste en la ejecución de los procedimientos que permitan la extracción de los datos para los periodos correspondientes necesarios para el proceso.
- Verificar calidad de datos: Se realiza con el fin de evitar el error que pueda originarse mientras se procesa los datos en el sistema.

Continuando con el enfoque de (Alhilman et al., 2014) se procedió a determinar las columnas de datos a ser utilizadas, para posteriormente proceder con la extracción de las bases de datos alocadas en los respectivos sistemas de la organización. Luego, elegimos y separamos las columnas cuyo contenido se utilizará para el modelado y predicción.

Los datos que son nulos o que no cumplen con el estándar fueron descartados. El proceso continuará con la construcción de un nuevo *dataset* de acuerdo con la siguiente estructura (ver figura 4.17).

Figura 4.17

Data set centralizado de los servicios de mantenimientos



B. Modelos de Clasificación

Una vez ya seleccionados los datos, se aplicaron dos técnicas de árbol de decisión con el objetivo de encontrar el modelo de clasificación más adecuado. Para el primero, se aplicó el algoritmo de *Random Forest*; mientras el segundo un árbol de clasificación (CART).

1. Modelo I - Algoritmo Random Forest

De acuerdo con la investigación de (Xu et al., 2017), es un algoritmo basado en la teoría del aprendizaje estadístico, que utiliza *Bootstrap* de forma aleatorio de re-muestreo con la finalidad de extraer varias versiones de conjuntos de muestras a partir de la data de entrenamiento.

El algoritmo de *Random Forest*, consiste en utilizar un conjunto de árboles de decisión con la finalidad de mejorar significativamente la precisión del modelo. Aquellos que se utilizan para la clasificación consiste en la coleccionar árboles individuales crecidos cada uno de una muestra de *Bootstrap* del mismo conjunto de datos.

La aplicación de dicho algoritmo para nuestro *dataset* se constituyó en los siguientes pasos:

- Paso 1: Utiliza técnicas de remuestreo de Bootstrap para generar k muestras. Estas muestras son 2/3 de los conjuntos de datos y el resto de los datos se llama *Out-of-*

bag (OOB), estos datos pueden utilizarse para las pruebas. Para nuestro modelo de *Random Forest* se utilizó el 60% de la data para entrenamiento y el 40 % para prueba.

- Paso 2: Utiliza las k muestras para formar k árboles de decisión. En cada nodo de cada árbol se selecciona aleatoriamente m características (m<m) en las M características, se sugiere empezar con m =$\sqrt{M}$ y luego disminuir o aumentar m hasta que se obtenga error mínimo para el conjunto de datos. Por último, elegir la mejor división de acuerdo con el criterio de Gini.

$$Gini\ (A_i)\ =\ 1\ -\sum_{i=1}^{n} p_i^2$$

Ecuación 13. Calcular la mejor división de acuerdo con el criterio de Gini (Xu, 2017).

Donde P_i representa la probabilidad de i-ésima instancia de clase, n es el número de clases; A_i representa el i-ésima característica. Se hizo una división de la data para determinar los conjuntos de entrenamiento y de prueba.

- Paso 3: Para predecir las muestras de prueba, y combinar los resultados de cada árbol; para finalmente determinar el resultado de acuerdo con el mecanismo de votación por mayoría.

2. Modelo II - Árbol de Clasificación

De acuerdo con (Xie et al., 2019), se utilizó un árbol de decisión de clasificación basado en el algoritmo CART, que consiste en podar el árbol para evitar el *overfitting*, permitiendo mejorar la clasificación de los datos.

El algoritmo CART consiste en la generación y poda del árbol de decisión. Para la construcción del modelo se utiliza el índice de Gini mínimo para elegir las mejores características.

Donde K es una clase, y la probabilidad que pertenezcan a la clase en P_k, el índice de Gini se define como:

$$Gini\ (p)\ =\ 1\ -\sum_{k=1}^{K} p_k^2$$

Ecuación 14. Calcular el índice de Gini (Xie et al., 2019)

4.2. Alcance

Luego de que las metodologías hayan sido aplicadas, se procederá a analizar los resultados obtenidos buscando identificar cuál de ellos consiguió mejor precisión a la hora obtener las predicciones. El alcance de esta propuesta se divide en lo siguiente:

- Obtención de un modelo óptimo para las predicciones del nivel de ventas para una marca de auto en específico con la que trabaje la organización.

- Obtención de un modelo que permita clasificar a aquellos clientes que estén próximos a realizar un servicio de taller, agrupados por características comunes para una marca de automóviles.

- Se utilizará Python como lenguaje de programación para la construcción de la red neuronal y R para la clasificación de árbol de decisión.

CAPÍTULO V: PRUEBAS Y RESULTADOS

5.1. Predicción de Ventas

Se implementaron los modelos de redes neuronales con el uso de *Anaconda*, la cual es una distribución de Python para *Scientific Computing*, dentro de los cuales destacan la aplicabilidad en *Data Mining.*

Se empleó la librería de Google conocida como *TensorFlow*. Esta se usó como *backend* para correr el API de Keras (Keras, 2017) , sobre el cual se construyeron finalmente los modelos a ser comparados. Keras brinda un modelo de red neuronal conocido como *Sequential,* el cual inicializa una matriz con pesos sinápticos calculados aleatoriamente, para luego ir acoplándose a las relaciones existentes de las variables de entrada. A partir de esto, se le puede ir agregando capas con las cantidades de neuronas requeridas, así como también seleccionar la función de activación que cumpla con mejor desempeño. A los parámetros mencionados se le dieron las siguientes especificaciones adicionales al modelo, tomando como referencia la literatura revisada:

Tabla 4.6

Parámetros Adicionales

Parámetros Adicionales	Valor
Número de interacciones	150
Batch Size	5
Función de pérdida	Mean Square Error
Optimizador	SGD
Cantidad de datos de entrenamiento (60% del total)	107
Cantidad de datos de entrenamiento (40% del total)	72

La métrica que se utiliza para le evaluación de los modelos será el *Mean Square Error* (MSE) ya que al ser una red neuronal de regresión la precisión se calcula a través de la función de costo, la cual estima el mejor modelo como el que tiene el menor error respecto a las predicciones realizadas

5.1.1. Primer Enfoque: Datos ordenados por fecha

Inicialmente, luego de tener los métodos construidos se realizaron las primeras ejecuciones para ver el comportamiento de la red. La métrica recomendada y revisada en la literatura de los autores para comprobar el desempeño de un modelo será el MSE (*mean square error*). La interpretación de este valor indica que aquel modelo que tenga el menor valor es el óptimo y preciso.

Tabla 5.7

Resultados del primer enfoque

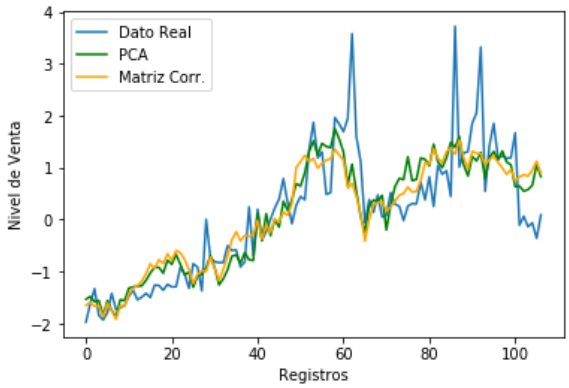
Entrenamiento (Valor MSE)	
Matriz Correlación	0.43908
PCA	0.39497
Pruebas (Valor MSE)	
Matriz Correlación	1.36216
PCA	0.68516

Se aprecia que los valores MSE de ambas técnicas de selección en la fase de entrenamiento son bastante cercanos, teniendo una diferencia de 0.0441 puntos. Sin embargo, en la fase de pruebas se aprecia que el modelo se vuelve demasiado impreciso. Se observa en la figura 5.19 un gráfico con la proyección de la data real, en comparación con lo encontrado por el modelo.

- Entrenamiento

  Figura 5.18

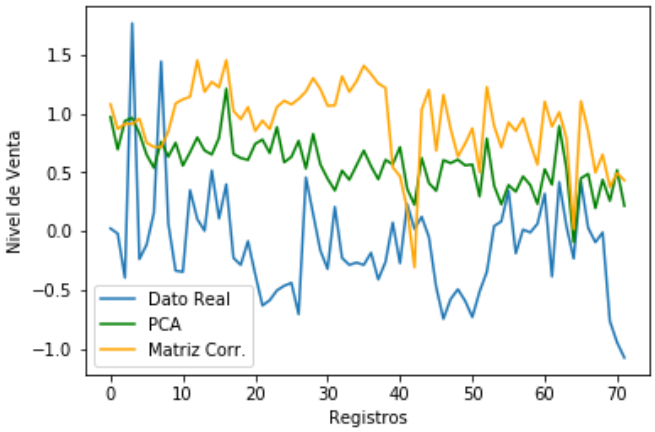
  Entrenamiento Enfoque 1



- Pruebas

Figura 5.19

Pruebas enfoque 1



5.1.2. Segundo Enfoque: Ordenamiento Aleatorio

En el estudio de (Lin & Tsai, 2016) se recomendaba que, para incrementar la precisión del modelo, se debería ordenar los datos aleatoriamente, ya que esto le permite al modelo conocer mejor la relación entre los datos sin condicionarlos a su valor en el tiempo. Dado que con este cambio en cada ejecución se tendría resultados distintos por el tema de la aleatoriedad, se procedió con 10 ejecuciones distintas, almacenando los resultados de cada una para posteriormente determinar cuál de ellos logró un mejor desempeño. Los resultados de esta nueva ronda fueron los siguientes:

- Fase de Entrenamiento (Ordenamiento Aleatorio)

Tabla 5.8

Resultados: Entrenamiento enfoque 2

Repetición	1		2		3		4		5	
Método	CORR	PCA	CORR	PCA	CORR	PCA	CORR	PCA	CORR	PCA
Valor MSE	0.30436	0.24899	0.31804	0.27938	0.32310	0.22527	0.36915	0.32074	0.32319	0.22505
Repetición	1		2		3		4		5	
Método	CORR	PCA	CORR	PCA	CORR	PCA	CORR	PCA	CORR	PCA
Valor MSE	0.41564	0.34305	0.32311	0.22504	0.41560	0.34291	0.32318	0.22512	0.41589	0.34293

Los resultados del nuevo entrenamiento muestran que, en todas repeticiones la red neuronal con origen de datos PCA tuvo una mejor performance que la red con origen de matriz de correlación (CORR), teniendo un valor de MSE menor. La diferencia porcentual promedio del error cuadrático entre ambos fue de 0.07528 puntos decimales a favor del modelo PCA.

Analizando las iteraciones realizadas, se observó que la técnica de optimización conocida como Gradiente Descendiente Estocástica (SGD), logró una convergencia rápida de los pesos sinápticos del modelo. Revisando las diferentes investigaciones, concluimos que la aplicación de un algoritmo de optimización más complejo, como el caso de ADAM, no aplicaría en nuestro caso de estudio, ya que el modelo de red neuronal presenta una cantidad de datos no muy amplia. Generalmente estas técnicas se aplican a modelos con cientos de miles de datos.

- Fase de Pruebas (Ordenamiento Aleatorio)

Tabla 5.9

Resultados: Pruebas enfoque 2

Repetición	1		2		3		4		5	
Método	CORR	PCA	CORR	PCA	CORR	PCA	CORR	PCA	CORR	PCA
Valor MSE	0.61194	0.45035	0.52487	0.41517	0.66624	0.48398	0.46494	0.36835	0.66621	0.48371
Repetición	1		2		3		4		5	
Método	CORR	PCA	CORR	PCA	CORR	PCA	CORR	PCA	CORR	PCA
Valor MSE	0.33643	0.30993	0.66604	0.48371	0.33649	0.30992	0.66617	0.48382	0.33626	0.31000

De lo obtenido se puede concluir que el modelo basado en PCA sigue siendo predominante a la de análisis de correlación, en lo que respecta a mejor desempeño. La diferencia porcentual promedio del error cuadrático entre ambos fue de 0.11767 puntos decimales a favor del modelo PCA.

En concordancia con la fase de pruebas, esta técnica sería la más adecuada para realizar predicciones con nuevos datos.

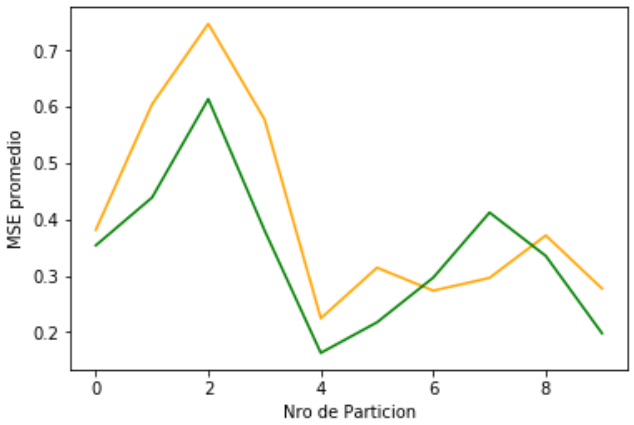
5.1.3. Validación Cruzada

Para validar que los resultados obtenidos en las ejecuciones de los métodos no presenten *overfitting*, nos apoyaremos con la técnica conocida como validación cruzada (*cross validation*) k-fold. Consistió en dividir el *dataset* completo en 10 particiones, de tal forma que se entrenan 9 y luego se prueba con la restante. Al igual que en la predicción, este proceso se repitió 10 veces con cada nuevo ordenamiento aleatorio generado. El rango de valores MSE obtenido como resultado de esta validación fue el siguiente:

- Resultados Red Corr: [0.41,0.45] MSE (0.17) Dev → Naranja
- Resultados Red PCA: [0.34, 0.36] MSE (0.13) Dev → Verde

Figura 5.20

Validación cruzada

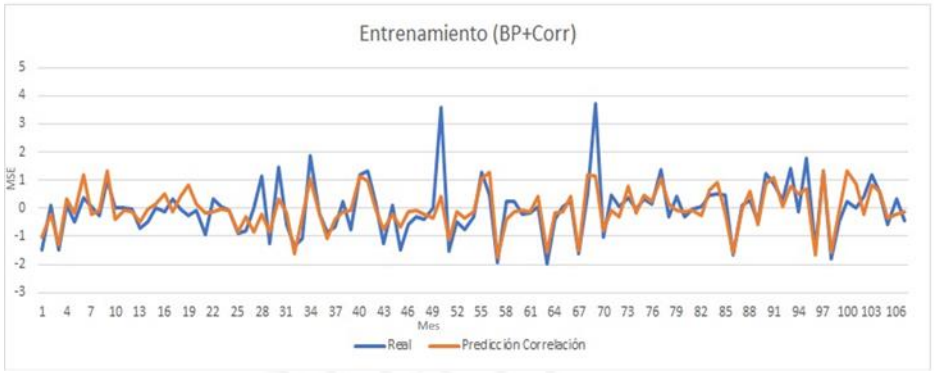


Comparando los rangos de los valores MSE encontrados por esta técnica, encontramos que algunas de las repeticiones realizadas en la predicción presentaban *overfitting*. Se infiere que el modelo número 4 (revisar Figura 5.24) sería el que obtuvo los valores más aproximados al rango de validación cruzada, por lo que sería el modelo con mejor desempeño que no presenta *overfitting.* A continuación, los gráficos correspondientes a los resultados encontrados en la repetición del modelo número 4:

- Matriz de Correlación – Entrenamiento

Figura 5.21

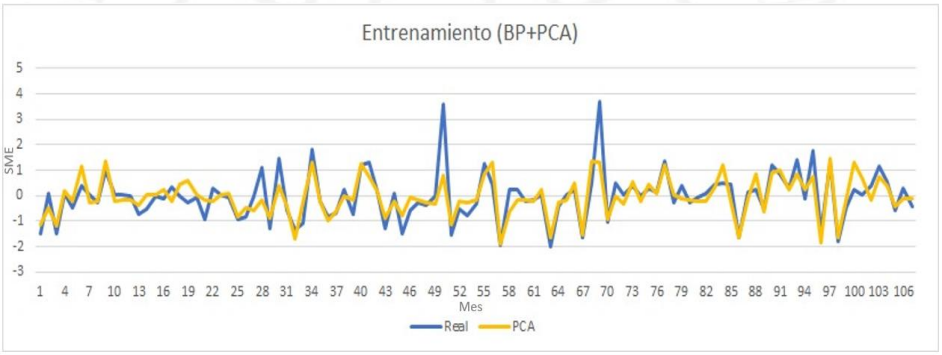
Entrenamiento Corr + BP



- PCA – Entrenamiento

Figura 5.22

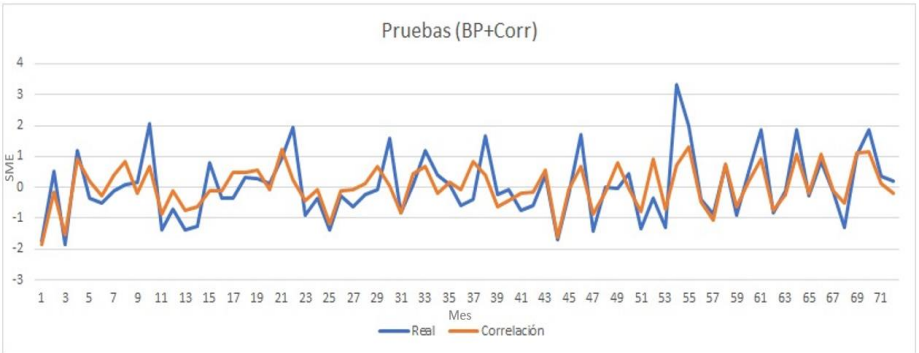
Entrenamiento PCA + BP



- Matriz de Correlación – Pruebas

Figura 5.23

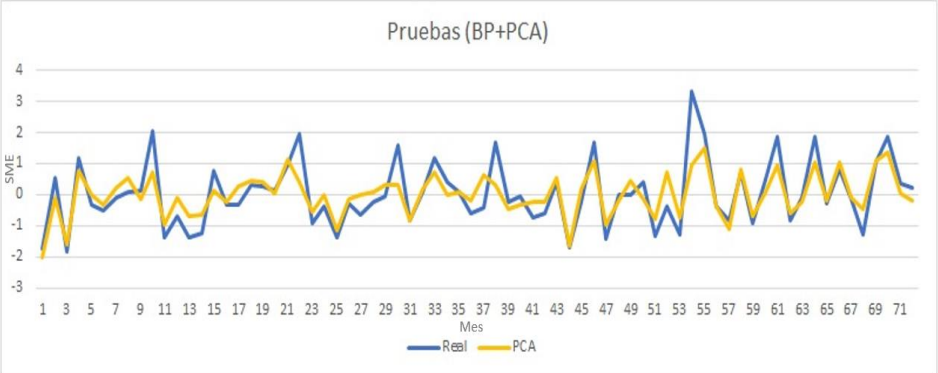
Pruebas Corr + BP



- PCA – Pruebas

Figura 5.24

Pruebas PCA + BP



5.1.4. Afinamiento de Hiperparámetros

Como se mencionó en la descripción de la metodología, el modelo creado necesita ser afinado a través de los hiperparámetros de configuración que nos brinda la API de Keras, y de esta manera mejorar los resultados obtenidos.

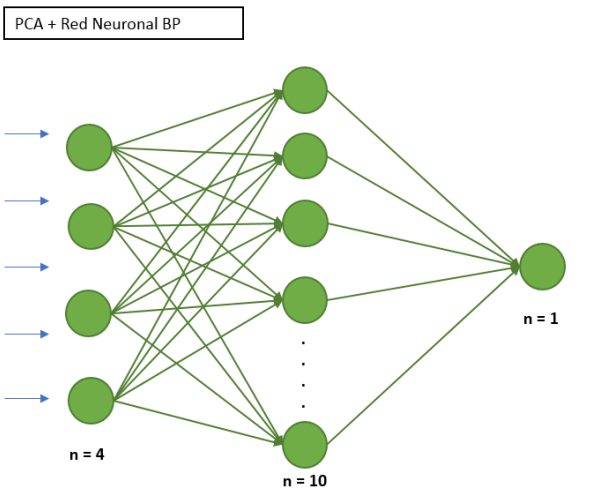
Para ello se utilizaron las librerías *KerasRegressor* y la función *GridSearchCV*. La primera de ellas nos pide pasar como parámetro una función que contemple al modelo S*equential* en su definición, mientras que la segunda pide como argumentos el modelo *KerasRegressor*, un diccionario de datos con los nombres de los hiperparámetros a afinar indicando el valor que tomarían, y finalmente la métrica a evaluar.

Se pueden seleccionar diferentes hiperparámetros al mismo tiempo, ya que la función se encarga de realizar todas las combinaciones posibles, brindando como resultado el valor de las métricas para cada una de ellas, así como también el mejor resultado con la descripción de los valores asignados como hiperparámetro.

En pseudocódigo se presenta la función del modelo original (ver figura 5.25):

Figura 5.25

Pseudocódigo planteado para el modelo inicial



Se generó una función para poder ejecutar el afinamiento con *GridSearch* un total de 50 veces. Del resultado obtenido se tomaron los 5 mejores grupos de parámetros que obtuvieron un menor MSE y se validó cual obtenía un mejor rendimiento con la data de prueba.

Figura 5.26

Pseudocódigo Planteado para el afinamiento de hiperparámetros

```
def hyperparameter_tuning(n=10,lr=0.1,OP='Adam',IM='Normal',AF='relu'):
        model_pca = Sequential()
        model_pca.add(Dense(n,input_dim=4,init_mode=IM,activation=AF))
        model_pca.add(Dense(1))
        model_pca.compile(loss='MSE', optimizer=OP,metrics=['MSE'])
        return model_pca

C = 0

while C < 50 :
        model_tuned = KerasRegressor(build_fn=hyperparameter_tuning, verbose=0)
        BZ = [20,40,60]
        EP = [10,50,100]
        LR = [0.01,0.1,0.3]
        OP = ['SGD', 'RMSprop', 'Adagrad', 'Adadelta', 'Adam', 'Adamax', 'Nadam']
        IM = ['uniform', 'lecun_uniform', 'normal','glorot_normal', 'glorot_uniform', 'he_normal', 'he_uniform']
        AF = ['softmax', 'softplus', 'softsign', 'relu', 'tanh', 'sigmoid', 'hard_sigmoid', 'linear']
        n = [1 , 5, 10 ,15 ,25 ,30 ]
        param_grid = dict(batch_size=BZ , epochs=EP , learn_rate=LR , optimizer=OP, init_mode = IM , activation = AF , neurons = n)
        grid = GridSearchCV(estimator=model_tuned , param_grid=param_grid , scoring='neg_root_mean_squared_error')
        grid_result = grid.fit(Train_Pca_X , Train_Y)
        best_score = grid_result.best_score_
        best_params=grid_result.best_params_
        C = C +1
```

Finalmente, los resultados obtenidos arrojaron como mejores parámetros los siguientes valores:

Tabla 5.10

Mejores resultados después de afinamiento

	RMSE	Batch_size	Epochs	Learn_rate	Optimizer	Init_mode	Activation	Neurons
1	0.7594	40	50	0.3	RMSprop	Lecun_uniform	Tanh	15
2	0.7645	40	50	0.1	SGD	He_uniform	Tanh	15
3	0.7886	40	50	0.3	SGD	He_uniform	Relu	15
4	0.7898	40	50	0.1	SGD	He_normal	Softplus	25
5	0.8145	60	50	0.1	RMSprop	Lecun_uniform	Tanh	20

Se seleccionó la terna de hiperparámetros con el RMSE más bajo, para posteriormente ejecutar predicciones nuevamente con el modelo de PCA. Bajo los dos enfoques vistos en la etapa anterior, el modelo se corrió esta vez contemplando los nuevos hiperparámetros obtenidos, y se obtuvo como resultado lo siguiente:

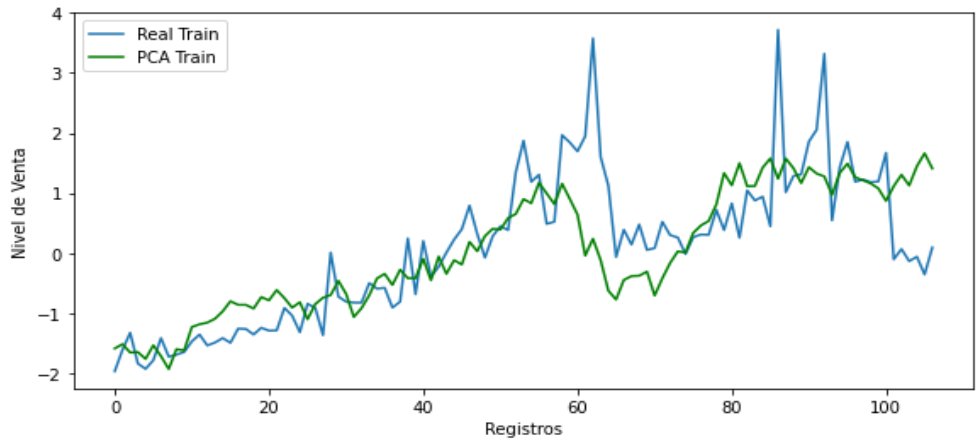
1er Enfoque: Datos ordenados por fecha

- Entrenamiento antes / después del afinamiento:

MSE: 0.4894

Figura 5.27

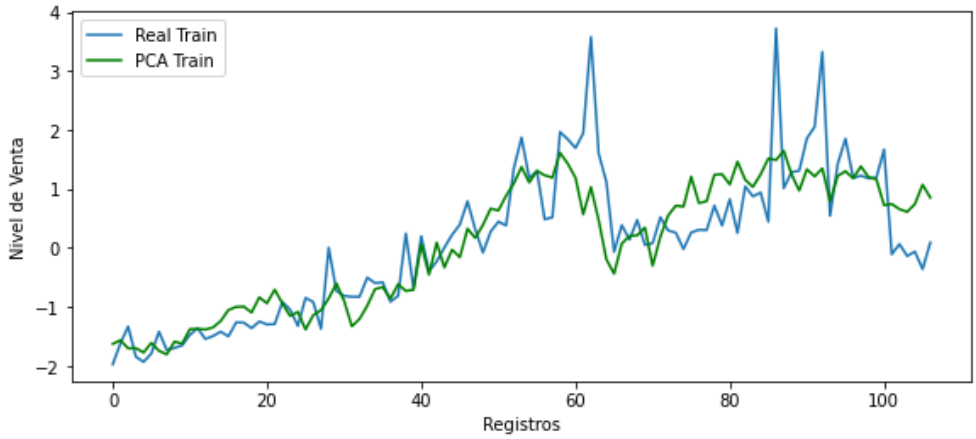
Entrenamiento BP sin afinamiento – 1er Enfoque



MSE: 0.4215

Figura 5.28

Entrenamiento BP con afinamiento – 1er Enfoque

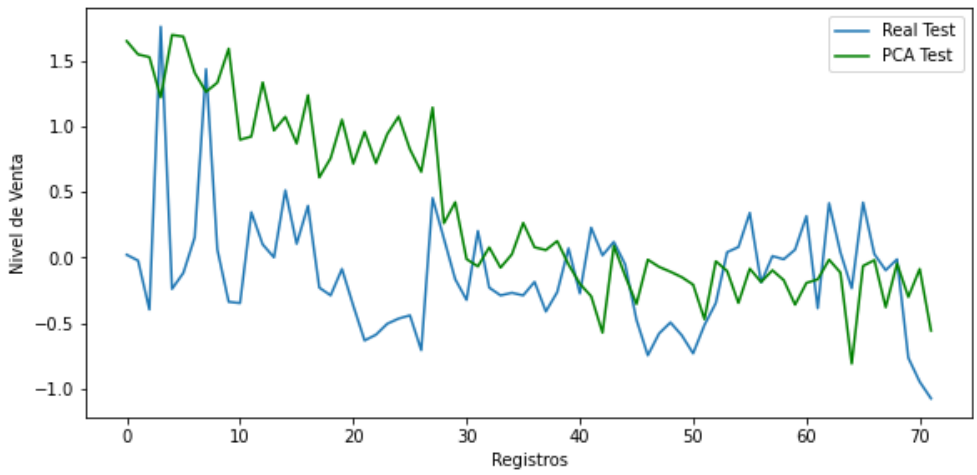


- Pruebas antes / después del afinamiento:

MSE: 0.6898

Figura 5.29

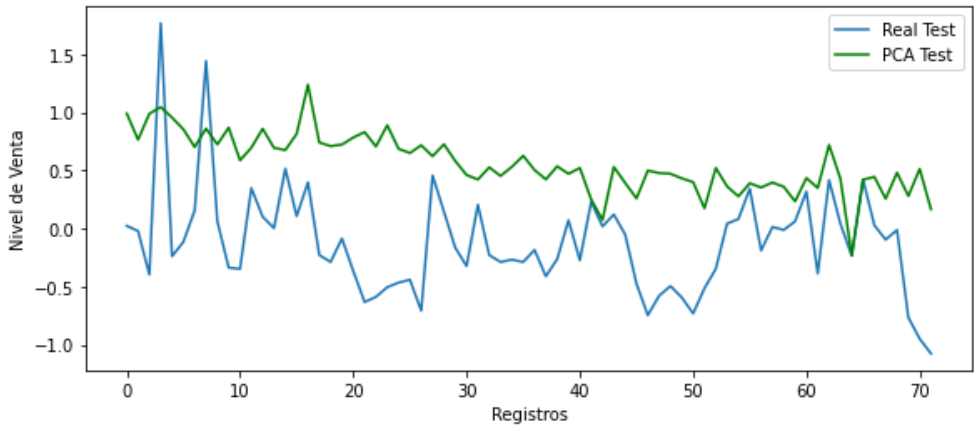
Pruebas BP sin afinamiento – 1er Enfoque



MSE: 0.5368

Figura 5.30

Pruebas BP con afinamiento – 1er Enfoque



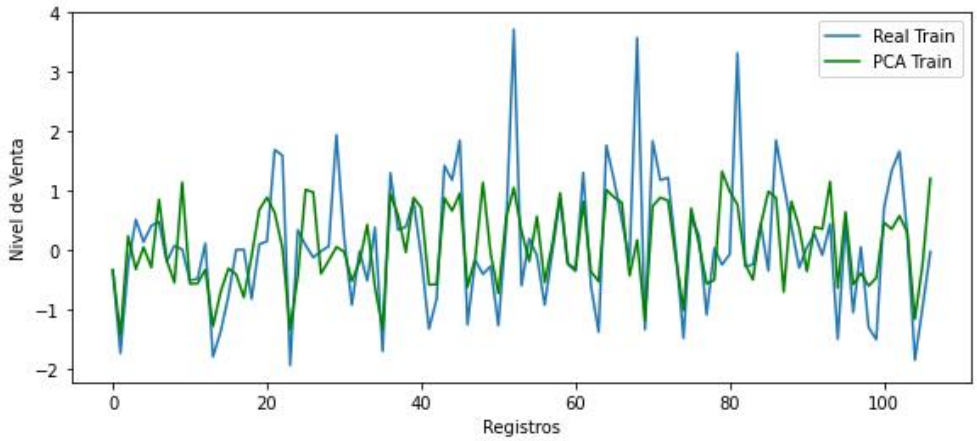
2do Enfoque: Datos ordenados aleatoriamente

- Entrenamiento antes/después del afinamiento:

MSE: 0.3351

Figura 5.31

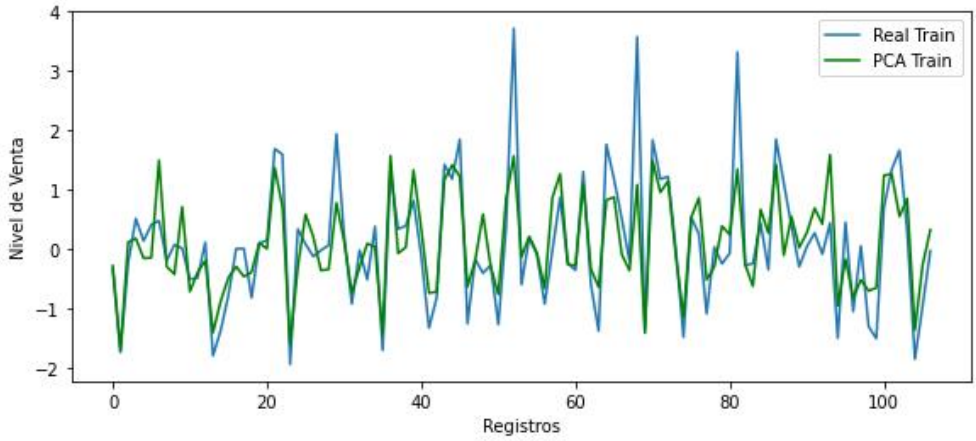
Entrenamiento BP sin afinamiento – 2do Enfoque



MSE: 0.2057

Figura 5.32

Entrenamiento BP con afinamiento – 2do Enfoque

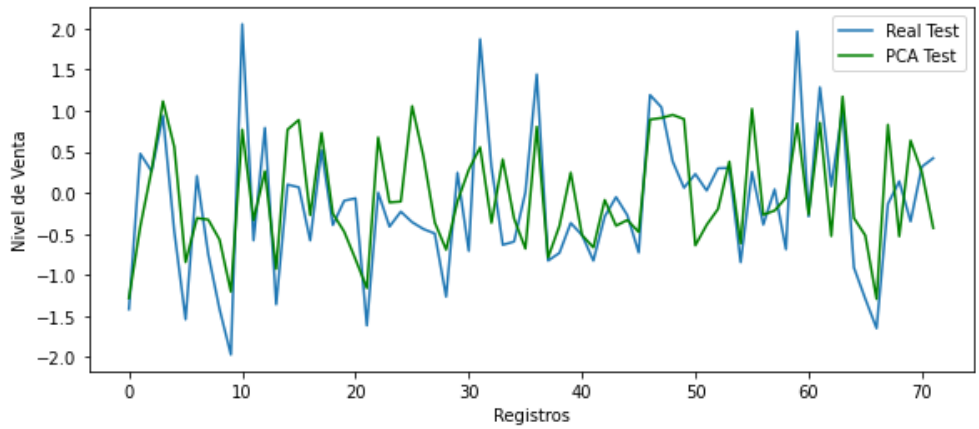


- Pruebas antes/después del afinamiento:

MSE: 0.3839

Figura 5.33

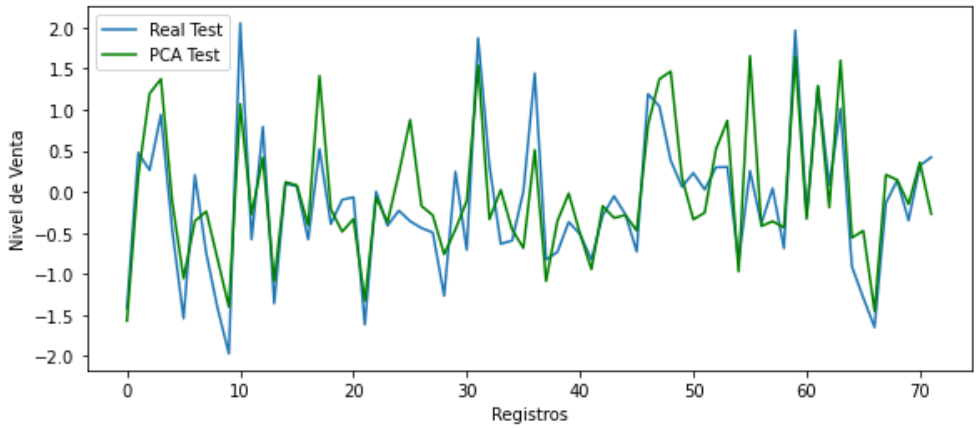
Pruebas BP con afinamiento – 2do Enfoque



MSE: 0.2412

Figura 5.34

Pruebas BP con afinamiento – 2do Enfoque



5.1.5. Comparación con Regresión Lineal Múltiple

Luego de determinar que el modelo de selección de datos más acertado fue el de PCA con la red neuronal de *feed forward*, se construyó un modelo de regresión lineal múltiple en el software Minitab, para comparar la eficacia de ambas técnicas aplicadas.

Se consideraron todas las variables de nuestro *dataset* original, para la construcción de este modelo, el cual dio un valor de MSE de 0.3263.

Se obtuvo como coeficientes de correlación los valores revisados en la tabla 5.11, así como el gráfico de las predicciones obtenidas frente a los valores reales (Figura 5.35)

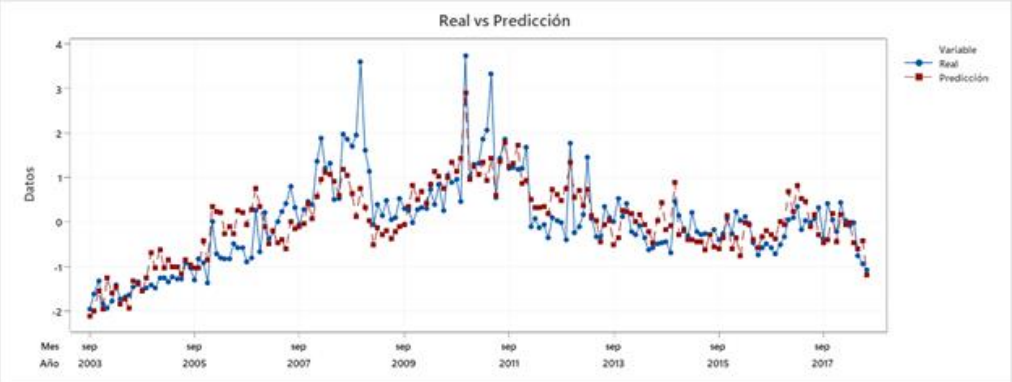
Tabla 5.11

Análisis de regresión lineal múltiple - Correlación

Término	Coef	EE del coef.	Valor T	Valor p	FIV
Constante	0.209	0.157	1.33	0.184	
CantVend	-0.583	0.133	-4.39	0.000	0.861
CantLcocales	1.207	0.166	7.25	0.000	13.50
CantProspectos	0.062	0.116	0.54	0.592	6.61
NivelSatisf	0.172	0.144	1.19	0.235	1.09
Tcambio	-0.448	0.067	-7.27	0.000	1.85
PBI	0.322	0.059	5.44	0.000	1.71
PartMercado	0.514	0.0739	6.96	0.000	2.66

Figura 5.35

Resultados Real Vs Predicción



Finalmente, para validar la consistencia del modelo de regresión obtenido se aplicó la técnica Durbin-Watson tal como lo trabajo (Abdollahian & Foroughi, 2005). En la cual se plantearon la hipótesis nula (Ho), la cual afirma que no existe autocorrelación; y la hipótesis H1, indicando lo contrario. Se trabajó con los residuos obtenidos con la siguiente Ecuación 15.

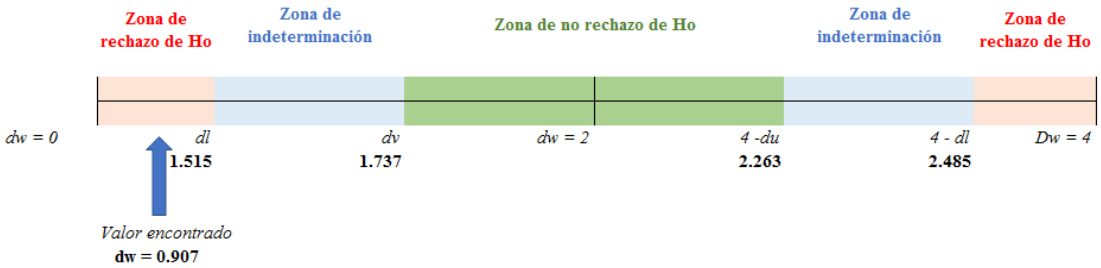
$$dw = \frac{\sum_{t=2}^{T}(e_t - e_{t-1})^2}{\sum_{t=1}^{T} {e_t}^2}$$

Ecuación 15. Cálculo de Durbin Watson (Abdollahian & Foroughi, 2005).

El valor obtenido para *dw* fue de 0.907, con la cual se realizó la distribución revisada en la figura 5.36. De acuerdo con el valor obtenido se debería rechazar la hipótesis nula (Ho) indicando que el modelo construido presenta autocorrelación. Se puede inferir que algunas de las columnas del *dataset* original no serían significativas para predecir de forma eficaz la variable dependiente.

Figura 5.36

Resultados de Durbin Watson



5.1.6. Resultados

Las ejecuciones del modelo sin afinamiento obtuvieron los siguientes resultados. Se trabajó con ambos enfoques, ya que se observó que con el ordenamiento aleatorio se obtenían mejores resultados.

Tabla 5.12

Resultados sin afinamiento

Fases	**1er Enfoque - Ordenamiento por Fecha**		**2 do Enfoque - Ordenamiento Aleatorio (10 ejecuciones)**	
Entrenamiento	MSE - M. Corr.	0.43908	MSE - Promedio M. Corr.	0.35313
	MSE – PCA	0.39497	MSE - Promedio PCA	0.27785
Pruebas	MSE - M. Corr.	1.36216	MSE - Promedio M. Corr.	0.52756
	MSE – PCA	**0.68516**	MSE - Promedio PCA	**0.40989**

Se tomó como *dataset* ganador al conjunto de dimensiones obtenido a través de PCA, ya que obtuvo un MSE menor en ambos enfoques. Los resultados luego del afinamiento hicieron que los resultados mejoraran considerablemente, reduciendo el error en casi un 40% respecto al modelo sin afinamiento.

Tabla 5.13

Resultados con afinamiento

Fases	**1er Enfoque - Ordenamiento por Fecha**		**2 do Enfoque - Ordenamiento Aleatorio (10 ejecuciones)**	
Entrenamiento	MSE - M. Corr.	0.4894	MSE - Promedio M. Corr.	0.3351
	MSE – PCA	0.4215	MSE - Promedio PCA	0.2057
Pruebas	MSE - M. Corr.	0.6898	MSE - Promedio M. Corr.	0.3839
	MSE – PCA	**0.5368**	MSE - Promedio PCA	**0.2412**

5.2. Asistencia al servicio de Mantenimiento

Los árboles de decisión fueron construidos para el servicio de mantenimiento de 5 km en la herramienta R Studio.

5.2.1 Random Forest

Para el modelo de *Random Forest* se utilizó 17 variables identificadas y se dividió la cantidad de registros; 60% data de entrenamiento y el 40% de prueba. La librería *randomForest* se usó para la construcción del modelo.

Para realizar el entrenamiento del modelo se obtuvo como valores por defecto. Para *mtry*=4, se emplea como valor la raíz cuadrada del número total de variables predictoras, *mtry* es el número de variables seleccionadas aleatoriamente en cada ramificación y n*tree*= 500, la cantidad de árboles a utilizar.

Tabla 5.14

OBB Error con 4 variables

mtry	**4**
OOB Error	5.15%

Utilizamos la matriz de confusión para comprobar que los datos predichos sean iguales a la variable independiente Servicio. Se obtiene la siguiente matriz

Tabla 5.15

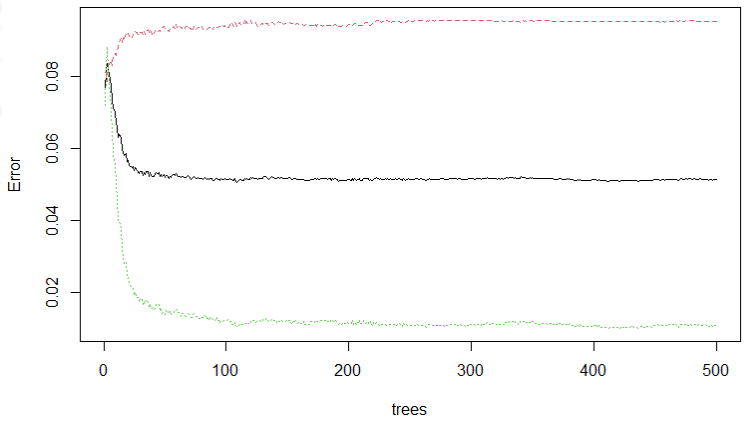
Matriz de confusión - Entrenamiento (Random Forest)

	NO	SI	Error de Clasificación
NO	3837	406	0.095
SI	49	4545	0.010

El siguiente gráfico muestra el OBB error vs el número de árboles. Se puede observar que se reduce el error a medida que se aumenta la cantidad de árboles. Cabe mencionar que, línea de color negro es *OOB Error* del modelo *Random Forest*, la línea verde es el % error de la variable observada cuando la variable objetivo es SI y la línea roja es el % error si la variable objetivo es NO.

Figura 5.37

OOB Error vs el número de árboles



Para las pruebas del modelo *Random Forest*, se obtuvo la siguiente matriz de confusión, teniendo como resultado una precisión del modelo 95.08%.

Tabla 5.16

Matriz de confusión - Pruebas (Random Forest)

	NO	SI	Error de Clasificación
NO	2485	31	0.012
SI	252	2995	0.007

Asimismo, se obtiene como resultado una precisión de 95.08% para el modelo Random Forest.

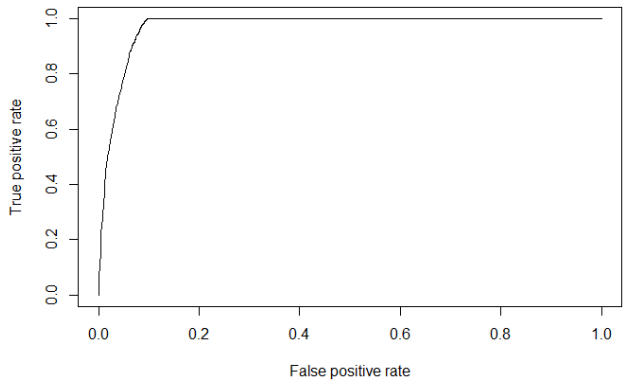
- **Curva ROC**

Este gráfico nos permite visualizar el rendimiento del modelo clasificador frente a todos los umbrales de clasificación.

Para el modelo de Random Forest se obtiene un AUC=0.98, se muestra la gráfica de Curva ROC.

Figura 5.38

Curva ROC - Random Forest sin afinamiento



- **Afinamiento de hiperparámetros**

Se realizó un afinamiento de hiperparámetros al modelo *Random Forest*, se utilizó *Grid Search* con la finalidad de encontrar la mejor combinación hiperparámetros que mejore la precisión del modelo.

El ajuste se realizó para los hiperparámetros:

- *ntree*: Número de árboles a utilizar
- *mtry*: Número de variables predictoras consideradas en cada división
- *max_depth*: Profundidad máxima que los árboles pueden generar

Se obtuvo los mejores hiperparámetros por *Out-of-bag error*

Tabla 5.17

Grid search basado en Out-of-bag-error

Ntree	Mtry	max_depth	OOB Error
5000	3	10	4.88%

Se obtuvo los mejores hiperparámetros por validación cruzada

Tabla 5.18

Grid search basado en validación cruzada

Ntree	Mtry	max_depth	Precisión
50	3	1	95.3%

Luego de identificar los mejores hiperparámetros, se realiza el reentrenamiento del modelo con los valores óptimos. Finalmente, se obtiene una precisión del modelo 95.31%.

Tabla 5.19

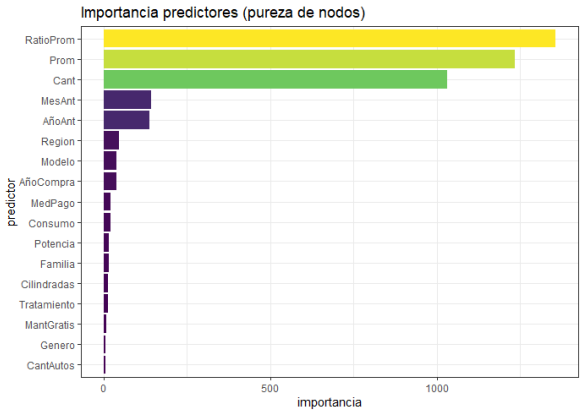
Matriz de confusión

	NO	SI	Error de Clasificación
NO	2467	0	0
SI	270	3026	0.08

El entrenamiento del modelo, con los mejores hiperparámetros encontrados, permite obtener las variables más importantes para el modelo, que son 3: ratio promedio, promedio, cantidad.

Figura 5.39

Importancia de variables predictores



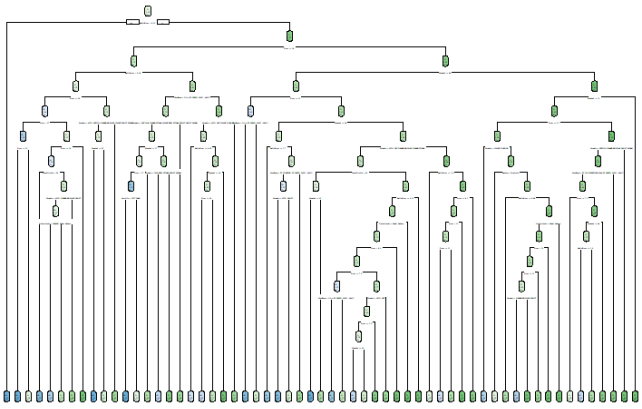
5.2.2. Árbol de Clasificación

Para el modelo de árbol de clasificación se utilizó el algoritmo CART. Asimismo, el modelo se construyó utilizando la librería *Rpart*. Se generaron dos grupos de datos; 60% para el entrenamiento y el 40% de prueba.

Para la fase de entrenamiento, se generó el modelo de árbol de decisión, cabe indicar que para la construcción del modelo empleó las 17 variables identificadas.

Figura 5.40

Árbol de clasificación



Para la fase de prueba, se generó el modelo de árbol de decisión y se obtuvo la siguiente matriz de confusión. Obteniendo como resultado una precisión del modelo del árbol de clasificación 94.22%.

Tabla 5.20

Matriz de confusión - Pruebas (Árbol de clasificación)

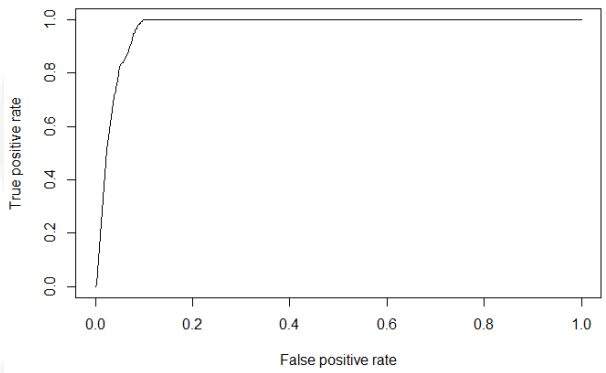
	NO	SI	Error de Clasificación
NO	2505	101	0.03
SI	232	2925	0.07

- **Curva ROC**

Para el modelo de clasificación se obtiene un AUC=0.968, se muestra la gráfica de Curva ROC.

Figura 5.41

Curva ROC - Árbol de Clasificación sin Poda

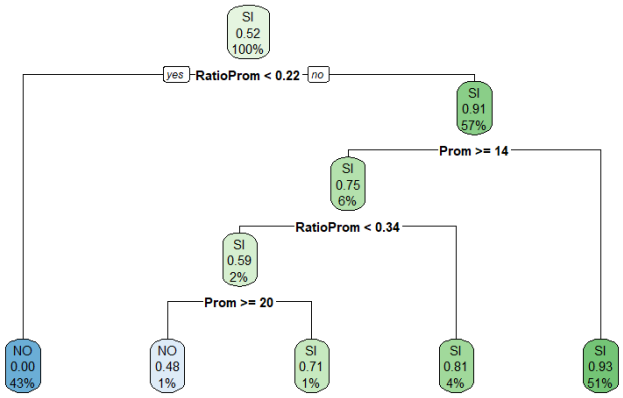


- **Podado de árbol**

El proceso de podado del árbol de clasificación se realizó con la finalidad de evitar el *overfitting*. Las variables que se consideran para la construcción del árbol son dos (promedio, ratio promedio).

Figura 5.42

Árbol de clasificación con podado



Se obtuvo la siguiente matriz de confusión. Teniendo como resultado una precisión del modelo de 95.14%.

Tabla 5.21

Matriz de confusión - Pruebas (Árbol de clasificación con podado)

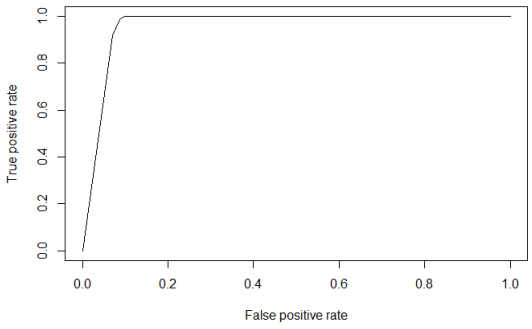
	NO	SI	Error de Clasificación
NO	2491	34	0.01
SI	246	2992	0.07

- **Curva ROC**

En el modelo de clasificación con poda se obtiene un AUC=0.960, se muestra la curva ROC.

Figura 5.43

Curva ROC - Árbol de clasificación con Poda



5.2.1 Resultados

En la tabla observamos las precisiones obtenidas a partir de los modelos construidos en los cuales se aplicaron técnicas de árboles de decisiones. Se puede indicar que el modelo *Random Forest* con optimización de hiperparámetros obtuvo mejor resultado de precisión.

Tabla 5.22

Resultado de la precisión de los modelos desarrollados

	Random Forest	RF con optimización de hiperparámetros	Árbol de clasificación	Árbol de clasificación con poda
Precisión	95.08%	95.31%	94.22%	95.14%

CONCLUSIONES

Se obtuvieron las siguientes conclusiones respecto a los objetivos específicos planteados:

Predicción de Ventas

- Se puede concluir que la metodología de (Lin & Tsai, 2016) obtuvo una menor performance sobre lo obtenido por el método de (Han, 2008). Se observó que en todas las ejecuciones, el método de selección de variables PCA presetanba menor valor de error cuadrático (MSE), lo que indica que los valores generados por la red neuronal se aproximaban mucho más al valor real del nivel de ventas de autos. A pesar de que el análisis de componente principal (PCA) es aplicado usualmente cuando se posee una gran cantidad de variables, para este caso particular el trabajar con las dimensiones más representativas logró buenos resultados teniendo como mejor valor un MSE de 0.3, siendo muy próximo al mejor valor obtenido en la fase de entrenamiento el cual fue 0.22.
- Un modelo de red neuronal necesita una buena cantidad de datos de entrenamiento, con la finalidad de que los pesos sinápticos que se calculan a través de cada iteración puedan ser más acertados y llegar a la convergencia con mejor precisión. A pesar de que solo se trabajó con 179 registros, los resultados de la fase de pruebas no se alejaron mucho luego de la validación con la fase de entrenamiento. Asimismo, podría considerarse algunas otras variables que puedan afectar a la decisión del cliente por comprar un auto, teniendo en cuenta que la técnica de selección de datos PCA se acopla muy bien cuando se tiene un gran cantidad de variables; sin dejar de lado aquellas que en este estudio se validó influyen de manera fehaciente en la determinación del nivel de ventas mensual.
- El afinamiento de parámetros permitió evaluar con mayor cobertura todas las posibles combinaciones existentes para nuestro modelo, obteniendo los valores más óptimos a ser configurados. Se pudo observar que el modelo ya cargado con los hiperparámetros tuvo mejor resultado comparado con el modelo inicial, tanto para el enfoque de ordenamiento por fecha así como para el de forma aleatoria.

- Al realizar la comparación con otro tipo de modelo predictivo, como es el de regresión lineal múltiple, se observa que este obtiene un valor MSE muy similar comparado con el valor más óptimo encontrado a partir de la red neuronal. Sin embargo, de acuerdo a lo validado con la prueba Durbin-Watson, este modelo tendría problemas debido a la autocorrelacción que existe entre las variables consideradas.
- En una primera versión de esta investigación, trabajando solo con un histórico de 3 años (41 registros), se obtenían mejores resultados con la técnica de regresión. Sin embargo, al aumentar la cantidad de datos del *dataset* para la versión final de este estudio, la precisión de este modelo dismuyó considerablemente, mientras que por el contrario el de red neuronal mejoró en comparación con su primera versión. De esto se puede inferir que la relación de las variables utilizadas dentro del *dataset* no presenta un comportamiento lineal. Se demuestra también que las redes neuronales tienen una mejor performance cuando se trabaja con grandes cantidades de datos, ya que eso brinda mayor nivel procesamiento y aprendizaje en la fase de entrenamiento.

Asistencia al Servicio de Mantenimiento

- Se ha realizado en esta investigación un método de predicción de servicios de mantenimiento de taller para la marca de automóviles más representativa en la organización de estudio. Debido a que la organización cuenta con 8 marcas distintas de autos. En el preprocesamiento de datos se identificaron 17 variables predictoras, que se utilizaron en ambos modelos propuestos.
- En la comparativa de metodologías para los árboles de decisión se utilizaron los algoritmos *Random Forest* y árbol de clasificación. Para mejorar los resultados de los modelos, al primer modelo se le realizó un afinamiento de hiperparámetros para obtener una mejor precisión del modelo (95.31%). Por otro lado, en el segundo modelo se realizó una poda del árbol mejorando la precisión (95.14%). Sin embargo, en comparación a los dos modelos finales el que obtuvo mejores resultados fue *Random Forest* con el afinamiento de hiperparámetros. Por ello, se puede concluir que es la mejor técnica por aplicar.

RECOMENDACIONES

Luego de haber realizado las construcciones de los modelos correspondientes a cada objetivo se detallan las siguientes recomendaciones sobre los distintos problemas encontrados.

- Para la aplicación de técnicas predictivas, como las mostradas en este estudio, se sugiere primero realizar un análisis descriptivo para tener una visión concreta de cómo están constituidos los datos que van a ser trabajados. El mayor beneficio de realizar esta validación previa es que permitirá determinar cuál es la técnica predictiva más adecuada de acuerdo con lo obtenido en el análisis previo.
- Apoyarse en estudios realizados sobre problemáticas similares nos resultó muy beneficioso, ya que se pudo tomar como línea base dichas investigaciones y partir de estas elaborar la nuestra. Se sugiere siempre revisar literatura de los artículos citados dentro de aquellos que se consideran para la investigación, ya que puede proporcionar información adicional que dentro del artículo original quizá no fue considerada.
- Uno de los problemas encontrados fue si debiéramos considerar todas las marcas de autos que ofrece servicio de taller, ya que a partir de esta pueden existir muchas variantes en el comportamiento de los clientes. Después, de un análisis descriptivo sobre la data trabajada, concluimos que cada marca de auto tiene variables y comportamientos distintos, por lo cual se eligió la marca de auto con la mayor demanda de clientes, a fin de poder acotar nuestro estudio
- En la parte del desarrollo, el algoritmo creado para el afinamiento de hiperparámetros tomaba un tiempo considerable por ejecución tanto para redes neuronales como los modelos de *Random Forest*, involucrando una importante cantidad de recursos computacionales.
- Finalmente, aplicar una técnica de validación (como en nuestro caso la validación cruzada) resulta muy beneficioso para poder identificar si el modelo logrado que obtenía una alta precisión presentaba *overfitting* o no.

REFERENCIAS

Alex Guazzelli, V. d. (26 de 11 de 2012). *IBM Developer Works.* Obtenido de https://www.ibm.com/developerworks/ssa/industry/library/ba-predictive-analytics1/#ibm-pcon

Amaya, J. A. (2010). TOMA DE DECISIONES GERENCIALES Métodos Cuantitativos. En J. A. Amaya. Colombia: ECOE EDICIONES.

Berlanga Silvente, V., Rubio Hurtado, M. J., & Vilà Baños, R. (2013). Cómo aplicar árboles de decisión en SPSS. Reire, Revista d'Innovació i Recerca en Educació, 65-79.

Continuum_Analytics. (2017). Anaconda. Obtenido de https://www.continuum.io/open-source-core-modern-software

Cristina Puello Correa, L. B. (2017). Correlación del diagnostico clinico, radiográfico e histológico de lesiones apicale dentales. Revista Odontológica Mexicana, 22-29.

Gestión, D. (11 de Enero de 2017). Se venderían 180,000 vehículos nuevos en el presente año, afirma Scotiabank. Diario Gestión.

Guo, Y., Hu, X., Zhou, Y., & Cheng, W. (2019). Research on Recomendation of Insurance Products Based on Random Forest.

IBM. (2017). IBM SPSS Decision Trees 20. Obtenido de IBM SPSS Decision Trees 20

INEI. (2012). INEI, Instituto Nacional de Estadistica e Informática. Obtenido de https://www.inei.gob.pe/estadisticas/indice-tematico/transport-and-communications/

Keras. (2017). Obtenido de https://keras.io/

Kingma, D., & Lei Ba, J. (Julio, 2015). ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION. ICLR 2015.

Lior, R., & Oded, M. (2014). DATA MINING WITH DECISION TREES, Theory and aplications . Boston: P. S. P. Wang (Northeastern Univ., USA).

Minitab. (2017). Minitab Products. Obtenido de http://www.minitab.com/es-mx/products/minitab/look-inside/

Python. (2017). Python - spyder 3.1.4. Obtenido de https://pypi.python.org/pypi/spyder

Simon, H. (2008). Naural Networks and Learning Machines. New Jersey: Pearson.

Tanagra. (2017). Obtenido de https://eric.univ-lyon2.fr/~ricco/tanagra/en/tanagra.html

Tensorflow, G. (2017). Tensorflow Apli Documentation. Obtenido de https://www.tensorflow.org/api_docs/

Tiantian, X., Runchuan, L., Xingjin, Z., Bing, Z., & Zongmin, W. (2019). Research on Heartbeat Classification Algorithm Based on CART Decision Tree. 2019 8th International Symposium on Next Generation Electronics (ISNE). doi:10.1109/ISNE.2019.8896650

Abdollahian, M., & Foroughi, R. (2005). Regression analysis of ozone data. International Conference on Information Technology: Coding and Computing (ITCC'05) - Volume II, 1, 174-178 Vol. 1. https://doi.org/10.1109/ITCC.2005.242

Alhilman, J., Rian, M. M., Marina, W., & Margono, K. (2014). Predicting and clustering customer to improve customer loyalty and company profit. 2014 2nd International Conference on Information and Communication Technology (ICoICT), 331–334. https://doi.org/10.1109/ICoICT.2014.6914087

Amral, N., Ozveren, C. S., & King, D. (2007). Short term load forecasting using Multiple Linear Regression. 2007 42nd International Universities Power Engineering Conference, 1192–1198.

Cruz, M. J. C. S., Montoya, M. H., Rivera, M. S. R., & Zúñiga, C. V. (2017). Plan Estratégico para el Sector de Comercio Automotriz de Vehículos Ligeros del PerúNo Title. Pontificia Universidad Catolica del Perú.

Han, M. (2008). Customer Segmentation Model Based on Retail Consumer Behavior Analysis. 2008 International Symposium on Intelligent Information Technology Application Workshops, 914–917. https://doi.org/10.1109/IITA.Workshops.2008.225

Hashmi, O. Z., & Sheikh, S. (2012). Impact of social attributes on Predictive Analytics in telecommunication industry. 2012 15th International Multitopic Conference (INMIC), 47–52. https://doi.org/10.1109/INMIC.2012.6511470

Hong, S. J., & Weiss, S. M. (2001). Advances in predictive models for data mining. Pattern Recognition Letters, 22(1), 55–61. https://doi.org/http://dx.doi.org/10.1016/S0167-8655(00)00099-4

Jain, A. K., Mao, J., & Mohiuddin, K. M. (1996). Artificial neural networks: a tutorial. Computer, 29(3), 31–44. https://doi.org/10.1109/2.485891

Kaneko Yuta; Yada, K. (2016). [IEEE 2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW) - Barcelona, Spain (2016.12.12-2016.12.15)] 2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW) - A Deep Learning Approach for the Prediction of Retail Sto. https://doi.org/10.1109/ICDMW.2016.0082

Katamaneni, M., Guttikonda, G., & Suneetha, M. (2018). Implementing of Decision Tree Algorithm using R-Studio and Java. 2018 Fourth International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB), 1–5. https://doi.org/10.1109/AEEICB.2018.8481001

Lin, K. Y., & Tsai, J. J. P. (2016). A Deep Learning-Based Customer Forecasting Tool. 2016 IEEE Second International Conference on Multimedia Big Data (BigMM), 198–205. https://doi.org/10.1109/BigMM.2016.85

Liu, T., Zhu, K., & Yu, S. (2009). China Human Capital Prediction Based on the PCA-BP Artificial Neural Networks. 2009 Second International Conference on Intelligent Computation Technology and Automation, 1, 67–70. https://doi.org/10.1109/ICICTA.2009.25

Moreira, M. W. L., Rodrigues, J. J. P. C., Oliveira, A. M. B., Saleem, K., & Neto, A. J. V. (2017). Predicting hypertensive disorders in high-risk pregnancy using the random forest approach. 2017 IEEE International Conference on Communications (ICC), 1–5. https://doi.org/10.1109/ICC.2017.7996964

Qin, Y., & Li, H. (2011). Sales forecast based on BP neural network. 2011 IEEE 3rd International Conference on Communication Software and Networks, 186–189. https://doi.org/10.1109/ICCSN.2011.6014419

Refaeilzadeh, P., Tang, L., & Liu, H. (2009). Cross-Validation. In L. LIU & M. T. ÖZSU (Eds.), Encyclopedia of Database Systems (pp. 532–538). Springer US. https://doi.org/10.1007/978-0-387-39940-9_565

Shankar, K., Zhang, Y., Liu, Y., Wu, L., & Chen, C. (2020). Hyperparameter Tuning Deep Learning for Diabetic Retinopathy Fundus Image Classification. IEEE Access, 8, 118164–118173. https://doi.org/10.1109/ACCESS.2020.3005152

Souza, G. C. (2014). Supply chain analytics. Business Horizons, 57(5), 595–605. https://doi.org/https://doi.org/10.1016/j.bushor.2014.06.004

Wang, T., Ye, X., Wang, L., & Li, H. (2012). Grid Search Optimized SVM Method for Dish-like Underwater Robot Attitude Prediction. 2012 Fifth International Joint Conference on Computational Sciences and Optimization, 839–843. https://doi.org/10.1109/CSO.2012.189

Xie, T., Li, R., Zhang, X., Zhou, B., & Wang, Z. (2019). Research on Heartbeat Classification Algorithm Based on CART Decision Tree. 2019 8th International Symposium on Next Generation Electronics (ISNE), 1–3. https://doi.org/10.1109/ISNE.2019.8896650

Xu, W., Zhang, J., Zhang, Q., & Wei, X. (2017). Risk prediction of type II diabetes based on random forest model. 2017 Third International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB), 382–386. https://doi.org/10.1109/AEEICB.2017.7972337

Yang, X.-C., Wu, J., Zhang, X.-H., & Lu, T.-J. (2008). Using decision tree and association rules to predict cross selling opportunities. 2008 International Conference on Machine Learning and Cybernetics, 3, 1807–1811. https://doi.org/10.1109/ICMLC.2008.4620698

Zorman, M., Masuda, G., Kokol, P., Yamamoto, R., & Stiglic, B. (2002). Mining diabetes database with decision trees and association rules. Proceedings of 15th IEEE Symposium on Computer-Based Medical Systems (CBMS 2002), 134–139. https://doi.org/10.1109/CBMS.2002.1011367

Ozveren, C. S., Sapeluk, A. T., & Birch, A. (2014). An investigation into using neuro-evolution of Augmenting Topologies (NEAT) for short term load forecasting (STFL). 2014 49th International Universities Power Engineering Conference (UPEC), 1–5. https://doi.org/10.1109/UPEC.2014.6934819

ANEXOS

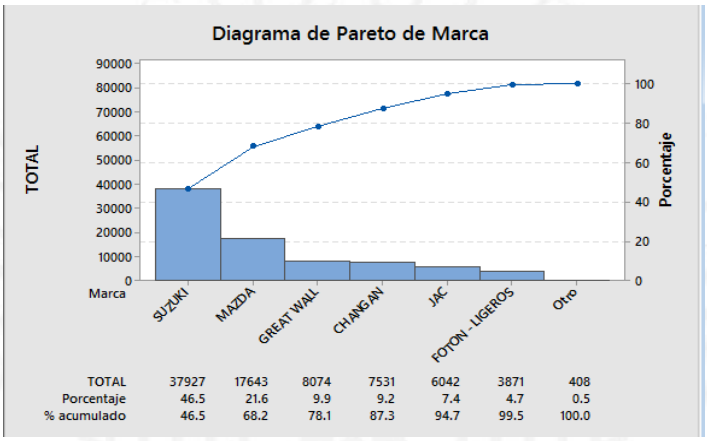
Análisis Descriptivos

En nuestra investigación realizamos un análisis descriptivo para poder realizar el estudio. Con el objetivo de enfocarnos en la marca más representativa en la organización.

El siguiente gráfico se muestra la cantidad de unidades vendidas por marca. Se observa que las marcas Suzuki, Mazda y Great Wall son aquellas que suman el 80%, por ello son las que generan mayor ingreso a la organización.

ANEXO 1:

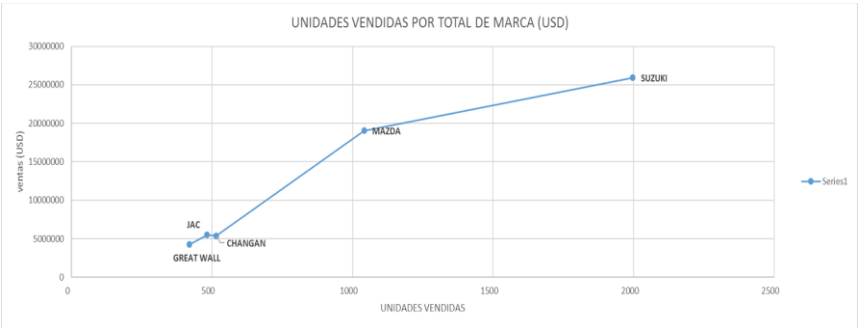
Diagrama de Pareto - Marcas



El siguiente gráfico se muestra las unidades vendidas por marca comparado con los ingresos generados por sus ventas (USD). Se puede concluir que la marca Suzuki es aquella que genera mayor ingreso en la organización.

ANEXO 2:

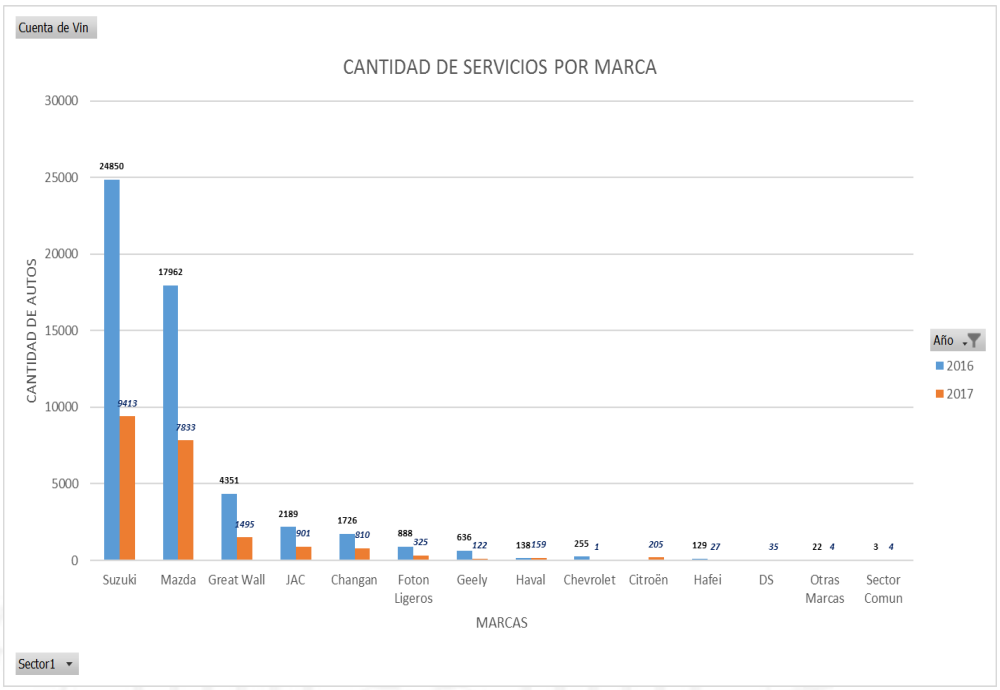
Unidades Vendidas por marca VS Ingresos (USD)



Para el servicio de post venta en la organización, el siguiente gráfico se observa que la marca que tiene mayor cantidad de atenciones es la marca Suzuki.

ANEXO 3:

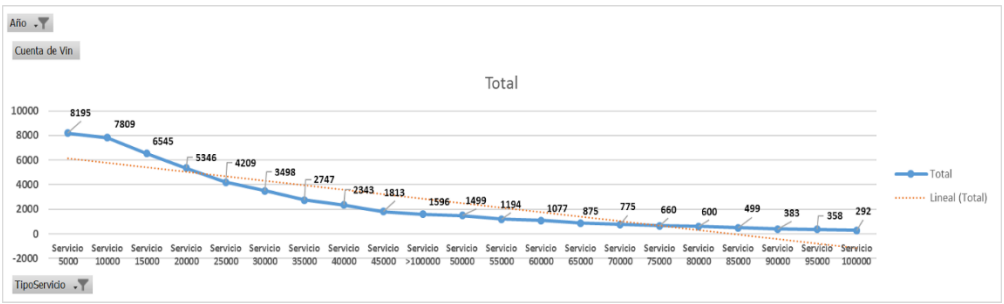
Servicio de mantenimiento por Marca



El siguiente gráfico se visualiza las cantidades de servicios realizados en la organización para los distintos tipos de servicios que empieza con el 5km hasta 100km. Al notar la gráfica se observa que hay un descenso cada vez que al auto le toca realizar un mantenimiento.

ANEXO 4:

Nro de Servicios de mantenimiento por tipo de KM



El siguiente gráfico es un historial de las cantidades de servicios de postventa realizado. Se puede apreciar que este año 2017 se encuentra por debajo a años anteriores.

ANEXO 5:

Histórico de servicios de mantenimientos