DIVISIÓN DE POSTGRADO

Título del trabajo:

Aplicación de técnicas de Inteligencia de Negocios para la predicción del churn de clientes en una empresa de telecomunicaciones

Línea de Investigación:

Tecnologías de la Información y la Comunicación

Modalidad de titulación:

Proyecto de desarrollo

Programa de Maestría

Maestría en Sistemas de Información mención en Inteligencia de Negocios

Título a obtener:

Magister en Sistemas de Información mención en Inteligencia de Negocios

Autor:

Ing. Jimmy Fernando Larrosa Zambrano

Tutor:

Ing. Fabricio Echeverría, MSC

Samborondón – Ecuador

2019

AGRADECIMIENTO

Agradezco a Dios, a mis padres, a mi esposa y a mi hija por este logro alcanzado en esta tapa de mi vida profesiona, los cuales han sido un pilar funadamental en mi vida.

Ing. Jimmy Larrosa Z.

DEDICATORIA

Dedico este Trabajo de Titulación principalmente a mis padres por ser ejemplo e inculcarme los valores y enseñanaza de que con perseverancia y dedicación se puede conseguir lo que uno se propone; a mi esposa, a mi hija por darme siempre todo su cariño, ayuda, respeto y confianza.

Ing. Jimmy Larrosa Z.

Universidad Ecotec

TRABAJO DE TITULACIÓN

CERTIFICACION DE REVISION FINAL

QUE EL PRESENTE PROYECTO DE DESARROLLO TITULADO:

APLICACIÓN DE TÉCNICAS DE INTELIGENCIA DE NEGOCIOS PARA LA PREDICCIÓN DEL CHURN DE CLIENTES EN UNA EMPRESA DE TELECOMUNICACIONES,

FUE REVISADO, SIENDO SU CONTENIDO ORIGINAL EN SU TOTALIDAD, ASÍ COMO EL CUMPLIMIENTO DE LOS REQUERIMIENTOS QUE SE DICTAN EN LA GUÍA DE ELABORACIÓN DEL TRABAJO DE TITULACIÓN, POR LO QUE SE AUTORIZA A: JIMMY FERNANDO LARROSA ZAMBRANO 0911965598, **QUE PROCEDA A SU PRESENTACION.**

Samborondón, 24 de Julio de 2019



TUTOR: Pedro Fabricio Echeverría Briones

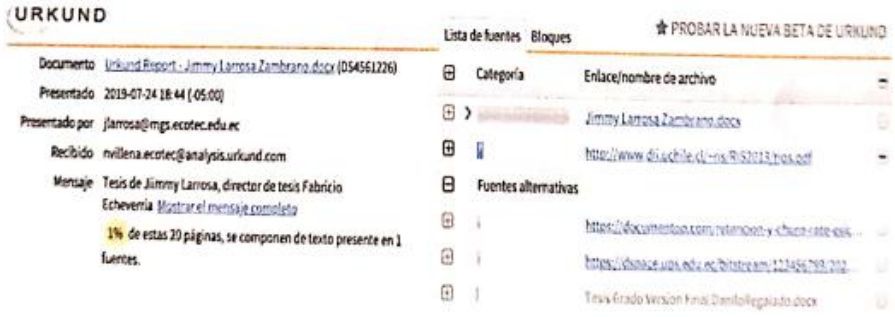
Universidad Ecotec

DIVISIÓN DE POSGRADO
PROCESO DE TITULACIÓN
CERTIFICADO DEL PORCENTAJE DE COINCIDENCIAS

Habiendo sido nombrado PEDRO FABRICIO ECHEVERRIA BRIONES tutor del trabajo de titulación "APLICACIÓN DE TÉCNICAS DE INTELIGENCIA DE NEGOCIOS PARA LA PREDICCIÓN DEL CHURN DE CLIENTES EN UNA EMPRESA DE TELECOMUNICACIONES" elaborado por JIMMY FERNANDO LARROSA ZAMBRANO, 0911965598, con mi respectiva supervisión como requerimiento parcial para la obtención del título de **APLICACIÓN DE TÉCNICAS DE INTELIGENCIA DE NEGOCIOS PARA LA PREDICCIÓN DEL CHURN DE CLIENTES EN UNA EMPRESA DE TELECOMUNICACIONES.**

Se informa que el mismo ha resultado tener un porcentaje de coincidencias (1 %) mismo que se puede verificar en el siguiente link:
https://secure.urkund.com/view/53127271-751880-133087#q1bKLVayijY3iNVRKs5Mz8tMy0xOzEtOVbly0DMwMDK1MLcwNrUAMgyMLE1MagE=

Adicional se adjunta print de pantalla de dicho resultado.


URKUND

Documento Urkund Report - Jimmy Larrosa Zambrano.docx (D54561226)
Presentado 2019-07-24 18:44 (-05:00)
Presentado por jlarrosa@mgs.ecotec.edu.ec
Recibido nvillena.ecotec@analysis.urkund.com
Mensaje Tesis de Jimmy Larrosa, director de tesis Fabricio Echeverría Mostrar el mensaje completo
1% de estas 20 páginas, se componen de texto presente en 1 fuentes.

Lista de fuentes Bloques ★ PROBAR LA NUEVA BETA DE URKUND

Categoría Enlace/nombre de archivo
Jimmy Larrosa Zambrano.docx
http://www.dii.uchile.cl/~rios/RIS2013.tros.pdf
Fuentes alternativas
https://documentop.com/retencion-y-churn-este-ejem...
https://dspace.ups.edu.ec/bitstream/123456789/703...
Tesis Grado Version Final DanielAlegado.docx

Pedro Fabricio Echeverría Briones
PROFESOR TUTOR DE LA CIENCIA

Resumen

El objetivo principal de este proyecto de investigación fue realizar la predicción del churn de clientes en una empresa de telecomunicaciones, analizando y utilizando los diferentes métodos de la minería de datos, la cual, es parte de la Inteligencia de Negocios. Los métodos escogidos para la predicción del churn de clientes son algoritmos de machine learning o aprendizaje automático tales como red neuronal, SVM (Support Vector Machines) o máquinas de vectores de soporte y KNN (K-nearest neighbors) o K vecinos más cercanos. Los algoritmos reciben como entrada los datos de los clientes que representan las variables independientes incluida la variable dependiente llamada churn. El método científico utilizado en este trabajo fue el método cuantitativo, que durante el desarrollo fue mostrando mejores resultados que los esperados debido a que cuando se introduce una variable independiente en una muestra poblacional, la variable dependiente se ve afectada incidiendo en los resultados finales.

Esta solución selecciona el mejor algoritmo de predicción en base a la precisión de las matrices de confusión y obtiene un listado de los posibles clientes que abandonarán la empresa. Así mismo se demuestra que el mejor algoritmo de predicción de los tres escogidos es el de red neuronal debido a que ofrece mejores resultados que incluyen una mejor precisión en su matriz de confusión. Finalmente mediante una página web se realiza el llamado al programa en lenguaje R y se muestran los resultados.

PALABRAS *CLAVES:* Inteligencia de Negocios, machine learning, minería de datos, matriz de confusión, predicción del churn.

Abstract

The main objective of this research project was to predict the churn of clients in a telecommunications company, analyzing and using the different methods of data mining, which is part of Business Intelligence. The methods chosen for the prediction of customer churn are machine learning or machine learning algorithms such as neural network, SVM (Support Vector Machines) or support vector machines and KNN (K-nearest neighbors) or K nearest neighbors. The algorithms receive as input the customers' data that represent the independent variables including the dependent variable called churn. The scientific method used in this work was the quantitative method, which during development was showing better results than expected because when an independent variable is introduced into a population sample, the dependent variable is affected, affecting the final results.

This solution selects the best prediction algorithm based on the accuracy of the confusion matrices and obtains a list of potential customers who will leave the company. It is also shown that the best prediction algorithm of the three chosen is the neural network because it offers better results that include better accuracy in its confusion matrix. Finally, through a web page the call to the program is made in R language and the results are shown.

KEYWORDS: Business Inteligence, machine learning, data minig, confusión matrix, churn prediction.

Tabla de Contenidos

INTRODUCCIÓN 1

Pregunta científica 2

Idea a defender 2

Objetivo general 2

Objetivos específicos 2

Justificación 2

CAPÍTULO 1 4

MARCO TEÓRICO 4

1.1 Modelo de predicción 6

1.1.1 Análisis predictivo 6

1.1.2 Modelo predictivo 6

1.2 Machine prediction 7

1.2.1 Redes neuronales 10

1.2.2 SVM 12

1.2.3 KNN 14

1.3 Variables del negocio 14

1.3.1 Modelo del Churn 15

1.3.2 El negocio de las telecomunicaciones 16

1.4 Modelo del Churn (Teórico) 17

CAPÍTULO 2 19

METODOLOGÍA 19

2.1. Comprensión del negocio 21

2.1.1 Determinación de los objetivos del negocio 21

2.1.1.2 Objetivos del negocio 21

2.1.2 Evaluación de la situación 22

2.1.3 Determinación de los objetivos de minería de datos 25

2.1.4 Producción de un plan del proyecto 26

2.2.1 Recopilación de datos iniciales 27

2.2.2 Descripción de los datos 28

2.2.3 Verificación de calidad de los datos 32

2.3 Preparación de los datos 32

2.3.1 Selección de los datos 32

2.3.2 Limpieza de los datos 33

2.3.3 Construcción de los datos 33

2.3.4 Integración de los datos 34

2.3.5 Formateo de los datos 34

2.4 Modelado 35

2.4.1 Elección de la técnica de modelado 35

2.4.2 Generación del plan de prueba 36
2.4.3 Construcción del modelo 37
2.4.4 Evaluación del modelo 43
2.5 Evaluación 44
2.5.1 Evaluación de los resultados 44
2.5.2 Revisión del proceso 45
2.5.3 Determinación de los próximos pasos 48
2.6 Distribución 48
2.6.1 Planificación de la Implantación 48
CAPÍTULO 3 50
VALIDACIÓN DE LOS RESULTADOS 50
3.1 Esquema de validación 51
3.1.1 Pre-experimentos 51
3.1.2 Cuasi-experimentos 52
3.1.3 Experimentos puros 52
3.2 Aplicación y descripción de los resultados 53
3.2.1 Pre-experimentos 53
3.2.2 Cuasi-experimentos 54
3.2.3 Experimentos puros 55
3.3 Impacto 58

3.3.1 Perspectivas económicas 58

3.3.2 Perspectivas sociales..... 59

Lista de referencias 63

Lista de tablas

Tabla 1 Matriz de riesgos del proyecto *23*

Tabla 2 Matriz de contingecias *25*

Tabla 3 Matriz de confusión de la red neuronal *39*

Tabla 4 Matriz de confusion de SVM *41*

Tabla 5 Matriz de confusión de KNN *43*

Tabla 6 Precisión de los modelos *44*

Tabla 7 Matriz de confusión obtenida con h2o.deeplearning con todos los clientes *53*

Tabla 8 Matriz de confusión obtenida con SVM con todos los clientes *53*

Tabla 9 Matriz de confusión obtenida con KNN con todos los clientes *54*

Tabla 10 Matriz de confusión de red neuronal con todos los clientes con caret *54*

Tabla 11 Matriz de confusión de SVM con todos los clientes con caret *55*

Tabla 12 Matriz de confusión de KNN con todos los clientes con caret *55*

Tabla 13 Matriz de confusión de red neuronal con datos balanceados *56*

Tabla 14 Matriz de confusión de red SVM con datos balanceados *56*

Tabla 15 Matriz de confusión de red KNN con datos balanceados *57*

Lista de figuras

Figura 1 Modelo predictive. 7

Figura 2 Red neuronal. Fuente: Computer Age Statistical Inference 11

Figura 3 Modelo structural de SVM linealmente separable 13

Figura 4 Modelo structural del KNN 14

Figura 5 Ciclo del Modelo CRISP-DM. Fuente: IBM Corporation 20

Figura 6 Plan del proyecto. Nota: Elaborado por el Autor 26

Figura 7 Modelo Entidad Relación. Nota: Elaborado por el Autor 29

Figura 8 Filtrado de registros. Nota: Elaborado por el Autor 33

Figura 9 Precisión de una matriz de confusión. 36

Figura 10 Algoritmo de red neuronal – Código en R. Nota: Elaborado por el Autor 38

Figura 11 Precisión de la matriz de confusión de la red neuronal. Nota: Elaborado por el Autor 39

Figura 12 Algoritmo SVM en R. Nota: Elaborado por el autor 40

Figura 13 Precisión de la matriz de confusión de SVM. Nota: Elaborado por el Autor 41

Figura 14 Algoritmo KNN en R. Nota: Elaborado por el Autor 42

Figura 15 Precisión de la matriz de confusión de KNN. Nota: Elaborado por el Autor 43

Figura 16 Algoritmo de red neuronal con paquete neuralnet en R. Nota: Elaborado por el Autor 46

Figura 17 Algoritmo de red neuronal con paquete h2o en R. Nota: Elaborado por el Autor 47

Figura 18 Página web prediccion_churn.php 57

Figura 19 Programa prediccion.php 58

INTRODUCCIÓN

Las empresas de telecomunicaciones toman en cuenta el abandono de clientes o el churn dado que es menos costoso retener a un cliente que adquirir uno nuevo. (Domínguez Doncel & Muñoz Vera, 2010, pág. 81). Por esta razón es muy importante conocer de antemano los posibles clientes que puedan abandonar la empresa para poder desarrollar estrategias de Marketing y poder retenerlos. Una forma de lograr la retención del cliente es poder predecir el churn del mismo en base a su comportamiento en la empresa.

El método científico utilizado fue el método cuantitativo debido a que se parte de muestras de datos de clientes, se los procesa, se analizan los resultados obtenidos y se introduce una variable independiente incidiendo en la variable dependiente; asimismo mediante el muestreo de datos se optimizan los resultados finales.

El presente trabajo se centra en la predicción del churn de clientes en base al comportamiento que han tenido a lo largo del ciclo de vida en la empresa. Se parte de una base de clientes adquiridos mediante scripts de consultas de bases de datos relacionales y mediante la metodología de CRISP-DM se va afinando el conjunto de datos hasta convertirlos en aptos para el procesamiento de los algoritmos de minería de datos. Los algoritmos utilizados en este trabajo son:

- Red neuronal
- SVM
- KNN

Estos algoritmos usan matrices de confusion, que ofrecen una determinada precisión que permiten escoger el mejor modelo tal y como se explica en la sección 2.5.1.

Pregunta científica

¿Cómo predecir la deserción del churn de cientes a partir de su comportamiento en una empresa de telecomunicaciones?

Idea a defender

Si se implementan las técnicas de Machine Learning en una empresa de telecomunicaciones a partir del comportamiento de sus clientes se podrá predecir su deserción.

Objetivo general

Desarrollar una solución de Inteligencia de Negocios en una empresa de telecomunicaciones para predecir la deserción de los clientes a partir de su comportamiento.

Objetivos específicos

- Contextualizar el marco teórico relacionado con el comportamiento de los clientes para la predicción del churn
- Diagnosticar el estado actual de las tecnologías, modelos y técnicas de la identificación del churn de clientes para identificar debilidades y elementos reutilizables
- Implementar una solución de Inteligencia de Negocios para mostrar la predicción del churn

Justificación

Los clientes son uno de los activos más críticos con los que cuenta una empresa porque pueden mantenerse en el mercado, aumentar el reconocimiento de la marca y a su vez crecer de forma sustentable. Es por ello que las empresas implementan diferentes técnicas para atraer

clientes y fidelizar los ya existentes. El churn de clientes es uno de los factores que más impactan de forma negativa sobre todo a las empresas que brindan servicios de telecomunicaciones.

Entre las técnicas más utilizadas para atraer, fidelizar y evitar el churn de clientes se pueden mencionar las promociones y beneficios temporales. Estas iniciativas se basan fundamentalmente en análisis superficiales que no responden en muchos de los casos a las preferencias usadas por los clientes. No se toman en cuenta atributos críticos como:

- El sector donde viven
- La situación socio económica
- Las preferencias familiares en función de las características del núcleo familiar
- El historial del cumplimiento de los pagos
- Nivel de aceptación de los beneficios ofrecidos

El limitado análisis de los atributos descritos anteriormente traen consigo que las estrategias de marketing, publicidad, ventas y fidelización de los clientes no tengan la efectividad deseada; y generan gastos de recursos materiales, financieros y desgastes en el personal sin percibir el crecimiento esperado.

Es por ello que es fundamental que las empresas de servicios como las telecomunicaciones implementen herramientas innovadoras que permitan atraer más clientes, fidelizar los ya existentes y disminuir el churn. Con ello implementar estrategias y campañas donde se diseñen productos y servicios afines a las necesidades que persiguen los clientes tomando en cuenta su perfil y el historial del ciclo de vida con la empresa.

CAPÍTULO 1

MARCO TEÓRICO

La necesidad de la humanidad por la digitalización de los procesos en las empresas y el consecuente aumento exponencial de los datos que esto conlleva, ha desembocado en el surgimiento inevitable y necesario de un término que está muy de moda en la actualidad denominado Big Data. Este término se deriva de otro muy familiar llamado Inteligencia de Negocios o BI (Business Intelligence). Es en realidad el término BI el que inició esta moda, la cual ha ido evolucionando en el tiempo hasta consolidarse como una tecnología necesaria para que las empresas se vuelvan competitivas.

Desde la década de los setenta con el uso de los computadores se usaron los llamados Sistemas de Procesamiento de Transacciones (Transaction Processing Systems, TPS) evolucionando hasta los Sistemas de Planificación de Recursos Empresariales (Enterprise Resource Planning, ERP), los Sistemas de Gestión de la Cadena de Suministros, SCM) y los sistemas de Gestión de las Relaciones con los Clientes (Customer Relationship Management, CRM). Al mismo tiempo que aparecieron los TPS's las empresas se vieron en la necesidad de procesar la información para el control de su gestión y para la toma de decisiones por lo que parecen los Sistemas de Información de Gestión (Management Information Systems, MIS), luego aparecen los Sistemas de Ayuda para la Decisión (Decision Support Systems, DSS); por la década de los ochenta llegan los Sistemas Expertos (Exert Systems, ES) los cuales son parte de la Inteligencia Artificial, los Sistemas de Información Ejecutivos (Executive Information Systems, EIS). Estos sistemas (MIS, EIS y DSS) fueron integrados en un solo término que se mantiene hasta hoy en día llamado Inteligencia de Negocios. Se define entonces la Inteligencia de Negocios como

el conjunto de aplicaciones diseñadas para apoyar la toma de decisiones (Roldán Salgueiro, Cepeda Carrión, & Galán González, 2012).

1.1 Modelo de predicción

1.1.1 Análisis predictivo

El análisis predictivo es la rama de la minería de datos que se dedica a la extracción de la información que existe en los datos para descubrir tendencias y patrones ocultos de comportamiento sobre cualquier fenómeno desconocido. Este análisis se basa en la identificación de relaciones entre variables en eventos pasados para explotarlas y predecir posibles resultados en futuras situaciones con el fin de optimizar la toma de decisiones. Para que el análisis predictivo sea posible es necesario disponer de una cantidad considerable de datos actuales y pasados, para poder determinar patrones de comportamiento y así poder obtener conocimiento (Espino Timón & Martínez Fontes, 2017, pág. 6).

1.1.2 Modelo predictivo

Un modelo predictivo es un instrumento que predice el comportamiento de un individuo y utiliza sus características como entrada para proporcionar una calificación predictiva como salida (Espino Timón & Martínez Fontes, 2017, pág. 12).

El análisis predictivo consiste en construir un modelo estadístico y usar los datos existentes para sugerir datos no disponibles. Para la creación de un modelo predictivo se utiliza una muestra disponible con atributos conocidos y un comportamiento conocido denominado conjunto de

entrenamiento, también se utiliza una muestra con atributos similares pero de comportamiento desconocido denominado conjunto de prueba (Espino Timón & Martínez Fontes, 2017, pág. 14).

1.2 Machine prediction

El aprendizaje de máquina es necesario y fundamental en el proceso de análisis predictivo ya que proporciona las técnicas de análisis de datos con los cuales se pueden descubrir relaciones entre las variables, que en un principio parecieran insignificantes pero con la aplicación de dichas técnicas se puede descubrir su importancia. Luego de haber establecido correlaciones entre las variables queda a criterio del ser humano interpretar las mismas de un modo correcto para no llegar a conclusiones equivocadas y se indentifican patrones de comportamiento que permiten crear un modelo predictivo (Espino Timón & Martínez Fontes, 2017, págs. 8, 9), tal como se muestra en la figura 1.



Figura 1 Modelo predictive.

Fuente: Predictive Analytics – The Power to predict who will click, buy lie or die

Hay que tener en cuenta que el modelo predictivo nunca proporcionará el 100% de acierto porque que a pesar de haberse repetido un patrón de comportamiento en el pasado, este no tiene por qué repetirse. A pesar de esto, siempre será mejor predecir con la ayuda de un modelo que

simplemente adivinar aunque dicho modelo proporciones porcentaje de aciertos bajos (Espino Timón & Martínez Fontes, 2017, pág. 10).

Hay dos tipos de aprendizaje automático, el aprendizaje supervisado y el no supervisado. El aprendizaje supervisado es aquel aprendizaje cuyo conjunto de datos de entrada contiene la salida de interés en donde se quiere que el método "aprenda" para predecir dicha salida a partir de nuevos datos. El aprendizaje automático que se usará en este trabajo será el aprendizaje supervisado. El objetivo del aprendizaje supervisado es aprender un modelo predictivo que asocie las características de los datos de entrada (como por ejemplo el tamaño de la casa, localización, tipo de piso, etc.) a una salida (como por ejemplo el precio) (Molnar, 2019, pág. 10).

Los pasos que siguen los algoritmos de aprendizaje automático supervisados son los siguientes:

1. Recolectar la mayor cantidad de datos posibles. Como se indicó anteriormente los datos de entrada deben contener la salida que se desea predecir. Para un predictor de crédito se necesitan datos pasados de los préstamos actuales, información como si el cliente estuvo en mora con sus préstamos y datos que ayudarán para la predicción como sus ingresos, historial de créditos pasados, etc. Para un estimador del valor de una casa se necesitarían los datos de ventas pasadas de casas e información actual de dicha casa como el tamaño, localización, etc.

2. Ingresar esta información en un algoritmo de aprendizaje automático que genere un modelo de detector de signos, un modelo de calificación crediticia o un estimador del valor de la vivienda.

3. Utilizar el modelo con nuevos datos. Integrar el modelo en un producto o proceso, como el auto autónomo, un proceso de solicitud de crédito o un sitio web del mercado de bienes raíces.

Las máquinas de aprendizaje superan al ser humano en muchas tareas como por ejemplo jugar ajedrez o en la predicción del clima. Incluso si la máquina de aprendizaje es tan buena como un ser humano o un poco peor en una tarea, existen grandes ventajas en términos de velocidad, reproducibilidad y escalado. Una vez que se implementa el modelo de aprendizaje automático la máquina puede completar una tarea mucho más rápido que los humanos, entrega de manera confiable resultados consistentes y puede ser copiado infinita cantidad de veces. La replicación de un modelo de aprendizaje automático a otra máquina es más rápida y de bajo costo mientras que el entrenamiento de un humano para una determinada tarea puede tomar décadas y es muy costoso. Una de las principales desventajas de utilizar el aprendizaje automático es que la información sobre los datos y la tarea que la máquina resuelve están ocultas en modelos más complejos. Se necesitan de millones de números para describir una red neuronal profunda y no hay manera de entender el modelo en su totalidad. Otros modelos como el de ramdon forest consisten de cientos de árboles de decisión que “votan” por las predicciones. Para comprender cómo se tomó una decisión, se tendrían que analizar los votos y las estructuras de cada uno de los cientos de árboles. Eso

simplemente no funciona, no importa lo inteligente que sea un humano o lo buena que sea su memoria. Los modelos con mejor rendimiento a menudo son combinaciones de varios modelos (también llamados conjuntos) que no pueden interpretarse, incluso si cada modelo individual pudiera interpretarse. Si se enfoca solo en el rendimiento, automáticamente se obtendrá más y más modelos difusos (Molnar, 2019, págs. 10, 11).

1.2.1 Redes neuronales

A mediados de los 80's hubo algo que sacudió a la comunidad de las Estadísticas Aplicadas debido a que con la introducción de las redes neuronales se marcó un cambio del modelado predictivo a la Informática y al aprendizaje automático. La red neuronal es un modelo altamente parametrizado, inspirado en la arquitectura del cerebro humano, el cual fue altamente promovido como un aproximador universal; es decir, una máquina que con suficientes datos puede aprender relación predictiva.

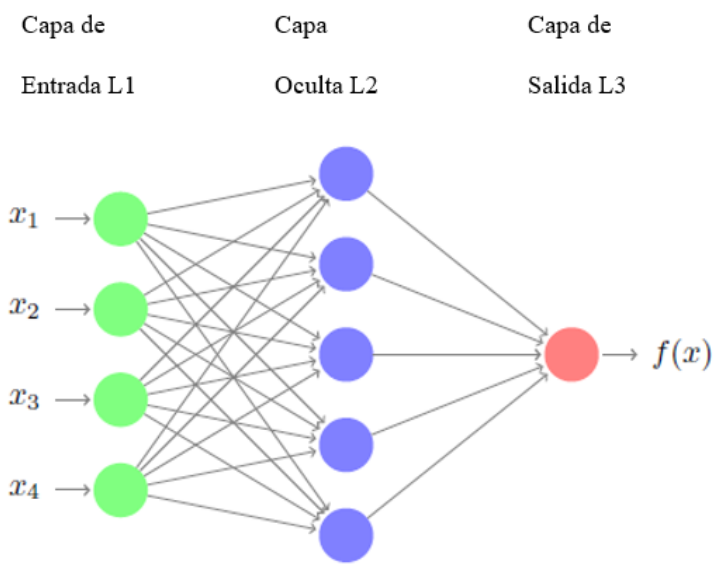


Figura 2 Red neuronal. Fuente: Computer Age Statistical Inference

La red neuronal de la figura 2 tiene cuatro predictores o entradas L1:

x_j,

cinco unidades o capas ocultas L2:

$$a_l = g(w_{l0}^{(1)} + \sum_{j=1}^{4} w_{lj}^{(1)} x_j),$$

y una única unidad de salida L3:

$$o = h(w_0^{(2)} + \sum_{l=1}^{5} w_l^{(2)} a_l)$$

Las unidades de memoria o neuronas aprenden automáticamente nuevos aspectos de los datos a través de un proceso llamado aprendizaje supervisado. Cada neurona a_l está conectada a la capa de entrada mediante un vector de parámetros o pesos $\{w_{lj}^{(1)}\}_1^p$ en donde (1) se refiere a la primera capa, *lj* se refiere a la j ésima variable y *l* ésima unidad. Los términos interceptos $w_{l0}^{(1)}$ son

llamados un *bias* mientras que la función *g* es una no linealidad tal como la función sigmoide $g(t) = 1 / (1 + e^{-t})$. La idea era que cada neurona aprendiera una simple función binaria de encendido/apagado y la función sigmoide es un compromiso suave y diferenciable. La capa final o de salida tiene también pesos y una función de salida *h*. Por regresión cuantitativa *h* es típicamente la función identidad y para una respuesta binaria es también sigmoide. Las redes neuronales típicamente se ajustan a la máxima probabilidad, generalmente con una variedad de formas de regularización. Los estadísticos consideran que una red neuronal es un modelo no lineal, no muy diferente de muchas otras generalizaciones de modelos lineales. Si bien esto puede ser cierto, las redes neuronales han sido de mucha ayuda porque se podrían ampliar y generalizar de varias maneras: muchas unidades ocultas en una capa, múltiples capas ocultas, compartir el peso, una amplia variedad de formas coloridas de regularización y algoritmos de aprendizaje innovadores para los conjuntos de datos masivos; y lo más importante, fueron capaces de resolver problemas en una escala muy superior a la que estaba acostumbrada la comunidad de estadísticas (Bradley & Trevor, 2017).

1.2.2 SVM

Las máquinas de soporte de vectores o en inglés (Support Vector Machines) utilizan planos complejos para poder encontrar la mejor división de las instancias que permita clasificarlas de una forma óptima. La formulación es un problema de minimización cuadrática con un número de variables igual al número de casos de entrenamiento. Las SVM usan la optimización de las matemáticas debido a que el problema que abordan, incluye la optimización de una función

convexa, esto significa que no tiene mínimos locales. Otra característica de este algoritmo es que no requiere información de la distribución del conjunto de datos, esto quiere decir que se busca el balance entre la certeza y la cantidad de datos a aceptar; esta combinación induce el origen del problema de minimización del riesgo estructural, cuya solución se deriva en un problema de optimización sobre encontrar el hiperplano separador entre las instancias, como se muestra en la figura 3 (Barrientos & Ríos, 2013, págs. 84-86) .

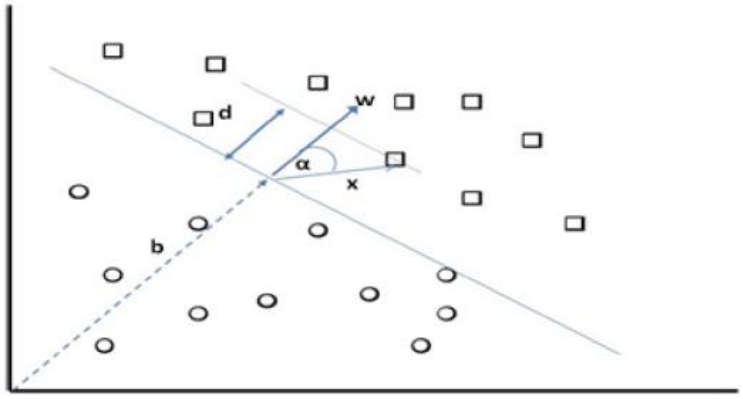


Figura 3 Modelo structural de SVM linealmente separable
Fuente: Aplicación de Minería de Datos para Predecir Fuga de Clientes en la Industria de las Telecomunicaciones

Donde la formulación para un caso linealmente separable es:

$$\text{Min}_{w} \tfrac{1}{2} w^{t} * w$$

$$\text{Sujeto a } y_i(x * w + b) - 1 \geq 0$$

$$\forall\, i = 1, \ldots l$$

1.2.3 KNN

El algoritmo del K vecino más cercano o KNN es considerado uno de los más simples porque no requiere de ningún parámetro fuera del número de vecinos a considerar. Este algoritmo reúne los K vecinos más cercanos y los hace votar con la particularidad de que la clase con más vecinos es la ganadora porque mientras más vecinos se consideran, menor es la tasa de error. Esa cercanía se mide con relación a un tipo de distancia por lo que se pueden obtener resultados diferentes dependiendo de la distancia escogida, dado que diferentes métricas definirán regiones diferentes. El esquema general se muestra en la figura 4 (Barrientos & Ríos, 2013, págs. 83, 84).

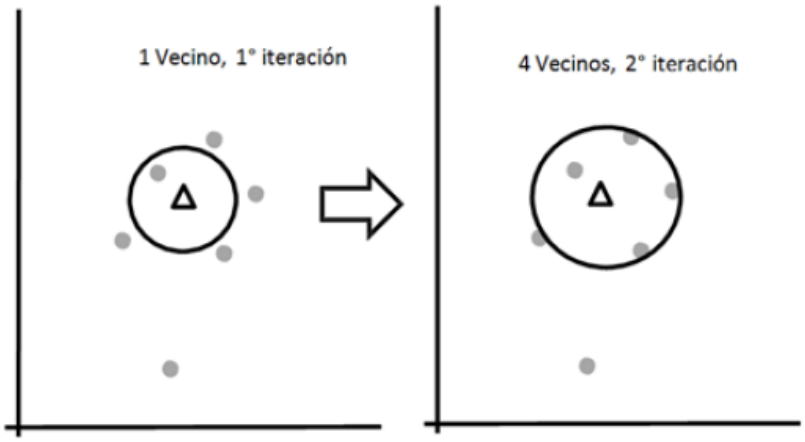


Figura 4 Modelo structural del KNN
Fuente: Aplicación de Minería de Datos para Predecir Fuga de Clientes en la Industria de las Telecomunicaciones

1.3 Variables del negocio

Las líneas de negocio presentes que se encuentran comúnmente en empresas de telecomunicaciones son: tráficos de larga distancia, telefonía local, internet, cargos de accesos,

servicios privados, facturas y cuentas corrientes de atención de clientes (referente a las solicitudes de atención y reclamos), clientes y contratos, modelos de operaciones, participación de mercado y suscriptores (Barrientos & Ríos, 2013, pág. 74).

Además de estos servicios, las empresas de telecomunicaciones han expandido su portafolio incluyendo los denominados servicios fijos tales como internet, televisión y con la aparición de la transformación digital se suman la analítica de datos, la nube y el internet de las cosas o IOT (Internet Of Things) convirtiéndolo en un sector muy atractivo por la gran cantidad de información que generan y almacenan sus empresas.

Una de las principales razones para que un cliente deje de comprar productos de una compañía es la disconformidad y la falta de políticas de retención efectivas expresadas en un mejor trato hacia el cliente (Barrientos & Ríos, 2013, pág. 74). Entre las disconformidades de los clientes traducidas a su vez en reclamos al ente regulador se destacan: por la suscripción del servicio, continuidad y calidad del servicio o bien por cobros irregulares.

1.3.1 Modelo del Churn

La minería de datos se define como una de las etapas de un proceso más genérico llamado Knowledge Discovery in Databases (KDD), el cual, es el proceso de análisis de bases de datos que busca encontrar relaciones inesperadas. En palabras simples significa encontrar relaciones no triviales en los datos. Este proceso iterativo consiste en cinco etapas, en donde la minería de datos es definida como una fase más de este procedimiento. Al tratarse de un proceso iterativo es posible

volver a una etapa previa en caso de que no se tengan resultados satisfactorios al final de una etapa (Barrientos & Ríos, 2013, pág. 76).

El modelo de minería de datos sigue las siguientes fases:

- Integración o Selección: En esta fase se seleccionan las variables y las fuentes a considerarse en el proceso completo dando como resultado la creación de un conjunto de datos como la base de estudio del proceso.
- Preprocesamiento: En esta fase se realiza el análisis y limpieza de los datos, además se produce el tratamiento de valores ausentes y los valores fuera de rango (outliers) mediante técnicas de imputación de datos.
- Transformación: En esta etapa se generan nuevas variables a partir de la naturaleza de las variables originales.
- Minería de datos: En esta etapa se aplica el análisis de datos para descubrir un algoritmo adecuado que encuentre patrones a partir de los mismos.
- Interpretación y evaluación: En esta última etapa intervienen las medidas de evaluación y trasposición de resultados técnicos a niveles comerciales, de tal forma que la aplicación del procedimiento converja a acciones correctivas en el negocio que den solución al fenómeno que se estudia (Barrientos & Ríos, 2013, págs. 76, 77).

1.3.2 El negocio de las telecomunicaciones

La telefonía celular llegó al Ecuador en el año 1993 con dos empresas llamadas PORTA CELULAR perteneciente a CONECEL y CELULAR POWER perteneciente a OTECEL. Con el

paso de tiempo ambas empresas fueron adquiridas por AMÉRICA MÓVIL y TELEFÓNICA respectivamente cambiando de nombre que ahora se conoce como CLARO y MOVISTAR.

Las empresas Porta y Movistar iniciaron con AMPS (sistema telefónico móvil avanzado), el cual, es un sistema de telefonía móvil de primera generación donde solo se permite la comunicación por voz analógica; luego migraron a TDMA (Time Division Multiple Access), que es una tecnología inalámbrica de segunda generación la misma que permite dar servicios de voz y datos; mientras Porta continuo hacia GSM, Movistar migró hacia CDMA, la cual, es una tecnología más avanzada porque pasa a ser digital y permite la incursión de dispositivos móviles más avanzados. En la actualidad las tres operadoras manejan GSM (sistema global para las comunicaciones móviles), la cual, es una red digital que permite la transferencia de voz y datos, además del envió y recepción de mensajes escritos y poder acceder al roaming (utilizar el mismo número en todo el mundo accediendo a plataformas de diferentes operadoras en cada país que vaya el cliente) (Valdivieso Flores, 2011).

Las compañías trabajan con la tecnología 4G y se habla de la red 5G y 6G que ofrecen velocidades superiores para integrar los servicios que se manejan en la actualidad como Cloud, Big Data e IOT convergiendo todos estos términos a la llamada “transformación digital”.

1.4 Modelo del Churn (Teórico)

De acuerdo a (Domínguez Doncel & Muñoz Vera, 2010, pág. 81) retener a un cliente resulta aproximadamente diez veces más barato que conseguir uno nuevo. Por eso debe ser una prioridad saber aplicar estrategias de retención y fidelización que consigan mantener y desarrollar a los clientes rentables y fieles.

Churn Rate es una métrica de fidelidad que complementa la de retención que mide la atracción. Es muy utilizada por las empresas cuyos clientes necesitan abonarse o suscribirse a un servicio o producto, por lo que se usa mucho en compañías de telecomunicaciones y editoriales (compañías telefónicas, de acceso a Internet, publicaciones periódicas, etc.).

El abandono de clientes, debido a la competencia o a la insatisfacción, es uno de los principales problemas a los que se enfrentan las compañías por loque analizar el Churn Rate es necesario para medir:

• El porcentaje de clientes que nos han dejado

• La calidad y rentabilidad de estos clientes

En realidad esta métrica debe desencadenar el análisis sobre la calidad y rentabilidad de los clientes que abandonan una empresa. Ninguna empresa puede permitirse dejar escapar a sus mejores clientes (Domínguez Doncel & Muñoz Vera, 2010, pág. 82).

CHURN = Clientes que se dieron de baja en el periodo / Clientes al inicio del periodo

Como se indicó anteriormente son varios los motivos por los que el cliente deserta de una empresa por lo que de alguna manera se tiene que ejecutar algún plan para evitarlo. Para aplicar un plan de retención de clientes primero se tiene que saber qué clientes son los que abandonarán a la compañía.

CAPÍTULO 2

METODOLOGÍA

La metodología utilizada en este trabajo es CRISP-DM que viene de Cross-Industry Standard Process for Data Mining, el cual es un método probado para orientar trabajos de minería de datos (IBM, 2012).

Como modelo de proceso, CRISP-DM ofrece un resumen del ciclo vital de minería de datos.

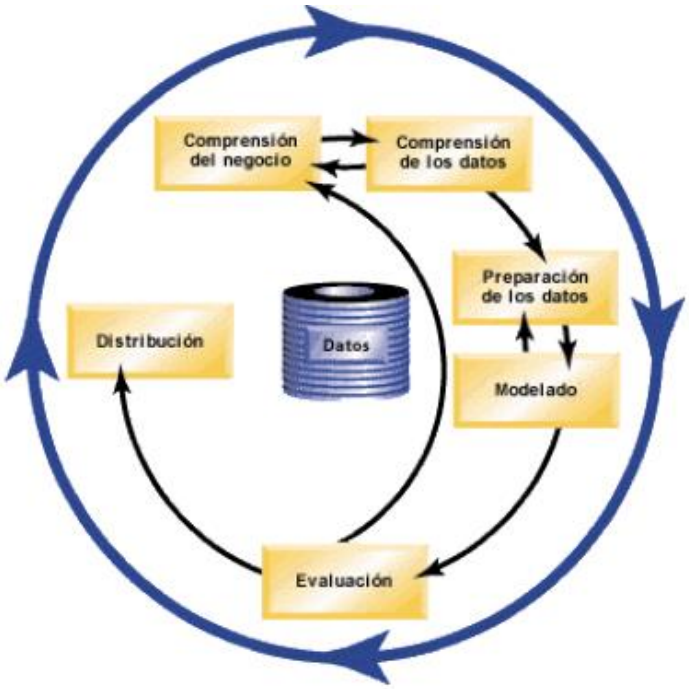


Figura 5 Ciclo del Modelo CRISP-DM. Fuente: IBM Corporation

De acuerdo a la figura 5, el ciclo vital del modelo consta de seis fases con flechas que indican las dependencias más importantes y frecuentes entre fases. Las secuencias de fases no es

estricta y la mayoría de proyectos avanzan y retroceden entre fases de ser necesario. Esto quiere decir que el modelo es flexible y se puede personalizar fácilmente (IBM, 2012).

A continuación se detalla cada fase de esta metodología.

2.1. Comprensión del negocio

2.1.1 Determinación de los objetivos del negocio

2.1.1.1 Backgorund

La empresa analizada es de información, comunicación, entretenimiento y brinda acceso al servicio móvil al territorio ecuatoriano poblado con productos y servicios de avanzada tecnología,Brinda a sus usuarios tecnología digital, GSM, 3G, HSPA+ y 4G LTE en las cuatro regiones del país y recientemente 4.5G.

2.1.1.2 Objetivos del negocio

Los objetivos del negocio son realizar la predicción de datos para los nuevos clientes de la empresa de tal forma que se logre hacer una predicción confiable a partir de los datos disponibles de dichos clientes. Partiendo de esta premisa se han definido los siguientes objetivos:

- Predecir los clientes que abandonarán la empresa a partir de sus datos
- Realizar la mejor predicción de los clientes que abandonarán la empresa

Este trabajo servirá para poder reaccionar ante la posible deserción de los clientes ofreciendo buenas estrategias de retención como otorgar 500MB de datos a dichos cliente. Esto

hará que la empresa se centre en la adquisición de nuevos clientes porque asegurará la permanencia de los clientes actuales.

2.1.1.3 Criterios de éxito del negocio

Los criterios de éxito del negocio son:

- La posibilidad de realizar predicciones sobre los nuevos clientes de la empresa con un alto grado de confiabilidad
- La posibilidad de realizar la mejor predicción

2.1.2 Evaluación de la situación

2.1.2.1 Inventario de recursos

La empresa utiliza principalmente bases de datos relacionales Oracle por lo que se obtuvo los datos usando sentencias SQL con PL/SQL Developer. Los algoritmos se desarrollaron en lenguaje R y los datos finales se obtuvieron usando un ETL (Extract, Transform and Load), programa de extracción transformación y carga desarrollado en lenguaje Perl.

Para el cálculo de la predicción del churn de esta solución se utilizó un computador personal con las siguientes características:

- Marca: Hewllet Packard
- Modelo: 15-da0015la
- Procesador: Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz 2.00 GHz
- Memoria RAM: 12.0 GB

- Capacidad de almacenamiento: 1 TB
- Tarjeta gráfica: NVIDIA GeForce MX130
- Sistema operativo: Microsoft Windows 10
- Monitor: PnP genérico

2.1.2.2 Requisitos, supuestos y restricciones

- Por cuestiones legales no se pudo obtener los datos reales de la empresa por lo que se usaron datos ficticios
- La empresa cuenta con una metodología de planificación de Proyectos
- Posee una política de calidad de datos de los clientes
- Maneja niveles de seguridad basado en roles de usuarios de empleados de la empresa

2.1.2.3 Riesgos y contingencia

En la tabla 1 se muestran los posibles riesgos que amenazan la consecusión del proyecto.

Tabla 1
Matriz de riesgos del proyecto

Riesgo	Probabilidad	Impacto
Falta de presupuesto	Baja	Alto
Fallas de hardware	Baja	Alto
No disponibilidad de fuentes de datos	Baja	Alto
Mala estimación de entregables	Baja	Alto
Desconocimiento del equipo en el desarrollo de proyectos	Baja	Alto

Nota: Elaborado por el Autor

La descripción de los riesgos de la tabla 1 se muestran a continuación:

- Falta de presupuesto

  Este riesgo puede darse debido a que no se destine el recurso económico necesario en el proyecto

- Fallas de hardware

  En el transcurso del proyecto pueden presentarse problemas en el hardware como daño por caídas de suministro eléctrico, climatización adecuada o fenómenos de la naturaleza

- No disponibilidad de fuentes de datos

  Sin la fuente de datos de clientes no es factible realizar la predicción

- Mala estimación de entregables

  Este riesgo tiene que ver con la disposición de los recursos necesarios en el tiempo establecido para su culminación en el tiempo acordado

- Desconocimiento del equipo en el desarrollo de proyectos

  Este riesgo es importante considerarlo por lo que se debe seleccionar el recurso humano con el conocimiento necesario en minería de datos

En la tabla 2 se describe la matriz de contingencia para mitigar los posibles riesgos presetados.

Tabla 2
Matriz de contingencia

Riesgo	Contingencia
Falta de presupuesto	Establecer un valor de contingencia que soporte una posible caída de flujo
Fallas de hardware	Revisiones y mantemnimiento constantes para evitar fallas de hardware y software
No disponibilidad de fuentes de datos	Realizar una copia de seguridad de la base de datos de manera local y otra en la nube para prevenir la falta de acceso
Mala estimación de entregables	Realizar un Project/Excel y determinar tiempos de desarrollo considerando el tiempo de los entregables
Desconocimiento del equipo en el desarrollo de proyectos	Realizar las reuniones con las personas que tengan mayor experiencia en las áreas involucradas del proyecto

Nota: Elaborado por el Autor

2.1.2.4 Costos y beneficios

- El costo total del Proyecto está valorado en $13880,00 descritos en la sección 2.1.4.1
- El beneficio que tendrá la empresa es poder retener a los posibles clientes que abandonarán la empresa, concentrándose en la aquisición de nuevos clientes

2.1.3 Determinación de los objetivos de minería de datos

2.1.3.1 Objetivos de la minería de datos

Predecir a los clientes que desertarán la empresa a partir de los datos del cliente.

2.1.3.2 Criterios de éxito de minería de datos

- De acuerdo a la minería de datos se establece como criterio de éxito la posibilidad de realizar predicciones de churn de clientes con un grado de confiabilidad de 75%

- El grado de confiabilidad dependerá del algoritmo que se emplee en el modelo de minería de datos

2.1.4 Producción de un plan del proyecto

2.1.4.1 Plan del Proyecto

El plan del proyecto se muestra en la figura 6.

Nombre de tarea	Costo	Duración	Comienzo	Fin	Predecesora	Nombres de los recursos
Pedicción del churn de clientes	**$13.880,00**	**33 días**	**lun 2/9/19**	**mié 16/10/19**		
Tareas del líder del proyecto	**$13.880,00**	**33 días**	**lun 2/9/19**	**mié 16/10/19**		**Líder de proyectos[25%]**
Análisis	**$6.160,00**	**19 días**	**lun 2/9/19**	**jue 26/9/19**		
Entrevistas con los expertos	$1.300,00	4 días	lun 2/9/19	jue 5/9/19		Ingeniero de proyectos 1;Ingeniero de proyectos 2
Análisis de datos	$2.260,00	7 días	vie 6/9/19	lun 16/9/19	4	Ingeniero de proyectos 1;Ingeniero de proyectos 2
Análisis de requisitos	$980,00	3 días	mar 17/9/19	jue 19/9/19	5	Ingeniero de proyectos 1;Ingeniero de proyectos 2
Ingeniería de Sistemas	$1.620,00	5 días	vie 20/9/19	jue 26/9/19	6	Ingeniero de proyectos 1;Ingeniero de proyectos 2
Desarrollo BI	**$3.400,00**	**11 días**	**vie 27/9/19**	**vie 11/10/19**		
Selección y depuración de datos	$820,00	5 días	vie 27/9/19	jue 3/10/19	7	Ingeniero de proyectos 1
Elaboración de algoritmos de Machine Learning	$960,00	6 días	vie 27/9/19	vie 4/10/19	7	Ingeniero de proyectos 2
Implementación de solución	$1.620,00	5 días	lun 7/10/19	vie 11/10/19	10	Ingeniero de proyectos 1;Ingeniero de proyectos 2
Documentación	**$1.020,00**	**3 días**	**lun 14/10/19**	**mié 16/10/19**		
Informes de finalización de cada fase	$180,00	1 día	lun 14/10/19	lun 14/10/19	11	Ingeniero de proyectos 1
Manuales de usuario	$160,00	1 día	lun 14/10/19	lun 14/10/19	11	Ingeniero de proyectos 2
Guias de administración	$180,00	1 día	mar 15/10/19	mar 15/10/19	13	Ingeniero de proyectos 1
Seguridad Informática	$160,00	1 día	mar 15/10/19	mar 15/10/19	14	Ingeniero de proyectos 2
Analisis de riesgo informático	$180,00	1 día	mié 16/10/19	mié 16/10/19	15	Ingeniero de proyectos 1
Acta de entrega y satisfacción	$160,00	1 día	mié 16/10/19	mié 16/10/19	16	Ingeniero de proyectos 2

Figura 6 Plan del proyecto. *Nota: Elaborado por el Autor*

2.1.4.2 Evaluación de herramientas y técnicas

Las herramientas y técnicas utilizadas en esta solución son:

- Lenguaje R (RStudio)

  Es un lenguaje sencillo y adecuado para minería de datos

- Lenguaje Perl

Es un lenguaje sencillo para manejar gran cantidad de datos en formato texto

- Paquetes de lenguaje R de red neuronal caret (Classification And REgresion training)

  Se usa este paquete porque consiste de un conjunto de funciones que tratan de simplificar el proceso para la creación de modelos predictivos (Kuhn, 2019)

- Paquete de lenguaje R de SVM de caret
- Paquete de lenguaje R de KNN de caret

2.2 Comprensión de los datos

2.2.1 Recopilación de datos iniciales

La predicción del churn se realiza de forma mensual, los datos utilizados en esta solución corresponden a los clientes postpago del mes de mayo que se usaron como datos de entrenamiento y del mes de junio usados como datos de prueba.

Los datos adquiridos son los siguientes:

- Tabla Personas:

  Los clientes están identificados por un número

- Tabla Contratos:

  Los contratos están identificados por un número

- Tabla Servicios:

  Los servicios están identificados por un código correspondiente al servicio, que en este caso será secuencial por las altas y bajas.

Los atributos específicos que se usaron son:

- Identificador del cliente
- Tipo de cliente
- Tipo de producto
- Forma de pago
- Región del cliente
- Última Facturación
- Churn

Las tablas de donde se seleccionaron los atributos son las siguientes:

- personas
- contratos
- servicios

En cuanto a la transformación, se codificaron los siguientes campos que contenían los caracteres y se les asignó un número para facilitar su integración con los algoritmos:

- Tipo de cliente
- Tipo de producto
- Forma de pago
- Región del cliente

2.2.2 Descripción de los datos

El modelo de base de datos es un modelo relacional de Oracle mostrado en la figura 7.

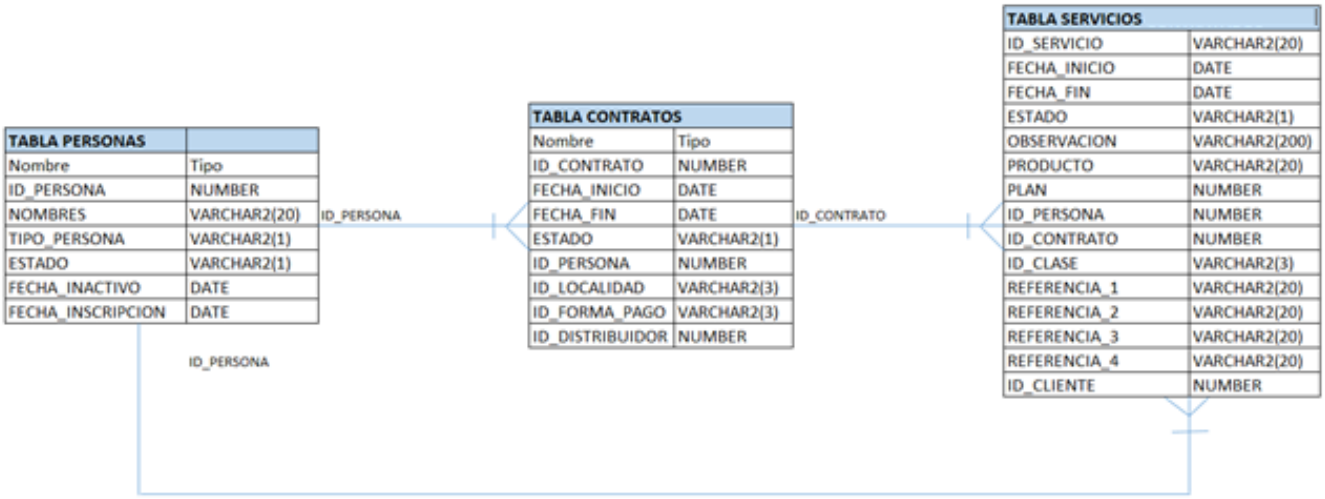


Figura 7 Modelo Entidad Relación. Nota: Elaborado por el Autor

Se puede apreciar en la figura 7 que para las activaciones solo se necesitan estas 3 tablas que se describen a continuación:

Tabla PERSONAS

Esta tabla es la que contiene a los clientes de la empresa y tiene como clave primaria el campo id_persona, a continuación se describen sus campos:

- Id_persona: Número secuencia que identifica a un cliente y es único para cada cliente
- Nombres: Campo tipo carácter iue identifica los nombres del cliente
- Tipo_persona: Campo tipo carater que identifica el tipo de persona, sea natural (N) o jurídica (J)
- Estado: Campo tipo carácter que identifica el estado del cliente, sea activo (A) o inactivo (I)

- Fecha_inactivo: Campo tipo fecha que identifica la fecha en que el cliente se inactivó
- Fecha_inscripcion: Campo tipo fecha que identifica la fecha en que el cliente se activó

Tabla CONTRATOS

Es la tabla que contiene los contratos de los clientes y tiene como clave primaria el campo id_contrato, a continuación se describen sus campos:

- id_contrato: Número secuencial que identifica el contrato del cliente y es un campo único para cada contrato
- fecha_inicio: Campo tipo fecha que identifica la fecha en que se activó el contrato
- fecha_fin: Campo tipo fecha que identifica la fecha en que se inactivó el contrato
- estado: Campo tipo carácter que identifica el estado del contrato, sea activo (A) o inactivo (I)
- id_persona: Campo tipo numérico que identifica al cliente del contrato y se trata de una clave foránea
- id_localidad: Campo tipo carácter que identifica la ciudad a la cual pertenece el contrato
- id_forma_pago: Campo tipo carácter que identifica la forma de pago del contrato
- id_distribuidor: Campo tipo numéricio que identifica al distribuidor que activó el contrato

Tabla SERVICIOS

Esta tabla contiene todos los servicios que un cliente tiene contratado y tiene con clave primaria el campo id_servicio, a continuación se detallan los campos:

- id_servicio: Campo tipo numérico que identifica el servicio que tiene un contrato y es un campo único para cada servicio
- fecha_inicio: Campo tipo fecha que identifica la fecha en que se activó el servicio
- fecha_fin: Campo tipo fecha que identifica la fecha en que se inactivó el servicio
- estado: Campo tipo carácter que identifica el estado del servicio, sea activo (A) o inactivo (I)
- observacion: Campo tipo carácter que identifica las observaciones hechas al servicio
- producto: Campo tipo carácter que identifica el tipo de producto del servicio, para este caso (TAR) son los tarifados y (AUT) son los autocontrolados, es decir, un híbrido entre postpago y prepago
- plan: Campo tipo numérico que identifica el identificador del plan del servicio
- id_persona: Campo tipo numérico que identifica al cliente del servicio y se trata de una clave foránea
- id_contrato: Campo tipo numérico que identifica el contrato del servicio y se trata de una clave foránea
- id_clase: Campo tipo carácter que identifica la tecnología del servicio ya sea TDMA = 1 o GSM = 2

- referencia_1: Campo tipo carácter que identifica el imei del equipo asociado al servicio
- referencia_2: Campo tipo carácter que identifica el modelo del equipo asociado al servicio
- referencia_3: Campo tipo carácter que identifica la homologaciónm del equipo asociado al servicio
- referencia_4: Campo tipo carácter que identifica la simcard asociada al servicio
- id_cliente: Campo tipo numérico que identifica un número asociado al servicio para la facturación

2.2.3 Verificación de calidad de los datos

Se partió de una base de registros de clientes, se realizó una depuración de datos con un ETL hecho con un script en lenguaje Perl. Este script descartó los campos nulos y también realizó formateo de datos para los campos tipo_persona, producto, forma_pago, región y una simulación de última facturación.

2.3 Preparación de los datos

2.3.1 Selección de los datos

Como se indicó anteriormente los campos seleccionados para el análisis de minería de datos son los siguientes:

- Tabla contratos
    - id_cliente
    - forma_pago
    - región

- churn

- Tabla clientes
  - tipo_persona
- Tabla servicios
  - Producto
- Última facturación

Dato simulado con un script en Perl

2.3.2 Limpieza de los datos

Para filtrar los datos no adecuados, se utilizó un script en Perl que selecciona los datos que son diferentes de nulo y que no cumplen con lo requerido. El esquema usado se muestra en la figura 8.



Figura 8 Filtrado de registros. Nota: Elaborado por el Autor

2.3.3 Construcción de los datos

Se simuló el campo facturación con un script en Perl.

2.3.4 Integración de los datos

No ha sido necesario la integración de nuevos datos debido a que con los existentes es suficiente para la predicción.

2.3.5 Formateo de los datos

Los siguientes campos han sido codificados:

- Tipo de persona
- Tipo de producto
- Forma de pago
- Región del cliente

La codificación utilizada se describe en el anexo1 y es la siguiente:

- tipo_persona
    - 1 -> N (Natural)
    - 2 -> J (Jurídica)
- producto
    - 1 -> AUT (Autocontrol)
    - 2 -> TAR (Tarifario)
- forma_pago
    - 1 -> BAN (Bancos)
    - 2 -> BAT (Transferencia)
    - 3 -> CFA (Contrafactura)

- 4 -> CFV (Forma de pago varios)
- 5 -> TCR (Tarjeta de crédito)
- 6 -> TCT (Tarjeta de débito)

- región
  - 1 -> GYE (Guayaquil), PTV (Portoviejo), MAC (Machala), BAB (Babahoyo), QUE (Quevedo), ESM (Esmeraldas), PLY (Playas), DUR (Durán), MNT (Manta), MIL (Milagro), JPJ (Jipijapa), CHN (Chone), SAL (Salinas), HQL (Huaquillas), TSG (Tosagua), DAU (Daule)
  - 2 -> UIO (Quito), CAY (Cayambe), CUE (Cuenca), STD (Santo Domingo), IBA (Ibarra), TUL (Tulcán), LOJ (Loja), AMB (Ambato), RBB (Riobamba), LTA (Latacunga), GUA (Gualaceo), CCA (Cotacachi), AZO (Azogues), MCH (Machachi)
  - 3 -> TEN (Tena), PYO (Puyo), MCS (Macas), ZAM (Zamora)
  - 4 -> SCZ (Santa Cruz)

2.4 Modelado

2.4.1 Elección de la técnica de modelado

Debido a que se trata de técnicas de predicción se seleccionaron las siguientes técnicas:

- Red neuronal
- SVM
- KNN

2.4.2 Generación del plan de prueba

Se utilizaron como datos de entrenamiento las altas y bajas del mes de mayo y como datos de prueba se utilizaron las altas y bajas del mes de junio hasta el día 23. La precisión de los algoritmos de predicción se obtiene de las matrices de confusión (Visa, Ramsay, Ralescu, & van der Knaap, 2011), tal como se muestra en la figura 9.

	PREDICTED NEGATIVE	PREDICTED POSITIVE
ACTUAL NEGATIVE	a	b
ACTUAL POSITIVE	c	d

$$Accuracy = \frac{a+d}{a+b+c+d}$$

Figura 9 Precisión de una matriz de confusión.
Fuente: (Visa, Ramsay, Ralescu, & van der Knaap, 2011)

Los significados de los valores de la matriz de confusión se describen a continuación:

- *a* es el número de predicciones negativas correctas
- *b* es el número de predicciones positivas incorrectas
- *c* es el número de predicciones negativas incorrectas
- *d* es el número de predicciones positivas correctas

La precisión de la matriz de confusión es más confiable siempre y cuando haya un balanceo de los datos con respecto al churn, es decir, cantidades similares de 0 y 1 por lo que previamente se usó un script en perl que balanceó los datos. El balanceo de los datos se da en función de las bajas por lo que en el mes de mayo hubo 16661 clientes que se fugaron por lo que se tomó aproximadamente la misma cantidad de altas. Para el mes de junio hubo 4993 clientes fugados y

al igual que el mes de mayo se tomó aproximadamente igual cantidad de altas por lo que finalmente se harán las predicciones en base a los datos balanceados del mes junio, es decir, sobre 9986 clientes.

2.4.3 Construcción del modelo

Para la construcción de todos los modelos se usó el paquete llamado caret (Kuhn, 2019), el cual contiene lo 3 métodos que se aplicaron en este trabajo. Asímismo para todos los métodos se siguió con la siguiente estrategia:

- Se balancearon los datos en función de las bajas tanto del mes de mayo como de junio dejando aproximadamente igual número de altas
- Se entrenó al algoritmo con el 70% de los datos balanceados del mes de mayo
- Los datos de prueba correspondieron al 30% de los datos balanceados de mayo
- Se usó el algoritmo con todos los datos balanceados del mes de junio
- Se calculó la precision de la matriz de confusión
- Se obtuvo un archivo con los clientes predichos

Los modelos se detallan a continuación:

- Red neuronal

El algoritmo de la red neuronal se muestra en la figura 10.

```
library(caret)

setwd("C:\\Documentos")

datos<-read.csv("clientes_mayo_balanceados.csv", sep=";")

intrain <- createDataPartition(y = datos$churn, p= 0.7, list = FALSE)

# Datos balanceados de entrenamiento de mayo
training <- datos[intrain,]

# Datos balanceados de prueba de mayo
testing <- datos[-intrain,]

training$churn = as.factor(training$churn)
trctrl <- trainControl(method = "repeatedcv", number = 5, repeats = 3)

base_1=training[,-1]

nnet <- train(as.factor(churn) ~ tipo_persona+producto+forma_pago+region+facturacion, data = base_1, method = "nnet",
                                    trControl=trctrl,
                                   preProcess = c("center", "scale"),
                                  tuneGrid = expand.grid(size=c(10), decay=c(0.1)))

pred_NNET <- predict(nnet, newdata = testing[,-c(1,7)])

# Matriz de confusion de datos balanceados del mes de mayo
confusionMatrix(pred_NNET, as.factor(testing$churn))

# Datos nuevos de junio
nuevos_datos<-read.csv("clientes_junio_balanceados.csv", sep=";")
testing_nuevo <- nuevos_datos

pred_NNET_nuevo <- predict(nnet, newdata = testing_nuevo[,-c(1,7)])

# Matriz de confusion de datos balanceados del me de junio
confusionMatrix(pred_NNET_nuevo, as.factor(testing_nuevo$churn))

datafinal=testing_nuevo
datafinal$prednnet=pred_NNET_nuevo
datafinal[,c('id_cliente', 'churn','prednnet')]
write.csv(datafinal, file='clientes_predichos_nnet.csv')
```

Figura 10 Algoritmo de red neuronal – Código en R. Nota: Elaborado por el Autor

El algoritmo de la figura 10 realiza lo siguiente:

- Toma los datos balanceados del mes de mayo
- Utiliza una partición de los datos de entrenamiento y prueba con los datos de mayo
- Aplica el método **nnet** del paquete caret con los datos de entrenamiento
- Realiza la predicción con los datos de prueba
- Obtiene la matriz de confusión de los datos de prueba
- Toma como datos de prueba los datos balanceados del junio
- Realiza la predicción de los datos de prueba

- Obtiene la matriz de confusión de los datos de prueba
- Obtiene un listado con los clientes fugados

Este algoritmo muestra la matriz de confusión en la tabla 3.

Tabla 3

Matriz de confusión de la red neuronal

	Reference	
Prediction	0	1
0	4845	1863
1	149	3129

Nota: Elaborado por el Autor

De acuerdo a la definición de la figura 9 en la sección 2.4.2, el cálculo de la precisión de la matriz de confusión de la tabla 3 se muestra en la figura 11.

$$\text{Precisión} = \frac{4845 + 3129}{4845 + 1863 + 149 + 3129} = 79{,}85\%$$

Figura 11 Precisión de la matriz de confusión de la red neuronal. *Nota: Elaborado por el Autor*

- **SVM**

El algoritmo de SVM se muestra en la figura 12.

```
library(caret)

setwd("C:\\Documentos")

datos<-read.csv("clientes_mayo_balanceados.csv", sep=";")

intrain <- createDataPartition(y = datos$churn, p= 0.7, list = FALSE)

# Datos balanceados de entrenamiento de mayo
training <- datos[intrain,]

# Datos balanceados de prueba de mayo
testing <- datos[-intrain,]

training$churn = as.factor(training$churn)
trctrl <- trainControl(method = "repeatedcv", number = 5, repeats = 3)

base_1=training[,-1]

svm_Linear <- train(as.factor(churn) ~ tipo_persona+producto+forma_pago+region+facturacion, data = base_1, method = "svmLinear",
                                       trControl=trctrl,
                                     preProcess = c("center", "scale"),
                                    tuneLength = 10)

pred_SVM <- predict(svm_Linear, newdata = testing[,-c(1,7)])

# Matriz de confusion de datos balanceados del mes de mayo
confusionMatrix(pred_SVM, as.factor(testing$churn))

# Datos nuevos de junio
nuevos_datos<-read.csv("clientes_junio_balanceados.csv", sep=";")
testing_nuevo <- nuevos_datos

pred_SVM_nuevo <- predict(svm_Linear, newdata = testing_nuevo[,-c(1,7)])

# Matriz de confusion de datos balanceados del me de junio
confusionMatrix(pred_SVM_nuevo, as.factor(testing_nuevo$churn))

datafinal=testing_nuevo
datafinal$predsvm=pred_SVM_nuevo
datafinal[,c('id_cliente', 'churn','predsvm')]
write.csv(datafinal, file='clientes_predichos_svm.csv')
```

Figura 12 Algoritmo SVM en R. Nota: Elaborado por el autor

El algoritmo de la figura 12 realiza lo siguiente:

- Toma los datos balanceados del mes de mayo
- Utiliza una partición de los datos de entrenamiento y prueba con los datos de mayo
- Aplica el método **svmLinear** del paquete caret con los datos de entrenamiento
- Realiza la predicción con los datos de prueba
- Obtiene la matriz de confusión de los datos de prueba
- Toma como datos de prueba los datos balanceados del junio
- Realiza la predicción de los datos de prueba

- Obtiene la matriz de confusión de los datos de prueba
- Obtiene un listado con los clientes fugados

La matriz de confusión obtenida se muestra en la tabla 4.

Tabla 4

Matriz de confusion de SVM

	Reference	
Prediction	0	1
0	4849	2176
1	145	2816

Nota: Elaborado por el Autor

De acuerdo a la definición de la figura 9 en la sección 2.4.2, el cálculo de la precisión de la matriz de confusión de la tabla 4 se muestra en la figura 13.

$$\text{Precisión} = \frac{4849 + 2816}{4849 + 2176 + 145 + 2816} = 76{,}76\%$$

Figura 13 Precisión de la matriz de confusión de SVM. Nota: Elaborado por el Autor

- **KNN**

  El algoritmo se muestra en la figura 14.

```
library(caret)

setwd("C:\\Documentos")

datos<-read.csv("clientes_mayo_balanceados.csv", sep=";")

intrain <- createDataPartition(y = datos$churn, p= 0.7, list = FALSE)

# Datos balanceados de entrenamiento de mayo
training <- datos[intrain,]

# Datos balanceados de prueba de mayo
testing <- datos[-intrain,]

training$churn = as.factor(training$churn)
trctrl <- trainControl(method = "repeatedcv", number = 5, repeats = 3)

base_1=training[,-1]

knn <- train(as.factor(churn) ~ tipo_persona+producto+forma_pago+region+facturacion, data = base_1, method = "knn",
                                    trControl=trctrl,
                                   preProcess = c("center", "scale"),
                                   tuneLength = 10)

pred_KNN <- predict(knn, newdata = testing[,-c(1,7)])

# Matriz de confusion de datos balanceados del mes de mayo
confusionMatrix(pred_KNN, as.factor(testing$churn))

# Datos nuevos de junio
nuevos_datos<-read.csv("clientes_junio_balanceados.csv", sep=";")
testing_nuevo <- nuevos_datos

pred_KNN_nuevo <- predict(knn, newdata = testing_nuevo[,-c(1,7)])

# Matriz de confusion de datos balanceados del me de junio
confusionMatrix(pred_KNN_nuevo, as.factor(testing_nuevo$churn))

datafinal=testing_nuevo
datafinal$predknn =pred_KNN_nuevo
datafinal[,c('id_cliente', 'churn','predknn')]
write.csv(datafinal, file='clientes_predichos_knn.csv')
```

Figura 14 Algoritmo KNN en R. Nota: Elaborado por el Autor

El algoritmo de la figura 14 realiza lo siguiente:

- Toma los datos balanceados del mes de mayo
- Utiliza una partición de los datos de entrenamiento y prueba con los datos de mayo
- Aplica el método **knn** del paquete caret con los datos de entrenamiento
- Realiza la predicción con los datos de prueba
- Obtiene la matriz de confusión de los datos de prueba
- Toma como datos de prueba los datos balanceados del junio

- Realiza la predicción de los datos de prueba
- Obtiene la matriz de confusión de los datos de prueba
- Obtiene un listado con los clientes fugados

La matriz de confusión se muestra en la tabla 5.

Tabla 5

Matriz de confusión de KNN

	Reference	
Prediction	0	1
0	4829	1972
1	165	3020

Nota: Elaborado por el Autor

De acuerdo a la definición de la figura 9 en la sección 2.4.2, el cálculo de la precisión de la matriz de confusión de la tabla 5 se muestra en la figura 15.

$$\text{Precisión} = \frac{4829 + 3020}{4849 + 1972 + 165 + 3020} = 78{,}60\%$$

Figura 15 Precisión de la matriz de confusión de KNN. Nota: Elaborado por el Autor

2.4.4 Evaluación del modelo

Para realizar una mejor comparativa la tabla 6 muestra de una mejor forma los valores obtenidos en las matrices de confusión.

Tabla 6
Precisión de los modelos

	Valores de las matrices				
Algoritmo	***a***	***b***	***c***	***d***	***Precisión***
Red neuronal	4845	1863	149	3129	79,85
SVM	4849	2176	145	2816	76,76
KNN	4849	1972	165	3020	78,6

Nota: Elaborado por el Autor

2.5 Evaluación

2.5.1 Evaluación de los resultados

En base a los resultados obtenidos en la tabla 6 se concluye que el mejor modelo es red neuronal que con 79,85% de precision, mostró el mejor resultado prediciendo 3129 clientes fugados y 149 posibles clientes que se fugarán. Los resultados también muestran que el segundo mejor modelo es el de KNN con una precision de 78,60% y con una predicción de 3020 clientes fugados y 165 posibles clientes que se fugarán.

Se desarrolló un solo programa en R que realiza el llamado de los 3 algoritmos del paquete caret, evalúa los resultados de las precisions de las 3 matrices de confusión, selecciona la mejor precision, predice los posibles clientes que se fugarán y los escribe en un archivo con formato csv.

Este programa en R es llamado por un programa desarrollado en Php que muestra los resultados luego de ser invocados por una página web hecha en Php.

2.5.2 Revisión del proceso

Durante el proceso de construcción del modelo se tuvo que evaluar varios métodos como por ejemplo en el caso de la red neuronal se usaron los paquetes neuralnet y h2o que se descartaron porque la precisión de la matriz de confusión y la predicción de los clientes fugados no fue aceptable. De igual forma se aplicaron los mismos criterios para los métodos SVM y KNN por lo que se optó usar los métodos que posee el paquete "caret" debido a que mostraron mejores resultados.

A continuación se indicarán los paquetes de red neuronal llamados neuralnet y h2o:

- Neuralnet

```
library(neuralnet)
library(ggplot2)

# Se establece directorio de trabajo
setwd("C:\\Documentos")

# Los datos de entrenamiento serán los clientes de mayo
mydata <- read.csv("clientes_junio_depurados.csv")
attach(mydata)

# Se normalizan los datos
# Normalización escalada
scaleddata<-scale(mydata)

# Normalización por max-min
normalize <- function(x) {
  return ((x - min(x)) / (max(x) - min(x)))
}

# Datos normalizados de mayo
maxmindf <- as.data.frame(lapply(mydata, normalize))
# Datos normalizados de junio

n=nrow(mydata)
porc_train=0.8
ntrain=as.integer(n*porc_train)

vtrain=sample(1:n, ntrain)

# Datos de entrenamiento
train=maxmindf[vtrain,]
# Datos de prueba
test=maxmindf[-vtrain,]

# Entrenando la red neuronal con neuralnet con datos de mayo
ann_model <- neuralnet(churn~id_cliente+tipo_persona+producto+forma_pago+region, data=train, hidden=c(2,1),
                       linear.output=FALSE, threshold=0.01)
ann_model$result.matrix
plot(ann_model)

temp_test <- subset(test, select = c("id_cliente","tipo_persona", "producto", "forma_pago", "region"))
head(temp_test)
ann_model.results <- compute(ann_model, temp_test)
results <- data.frame(actual = test$churn, prediction = ann_model.results$net.result)

#Matriz de confusión
roundedresults<-sapply(results,round)
roundedresultsdf=data.frame(roundedresults)
attach(roundedresultsdf)
table(actual,prediction)
```

Figura 16 Algoritmo de red neuronal con paquete neuralnet en R. Nota: Elaborado por el Autor

El algoritmo de la figura 16 realiza lo siguiente:

- Toma los datos del mes de mayo
- Utiliza una partición de los datos de entrenamiento y prueba con los datos de mayo
- Aplica el método **neuralnet** del paquete neuralnet con los datos de entrenamiento

- Realiza la predicción con los datos de prueba
- Obtiene la matriz de confusión de los datos de prueba
- Obtiene la matriz de confusión de los datos de prueba

- H2o

```
library(caTools)
library(h2o)
library(ROCR)

setwd("C:\\Documentos")
  churn_mayo<-read.csv("clientes_mayo_depurados.csv")
  churn_junio<-read.csv("clientes_junio_depurados.csv")
n=nrow(churn_junio)
colnames(churn_mayo)[6] <- "churn"
colnames(churn_junio)[6] <- "churn"

churn_mayo$churn <- factor(churn_mayo$churn, levels = c("0", "1"), labels = c("No", "Si"))
churn_junio$churn <- factor(churn_junio$churn, levels = c("0", "1"), labels = c("No", "Si"))


churn_mayo[, c(1:5)] <- scale(churn_mayo[, c(1:5)])
churn_junio[, c(1:5)] <- scale(churn_junio[, c(1:5)])


set.seed(1234)
training_set <- churn_mayo
test_set <- churn_junio

precision=0
intentos=0
h2o.init(nthreads = -1)

classifier = h2o.deeplearning(y = 'churn',
                              training_frame = as.h2o(training_set),
                              activation = 'RectifierWithDropout',
                              hidden = c(5, 5, 5),
                              epochs = 100,
                              train_samples_per_iteration = -2)


prob_pred <- h2o.predict(classifier, newdata = as.h2o(test_set))
y_pred <- as.vector(ifelse(prob_pred$predict == 'No', 0, 1))
y_test_set <- ifelse(test_set$churn == 'No', 0, 1)

cm <- table(y_test_set, y_pred)
cm

h2o.shutdown()
```

Figura 17 Algoritmo de red neuronal con paquete h2o en R. Nota: Elaborado por el Autor

El algoritmo de la figura 17 realiza lo siguiente:

- Toma los datos del mes de mayo
- Toma los datos del mes de junio
- Utiliza una partición de los datos de entrenamiento con los datos de mayo y prueba con los datos del mes de junio
- Aplica el método **h2o.deeplearning** del paquete h2o con los datos de entrenamiento
- Realiza la predicción con los datos de prueba
- Obtiene la matriz de confusión de los datos de prueba

2.5.3 Determinación de los próximos pasos

El siguiente paso consiste en ejecutar la etapa de distribución para cumplir con los objetivos comerciales definidos en la sección 2.1.1.2.

2.6 Distribución

La distribución es un proceso que consiste en utilizar nuevos conocimientos, como por ejemplo de minería de datos, para implementar las mejoras de la organización. (IBM, 2012).

2.6.1 Planificación de la Implantación

Para la implantación de este proyecto es necesario tener acceso a datos reales, es decir, a toda la base de clientes de la empresa porque se indicó que los datos son ficiticios. Los pasos para implantar la solución son los siguientes:

- Obtener los datos de altas y bajas de clientes de 2 meses en 2 archivos con formato csv separados con “;” con la información siguiente:

    - id_cliente
    - tipo_persona
    - producto
    - forma_pago
    - region
    - facturacion
    - churn
- Los archivos deben llamarse de la siguiente manera:
    - clientes_mes1.csv
    - clientes_mes2.csv
- Colocar ambos archivos csv en un directorio llamado Documentos dentro de la unidad C:\
- Colocar los siguientes scripts de R y perl en el directorio anteriormente indicado:
    - modelos_de_prediccion.R
    - balancea.pl
    - selecciona_fugados.pl
- Ejecutar el modelo con los datos descritos anteriormente
- Obtener el listado de los posibles clientes a fugarse producto de la predicción que realiza esta solución en un archivo llamado clientes_a_fugarse_nnet.csv

CAPÍTULO 3

VALIDACIÓN DE LOS RESULTADOS

3.1 Esquema de validación

En este trabajo se aplicaron los métodos cuantitativos, los cuales permitieron demostrar que la solución propuesta resuelve el problema planteado. A lo largo del desarrollo de esta solución se utilizaron las siguientes formas de validación cuantitativas:

- Pre-experimentos
- Cuasi-experimentos
- Experimentos puros

Se probó con una selección de dos muestras de clientes cuyas variables independientes no presentaban una relación correcta con la variable dependiente llamada churn porque los resultados no eran los adecuados ni confiables. Luego se ingresó la variable independiente facturación del cliente, se separaron los grupos de tal manera que se tengan grupos balanceados con respecto al churn, es decir, grupos con cantidades similares de altas y bajas. Luego se obtuvieron resultados aceptables y confiables en base a las matrices de confusion y se construyó un solo algoritmo que abarcó toda la solución.

3.1.1 Pre-experimentos

En esta solución se trabajó con las alta y bajas tanto del mes de mayo como de junio inicialmente sin alterar el contenido de los datos obteniendo resultados poco satisfactorios aparte de incluir algoritmos que tomaban mucho tiempo en ejecutarse como por ejemplo los algoritmos independientes del paquete caret como neuralnet, h2o.depplearning, SVM y KNN.

La medida con la cual se consideraron resultados poco satisfactorios fueron las precisiones de sus matrices de confusión debido a que mostraron resultados sin relación.

3.1.2 Cuasi-experimentos

Luego de obtener los resultados sin relación como se indicó anteriormente y con el propósito de mejoralos, se ingresó una nueva variable independiente llamada facturación simulando los valores facturados de los clientes y se utilizó el paquete caret con lo que se obtuvo resultados un poco más razonables.

3.1.3 Experimentos puros

Finalmente luego de revisar los datos anteriores y considerando que dichos datos no eran muy confiables se usó un balanceo de datos, que como se indicó anteriormente consiste en tomar igual cantidad de altas en función de las bajas existentes tanto del mes de mayo como de junio. Para el caso del mes de mayo hubo 16661 clientes fugados por lo que se tomó un número similar de altas, mientras que para el mes de junio hubo 4993 clientes fugados que con un número aproximado de altas se obtuvo un total de 9986 clientes.

3.2 Aplicación y descripción de los resultados

3.2.1 Pre-experimentos

- **Red neuronal**

A pesar de que la precisión de la matriz de confusión del algoritmo h2o.deeplearning fue alta, se obtuvieron valores predichos de clientes fugados considerados muy bajos debido al total de clientes tal como se muestra en la tabla 7.

Tabla 7

Matriz de confusión obtenida con h2o.deeplearning con todos los clientes

	y_pred	
y_test_set	0	1
0	186491	3244
1	4545	448

Nota: Elaborado por el Autor

- **SVM**

Similarmente como sucedió con el algoritmo de red neuronal que también dio como resultado una muy buena precisón de la matriz de confusión pero ningún cliente fugado tal como se muestra en la tabla 8.

Tabla 8

Matriz de confusión obtenida con SVM con todos los clientes

test_pred	0	1
0	189735	4993
1	0	0

Nota: Elaborado por el Autor

- **KNN**

Los resultados del algoritmo KNN no son la excepción debido a que la precision obtenida de la matriz de confusión fue alta con un número de 102 clientes fugados tal como se muestra en la tabla 9.

Tabla 9

Matriz de confusión obtenida con KNN con todos los clientes

knn.prd	0	1
0	183861	4891
1	5874	102

Nota: Elaborado por el Autor

3.2.2 Cuasi-experimentos

- **Red neuronal**

A pesar que la precisión de la matriz de confusión de la tabla 10 fue alta, la predicción de clientes fugados fue de 722, lo cual sigue siendo muy poco para la cantidad de clientes totales del mes de junio.

Tabla 10

Matriz de confusión de red neuronal con todos los clientes con caret

	Reference	
Prediction	0	1
0	186494	4271
1	3241	722

Nota: Elaborado por el Autor

- **SVM**

Para este caso la precision de la matriz de confusión mostrada en la tabla 11 fue muy baja, con una predicción de clientes fugados de de 2817, que al igual que el algoritmo anterior sigue siendo muy poco para la cantidad total de clientes.

Tabla 11

Matriz de confusión de SVM con todos los clientes con caret

	Reference	
Prediction	0	1
0	79552	2176
1	110183	2817

Nota: Elaborado por el Autor

- **KNN**

A pesar que la precisión de la matriz de confusión de la tabla 12 tambié fue alta, la predicción de clientes fugados fue de 675, que continua siendo muy poco para la cantidad de clientes.

Tabla 12

Matriz de confusión de KNN con todos los clientes con caret

	Reference	
Prediction	0	1
0	186553	4318
1	3182	675

Nota: Elaborado por el Autor

3.2.3 Experimentos puros

- **Red neuronal**

Con la matriz de confusión mostrada en la tabla 13, definitivamente los resultados obtenidos con un muestreo de datos balanceados son muy aceptables y confiables. Esto se concluye debido a la precisión de dicha matriz de 79,85%, mostrada en la tabla 6 de la sección 2.4.4 y a su predicción de 3062 clientes fugados y 139 posibles clientes que se fugarán.

Tabla 13

Matriz de confusión de red neuronal con datos balanceados

	Reference	
Prediction	0	1
0	4855	1930
1	139	3062

Nota: Elaborado por el Autor

- **SVM**

La matriz de confusión de SVM se muestra en la tabla 14 y al igual que el algoritmo anterior los resultados son aceptables y confiables con una precisión mostrada en la tabla 6 de la sección 2.4.4 de 76,76%, con 2816 clientes fugados y 145 posibles clientes que se fugarán.

Tabla 14

Matriz de confusión de red SVM con datos balanceados

	Reference	
Prediction	0	1
0	4849	2176
1	145	2816

Nota: Elaborado por el Autor

- **KNN**

La matriz de confusión de KNN se muestra en la tabla 15 y al igual que el algoritmo anterior los resultados son aceptables y confiables con una precisión mostrada en la tabla 6 de la sección 2.4.4 de 78,56%, con 3031 clientes fugados y con 180 posibles clientes que se fugarán.

Tabla 15

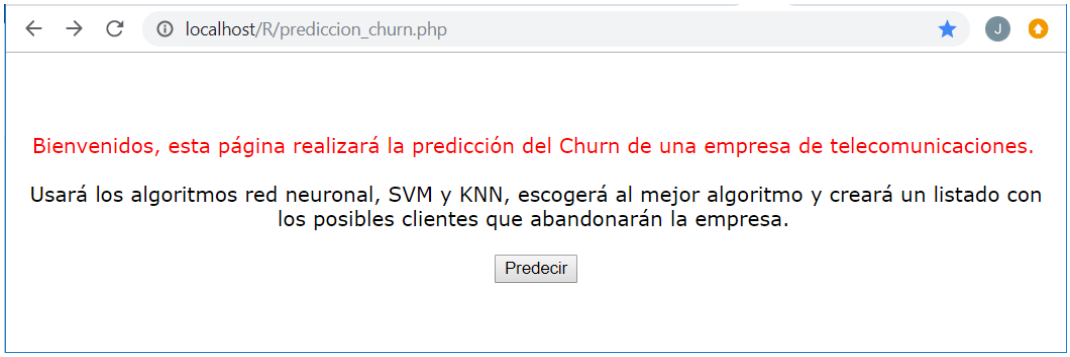
Matriz de confusión de red KNN con datos balanceados

	Reference	
Prediction	0	1
0	4814	1961
1	180	3031

Nota: Elaborado por el Autor

Finalmente, se unifican los 3 algoritmos en un solo programa en R que selecciona la mejor predicción en base a la precisión de la matriz de confusión y obtiene un listado de los posibles clientes a fugarse. Este programa en R llamado modelos_de_prediccion.R es invocado por una página web hecha en Php llamada predicción_churn.php que a su vez llama a otro programa en php llamado prediccion.php. La página web se muestra en la figura 18.

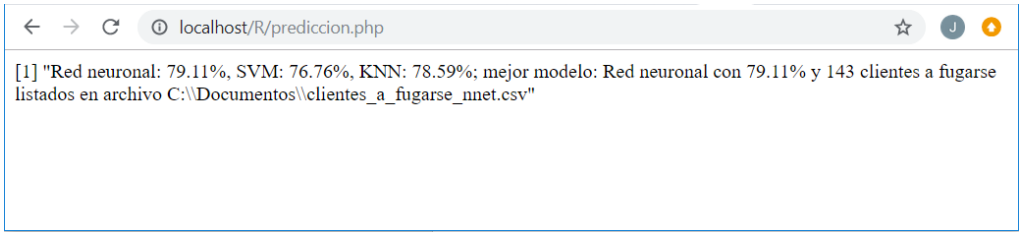
Figura 18 Página web prediccion_churn.php



Nota: Elaborado por el Autor

Los resultados de la predicción se muestran en la figura 19.

Figura 19 Programa prediccion.php



Nota: Elaborado por el Autor

Como se puede observar los datos mejoraron conforme se ingresó una nueva variable independiente a las muestras y seleccionando los datos de tal manera que se balanceran con respecto a la variable dependiente llamada churn y que corresponden a los clientes fugados, obteniendo una cantidad similar entre fugados y no fugados. En la figura 28 se muestra el mejor modelo de predicción que fue el de la red neuronal con una precision de 79,94%, la matriz de precision de la red neuronal, así como la cantidad de clientes que posiblemente se fugarán de una empresa y que este caso es de 145. Estos clientes se detallan en un archivo llamado "clientes_a_fugarse_nnet.csv".

3.3 Impacto

3.3.1 Perspectivas económicas

Desde la perspectiva económica, esta solución tiene un impacto positivo dado que con los resultados obtenidos se podrá establecer una adecuada estrategia para lograr la retención del

posible cliente predicho como fugable, logrando así concentrarse el negocio en la adquisición de nuevos clientes.

3.3.2 Perspectivas sociales

Desde esta prespectiva, esta solución tiene un impacto positivo debido a que con la retención de los posibles clientes a fugarse, sentirán un sentido de pertenencia a la empresa así como de satisfacción, puesto que se sentirán como clientes importantes. Esto hará que al transmitirse la satisfacción del cliente, atraigan a más clientes.

CONCLUSIONES

- Las técnicas de minería de datos establecen que existe una relación entre las variables independientes correspondientes a los datos de alguna muestra, que en este caso corresponden a los clientes con las variable dependiente que en este caso es el churn; y en base a estas relaciones se pueden establecer las predicciones midiendo sus resultados.
- En base a los resultados obtenidos se puede ir midiendo dichos resultados a partir de las técnicas de minería de datos logrando cada vez ir mejorándolos añadiendo nuevas variables independientes o aplicando ciertas técnicas de muestreo de datos hasta lograr resultados confiables.
- En esta solución se emplean los métodos de red neuronal, SVM y KNN para lograr la mejor predicción posible.
- Con la técnica de precision de la matriz de confusión, el programa elige al mejor algoritmo de los tres que resultó ser el algoritmo de red neuronal con una precisión de 79,94%.
- La solución crea un listado de los clientes que se predicen como posibles fugables a través de un archivo csv escrito en un directorio llamado Documentos dentro de la unidad C:\.

RECOMENDACIONES

- Se recomienda utilizar datos reales para obtener resultados más óptimos
- Los archivos de los clientes deben estar en formato csv separados con ";" correspondienes a las altas y las bajas de clientes de 2 meses (1 mes por archivo) con los siguientes campos:
    - id_cliente (código numérico que identifica al cliente)
    - tipo_persona (dato numérico con valores 1=persona natural, 2=persona jurídica)
    - producto (dato numérico con valores 1=tarifados, 2=autocontrolados)
    - forma_pago (dato numérico con valores que van del 1 al 6 representando las diferentes formas de pago)
    - region (dato numérico con valores que van del 1 al 4 representando a las regiones del Ecuador)
    - facturacion (valor numérico)
    - churn (valor numérico con vlaores 0=cliente no fugado, 1=cliente fugado)
- Los siguientes programas en lenguaje R y Perl deben estar en un directorio llamado "Documentos" dentro de la unidad C:\:
    - modelos_de_prediccion.R
    - balancea.pl
    - selecciona_fugados.pl

- Debido a que la solución completa consta de una página web para realizar la predicción, los siguientes programas desarrollados en Php deben ubicarse en un directorio llamado R dentro del directorio htdocs:
    - predicción_churn.php:

      Página web principal que invoca al programa predicción.php
    - predicción.php:

      Página que invoca al programa en R llamado modelos_de_prediccion.R para reealizar la predicción y muestra los resultados por web

Lista de referencias

Barrientos, F., & Ríos, S. A. (Septiembre de 2013). Aplicación de minería de datos para predecir fuga de clientes en la industria de las telecomunicaciones. *Ingeniería de Sistemas*.

Bradley, E., & Trevor, H. (2017). *Computer Age Statistical Inference.*

Domínguez Doncel, A., & Muñoz Vera, G. (2010). *Métricas del Marketing.* Madrid: Esic.

Espino Timón, C., & Martínez Fontes, X. (16 de Enero de 2017). *Análisis predictivo: técnicas y modelos utilizados y aplicaciones del mismo - herramientas Open Source que permiten su uso.* Universitat Oberta de Catalunya.

IBM. (2012). *Repositorio público de IBM.* Obtenido de ftp://public.dhe.ibm.com/software/analytics/spss/documentation/modeler/15.0/es/CRISP-DM.pdf

Jiangsheng, Y. (Enero de 2002). *ResearchGate.* Obtenido de https://www.researchgate.net/publication/242081538_Method_of_k-Nearest_Neighbors

Kuhn, M. (Abril de 2019). *GitHub.* Obtenido de https://cran.r-project.org/web/packages/caret/caret.pdf

Molnar, C. (2019). *Interpretable Machine Learning.* Lulu.

Pedrosa, J., & Murata, N. (Febrero de 2000). *ResearhGate.* Obtenido de https://www.researchgate.net/publication/3857587_Optimization_on_Support_Vector_Machines

Roldán Salgueiro, J. L., Cepeda Carrión, G., & Galán González, J. L. (Enero de 2012). Los sistemas de inteligencia de negocio como soporte a los procesos de toma de decisiones en las organizaciones. *Los sistemas de inteligencia de negocio como soporte a los procesos de toma de decisiones en las organizaciones*. Sevilla, España: Papeles de Economía Española.

Valdivieso Flores, D. (Noviembre de 2011). *Repositorio Institucional de la Universidad Politécnica Salesiana.* Obtenido de https://dspace.ups.edu.ec/bitstream/123456789/2020/1/UPS-QT00095.pdf

Visa, S., Ramsay, B., Ralescu, A., & van der Knaap, E. (Abril de 2011). *CEUR Workshop Proceedings.* Obtenido de http://ceur-ws.org/Vol-710/paper37.pdf