

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL



FACULTAD DE CIENCIAS NATURALES Y MATEMÁTICAS

DEPARTAMENTO DE MATEMÁTICAS

**PROYECTO DE GRADUACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE
“MAGISTER EN CONTROL DE OPERACIONES Y GESTIÓN LOGÍSTICA”**

TEMA

**DISEÑO E IMPLEMENTACIÓN DE UN MODELO DE CLASIFICACIÓN PARA LA
DETECCIÓN DE TRANSACCIONES INUSUALES EN LA CONDUCTA HABITUAL
DE UN TARJETAHABIENTE EN UNA ENTIDAD DEL SISTEMA FINANCIERO**

AUTOR

ING. DIEGO ARMANDO ROMERO CARRIÓN

DIRECTOR DE TESIS

CARLOS ANÍBAL SUÁREZ HERNÁNDEZ

GUAYAQUIL – ECUADOR

2016

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL



FACULTAD DE CIENCIAS NATURALES Y MATEMÁTICAS

DEPARTAMENTO DE MATEMÁTICAS

**PROYECTO DE GRADUACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE
“MAGISTER EN CONTROL DE OPERACIONES Y GESTIÓN LOGÍSTICA”**

TEMA

**DISEÑO E IMPLEMENTACIÓN DE UN MODELO DE CLASIFICACIÓN PARA LA
DETECCIÓN DE TRANSACCIONES INUSUALES EN LA CONDUCTA HABITUAL
DE UN TARJETAHABIENTE EN UNA ENTIDAD DEL SISTEMA FINANCIERO**

AUTOR

ING. DIEGO ARMANDO ROMERO CARRIÓN

DIRECTOR DE TESIS

CARLOS ANÍBAL SUÁREZ HERNÁNDEZ

GUAYAQUIL – ECUADOR

2016

DEDICATORIA

A mis padres, hermanas y seres especiales.

Diego R.

AGRADECIMIENTO

A Dios por haber permitido terminar este gran reto.

Un agradecimiento sincero a todos quienes me han dado un apoyo incondicional,
deseándoles lo mejor y reiterándoles toda mi retribución desde ya.

Hago parte de este agradecimiento a Aníbal S., Elkin A., César L., Andrea M.
Anita, Diana, Giglia T., José, Nicole, Jorge, Laurita y Pepe N.

Diego R.

DECLARACIÓN EXPRESA

La responsabilidad por los hechos y doctrinas expuestas en este Proyecto de Graduación, me corresponde exclusivamente; el patrimonio intelectual del mismo, corresponde exclusivamente a la **Facultad de Ciencias Naturales y Matemáticas, Departamento de Matemáticas** de la Escuela Superior Politécnica del Litoral.

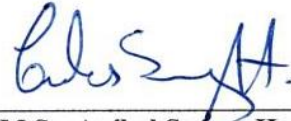


Ing. Diego Romero Carrión

TRIBUNAL DE GRADUACIÓN



Sandra García, Ph.D.
PRESIDENTE DEL TRIBUNAL



M.Sc. Aníbal Suárez Hernández
DIRECTOR DE PROYECTO



M.Sc. Elkin Angulo Ramírez
VOCAL DEL TRIBUNAL

**FIRMA DEL AUTOR DEL PROYECTO DE
GRADUACIÓN**



Ing. Diego Romero Carrión

Índice de contenido

DEDICATORIA	II
AGRADECIMIENTO	III
DECLARACIÓN EXPRESA	IV
TRIBUNAL DE GRADUACIÓN.....	V
FIRMA DEL AUTOR DEL PROYECTO DE GRADUACIÓN	VI
ÍNDICE DE CONTENIDO.....	VII
ÍNDICE DE ILUSTRACIONES	IX
ÍNDICE DE TABLAS	X
CAPÍTULO I	1
EL PROBLEMA.....	1
1.1. INTRODUCCIÓN	1
1.2. JUSTIFICACIÓN DEL PROBLEMA	2
1.3. OBJETIVOS GENERALES Y ESPECÍFICOS	4
1.3.1. Objetivo General.....	4
1.3.2. Objetivos Específicos	5
1.4. HIPÓTESIS	5
CAPÍTULO II.....	1
MARCO TEÓRICO.....	1
2.1. EL FRAUDE CON TARJETAS DE CRÉDITO	1
2.1.1. Formas de realizar transacciones fraudulentas con tarjeta de crédito	1
2.2. INTELIGENCIA ARTIFICIAL Y REDES NEURONALES	3
2.2.1. Reseña Histórica	3
2.2.2. Red Neuronal	4
2.2.3. Características.....	4
2.2.4. Aprendizaje.....	11
2.2.5. Elección del conjunto inicial de pesos	12
2.2.6. Detención del proceso de aprendizaje.....	12
2.2.7. Principales Topologías de redes neuronales	13
2.3. SITUACIÓN ACTUAL.....	17
2.3.1. Institución objeto de estudio	17

2.3.2.	Tarjeta de Crédito	18
2.4.	ANÁLISIS EXPLORATORIO DE LA INFORMACIÓN	19
2.5.	MONTO DE PÉRDIDA POR FRAUDE.....	19
2.5.1.	Fraude detectado por canal	20
CAPÍTULO III		29
APLICACIÓN DE LA METODOLOGÍA		29
3.1.	METODOLOGÍA	29
3.2.	POBLACIÓN Y MUESTRA	29
3.3.	VARIABLES DEL ESTUDIO	30
3.3.1.	Selección de las Variables de entrada	33
3.3.2.	Exclusión de Variables	34
3.4.	CODIFICACIÓN DE LOS DATOS DE ENTRADA	34
3.5.	EXPLICACIÓN DEL MODELO	38
3.5.1.	Red Multicapa.....	39
3.5.2.	Algoritmo BackPropagation o Retropropagación.....	40
3.5.3.	Calculo de pesos sinápticos y polarización.....	41
3.5.4.	Cálculo de la sensibilidad	44
3.5.5.	Función de activación	47
3.6.	ENTRENAMIENTO DE LA RED NEURONAL	48
3.7.	EJECUCIÓN DEL MODELO.....	52
CAPÍTULO IV.....		63
PROPUESTA DE MEJORA		63
4.1.	MODELO DE CLASIFICACIÓN	63
4.1.1.	Pesos sinápticos del modelo	63
4.2.	RESULTADOS OBTENIDOS.....	65
4.3.	FUNCIONAMIENTO DE LA RED NEURONAL EN TIEMPO REAL.	68
CONCLUSIONES		69
RECOMENDACIONES		70
ANEXOS		71
ANEXO 1 CODIFICACIÓN Y TRANSFORMACIÓN DE LA VARIABLE PAÍS.....		71
ANEXO 2 CODIFICACIÓN Y TRANSFORMACIÓN (TX) DE MCC		74

Índice de Ilustraciones

Ilustración 1 Componentes de una red neuronal	6
Ilustración 2 Función Sigmoide Lineal	8
Ilustración 3 Función Sigmoide	9
Ilustración 4 Función Sigmoide Hiperbólica	10
Ilustración 5 Descripción grafica de una red	16
Ilustración 6 Plástico Tarjeta Internacional	18
Ilustración 7 Monto Fraude por marca de tarjeta 2014-2015.....	20
Ilustración 8 Fraude detectado por canal	21
Ilustración 9 Fraude detectado por canal Internet	22
Ilustración 10 Comparativo por marca 2014-2015	23
Ilustración 11 Fraude por marca total	24
Ilustración 12 Fraude encontrado por tipo de entrada.....	25
Ilustración 13 Fraude encontrado por tipo de tarjeta	26
Ilustración 14 Fraude Top Países	27
Ilustración 15 Tabla 9 Fraude top Mcc	28
Ilustración 16 Transformación Lineal	36
Ilustración 17 Función de Activación	38
Ilustración 18 Representación Red Neuronal.....	39
Ilustración 19 Representación Red Multicapa	40
Ilustración 20 Comportamiento del error - Entrenamiento 1	53
Ilustración 21 Comportamiento del error - Entrenamiento 2.....	54
Ilustración 22 Comportamiento del error - Entrenamiento 3.....	55
Ilustración 23 Comportamiento del error - Entrenamiento 4.....	56
Ilustración 24 Comportamiento del error - Entrenamiento 5.....	57
Ilustración 25 Comportamiento del error - Entrenamiento 6.....	58
Ilustración 26 Comportamiento del error - Entrenamiento 7.....	59
Ilustración 27 Comportamiento del error - Entrenamiento 8.....	60
Ilustración 28 Comportamiento del error - Entrenamiento 9.....	61
Ilustración 29 Comportamiento del error - Entrenamiento 10.....	62

Índice de Tablas

Tabla 1 Monto de Fraude por marca de tarjeta 2014-2015.....	19
Tabla 2 Fraude detectado por canal POS	20
Tabla 3 Fraude detectado por canal internet	21
Tabla 4 Comparativo por marca 2014-2015	22
Tabla 5 Fraude por marca total	23
Tabla 6 Fraude encontrado por tipo de entrada.....	24
Tabla 7 Fraude encontrado por tipo de tarjeta	25
Tabla 8 Fraude Top Países	26
Tabla 9 Fraude top Mcc	27
Tabla 10 Código Actual Marca.....	30
Tabla 11 Código actual tipo de tarjeta	31
Tabla 12 Modos de entrada.....	31
Tabla 13 Codificación binaria Marca.....	35
Tabla 14 Codificación Modo de entrada.....	35
Tabla 15 Codificación tipo de tarjeta.....	35
Tabla 16 Codificación para entrada a red neuronal marca.....	37
Tabla 17 Codificación para entrada a red neuronal Modo de entrada	37
Tabla 18 Datos primer entrenamiento.....	52
Tabla 19 Datos segundo entrenamiento	54
Tabla 20 Datos tercer entrenamiento	55
Tabla 21 Datos cuarto entrenamiento	56
Tabla 22 Datos quinto entrenamiento	57
Tabla 23 Datos sexto entrenamiento.....	58
Tabla 24 Datos séptimo entrenamiento.....	59
Tabla 25 Datos octavo entrenamiento.....	60
Tabla 26 Datos noveno entrenamiento.....	61
Tabla 27 Datos décimo entrenamiento.....	62
Tabla 28 Resultados W1 parte 1	63
Tabla 29 Resultados W1 parte 2	64
Tabla 30 Resultados Pesos sinápticos W2	64

Tabla 31 Comparativo Sistema Actual / Red Neuronal (Alertas).....	66
Tabla 32 Comparativo Sistema Actual / Red Neuronal (Falsos Positivos)	67
Tabla 33 Transacción Sospechosa	68
Tabla 34 Transacción No Sospechosa	68

Capítulo I

El problema

1.1. Introducción

En los últimos años, la industria de las tarjetas de crédito ha ido creciendo de manera significativa, por ende, los niveles de seguridad exigibles por la Superintendencia de Bancos y las franquicias son cada vez más exigentes. En Ecuador desde el año 2010, la Superintendencia de Bancos resolvió que todas las tarjetas de crédito deberán implementar la tecnología chip con el fin de repotenciar la seguridad de la información de datos del cliente con la finalidad de evitar fraudes, sin embargo, el robo de información y el fraude no se queda atrás y también ha ido evolucionando; ya que hoy en día la tarjeta de crédito es utilizada en diferentes canales, las 24 horas del día y los 365 días del año. Por las franquicias que las representan, una tarjeta puede ser usada a nivel mundial, de manera que usurpadores están cada día buscando vulnerabilidades, como robo de identidad, phishing, entre otros, esperando el momento para cometer el denominado fraude.

En la actualidad, las empresas emisoras de tarjetas de crédito cuentan con cierto número de alertas de transacciones sospechosas, que son notificadas al cliente por medios electrónicos y otras por llamadas telefónicas, y en otros casos no son notificadas, también existe la situación en que los clientes reportan transacciones no realizadas; según sea el caso, estas transacciones son registradas por un recurso humano que luego de una investigación resuelve si ha existido o no un fraude, determinando el fraude potencial y fraude neto; esta actividad ocasiona una serie de costos operativos y a su vez causa en el mayor de los casos molestias a los clientes, ya que en ocasiones las transacciones no son aprobadas por sospecha de fraudes cuando en la mayoría de casos si es el mismo cliente quien realiza la compra.

Para ello, las instituciones, deben estar preparadas para mantener el control de las operaciones de manera eficaz, logrando resultados al menor costo, por ello las empresas emisoras de tarjetas de crédito han utilizado la tecnología para monitorear a los clientes, y a su vez requieren ingenierías que permitan a la tecnología operar al menor costo y con mayor resultado.

En la actualidad existen varios productos que contribuyen a la prevención de fraudes, principalmente como Sentinel y Monitor plus, pero ambos necesitan el conocimiento de datos para poder parametrizar las herramientas. Por ello, este estudio plantea obtener este conocimiento, mediante el uso de las redes neuronales, de manera que permitan contribuir a la clasificación de transacciones como sospechosas o no, mediante el uso del perceptron multicapa que analizaremos en el desarrollo de este proyecto.

Dada la popularidad de este medio de transacción financiera, es necesario implementar herramientas ya sean sistemáticas o de cualquier otra índole que permitan verificar el uso fidedigno del cliente de la tarjeta y emitir una señal de alerta en el momento que las transacciones no se den acorde a lo que acostumbra registrar el sistema, de esta manera se estaría mitigando el fraude por transacciones ilícitas o de dudosa procedencia.

1.2. Justificación del problema

El constante crecimiento del mercado de tarjetas no es un hecho fortuito, a lo largo del tiempo, este medio de pago nos ha proporcionado numerosos beneficios ya sea por su conveniencia y/o seguridad de pago, su generalizada aceptación en todo tipo de comercios (incluso cibernéticos), su proceso de pago rápido y todo tipo de servicios de valor agregado (seguros, promociones, descuentos, entre otros).

En la actualidad los fraudes con tarjetas de crédito, a partir de robo o uso indebido de personas no propietarias de las mismas, están volviendo desconfiado al cliente hacia las entidades emisoras de tarjetas de crédito. A partir de esta realidad las emisoras de las tarjetas de crédito han visto la necesidad de tomar acciones para evitar que los usuarios sean perjudicados, ya sea comunicando por SMS el efecto de la transacción, integrar chips en las tarjetas, usar claves biométricas, doble factor de seguridad, llamada al cliente, entre otras.

¿Pero qué sucede cuando le roban la tarjeta al usuario, o aun peor, le es clonada? Pues las compras sin precedentes de los delincuentes y transacciones exageradas en pocas horas provocan que el cliente pierda todo su crédito.

De acuerdo a la política de la empresa objeto de este estudio, que es una institución que funciona bajo la ley del Sistema Financiero, indica que el usuario es informado en los estados de cuenta de la tarjeta luego de haber transaccionado en un ciclo transaccional (periodo de consumos), y muchas veces al cliente en un suceso de fraude, al cliente le toca pagar la deuda si ni siquiera haber adquirido el servicio o producto. Una vez cometido el delito, ya es muy tarde para reaccionar o para tomar alguna acción que concluya en la devolución de los valores utilizados por los defraudadores.

A pesar de que se pueden tomar muchas normas de seguridad para mitigar o disminuir el riesgo a fraudes y robos mediante tarjetas de crédito, el cliente no tiene una protección en tiempo real, una protección a partir de algún artilugio tecnológico o sistemático que advierta de transacciones sospechosas que el cliente por lo general no realiza o que jamás realiza. Es aquí donde podemos utilizar a la tecnología informática y a la programación de procesos para implementar una herramienta eficaz, que aunque no evite el fraude o robo, si puede mitigar de gran manera el robo o fraude por este medio.

La mayoría de los emisores de tarjeta en el país no cuentan con una herramienta que les permita constatar en tiempo real transacciones sospechosas realizadas con una tarjeta de crédito a partir del estándar normal del usuario de esta.

Por ello este proyecto pretende incrementar la eficiencia operativa mediante la automatización del estudio de un cliente a través de redes neuronales permitiendo conocer el comportamiento normal de un cliente y establecer un margen de probabilidad ante un fraude y en general de las transacciones no acorde a las que con normalidad realiza el titular de la tarjeta.

Además con este trabajo se contribuye con el cumplimiento de la normativa emitida por la Superintendencia de Bancos en el punto 4.3.8.7:

“Establecer procedimientos para monitorear, controlar y emitir alarmas en línea que informen oportunamente sobre el estado de los canales electrónicos, con el fin de identificar eventos inusuales, fraudulentos o corregir las fallas.”

1.3. Objetivos generales y específicos

1.3.1. Objetivo general

Diseñar e implementar un modelo de clasificación mediante aplicación de redes neuronales para la detección de transacciones sospechosas en la conducta habitual de un tarjetahabiente en una entidad del Sistema Financiero.

1.3.2. Objetivos específicos

- Identificar las principales variables que intervienen en la concentración del fraude.
- Definir la topología que permita el entrenamiento de una red neuronal con el método de retropropagación o backpropagation.
- Identificar los pesos sinápticos que hacen que el modelo funcione con un mínimo error cuadrático.
- Presentar el modelo de red neuronal con las variables y ponderaciones que permite la clasificación de transacciones.
- Conocer las principales reglas de negocio que permiten a otras herramientas incorporarlas en su motor de monitoreo.

1.4. Hipótesis

La aplicación de un modelo de clasificación mediante redes neuronales se permite determinar el entrenamiento para aprender de transacciones fraudulentas y poder categorizarlas como sospechosas.

- **Variable independiente:** Entrenamiento de redes neuronales.
- **Variable dependiente:** Clasificación de la transacción.

Capítulo II

Marco teórico

2.1. El fraude con tarjetas de crédito

Por lo general el robo es la forma más frecuente de fraudes realizados con una tarjeta de crédito, este puede darse de varias maneras, desde lo más básico como rebuscar en la basura datos y papeles hasta el “pirateo” de vanguardia tecnológica. Por ejemplo un ladrón podría encontrar facturas al revolver la basura y encontrar información valiosa para su red de fraude y hacer compras. Por otro lado podría darse un incidente donde un sitio web pueda ser vulneralizado y sea sustraída la información de sus clientes.

Pueden suceder cosas más complejas aún tal como un mesero que le tome una foto a la tarjeta de crédito de un cliente con el fin de usar esa información y comprar cosas e inclusive para crear otras cuentas. Así mismo, podría suscitarse que se reciban llamadas ofreciendo viajes, ofertas y regalos con el único fin de pedirle a un cliente ingenuo o desinformado, los datos de sus tarjetas de crédito para de esta manera el delincuente realizar compras y robar. (Comisión federal del Comercio, 2014).

2.1.1. Formas de realizar transacciones fraudulentas con tarjeta de crédito

2.1.1.1. Skimming o clonación de tarjetas

Es la clonación de una tarjeta de crédito a través de un mecanismo de banda magnética que copia la información que contiene dicha tarjeta. Entonces el delincuente puede utilizar esta tarjeta como si fuese el titular realizando transacciones de todo tipo, desde centro comerciales hasta cajeros automáticos.

Este tipo de fraude ha disminuido casi en su totalidad gracias a la utilización de un microchip que se ha integrado a las tarjetas de crédito y débito. (Ferro Veiga, 2010)

2.1.1.2. Robo de datos

Esto sucede cuando se roba la información de la tarjeta al llevar a cabo una transacción o a través de un recibo; dicha información la utilizan para realizar compras a distancia, ya sea por teléfono, internet u otro medio. (CNN Expansión, 2012)

2.1.1.3. Pérdida o robo de la tarjeta

Esto acontece cuando una persona ha encontrado una tarjeta, sin ser el usuario, realizando compras y transacciones fingiendo ser el titular. (CNN Expansión, 2012)

2.1.1.4. Robo de tarjeta antes de la entrega al titular

Sucede cuando una tarjeta es sustraída antes de llegar al lugar de residencia del titular, esto por lo general acontece cuando el emisor o entidad financiera envía la tarjeta por correo y no cuenta con soportes de recibo que permita establecer como dada la entrega al usuario real. (CNN Expansión, 2012)

Estas son algunas de las tantas modalidades del uso fraudulento de tarjetas de crédito y débito, ya que a medida que avanza la tecnología y las maneras de evitar el fraude los delincuentes se las ingenian para crear nuevas maneras de replicar o clonar la información de éstas e incluso de robarlas de manera casi imperceptible.

El fraude existe en muchas formas y cada vez más inteligentes, sin embargo la cuestión es poder mitigarlas, para el efecto existen muchas herramientas que pueden considerarse para la clasificación de transacciones inusuales tales como:

- Lógica Difusa
- Regresión Lineal
- Cadenas de Markovs
- Redes Neuronales

Para este proyecto se ha seleccionado el uso de redes neuronales para aprendizaje no supervisado de la información.

2.2. Inteligencia Artificial y Redes Neuronales

2.2.1. Reseña Histórica

Los primeros datos presentados de los modelos de redes neuronales artificiales se estiman del año 1943, desarrollado por los neurólogos Warrem McCulloch y Walter Pitts. Años después, en 1949, Donald Hebb desarrolló varias ideas sobre lo que era el aprendizaje neuronal, lo cual quedo patentado luego como la “regla de Hebb”. Ya para 1958, Rosenblatt puso en desarrollo el perceptrón simple, y en el año de 1960, Widrow y Hoff desarrollaron la primera aplicación en la industria con su proyecto ADALINE, lo cual supuso una revolución en este ámbito. (Grupo 5 - RAI - UC3M 2011/2012, 2012).

Con el pasar de las décadas las investigaciones fueron reduciéndose paulatinamente debido a la falta de modelos de aprendizaje y a las limitaciones sobre el perceptrón que introdujeron Minsky y Papert. Pero en los años 80 las investigaciones volvieron a resurgir gracias al desarrollo de la red de Hopfield, básicamente por el algoritmo de aprendizaje de retroprogramación (BackPropagation), idea llevada a cabo por Rumelhart y McClelland en el año de 1986. Para este entonces ya se desarrollaban ideas del perceptrón multicapas. (Grupo 5 - RAI - UC3M 2011/2012, 2012)

2.2.2. Red Neuronal

Una red neural artificial es un modelo matemático entrenable que permite encontrar relaciones funcionales entre dos conjuntos de datos. (Universidad Católica Andrés Bello, 2010).

El concepto de neurona artificial se deriva de los primeros modelos que los psicólogos presentaron sobre el funcionamiento de las neuronas biológicas en cuanto a dos aspectos básicos: la respuesta a estímulos y el aprendizaje. Un perceptrón, o neurona artificial, es la unidad de procesamiento fundamental de toda red neuronal. (Universidad Católica Andrés Bello, 2010)

Lo que hace una red neuronal es simular las redes neuronales biológicas para procesar datos de manera inteligente y con resultados perceptibles a la lógica.

2.2.3. Características

Las Redes Neuronales Artificiales, ANN (Artificial Neural Networks) están inspiradas en las redes neuronales biológicas del cerebro humano. Están constituidas por elementos que se comportan de forma similar a la neurona biológica en sus funciones más comunes. Estos elementos están organizados de una forma parecida a la que presenta el cerebro humano. (Basogain Olabe, 2011)

Las ANN al margen de "parecerse" al cerebro presentan una serie de características propias del cerebro. Por ejemplo las ANN aprenden de la experiencia, generalizan de ejemplos previos a ejemplos nuevos y abstraen las características principales de una serie de datos. (Basogain Olabe, 2011)

Aprender: adquirir el conocimiento de una cosa por medio del estudio, ejercicio o experiencia. Las ANN pueden cambiar su comportamiento en función del entorno. Se les muestra un conjunto de entradas y ellas mismas se ajustan para producir unas salidas consistentes. (Basogain Olabe, 2011)

Generalizar: extender o ampliar una cosa. Las ANN generalizan automáticamente debido a su propia estructura y naturaleza. Estas redes pueden ofrecer, dentro de un margen, respuestas correctas a entradas que presentan pequeñas variaciones debido a los efectos de ruido o distorsión. (Basogain Olabe, 2011)

Abstraer: aislar mentalmente o considerar por separado las cualidades de un objeto. Algunas ANN son capaces de abstraer la esencia de un conjunto de entradas que aparentemente no presentan aspectos comunes o relativos. (Basogain Olabe, 2011)

2.2.3.1. Elementos Básicos de una red neuronal

Una red neuronal está formada por neuronas interconectadas entre si y arregladas en tres capas (aunque esto último es variable). Los datos de la red entran por medio de la capa denominada “capa de entrada”, luego pasan a través de una “capa oculta” para salir finalmente por una “capa de salida”. Por lo general la capa oculta está constituida por varias capas. (Soltani Halvaiee & Kazem Arkabo, 2014)

Una neurona en sí misma tiende a ser muy pequeña para resolver problemas y cálculos, en todo caso el conjunto de ellas lograrán cumplir con el objetivo de resolver problemas complejos. (Soltani Halvaiee & Kazem Arkabo, 2014)

2.2.3.2. Función de entrada

La llamada función de entrada implica a la neurona que trata a varios valores de entrada como si estos fueran uno solo, por lo cual se le denomina como entrada global. El problema más complejo en este punto de entender la programación es el hecho de poder combinar estas simples entradas (in_{i1}, in_{i2}, \dots) dentro de la entrada global. Dicho objetivo es logrado a través de la función de entrada, aplicada a partir del vector entrada. Esta función puede describirse así:

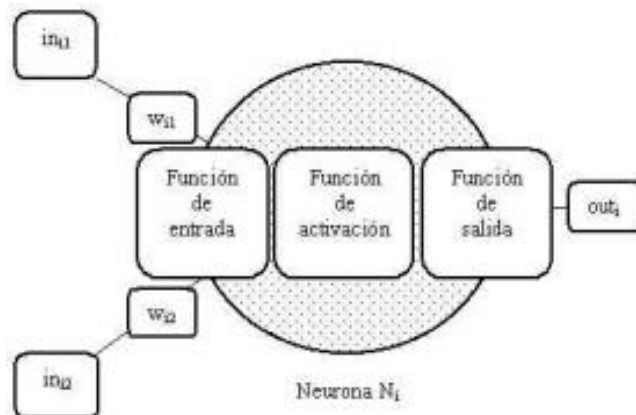
$$input_i = (in_{i1} * w_{i1}) * (in_{i2} * w_{i2}) * \dots * (in_{in} * w_{in})$$

De donde:

- * Es el operador apropiado (ejemplo de esto: sumatoria, máximo, etc.)
- n es el número de entradas que se tiene
- N_i y w_i al peso

Los valores existentes de entrada se multiplican por los pesos que anteriormente fueron ingresados a la neurona. Como los pesos no se encuentran restringidos cambian la medida de cómo influyen los valores de entrada. (Soltani Halvaiee & Kazem Arkabo, 2014)

Ilustración 1 Componentes de una red neuronal



Fuente: Neural Networks: Basics and Applications
Elaborado por: R. Lakes and D.Mack, 1998.

La nomenclatura utilizada en la Figura es la siguiente: $ini1$ = entrada número 1 a la neurona N_i ; w_{i1} = peso correspondiente a $ini1$; $ini2$ = entrada número 2 a la neurona N_i ; w_{i2} = peso correspondiente a $ini2$; y out_i = salida de la neurona N_i . El conjunto de todas las n entradas $ini = (ini1, ini2, \dots, inin)$ es comúnmente llamado “vector entrada”. (Matich, 2001)

Algunas de las funciones de entrada más comúnmente utilizadas y conocidas son:

- 1) Sumatoria de las entradas pesadas: es la suma de todos los valores de entrada a la neurona, multiplicados por sus correspondientes pesos.

$$\sum_j (n_{ij}w_{ij}), \quad \text{con } j = 1, 2, \dots, n$$

- 2) Producto de las entradas pesadas: es el producto de todos los valores de entrada a la neurona, multiplicados por sus correspondientes pesos.

$$\prod_j (n_{ij}w_{ij}), \quad \text{con } j = 1, 2, \dots, n$$

- 3) Máximo de las entradas pesadas: solamente toma en consideración el valor de entrada más fuerte, previamente multiplicado por su peso correspondiente.

$$\prod_j (n_{ij}w_{ij}), \quad \text{con } j = 1, 2, \dots, n$$

2.2.3.3. Función de activación

Las neuronas de la red neuronal tienen diferentes estados de activación, llegando a tomar cualquier valor dentro de un conjunto establecido. (Matich, 2001)

Esta función es la que calcula el estado de actividad de una neurona; lo hace al transformar la escala global (menos el umbral Θ_1) dentro de un valor de activación, donde el rango por lo general es de (0 a 1) o de (-1 a 1). Esto sucede ya que puede estar o totalmente activa o totalmente inactiva. (Matich, 2001)

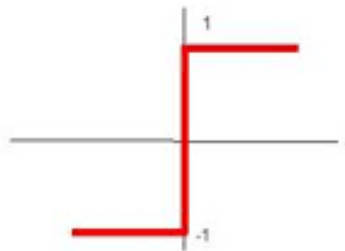
Una función de activación es una función de la entrada global (gin_i) restando el umbral (Θ_1). Las funciones habituales de activación son las siguientes:

Función lineal:

$$f(x) = \begin{cases} -1 & x \leq -1/a \\ a * x & -1/a < x < 1/a \\ 1 & x \geq 1/a \end{cases}$$

con $x = gin_i - \Theta_i$, y $a > 0$.

Ilustración 2 Función Sigmoide Lineal



Fuente: Neural Networks: Basics and Applications
Elaborado por: R. Lakes and D.Mack, 1998.

Los valores de salida obtenidos por medio de esta función de activación serán:

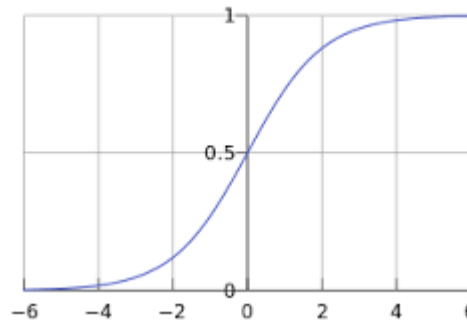
$a \cdot (gin_i - \Theta_i)$, cuando el argumento de $(gin_i - \Theta_i)$ esté comprendido dentro del rango $(-1/a, 1/a)$. (Matich, 2001)

Por encima o por debajo de esta zona se fija la salida en 1 o -1, respectivamente. Cuando $a = 1$ (siendo que la misma afecta la pendiente de la gráfica), la salida es igual a la entrada.

Función sigmoidea:

$$f(x) = \frac{1}{1 + e^{-gx}}, \text{ con } x = gin_i - \Theta_i.$$

Ilustración 3 Función Sigmoide



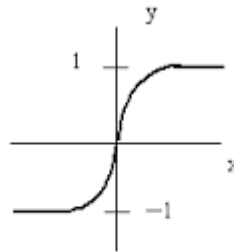
Fuente: Neural Networks: Basics and Applications
Elaborado por: R. Lakes and D.Mack, 1998

Los valores de salida que proporciona esta función están comprendidos dentro de un rango que va de 0 a 1. Al modificar el valor de g se ve afectada la pendiente de la función de activación.

Función tangente hiperbólica:

$$f(x) = \frac{e^{gx} - e^{-gx}}{e^{gx} + e^{-gx}}, \text{ con } x = gin_i - \Theta_i.$$

Ilustración 4 Función Sigmoide Hiperbólica



Función Tangencial Hiperbólica

Fuente: Neural Networks: Basics and Applications
Elaborado por: R. Lakes and D.Mack, 1998

Los valores de salida de la función tangente hiperbólica están comprendidos dentro de un rango que va de -1 a 1. Al modificar el valor de g se ve afectada la pendiente de la función de activación.

2.2.3.4. Función de salida

La función de salida es el último eslabón en el procesamiento de la información en una red neuronal. La salida de la neurona es el valor resultante de esta función i (*out*); es decir que la función de salida será la que determinará el valor a transferir para las neuronas vinculadas. Si la función de activación se encuentra por debajo de cierto umbral determinado, no habrá salida que pase la neurona que le sigue. No todo valor es permitido como una entrada para una neurona, por esto los valores de salida deben comprender valores cuyos rangos estén dentro de $[0, 1]$ o $[-1, 1]$. Igualmente pueden ser binarios $\{0, 1\}$ o $\{-1, 1\}$. (Matich, 2001)

Dos de las funciones de salida más comunes son:

- Ninguna: este es el tipo de función más sencillo, tal que la salida es la misma que la entrada. Es también llamada función identidad.
- Binaria:

$$\begin{cases} 1 & \text{si } act_i \geq \xi_i \\ 0 & \text{de lo contrario} \end{cases}, \text{ donde } \xi_i \text{ es el umbral.}$$

2.2.4. Aprendizaje

Como se ha establecido, los datos de entrada se procesan en la red neuronal con el fin de lograr una salida. Una red neuronal tiene la finalidad de aprender a calcular correctamente la salida para cada arreglo o vector de entrada, dicho proceso es denominado proceso de entrenamiento o acondicionamiento. Estos datos empleados para que la red neural aprenda se los llama conjunto de datos de entrenamiento. (Matich, 2001)

2.2.4.1. Aprendizaje Supervisado

Este tipo de aprendizaje es aquel donde se realiza un entrenamiento controlado por un agente externo (supervisor, maestro), este determinará la respuesta que una red tendría dependiendo de los datos de entrada. El supervisor será el que controle la salida de la red, si sucede que esta no tiene coincidencia con la deseada, se procederá a modificar los pesos de las conexiones, con la finalidad de que la salida obtenida se aproxime a la deseada. (Matich, 2001)

Se considera por lo general tres formas de ponerlo en marcha:

- Aprendizaje por corrección de error
- Aprendizaje por refuerzo
- Aprendizaje estocástico
-

2.2.4.2. Aprendizaje no supervisado

También conocido como auto supervisado, no necesitan de un supervisor para poder controlar el peso de las conexiones entre sus neuronas. La red no recibe información alguna por parte del entorno que le indique si la salida que se ha generado en respuesta a una determinada entrada está o no correcta.

Estas redes deben siempre incluir datos de entrada donde estas puedan encontrar las características, regularidades, correlaciones o categorías donde puedan operar sin supervisión.

Los algoritmos de aprendizaje no supervisados por lo general comprenden dos tipos que dan cabida a los siguientes tipos de aprendizaje:

- Aprendizaje Hebbiano
- Aprendizaje competitivo y comparativo

2.2.5. Elección del conjunto inicial de pesos

Con anticipación al proceso de entrenamiento se deberá determinar un estado inicial, es decir que se deberá escoger un punto inicial de peso para las diversas conexiones entre las neuronas de la red. Dicho objetivo se puede alcanzar bajo varios criterios; uno de ellos sería darle un peso aleatorio a cada conexión, ubicándose estos dentro de cierto intervalo. Por lo general un intervalo del tipo $[-n, n]$ donde n es un número natural positivo.

Es necesario aclarar que durante el entrenamiento los pesos no se encontrarán restringidos al intervalo referenciado.

2.2.6. Detención del proceso de aprendizaje

Es necesario determinar una condición de detención para establecer el momento en que el aprendizaje se debe detener.

Por lo general se detiene el entrenamiento cuando el cálculo del error cuadrado sobre todos los ejemplos de entrenamiento ha llegado a alcanzar un valor mínimo o también cuando en cada uno de los ejemplos de datos, el observador determina que está por debajo de un determinado umbral. Para controlar estos procesos, las redes neuronales en su mayoría lo que hacen es mostrar dichos errores a través de gráficos especiales, los cuales no son empleados por el aprendizaje si no que solo tienen la finalidad de indicar este suceso dentro del proceso en sí mismo.

Otra condición de detención del aprendizaje puede ser cuando un cierto número de ciclos y/o pasos de entrenamiento hayan sido completamente corridos.

Luego de alcanzarse la condición de detención, los pesos no se volverán a cambiar. Entonces podemos decir que la transformación de los datos de entrada a los de salida está resuelta. Esto se puede interpretar como una función f oculta en el conjunto de la red neuronal. Esta función es exactamente la “instrucción” de cómo la salida será calculada a partir de una constelación (vector) de entrada.

El orden en que los ejemplos de entrenamiento se presentan a la red neuronal es otro tema importante. En general se ha observado que en la mayoría de los casos es beneficioso realizarlo en forma aleatoria.

2.2.7. Principales Topologías de redes neuronales

La forma topológica de una red neuronal está determinada por la organización y disposición de las neuronas en ésta, las cuales forman capas o agrupaciones de neuronas por lo general alejadas de la entrada y la salida de ésta red. Según lo explicado, los parámetros fundamentales de la red son: número de capas, número de neuronas por capas, grado de conectividad y el tipo de conexiones entre las neuronas.

2.2.7.1. Red Monocapa

En este tipo de red se determinan conexiones entre las neuronas las cuales pertenecen a la capa única que contribuye la red. Estas redes se emplean por lo regular en tareas relacionadas con la autoasociación, es decir regenerar información de entrada que se obtiene de la red de forma incompleta o distorsionada.

2.2.7.2. Red Multicapa

Este tipo de redes son aquellas que tienen un conjunto de neuronas en varios (2, 3 etc.) niveles o capas. En estos casos una manera de distinguir la capa a la que pertenece una neurona es fijarse en el origen que tienen las señales que recibe a la entrada y el destino que ha tenido la señal. Todas las neuronas de una capa normalmente reciben señales de entrada desde una capa anterior, la cual por lo regular es la cercana a la entrada de la red.

Sin embargo, en un gran número de estas redes también existe la posibilidad de conectar la salida de las neuronas de capas posteriores a la entrada de capas anteriores; a estas conexiones se las denomina conexiones hacia atrás o feedback.

Estas dos posibilidades permiten distinguir entre dos tipos de redes con múltiples capas: las redes con conexiones hacia adelante o redes feedforward, y las redes que disponen de conexiones tanto hacia adelante como hacia atrás o redes feedforward/feedback.

2.2.7.3. Algoritmo Backpropagation

El nombre de backpropagation tiene su origen en la forma en el que el error es propagado hacia atrás a través de la red neuronal, es decir el error se propaga hacia atrás desde la capa de salida. Esto hace que los pesos sobre las conexiones de las neuronas situadas en las capas ocultas cambien durante el entrenamiento. (Matich, 2001)

Este cambio de los pesos en las conexiones de las neuronas a más de intervenir sobre la entrada global, interviene en la activación y por ende en la salida de una neurona. Consecuentemente, es de gran beneficio considerar las variaciones de la función activación al cambiar el valor de los pesos. A esto se lo llama sensibilidad de la función activación, conforme al cambio en los pesos.

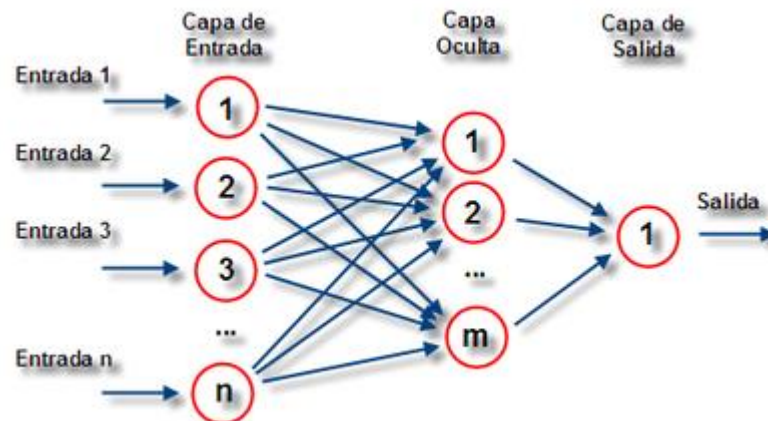
2.2.7.4. Constitución de una red neuronal

Una red neuronal está conformada por la interconexión de varios perceptrones. La estructura de dicha interconexión en sus distintos niveles de estructuración se denomina arquitectura de la red. (Lakes & Mack, 1998)

La forma en que se presenta el tipo de información provista en los datos y la naturaleza de la simulación determina la forma de aprendizaje que debe ser utilizada y a la vez la arquitectura más acorde a la realidad. (Universidad Católica Andrés Bello, 2010)

Independiente de la arquitectura de la red y de su aprendizaje, el perceptrón como unidad básica de procesamiento es básicamente la misma en cualquier tipo de red neural artificial. (Lakes & Mack, 1998)

Ilustración 5 Descripción grafica de una red



Fuente: Redes Neuronales: Conceptos Básicos y Aplicaciones
Elaborado por: Matich, Damián Jorge, 2011

En el gráfico se puede apreciar que una red neuronal artificial funciona de manera similar o análoga a una biológica, ya que esta recibe varias entradas o estímulos y producen una respuesta o salida. La neurona artificial análoga a la biológica se encarga de interceptar las soluciones generadas por las neuronas de la capa de entrada logrando así discriminar los datos que no corresponden y presentando como salida los que si deberían corresponder. (Universidad Católica Andrés Bello, 2010)

2.2.7.5. Aplicaciones

Las características especiales de los sistemas de computación neuronal permiten que sea utilizada esta nueva técnica de cálculo en una extensa variedad de aplicaciones.

La computación neuronal provee un acercamiento mayor al reconocimiento y percepción humana que los métodos tradicionales de cálculo. Las redes neuronales artificiales presentan resultados razonables en aplicaciones donde las entradas presentan ruido o las entradas están incompletas. Algunas de las áreas de aplicación de las ANN son las siguientes:

- ❖ Análisis y Procesado de señales
- ❖ Reconocimiento de Imágenes
- ❖ Control de Procesos
- ❖ Filtrado de ruido
- ❖ Robótica
- ❖ Procesado del Lenguaje
- ❖ Diagnósticos médicos

2.3. Situación Actual

2.3.1. Institución objeto de estudio

La institución, una entidad del sistema financiero ecuatoriano cuya parte de su negocio es la emisión y administración de tarjetas de crédito, se apega a una estrategia que consiste en el crecimiento de todos los segmentos de la economía, con productos y servicios que atiendan las necesidades de los clientes. El mandato de su administración es encontrar procesos más eficientes, a menores costos, y de impacto directo.

Actualmente esta entidad persigue agilizar las transacciones pero, al mismo tiempo, ofrecer soluciones crediticias y de servicio a sus clientes. Todo dentro de un esquema de eficiencia y rentabilidad.

En banca de personas, además de ofrecer servicios pioneros como Banca Telefónica e Internet Banking el Banco se ha destacado por revolucionar –desde 2004- el mercado de hipotecas. Esto lo ha logrado al disminuir las tasas de interés para créditos de vivienda, con productos como Hipoteca 10, 9 y 8. En el segmento empresarial, el banco ha desarrollado el producto Orden de Cobros y Pagos (OCP).

Este sistema, de cash management, permite a las empresas el manejo eficiente de sus cuentas por cobrar (con clientes) y de las cuentas por pagar (con proveedores,

empleados, transportistas, contratistas y accionistas), a través de información enviada al banco, vía Internet, sin el riesgo que supone el transporte de valores.

En el mediano plazo, el banco tiene planes para incursionar en nuevos segmentos, como el microcrédito, en sus rangos altos. Además busca aumentar y reforzar los canales de distribución, mediante la implementación de una nueva red de puntos de atención en establecimientos comerciales y una importante penetración en los segmentos de la población que aún no tienen acceso a la banca, a través de medios electrónicos.

2.3.2. Tarjeta de Crédito

Ilustración 6 Plástico Tarjeta Internacional



Fuente: PacifiCard – Banco del Pacífico

Elaborado: PacifiCard – Banco del Pacífico

La tarjeta de crédito es un medio de pago que permite a los clientes realizar consumos, diferirlos y administrar una línea referencial (cupó) que de manera responsable puede ser útil y necesaria para un cliente.

2.4. Análisis exploratorio de la información

Para iniciar el análisis exploratorio de la información se solicita a la entidad la información histórica reportada en el último año a la alta dirección, en la que se recopila información relevante del año 2014 y 2015 para el inicio del estudio.

2.5. Monto de Pérdida por fraude

Dentro del análisis operativo se obtiene la siguiente información respecto al año 2014 y 2015 en relación al fraude detectado y fraude potencial.

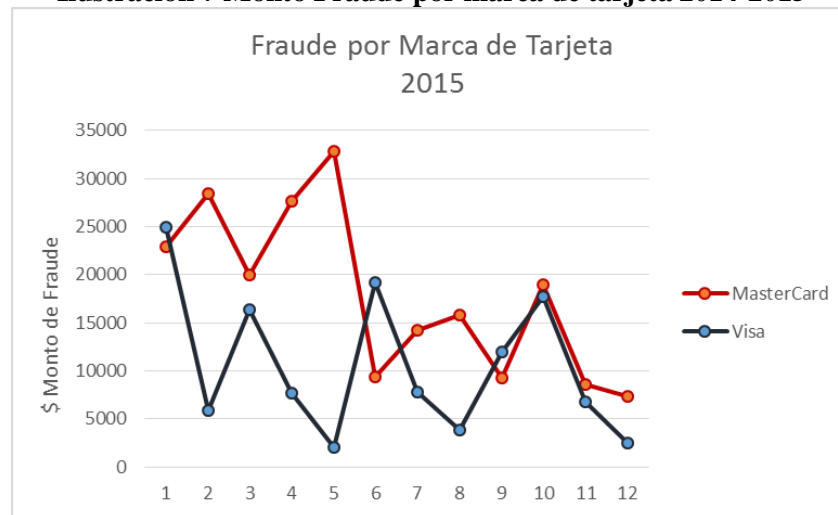
Tabla 1 Monto de Fraude por marca de tarjeta 2014-2015

Pérdida	MasterCard		Visa	
	2014	2015	2014	2015
ENE	67596,17	22932,25	20859,73	24946,14
FEB	41398,79	28467,9	32517,15	5922,21
MAR	65047,29	19936,65	18066,14	16383,04
ABR	55962,88	27681,39	9967,76	7656,19
MAY	33108,29	32798,21	3685,49	2052,36
JUN	32133,86	9408,37	10785,84	19159,08
JUL	31652,05	14229,75	11099,32	7848,64
AGO	32329,3	15855,82	14760,38	3886,33
SEP	34172,18	9323,41	24234,93	12003,01
OCT	35181,95	18935,74	18010,12	17711,12
NOV	38775,16	8617,49	9366,14	6835
DIC	42672,94	7347,54	28572,64	2512,42

Fuente: Tomado de entidad financiera, 2015
Elaborado por: Diego Romero

Se evidencia que el monto es alto en la marca MasterCard, y en el año 2015 disminuye el monto de fraude, por la aparición de la tecnología chip.

Ilustración 7 Monto Fraude por marca de tarjeta 2014-2015



Fuente: Tomado de Entidad financiera, 2015
Elaborado por: Diego Romero

2.5.1. Fraude detectado por canal

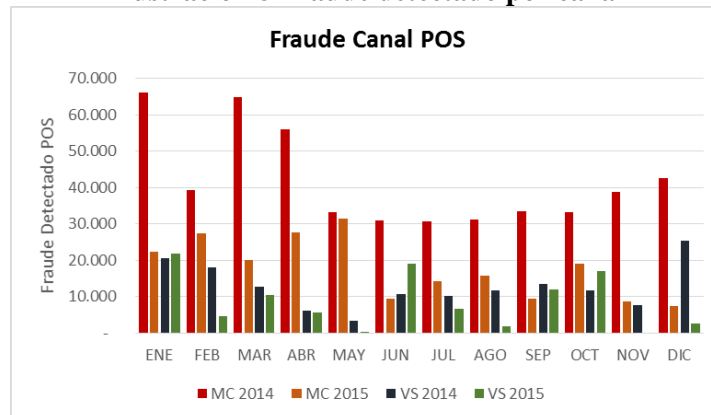
Tabla 2 Fraude detectado por canal POS

POS	MasterCard		Visa	
	2014	2015	2014	2015
ENE	66.189	22.434	20.459	21.951
FEB	39.298	27.399	18.069	4.672
MAR	64.837	19.937	12.665	10.550
ABR	55.963	27.681	6.190	5.731
MAY	33.108	31.538	3.472	137
JUN	30.980	9.408	10.726	19.159
JUL	30.704	14.230	10.210	6.648
AGO	31.130	15.856	11.791	1.760
SEP	33.544	9.323	13.507	11.870
OCT	33.292	18.936	11.724	17.110
NOV	38.775	8.617	7.759	-
DIC	42.673	7.348	25.469	2.512

Fuente: Tomado de entidad financiera, 2015
Elaborado por: Diego Romero

El fraude detectado en el canal POS (Point Of Sale – Punto de Venta) es importante en la marca MasterCard, cuyo monto asciende en el 2015 a 7348.

Ilustración 8 Fraude detectado por canal



Fuente: Tomado de entidad financiera, 2015

Elaborado por: Diego Romero

Tabla 3 Fraude detectado por canal internet

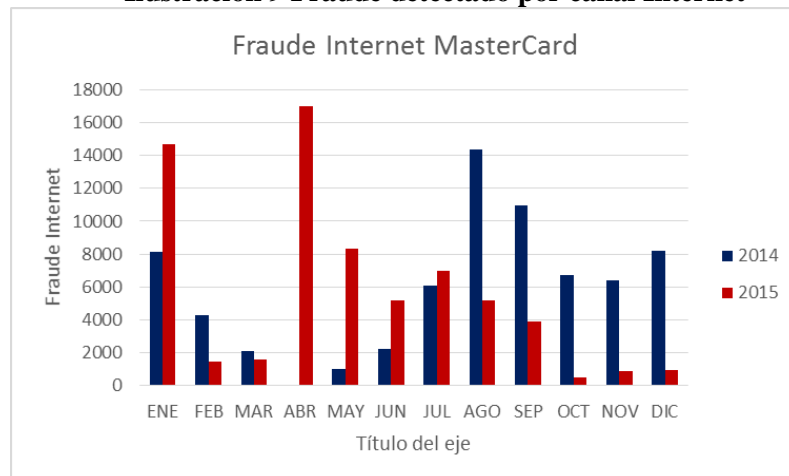
Internet	MasterCard		Visa	
	2014	2015	2014	2015
ENE	8122,3	14692,29	8122,3	14692,29
FEB	4299,86	1480,68	4299,86	1480,68
MAR	2097,8	1605,32	2097,8	1605,32
ABR	48,12	16974,92	48,12	16974,92
MAY	999,62	8338,51	999,62	8338,51
JUN	2250,41	5184,02	2250,41	5184,02
JUL	6092,04	6976,39	6092,04	6976,39
AGO	14361,15	5205,49	14361,15	5205,49
SEP	10958,66	3899,43	10958,66	3899,43
OCT	6749,03	520,61	6749,03	520,61
NOV	6386,42	882,7	6386,42	882,7
DIC	8194,16	951,8	8194,16	951,8

Fuente: Tomado de entidad financiera, 2015

Elaborado por: Diego Romero

El fraude detectado en el canal Internet es importante en la marca MasterCard, sin embargo, se disminuye por la aparición de chip.

Ilustración 9 Fraude detectado por canal Internet



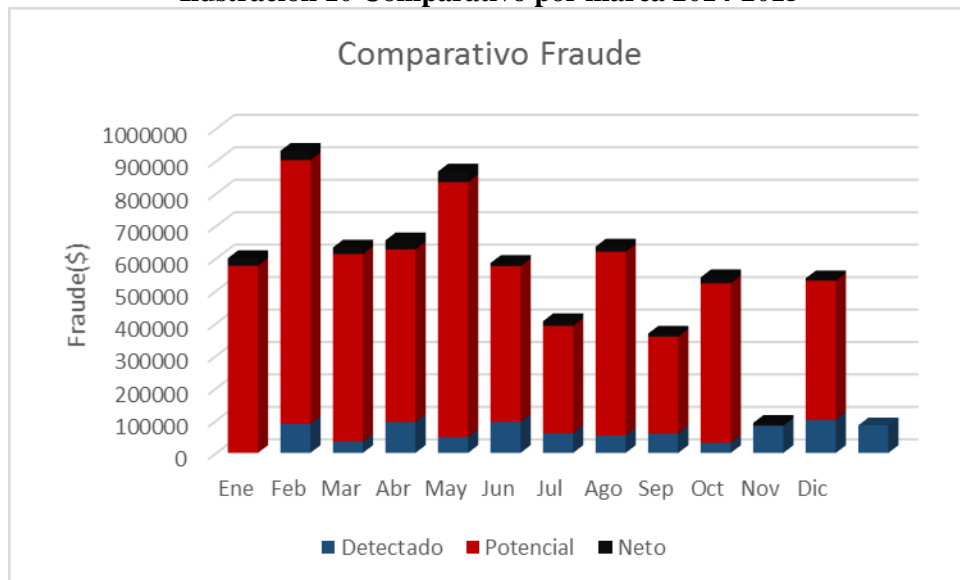
Fuente: Tomado de entidad financiera, 2015
Elaborado por: Diego Romero

Tabla 4 Comparativo por marca 2014-2015

Comparativo	MasterCard			Visa		
	Detectado	Potencial	Neto	Detectado	Potencial	Neto
Ene	88982,64	577389,13	22932,25	41722,89	295047,58	24946,1
Feb	35082,54	813659,99	28467,9	24147,63	249871,88	5922,21
Mar	95491,77	578285,04	19936,65	38523,03	340607,35	16383
Abr	49011,6	532051,04	27681,39	45786,83	176583,74	7656,19
May	95884,43	785842	32798,21	37559,54	222197,76	2052,36
Jun	61145,9	480267,53	9408,37	59870,09	225276,05	19159,1
Jul	53751,55	330757,36	14229,75	37857,7	184982,74	7848,64
Ago	60057,43	566710,2	15855,82	28275,51	189036,5	3886,33
Sep	31340,98	298376,99	9323,41	13430,26	141765,99	12003
Oct	84068,23	491256,55	18935,74	20172,74	208213,7	17711,1
Nov	102239	4	8617,49	33201,48	246265,67	0
Dic	85092,53	428760,81	7347,54	21695,75	154063,49	2512,42

Fuente: Tomado de entidad financiera, 2015
Elaborado por: Diego Romero

Ilustración 10 Comparativo por marca 2014-2015



Fuente: Tomado de entidad financiera, 2015

Elaborado por: Diego Romero

- **Fraude Potencial:** Es el Cupo Disponible del cliente para seguir transaccionando.
- **Fraude Detectado:** Es el monto de la transacción por el cual se detectó el fraude.
- **Fraude Neto:** Es la pérdida neta en relación al fraude detectado.

Tabla 5 Fraude por marca total

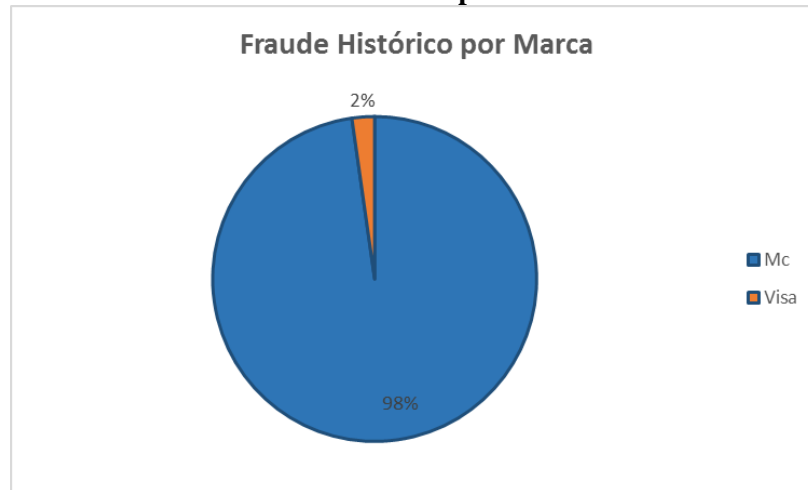
Marca	Porcentaje
Mc	98%
Visa	2%
Total general	100%

Fuente: Tomado de entidad financiera, 2015

Elaborado por: Diego Romero

Como se explica en la tabla 5, la concentración del fraude se encuentra en la marca MasterCard.

Ilustración 11 Fraude por marca total



Fuente: Tomado de entidad financiera, 2015
Elaborado por: Diego Romero

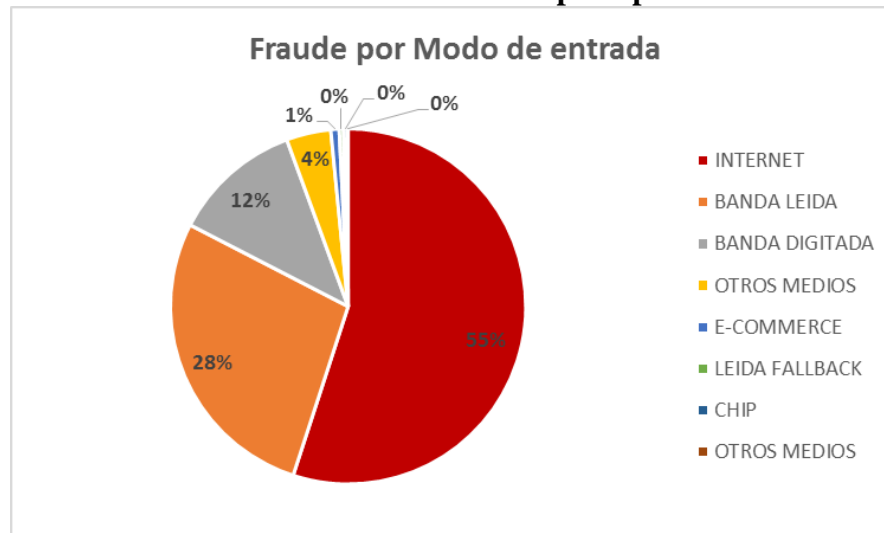
Tabla 6 Fraude encontrado por tipo de entrada

Modo de entrada	%
INTERNET	55,0%
BANDA LEIDA	27,6%
BANDA DIGITADA	11,8%
OTROS MEDIOS	4,1%
E-COMMERCE	0,8%
LEIDA FALLBACK	0,4%
CHIP	0,3%
OTROS MEDIOS	0,1%
Total general	100%

Fuente: Tomado de entidad financiera, 2015
Elaborado por: Diego Romero

La concentración del fraude se encuentra principalmente en el canal Internet, posteriormente en banda leída.

Ilustración 12 Fraude encontrado por tipo de entrada



Fuente: Tomado de entidad financiera, 2015
Elaborado por: Diego Romero

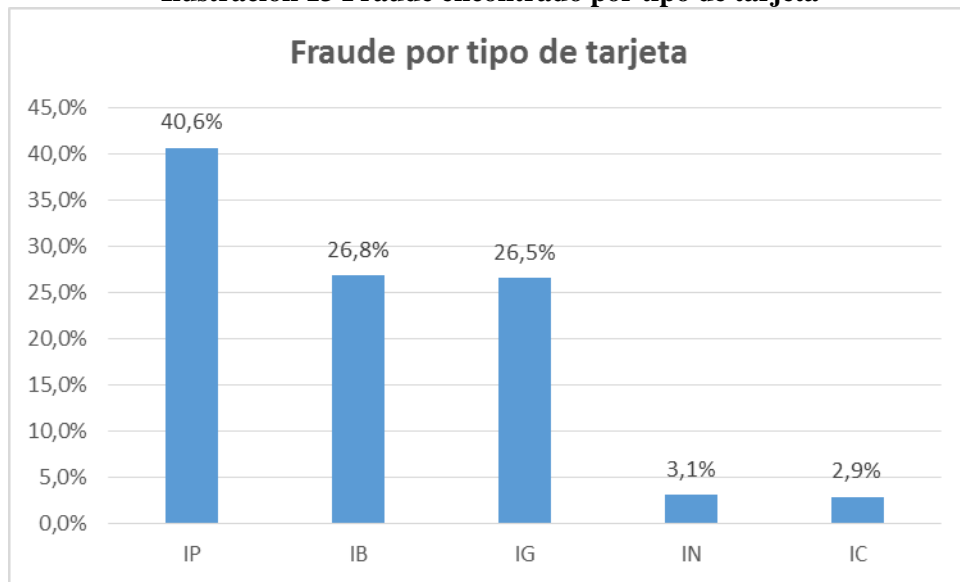
Tabla 7 Fraude encontrado por tipo de tarjeta

Tipo de tarjeta	%
IP	40,6%
IB	26,8%
IG	26,5%
IN	3,1%
IC	2,9%
Total general	100,0%

Fuente: Tomado de entidad financiera, 2015
Elaborado por: Diego Romero

La concentración de fraude por tipo de tarjeta, se encuentra en los clientes platinum (IP), por su característica principal de viajeros y cupo disponible altos.

Ilustración 13 Fraude encontrado por tipo de tarjeta



Fuente: Tomado de entidad financiera, 2015
Elaborado por: Diego Romero

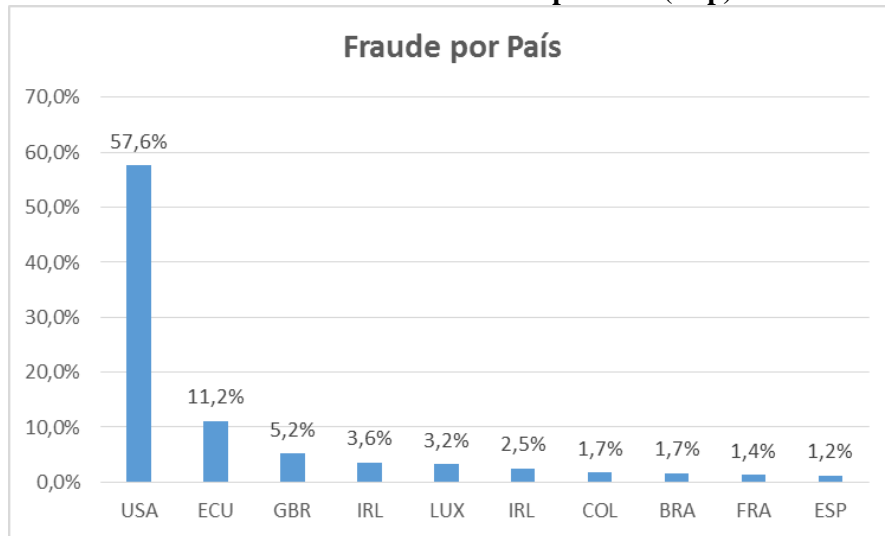
Tabla 8 Concentración de Fraude por Países

País	%
USA	57,6%
ECU	11,2%
GBR	5,2%
IRL	3,6%
LUX	3,2%
IRL	2,5%
COL	1,7%
BRA	1,7%
FRA	1,4%
ESP	1,2%

Fuente: Tomado de entidad financiera, 2015
Elaborado por: Diego Romero

La concentración de fraude por país, se encuentra en el país USA con un 57.6%, Ecuador 11.2%, Gran Bretaña 5.2%.

Ilustración 14 Fraude por País (Top)



Fuente: Tomado de entidad financiera, 2015
Elaborado por: Diego Romero

Tabla 9 Fraude top MCC

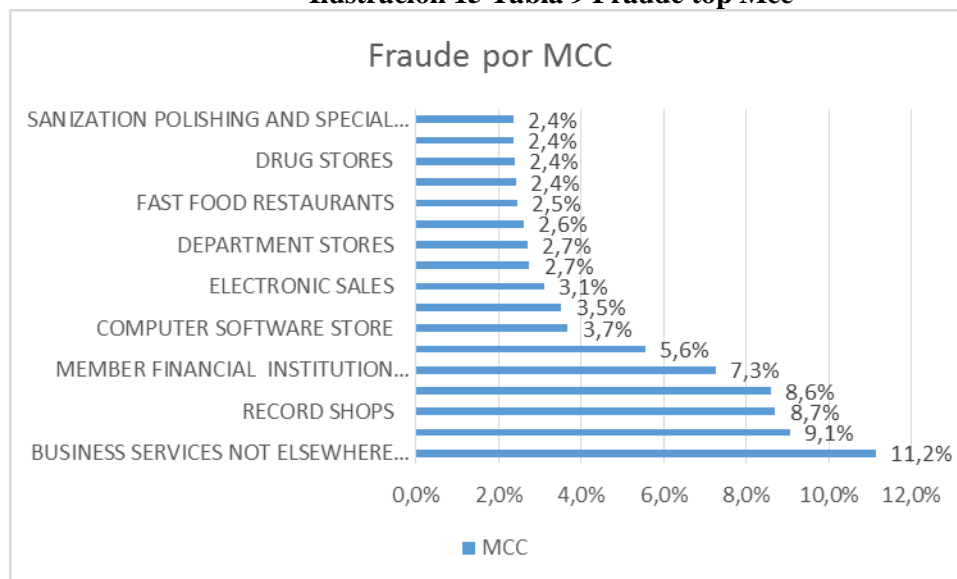
MCC	%
7399	11,2%
7311	9,1%
5735	8,7%
5411	8,6%
6011	7,3%
4722	5,6%
5734	3,7%
5999	3,5%
5732	3,1%
5968	2,7%
5311	2,7%
5691	2,6%
5814	2,5%
5964	2,4%
5912	2,4%
5541	2,4%
4816	2,4%

Fuente: Tomado de entidad financiera, 2015
Elaborado por: Diego Romero

MCC: Merchant Category Commerce, es una nomenclatura utilizada en medios de pago para identificar el giro de negocio del establecimiento (comercio) donde se realiza la compra.

Los datos reflejados en la tabla 9, representan el giro de negocio donde ocurre la mayor cantidad de fraudes por MCC.

Ilustración 15 Tabla 9 Fraude top Mcc



Fuente: Tomado de entidad financiera, 2015

Elaborado por: Diego Romero

Se puede apreciar en la ilustración que el MCC 7399 con mayor porcentaje de fraude es el de Business Services Not Elsewhere (Servicios No Comerciales), le sigue Record shops (Tiendas de Discos) con el 9,1%.

Capítulo III

Aplicación de la Metodología

3.1. Metodología

La metodología de este estudio comprende principalmente el análisis de la información actual, para proceder a la selección y descarte de las variables a utilizar; luego se diseñará el algoritmo de red neuronal para el entrenamiento del modelo, posteriormente se realizarán las pruebas del modelo matemático y se obtendrán sus resultados.

3.2. Población y Muestra

La población a tomar en este estudio corresponderá al histórico de datos correspondiente al fraude desde el año 2012 al 2015, en el que luego de una depuración de datos (bases de datos basura, con letras cuando deben ser número, campos con caracteres inválidos y datos aberrantes), se obtuvo un total de 10459 registros.

A partir de 10459 registros, se toma una muestra, mediante muestreo aleatorio simple, para el entrenamiento del perceptrón, que ha sido calculada de la siguiente manera:

$$n = \frac{Z^2 pqN}{NE + Z^2 pq}, \text{ donde:}$$

- N: tamaño de la muestra
- Z: Nivel de Confianza
- P: variabilidad positiva
- Q: Variabilidad negativa
- N: Tamaño de la población
- E: Precisión del error

Reemplazando los valores, se obtiene que n es:

$$n = \frac{(1.96)^2(0.5)(0.5)(10459)}{(10459)(0.05)^2 + (1.96)^2(0.5)(0.5)} = 400$$

- | | | |
|-----------------------------|---|-------|
| ▪ N: tamaño de la muestra | → | 400 |
| ▪ Z: Nivel de Confianza 95% | → | 1.96 |
| ▪ P: variabilidad positiva | → | 0.5 |
| ▪ Q: Variabilidad negativa | → | 0.5 |
| ▪ N:Tamaño de la población | → | 10459 |
| ▪ E: Precisión del error | → | 0.05 |

Se puede observar que se eligió 0.5 tanto para P, como para Q, dado que se eligen sobre un total conocido de las transacciones son fraudulentas, se toma el 50% para equilibrar la participación de los datos, lograr un aprendizaje más rápido y desviación más precisa.

3.3. Variables del estudio

Las variables con las que cuenta la base de datos son:

- **Nº tarjeta:** Es el número que se indica en el plástico de la tarjeta, cuya numeración actualmente tiene 16 dígitos.
- **Marca:** Corresponde si la tarjeta es de la franquicia MasterCard y Visa.

Tabla 10 Código Actual Marca

Marca	Descripción
mc	MasterCard
vs	Visa

Fuente: Tomado de entidad financiera, 2015

Elaborado por: Diego Romero

- **Tipo Tarjeta:** Representa el tipo de producto asociado a la marca, puede ser Internacional, Gold, Platinum, Black e Infinite.

▪

Tabla 11 Código actual tipo de tarjeta

Tipo Tarjeta	Descripción
IB	Black
IC	Cash
IG	Gold
IN	Inter
IP	Platinum

Fuente: Tomado de entidad financiera, 2015
Elaborado por: Diego Romero

- **Modo Entrada:** Corresponde a la forma de cómo ingresa la información, ya sea por POS, ATM, Internet, e-commerce.

Tabla 12 Modos de entrada

Modo de entrada
INTERNET
BANDA LEIDA
BANDA DIGITADA
OTROS MEDIOS
E-COMMERCE
LEIDA FALLBACK
CHIP

Fuente: Tomado de entidad financiera, 2015
Elaborado por: Diego Romero

- **País de transacción:** Está representado por la tabla países (Ver Anexo#1)
- **MCC:** Corresponde al giro de negocio del establecimiento donde se realiza la transacción. (Ver Anexo #2)
- **Valor:** Monto de la transacción

- **Día de Autorización:** Día de la semana en que se realizó la transacción.
- **Hora:** Hora en que se realizó la transacción.
- **F. Autorización:** Fecha en que se autorizó la transacción
- **Estado Aut:** Estado de la autorización
- **Respuesta:** Respuesta de la autorización, S=Si o N=No.
- **Razón Neg:** En caso de ser negada la transacción, se especifica la razón de negación.
- **Tipo Alerta:** Corresponde al tipo de alerta generado por la transacción.
- **# de Trxs. en 10 minutos mismo mcc mismo canal:** Muestra el número de veces que el cliente ha transaccionado en menos de 10 minutos.
- **Monto Acum. en trx. en 10 minutos:** Refleja el monto acumulado en transacciones del mismo cliente y mismo canal en menos de 10 minutos.
- **# de trxs. en 60 minutos:** Muestra el número de veces que el cliente ha transaccionado en menos de 60 minutos.
- **Monto Acum. en 60 minutos:** Refleja el monto acumulado en transacciones del mismo cliente y mismo canal en menos de 60 minutos.
- **Nivel:** Nivel del Fraude, se lo otorga el Ejecutivo de Riesgos.

- **F.Total:** Es el monto de la transacción por fraude.
- **F.Potencial:** Es el disponible que tenía el cliente al momento del fraude.
- **País del Fraude:** País en el que se encuentra el establecimiento donde se realizó el fraude.
- **Hora Bloqueo:** Hora en que se realizó un bloqueo de la tarjeta.

3.3.1. Selección de las Variables de entrada

De acuerdo a los resultados exploratorios donde se evidenció la concentración del fraude (capítulo dos), se puede obtener que las variables que más aportan para determinar si las características de una transacción son sospechosas o no, son las siguientes:

- Marca
- Tipo Trj.
- Modo Entrada
- País de transacción
- Mcc
- Valor
- # de trxs en 10 minutos mismo mcc mismo canal
- Monto Acum en trx en 10 minutos

Así mismo se puede indicar que estas variables son las que definen una transacción. Con este grupo de variables, se logra una ventaja, ya que si se seleccionan todas las variables, el proceso de aprendizaje puede tomar más tiempo y la red tendrá complejidad en hallar una solución general.

3.3.2. Exclusión de Variables

Las siguientes variables no conformarán parte del entrenamiento de la red neuronal, ya que se consideró que no son representativos para el modelo debido a que son muy comunes y actualmente son solo informativos.

- N° tarjeta
- Día de Autorización
- F. Autorización
- Estado Aut.
- Respuesta
- Razón Neg.
- Tipo Alerta
- # de trxs en 60 minutos
- Monto en la trx N
- Monto Acum en 60 minutos
- Nivel
- F.Total
- F.Potencial
- Pais del Fraude
- Hora Bloqueo

3.4. Codificación de los datos de entrada

Una vez seleccionada las variables, se deben codificar, para ello, se ha utilizado el siguiente proceso, para las variables que contengan dos respuestas se aplicará la codificación binaria, y para las variables con números reales y, cualitativos con mayor a dos valores de respuesta, se utilizará la transformación lineal, para convertir su valores en el rango 0 y 1.

Para la codificación se ha realizado el siguiente método:

1. Se asigna codificación binaria a las siguientes variables discretas o simbólicas:
 - Marca

Tabla 13 Codificación binaria Marca

Marca	Código
Mc	1
Visa	0

Fuente: Tomado de entidad financiera, 2015
Elaborado por: Diego Romero

- Modo de Entrada

Tabla 14 Codificación Modo de entrada

Modo de entrada	Código
BANDA DIGITADA	1
BANDA LEIDA	2
CHIP	3
E-COMMERCE	4
INTERNET	5
LEIDA FALLBACK	6
OTROS MEDIOS	7

Fuente: Tomado de entidad financiera, 2015
Elaborado por: Diego Romero

- Tipo de Tarjeta

Tabla 15 Codificación tipo de tarjeta

Tipo de tarjeta	Código
IN	1
IG	2
IP	3
IB	4
IC	5

Fuente: Tomado de entidad financiera, 2015
Elaborado por: Diego Romero

- Para las variables MCC y País, por la cantidad de datos, se muestra su codificación en el Anexo #2.
2. Una vez que se obtiene la codificación, se calcula la transformación lineal de los códigos para realizar el escalamiento a valores entre 0 y 1; que son los que debe recibir la red neuronal para el entrenamiento.

Ilustración 16 Transformación Lineal



Fuente: (Lakes & Mack, 1998)
Elaborado por: Diego Romero

Se parte de una función de transformación, a la que se denomina T:

T: datos originales (x) → datos codificados (x_{nuevos})

$$T(x_{\text{nuevos}}) = a \cdot x + b$$

a = pendiente

b = ordenada

Para este modelo, se explica la siguiente forma general de la transformación lineal:

$$a = \frac{\text{valor} - \text{mínimo en intervalo codificado}}{\text{máximo de datos originales} - \text{mínimo de datos originales}}$$

Tomando como ejemplo el tipo de tarjeta, aplicando la fórmula, se obtienen los siguientes resultados; para el caso del tipo IG, se obtiene que:

$$a = \frac{(2)-(1)}{(5)-(1)}=0.25, \text{ entonces IG es igual a } 0.25$$

Tabla 16 Codificación para entrada a red neuronal marca

Tipo de tarjeta	Código	T(x)
IN	1	0
IG	2	0,25
IP	3	0,5
IB	4	0,75
IC	5	1

Fuente: Tomado de entidad financiera, 2015

Elaborado por: Diego Romero

Entonces, aplicando la fórmula para las demás variables, se tiene también la transformación para el modo de entrada:

Tabla 17 Codificación para entrada a red neuronal Modo de entrada

Modo de entrada	Código	T(x)
BANDA DIGITADA	1	0,00
BANDA LEIDA	2	0,14
CHIP	3	0,29
E-COMMERCE	4	0,43
INTERNET	5	0,57
LEIDA FALLBACK	6	0,71
OTROS MEDIOS	7	0,86
OTROS MEDIOS	8	1,00

Fuente: Tomado de entidad financiera, 2015

Elaborado por: Diego Romero

Para el caso de las variables MCC y País, por la cantidad de datos, se adjunta su transformación lineal en el Anexo # 3.

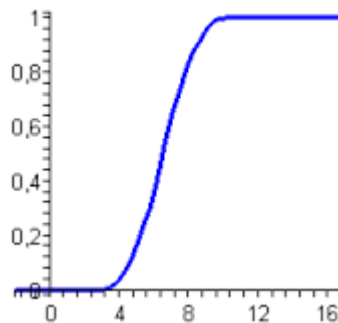
3.5. Explicación del Modelo

Para el diseño de la red neuronal se ha utilizado el modelo de perceptrón con topología multicapa, empleando a su vez el algoritmo de Retropropagación o también conocido como Backpropagation.

La red neuronal, necesita tener definida una función de activación, en la que se ha considerado para este análisis la función sigmoide.

$$f(x) = \frac{1}{(1+e^{-x})}, x \in [0, 1]$$

Ilustración 17 Función de Activación



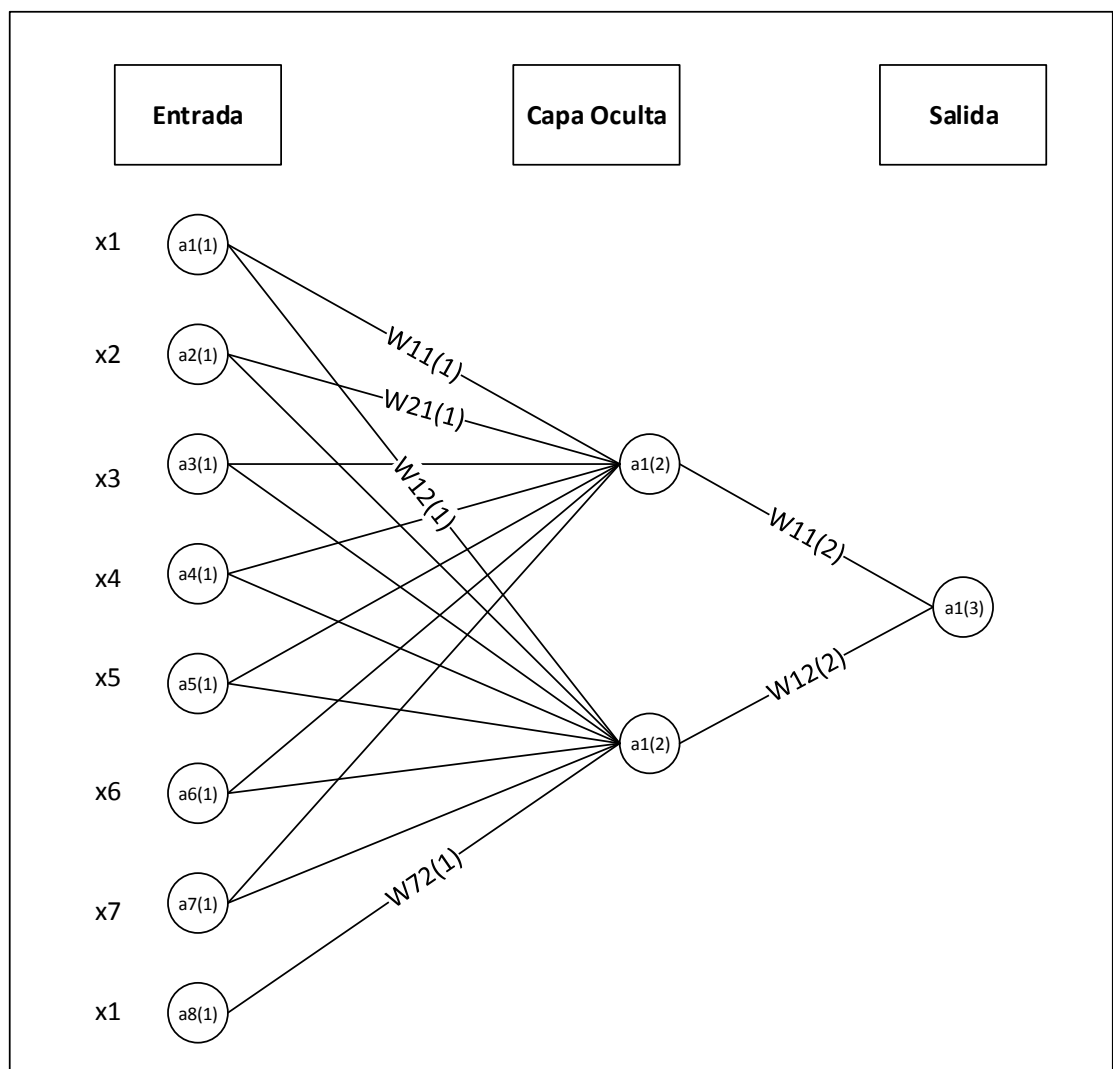
Fuente: (Lakes & Mack, 1998)

Elaborado por: Diego Romero

3.5.1. Red Multicapa

Se plantea gráficamente el primer escenario del perceptrón con el que se propone establecer una representación formal matemática de manera detallada para luego definir una notación general, en la siguiente figura se observa la red neuronal para definir las fórmulas.

Ilustración 18 Representación Red Neuronal

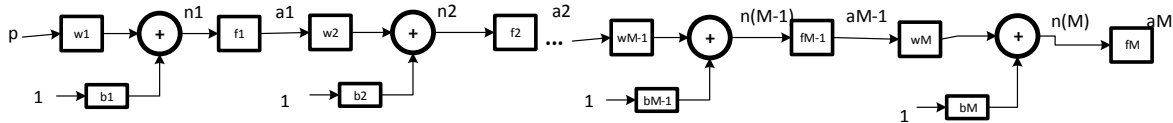


Fuente: Tomado de entidad financiera, 2015
Elaborado por: Diego Romero

3.5.2. Algoritmo BackPropagation o Retropropagación

El algoritmo de Backpropagation es una red neuronal multicapa, donde se utiliza 1 o varias capas en su ejecución hasta encontrar una solución óptima.

Ilustración 19 Representación Red Multicapa



Fuente: (Soltani Halvaiee & Kazem Arkabo, 2014)

Elaborado por: Diego Romero

- El número de capas está representado por la variable a_i ($i=1,2,\dots,M$), donde M es el número de capas.
- La entrada p está compuesta por el registro que contiene las variables(x) en la entrada de datos.
- Los pesos sinápticos de cada capa están representados por w_i ($1,2,\dots,M$).
- Las neuronas por cada capa están representadas por n_j ($j=1,2,\dots, M$).

Nota: Se puede representar a cada neurona con un superíndice k indicando el número de capa donde se encuentra, sería a^k .

La expresión de la neurona está representada de la siguiente manera:

$(P_1,t_1), (P_2,t_2), \dots, (P_q,t_q), \dots, (P_Q,t_Q)$; donde P es el vector de entrada y t es el vector de salida, y Q es el número de patrones de aprendizaje en la matriz de datos.

La expresión de salida de la multicapa, es la siguiente:

$$a^M = f^M (w^M f^{M-1} (w^{M-1} \dots f^2 (w^2 f^1 (w^1 p + b^1) + b^2) + \dots b^{M-1}) + b^M)$$

Y para denotar la expresión de manera compacta se utiliza la siguiente función recursiva:

$$a^m = f^m(w^m a^m - 1 + b^m) \quad m=1,2,3,\dots,M$$

$$a^m=p; \quad m=0$$

3.5.3. Cálculo de pesos sinápticos y polarización

Para calcular los pesos sinápticos y polarización de las capas, se entiende que X representa los pesos sinápticos y polarización de una capa.

$$P_q \rightarrow \text{Multicapa}(X_1, \dots, X_M) \rightarrow a_q^M$$

Calculamos luego el error:

$$e_q = t_q - a_q^M \rightarrow e_q^T e_q$$

El objetivo es minimizar el error cuadrático para el ejemplo q :

$$F(X_1, X_2, \dots, X_M) = e_q^T e_q, \text{ donde } X^m = [w^m \ b^m]^T$$

Vemos que tenemos varios errores cuadráticos por cada conjunto de X , por ello es más difícil utilizando un método tradicional, entonces el algoritmo de retropropagación utiliza el método del descenso por gradiente para hallar los pesos sinápticos que minimizan el error cuadrático de cada ejemplo, de la forma como sigue:

$$X^m = X^m - \alpha \frac{dF}{dX^m} \mid X^m = X_0^m$$

Tomando en cuenta que X contiene los pesos y polarizaciones $X^m = [w^m \ b^m]^T$ Extraemos la ecuación del descenso del gradiente transpuesto, donde el gradiente no es más que una matriz jacobiana y se representa el desarrollo de la siguiente manera:

$$(X^m)^T = (X^m)^T - \alpha \frac{dF^T}{dX^m}$$

$$\frac{\partial F}{\partial X^m} = \left[\frac{\partial F}{\partial X_1^m} \ \dots \ \frac{\partial F}{\partial X_{S^m}^m} \right]$$

La primera columna de la matriz representa las variaciones del ECM respecto a los pesos sinápticos y polarización de la neurona correspondiente.

Para calcular el gradiente respecto a la matriz X , hay que calcular cada uno de los gradientes para cada uno de los x_1, x_2, \dots, x_s ; y se lo puede descomponer mediante la regla de la cadena, de la siguiente manera:

$$\frac{\partial F}{\partial x^i} = \frac{\partial F}{\partial x_i} \frac{\partial F}{\partial n_i}$$

La entrada neta está dada por el producto de los dos vectores $x_i^T z$, por lo tanto se calcula la primera derivada de la entrada neta (n_i) respecto a $(X_i)^T$ y su resultado es z :

$$n_i = x_i^T z$$

La segunda derivada llevará un nombre particular llamado sensibilidad, es decir a cualquier derivada del error cuadrático respecto a cualquier entrada neta, en cualquier capa, tendrá este nombre “Sensibilidad” s_i .

$$\frac{\partial F}{\partial x^i} = \frac{\partial F}{\partial x_i} \frac{\partial F}{\partial n_i} = z s_i$$

Ahora sustituimos los gradientes en la matriz jacobiana, quedando así:

$$\frac{\partial F}{\partial X^m} = [zS_1 \quad zS_2 \quad \dots \quad zS_{S^m}]^m$$

Si extraemos el vector z, quedando un vector horizontal de sensibilidades de la capa m, y llegamos a la siguiente expresión:

$$z^m (s^m)^T$$

Ahora z es la entrada aumentada con un 1 en el último elemento, por lo que representando de manera matricial se obtiene:

$$z^m (s^m)^T = \begin{bmatrix} a^{m-1} \\ 1 \end{bmatrix} (s^m)^T = \begin{bmatrix} a^{m-1} (s^m)^T \\ (s^m)^T \end{bmatrix}$$

Para representación horizontal, transponemos el gradiente, quedando la siguiente expresión:

$$\frac{\partial F^T}{\partial X^m} = [s^m (a^{m-1})^T \quad s^m]$$

Ahora sustituimos el resultado en la expresión de descenso por gradiente:

$$(X^m)^T = (X^m)^T - \alpha \frac{dF^T}{dX^m}$$

$$[W^m \quad b^m] = [W^m \quad b^m] - \alpha [s^m (a^{m-1})^T \quad s^m]$$

Como se puede apreciar, existen dos ecuaciones, las mismas que se pueden desacoplar, una para los pesos sinápticos y otra para las polarizaciones y se obtiene:

$$W^m = W^{m-1} - \alpha s^m (a^{m-1})^T$$

$$b^m = b^{m-1} - \alpha s^m$$

Para $\forall m[1, 2, \dots, M]$

Las ecuaciones descritas actualizan los pesos sinápticos y polarización para todas las capas, desde 1 hasta m; el único problema hasta el momento es la inclusión de la variable sensibilidad, y no se conoce aún su igualdad. Se describe en 3.5.4.

3.5.4. Cálculo de la sensibilidad

Para calcular las sensibilidades, se requiere una función recursiva que permita calcular cada una de las sensibilidades de cada capa.

$$\frac{\partial F}{\partial n^{m-1}}$$

Se puede expresar la siguiente derivada utilizando la regla de la cadena en términos de la sensibilidad de la capa posterior a m-1.

$$\frac{\partial F}{\partial n^{m-1}} = \frac{\partial n^m}{\partial n^{m-1}} \frac{\partial F}{\partial n^m}$$

Podemos sustituir las sensibilidades por;

$$s^{m-1} = \frac{\partial n^m}{\partial n^{m-1}} s^m$$

Se puede observar que hay una función recursiva, ya que la sensibilidad de la capa anterior s^{m-1} se puede deducir de la capa posterior s^m , siempre y cuando también se conozca el resultado de la derivada $\frac{\partial n^m}{\partial n^{m-1}}$.

Esta función es la que le da al algoritmo el nombre de retropropagación, porque se va a propagar la sensibilidad de la última capa hasta la primera capa mediante la ecuación en cuestión.

Se procede a calcular la primera derivada de la entrada neta de la capa m respecto de la capa m-1. Las entradas netas de la capa m están representadas mediante la siguiente ecuación:

$$n^m = W^m f^{m-1}(n^{m-1}) + b^m$$

Se utiliza la regla de la cadena para su resolución:

$$\frac{\partial n^m}{\partial n^{m-1}} = \frac{\partial f^{m-1}}{\partial n^{m-1}} \frac{\partial n^m}{\partial n^{m-1}} = F^{.m-1} (n^{m-1})(W^m)^T$$

Se obtiene como resultado una función recursiva que permitirá conocer cada una de las sensibilidades:

$$s^{m-1} = F^{.m-1} (n^{m-1})(W^m)^T s^m$$

Para $\forall m[1,2, \dots, M]$

Para que la función recursiva se ejecute debe partir de la sensibilidad de la última capa, por lo tanto para calcular la su valor de tiene que:

$$s^m = \frac{\partial F}{\partial n^M}$$

La función F es el producto punto del vector error:

$$F = e_q^T e_q$$

El vector error está, dado por:

$$e_q = t_q - f^M(n^M)$$

Se evidencia que se identifican tres dependencias, donde el cálculo de la derivada de F depende del vector error, el vector error depende de la función de activación y la función de activación depende de la entrada neta; entonces se procede a utilizar la regla de la cadena:

$$s^m = \frac{\partial F}{\partial n^M} = \frac{\partial f^M}{\partial n^M} \frac{\partial e_q}{\partial n f^M} \frac{\partial F}{\partial e_q}$$

Se puede computar las últimas dos derivadas y se obtiene el siguiente resultado:

$$s^m = \frac{\partial F}{\partial n^M} = \frac{\partial f^M}{\partial n^M} \frac{\partial e_q}{\partial n f^M} \frac{\partial F}{\partial e_q} = \frac{\partial f^M}{\partial n^M} (-1)(2e_q)$$

Entonces:

$$s^m = \frac{\partial F}{\partial n^M} = \frac{\partial f^M}{\partial n^M} \frac{\partial e_q}{\partial n f^M} \frac{\partial F}{\partial e_q} = \frac{\partial f^M}{\partial n^M} (-1)(2e_q) = -2F.^M(n^M)e_q$$

$$s^m = \frac{\partial F}{\partial n^M} = \frac{\partial f^M}{\partial n^M} \frac{\partial e_q}{\partial n f^M} \frac{\partial F}{\partial e_q} = \frac{\partial f^M}{\partial n^M} (-1)(2e_q) = -2F.^M(n^M)e_q$$

$$s^m = -2F.^M(n^M)e_q$$

Ahora solo se debe calcular el valor de “F.” Es la derivada de la función de activación (Función vectorial) respecto al vector de entrada neta.

$$F.^m (n^m) = \frac{\partial f^m}{\partial n^m}$$

Esta función genera una matriz, que contiene los gradientes de f_1 respecto a la entrada n , así hasta la última neurona. Se toma en cuenta que la función de activación de la neurona 1, solo depende de la neurona 1 y así con cada una; por ello se genera una matriz diagonal, solo hay dependencias con la entrada neta que le corresponde, y se la denota como “diag”.

$$F.^m (n^m) = \frac{\partial f^m}{\partial n^m} \left[\frac{\partial f_1^m}{\partial n^m} \dots \frac{\partial f_{s^m}^m}{\partial n^m} \right] = \begin{bmatrix} \frac{\partial f_1^m}{\partial n_1^m} & 0 & \dots & 0 \\ 0 & \frac{\partial f_2^m}{\partial n_2^m} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \frac{\partial f_{s^m}^m}{\partial n_{s^m}^m} \end{bmatrix}$$

$$= \text{diag} \left(\frac{\partial f_1^m}{\partial n_1^m} \right)$$

3.5.5. Función de activación

Como se ha identificado se requiere calcular la derivada de la función de activación respecto a la entrada neta, por ello se debe utilizar una función continua, de las cuales existen tres, y para cada una de ellas se debe calcular la F.

Función Sigmoide - Logística

$$a_i = f_i(n_i) = \frac{1}{1 + e^{-n_i}} \rightarrow \frac{\partial f_i}{\partial n_i} = (1 - a_i)a_i$$

$$F.^m (n^m) = \text{diag}((1 - a_i^m)a_i^m)$$

Función Sigmoide Tangencial

$$a_i = f_i(n_i) = \frac{e^{n_i} - e^{-n_i}}{e^{n_i} + e^{-n_i}} \rightarrow \frac{\partial f_i}{\partial n_i} = (1 - a_i^2)$$

$$F.^m(n^m) = \text{diag}((1 - a_i^m)^2)$$

Función Sigmoide Lineal

$$a_i = f_i(n_i) = 1$$

3.6. Entrenamiento de la Red Neuronal

Una vez analizado el proceso de aprendizaje formal de la red neuronal, se procede a realizar el siguiente pseudocódigo que permite sintetizar las fórmulas matemáticas descritas en el numeral anterior, y así lograr el entrenamiento de la red neuronal.

Iniciar aleatoriamente W^m y b^m para todo m

Para época=1 a N_{epocas}

1. Propagar hacia adelante

$$a^m = f^m(w^m a^{m-1} + b^m)$$

2. Propagar hacia atrás

$$e_q = t_q - a^M(n^M)$$

$$s^m = -2F.^M(n^M)e_q$$

$$s^{m-1} = F.^{m-1}(n^{m-1})(W^m)^T s^m$$

3. Actualizar los pesos

$$W^m = W^m - \alpha s^m (a^{m-1})^T$$

$$b^m = b^m - \alpha s^m$$

Para el entrenamiento de la red Neuronal, se plantea el desarrollo de las formulas descritas en el numeral anterior, sin embargo a esta instancia se representa de manera computacional.

```
clc;
disp('Entrenamiento de los datos de conocimiento');
filename=('C:\Users\pc\Documents\DIEGO\Proyectos Universitarios\Tesis
Diego\entrenamiento.xlsx');
%Patron de entrada
A=xlsread('C:\Users\pc\Documents\DIEGO\Proyectos Universitarios\Tesis
Diego\entrenamiento.xlsx')
%fA,cA=size(A);%tamaño de la matriz de entrada
fprintf('La matriz de entradas tiene %d datos \n', fA);
fprintf('La matriz de entradas tiene %d variables \n\n', cA);
%Patron de salida
B=xlsread('C:\Users\pc\Documents\DIEGO\Proyectos Universitarios\Tesis
Diego\entrenamiento.xlsx',2);
[fB,cB]=size(B);%tamaño de la matriz de entrada
fprintf('La matriz de salida tiene %d datos \n\n\n', fB);
%Patrones de entrada
%plot(A)

fprintf('Iniciando la arquitectura de la red. Espere... \n')

%Calculando el numero de nodos ocultos
%sl=fix((fA+fB)/2);
%nodo de s
%s2=fB ;
```

```
%numero de neuronas en la capa oculta
n2=input('Digite el numerode neuronas capa 2');
ep=input('Ingrese ventana de valores iniciales')
%valores de iniciación
w1=ep*rand(cA,n2)
b1=ones(1,n2);
w2=ep*rand(n2,1);
b2=1;
alfa=input('Digite tolerancia de error entre 0,1 y 0,9');
epoc=input('Digite cantidad de épocas');
Q=length(A)
for epocas=1:epoc
    suma=0;
    for i=1:Q
        disp(i)
        %Propagacion entrada - salida
        aa=A(i,:)
        bb=w1
        cc=aa*bb
        cc=cc+b1
        a1=tansig(cc)
        w2
        dd=a1*w2
        b2=b2
        dd=dd+b2
        a2(i)=tansig(dd)
        disp('Retropropagacion de la sensibilidad')
        e=B(i)-a2(i)
        s2=-2*(1-a2(i)^2)*e
        a1=a1.^2
        ww2=w2
        %ww2=w2'
        ss2=s2
        multi=ww2*s2
```

```
ddd=diag(1-a1)
s1=ddd*multi
%s1=diag(1-(a1.^2))*w2'*s2(no activar)
disp('Actualización de pesos')
w2=w2
alfa=alfa
s2=s2
a1=a1
alfas2a1=alfa*s2*a1
w2=w2-(alfas2a1)'
%w2=w2-alfa*s2*a1 (no activar)
b2=b2
alfas2=alfa*s2
b2=b2-alfas2
w1=w1
alfa=alfa
s1A=alfa*s1*A(i,:)
s1A=s1A'

w1=w1-s1A
%w1=w1-alfa*s1*A(i,:); (no activar)
b1=b1-(alfa*s1)'
suma=suma+e^2;
end
%error
emedio(epocas)=suma/Q
end
figure, subplot(1,2,1), plot(emedio)

%Verificación de la respuesta
for i=1:Q
aa=A(i,:);
bb=w1;
cc=aa*bb;
```

```
cc=cc+b1;
a1=tansig(cc);
a1w2=a1*w2;
factor=a1w2+b2;
a(i)=tansig(factor);
end
disp('verificación')
disp(a)
```

3.7. Ejecución del Modelo

En la ejecución se tomará en cuenta el número de neuronas, valores iniciales, valor del error estimado.

Este modelo está diseñado con tres capas, y se irán anotando las corridas de acuerdo a los patrones ingresados.

Primer Entrenamiento

Tabla 18 Datos primer entrenamiento

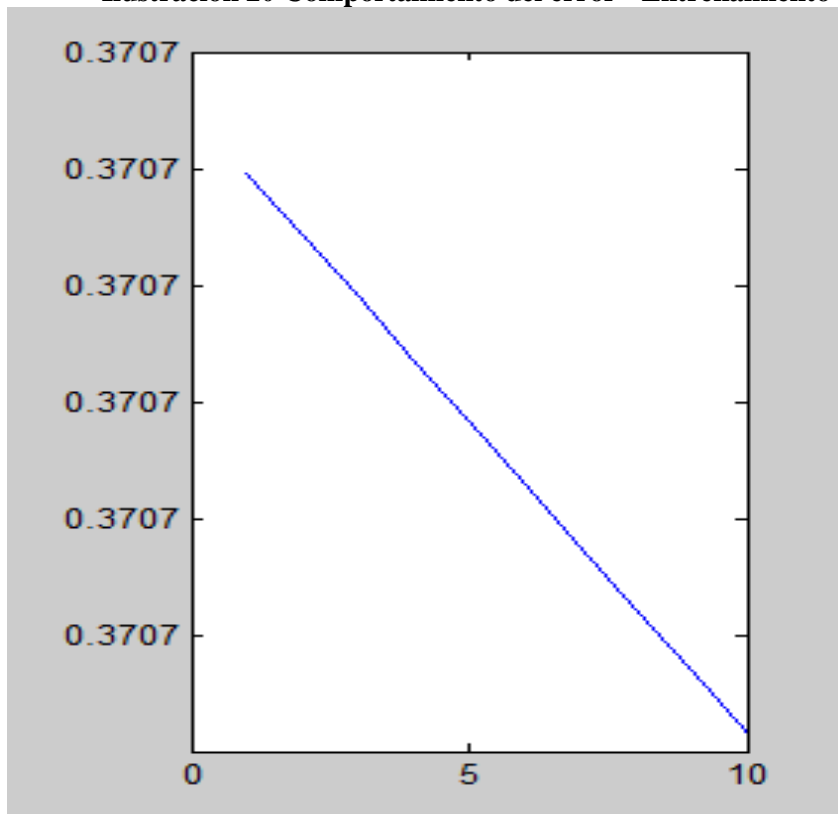
Capas	Neuronas	Valor inicial	Error Permitido	#Épocas
3	5	1	0.20	10

Fuente: Entrenamiento Red Neuronal, 2016

Elaborado por: Diego Romero

Como resultado de la ejecución, podemos evidenciar que la red con 5 neuronas, y 10 épocas consideradas para su entrenamiento tiene un error cuadrático medio de 0.3707, el tiempo de ejecución es 40 segundos.

Ilustración 20 Comportamiento del error - Entrenamiento 1



Fuente: Entrenamiento Red Neuronal, 2016
Elaborado por: Diego Romero

Se puede observar en la gráfica que el error desciende pero hasta en un mínimo de 0.3707, lo que no se aproxima al error esperado de 0.20.

Segundo entrenamiento

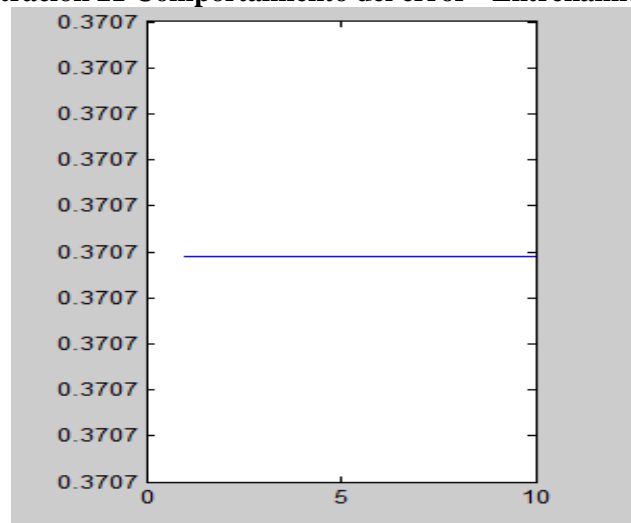
Tabla 19 Datos segundo entrenamiento

Capas	Neuronas	Valor inicial	Error Permitido	#Épocas
3	10	1	0.20	10

Fuente: Entrenamiento Red Neuronal, 2016
Elaborado por: Diego Romero

Como resultado de la ejecución, podemos evidenciar que la red con 10 neuronas, y 10 épocas consideradas para su entrenamiento tiene un error cuadrático medio de 0.3707, el tiempo de ejecución es 63 segundos.

Ilustración 21 Comportamiento del error - Entrenamiento 2



Fuente: Entrenamiento Red Neuronal, 2016
Elaborado por: Diego Romero

Se puede observar en la gráfica que el error es constante en 0.3707, lo que no se aproxima al error esperado de 0.20.

Tercer Entrenamiento

Tabla 20 Datos tercer entrenamiento

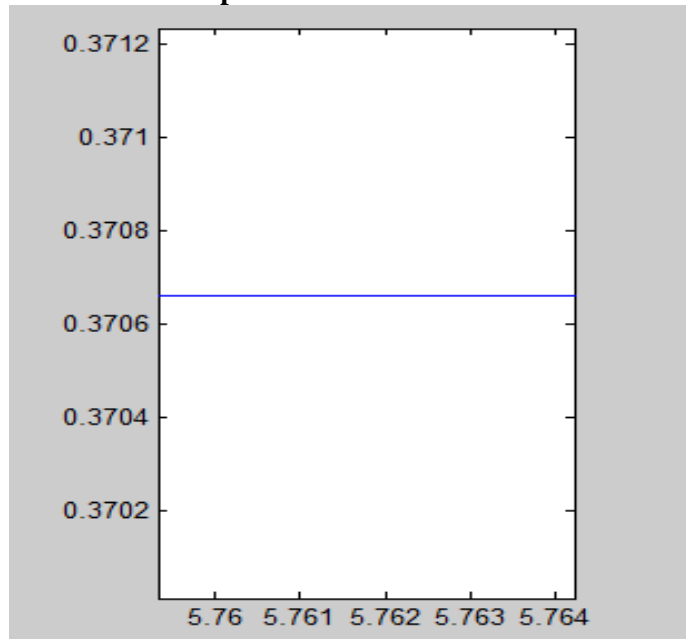
Capas	Neuronas	Valor inicial	Error Permitido	#Epocas
3	20	1	0.20	10

Fuente: Entrenamiento Red Neuronal, 2016

Elaborado por: Diego Romero

Como resultado de la ejecución, podemos evidenciar que la red con 20 neuronas, y 10 épocas consideradas para su entrenamiento tiene un error cuadrático medio de 0.3707, el tiempo de ejecución es 65 segundos.

Ilustración 22 Comportamiento del error - Entrenamiento 3



Fuente: Entrenamiento Red Neuronal, 2016

Elaborado por: Diego Romero

Se puede observar en la gráfica que el error continúa de forma constante, en 0.3707, lo que se evidencia claramente que no es una solución óptima.

Cuarto Entrenamiento

Tabla 21 Datos cuarto entrenamiento

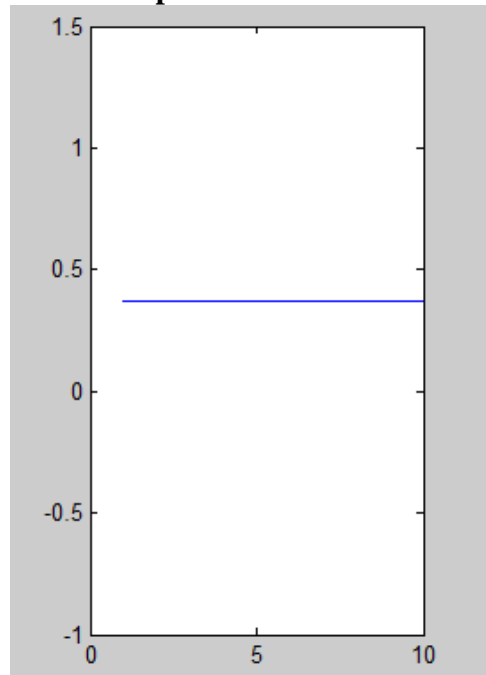
Capas	Neuronas	Valor inicial	Error Permitido	#Epocas
3	30	1	0.20	10

Fuente: Entrenamiento Red Neuronal, 2016

Elaborado por: Diego Romero

Como resultado de la ejecución, podemos evidenciar que la red con 30 neuronas, y 10 épocas consideradas para su entrenamiento tiene un error cuadrático medio de 0.3707, el tiempo de ejecución es 89 segundos.

Ilustración 23 Comportamiento del error - Entrenamiento 4



Fuente: Entrenamiento Red Neuronal, 2016

Elaborado por: Diego Romero

Se puede observar en la gráfica sigue siendo constante en 0.3707, lo que se evidencia claramente que no es una solución óptima.

Quinto entrenamiento

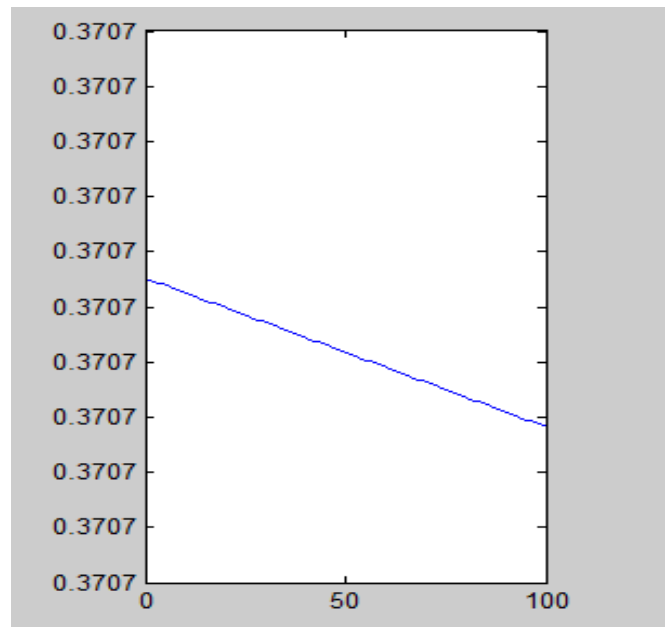
Tabla 22 Datos quinto entrenamiento

Capas	Neuronas	Valor inicial	Error Permitido	#Épocas
3	10	1	0.20	100

Fuente: Entrenamiento Red Neuronal, 2016
Elaborado por: Diego Romero

Como resultado de la ejecución, podemos evidenciar que la red con 10 neuronas, y 100 épocas consideradas para su entrenamiento tiene un error cuadrático medio de 0.3707, el tiempo de ejecución es 12 minutos.

Ilustración 24 Comportamiento del error - Entrenamiento 5



Fuente: Entrenamiento Red Neuronal, 2016
Elaborado por: Diego Romero

Se puede observar en la gráfica que el error desciende pero se mantiene finalmente en 0.3707, lo que se evidencia claramente que no es una solución óptima.

Sexto Entrenamiento

Tabla 23 Datos sexto entrenamiento

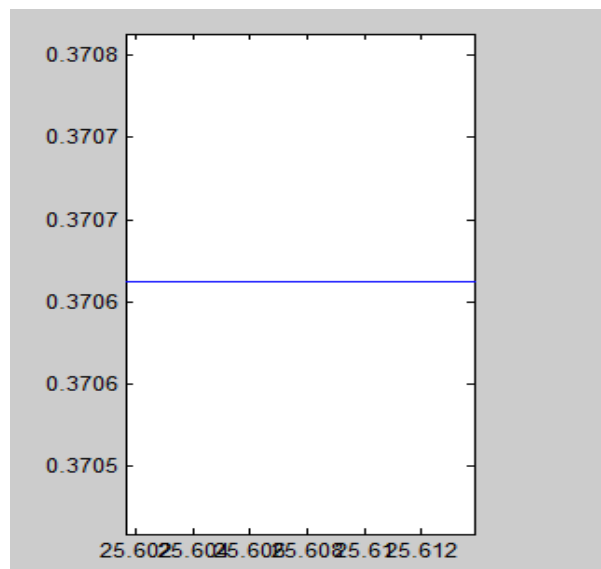
Capas	Neuronas	Valor inicial	Error Permitido	#Epocas
3	20	1	0.20	100

Fuente: Entrenamiento Red Neuronal, 2016

Elaborado por: Diego Romero

Como resultado de la ejecución, podemos evidenciar que la red con 20 neuronas, y 100 épocas consideradas para su entrenamiento tiene un error cuadrático medio de 0.3707, el tiempo de ejecución es 2.58 minutos.

Ilustración 25 Comportamiento del error - Entrenamiento 6



Fuente: Entrenamiento Red Neuronal, 2016

Elaborado por: Diego Romero

Se puede observar en la gráfica que el error vuela a mantenerse en 0.3707, lo que se evidencia claramente que no es una solución óptima.

Séptimo Entrenamiento

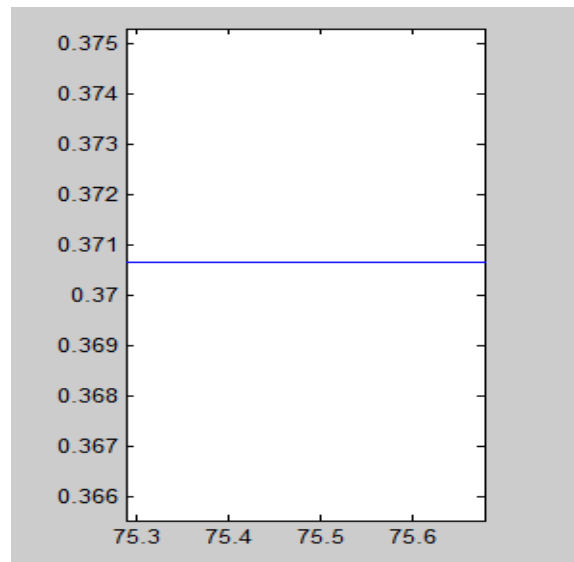
Tabla 24 Datos séptimo entrenamiento

Capas	Neuronas	Valor inicial	Error Permitido	#Épocas
3	30	1	0.20	100

Fuente: Entrenamiento Red Neuronal, 2016
Elaborado por: Diego Romero

Como resultado de la ejecución, podemos evidenciar que la red con 30 neuronas, y 100 épocas consideradas para su entrenamiento tiene un error cuadrático medio de 0.3705, el tiempo de ejecución es 12 minutos.

Ilustración 26 Comportamiento del error - Entrenamiento 7



Fuente: Entrenamiento Red Neuronal, 2016
Elaborado por: Diego Romero

Se puede observar en la gráfica que el error continúa en 0.3705, lo que se evidencia claramente que no es una solución óptima.

Octavo Entrenamiento

Tabla 25 Datos octavo entrenamiento

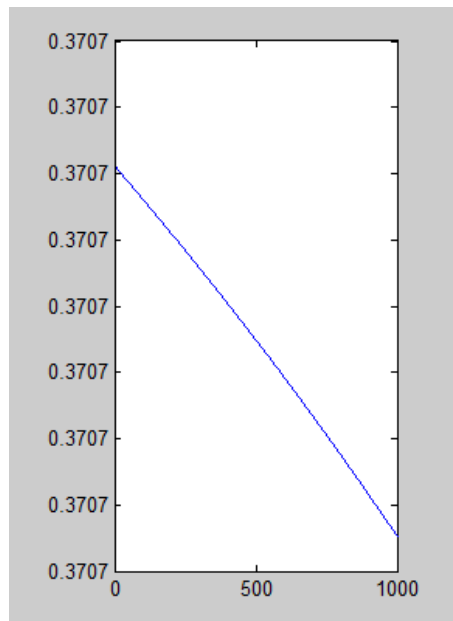
Capas	Neuronas	Valor inicial	Error Permitido	#Épocas
3	10	1	0.20	1000

Fuente: Entrenamiento Red Neuronal, 2016

Elaborado por: Diego Romero

Como resultado de la ejecución, podemos evidenciar que la red con 10 neuronas, y 1000 épocas consideradas para su entrenamiento tiene un error cuadrático medio de 0.3705, el tiempo de ejecución es 58 minutos.

Ilustración 27 Comportamiento del error - Entrenamiento 8



Fuente: Entrenamiento Red Neuronal, 2016

Elaborado por: Diego Romero

Se puede observar en la gráfica que el error desciende pero hasta en un mínimo de 0.3705, lo que se evidencia claramente que no es una solución óptima.

Noveno Entrenamiento

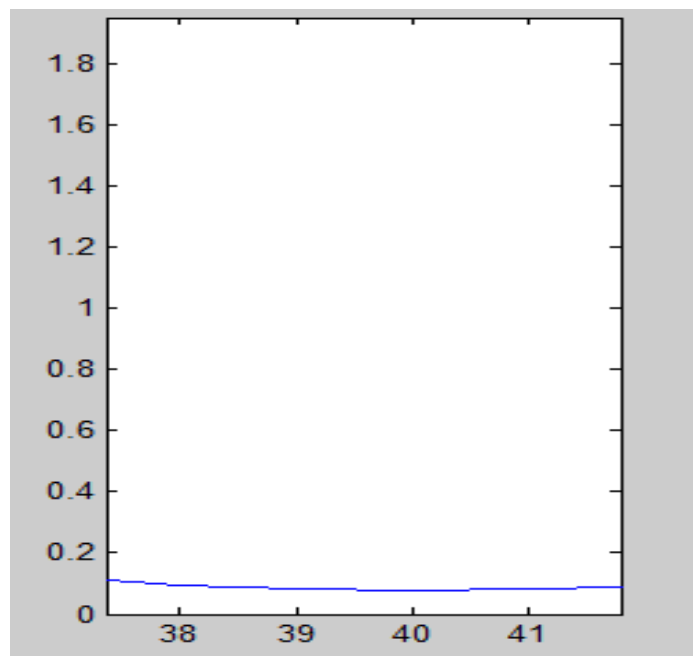
Tabla 26 Datos noveno entrenamiento

Capas	Neuronas	Valor inicial	Error Permitido	#Épocas
3	20	1	0.20	1000

Fuente: Entrenamiento Red Neuronal, 2016
Elaborado por: Diego Romero

Como resultado de la ejecución, podemos evidenciar que la red con 20 neuronas, y 1000 épocas consideradas para su entrenamiento tiene un error cuadrático medio de 0.1605, el tiempo de ejecución es 26 minutos.

Ilustración 28 Comportamiento del error - Entrenamiento 9



Fuente: Entrenamiento Red Neuronal, 2016
Elaborado por: Diego Romero

Se puede observar en la gráfica que el error desciende pero hasta en un mínimo de 0.1605, lo que se puede considerar una respuesta óptima.

Décimo Entrenamiento

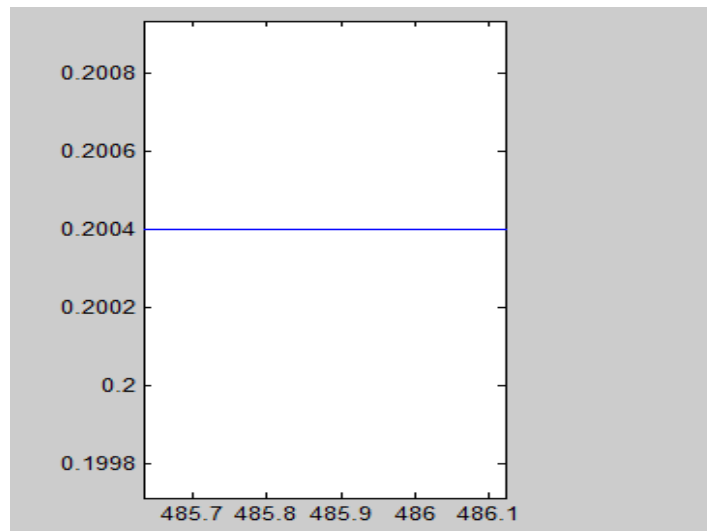
Tabla 27 Datos décimo entrenamiento

Capas	Neuronas	Valor inicial	Error Permitido	#Épocas
3	30	1	0.20	1000

Fuente: Entrenamiento Red Neuronal, 2016
Elaborado por: Diego Romero

Como resultado de la ejecución, podemos evidenciar que la red con 30 neuronas, y 1000 épocas consideradas para su entrenamiento tiene un error cuadrático medio de 0.2004, el tiempo de ejecución es 16.41 minutos.

Ilustración 29 Comportamiento del error - Entrenamiento 10



Fuente: Entrenamiento Red Neuronal, 2016
Elaborado por: Diego Romero

Se puede observar en la gráfica que el error consigue un mínimo de 0.2004, lo que se evidencia claramente que no es una solución óptima al compararla con la el resultado de la gráfica 28.

Capítulo IV

Propuesta de Mejora

4.1. Modelo de clasificación

4.1.1. Pesos sinápticos del modelo

Los pesos sinápticos que debe considerar el modelo de red neuronal para el aprendizaje son los siguientes:

4.1.1.1. Pesos para W1

A continuación se muestran los pesos a considerar en para la puesta en marcha de la red neuronal, representados en una matriz de 8x30, Matlab muestra la información de la siguiente manera:

Tabla 28 Resultados W1 parte 1

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	10.344	0.5421	13.726	0.5226	0.3359	0.1504	0.2754	11.196	0.9625	18.378	15.269	0.0146	15.177	16.351	0.4245
2	0.6788	0.9100	17.936	17.786	0.4634	0.0130	18.663	0.4176	0.9521	16.169	10.436	13.505	14.482	17.588	12.667
3	18.515	18.907	17.888	0.2935	12.146	0.3857	13.232	0.1046	0.9275	0.8821	0.4678	0.0397	17.505	0.1417	0.0840
4	0.1422	16.102	12.674	0.3114	0.0437	0.6950	0.1434	0.6138	15.988	0.6670	0.9656	0.2023	0.5138	13.729	16.927
5	0.0860	15.005	10.915	0.8966	11.519	16.687	17.838	0.8359	0.7261	0.2381	16.784	17.641	19.596	0.8900	12.226
6	0.1079	0.8096	16.111	0.7048	0.4712	0.7859	16.400	0.7671	14.717	0.8920	15.869	12.102	13.485	0.3696	10.113
7	0.5923	17.172	16.966	0.3510	0.8372	11.159	14.641	17.562	18.779	0.4895	10.479	10.648	0.0687	13.015	16.663
8	0.8775	11.239	17.190	0.5241	12.767	0.1963	0.4045	18.987	16.664	0.2450	0.7974	10.394	15.017	0.6277	14.965

Fuente: Entrenamiento Red Neuronal, 2016

Elaborado por: Diego Romero

Tabla 29 Resultados W1 parte 2

	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	19.820	0.2224	12.901	11.152	0.6308	14.864	0.1043	0.9262	0.9040	0.1329	0.5510	0.4652	0.3398	10.817	0.7867
2	0.1353	13.656	0.5396	0.9613	14.814	0.1937	0.5919	0.7017	15.415	19.026	16.974	18.993	0.2111	0.4444	11.401
3	0.1043	14.232	13.471	16.210	0.9439	0.9704	12.182	16.447	14.175	0.7831	0.0857	13.042	16.014	0.7849	13.456
4	16.222	17.713	19.032	16.760	11.927	18.321	19.432	16.383	14.744	13.991	10.023	14.973	19.509	0.9905	12.154
5	15.750	0.9658	11.865	0.0687	12.540	11.840	0.5282	0.6195	16.815	0.7424	16.726	0.2008	0.4747	0.0143	0.9550
6	15.375	17.793	0.5971	0.7553	0.9405	12.347	0.8240	17.675	16.713	0.7486	10.563	0.0054	13.710	14.739	16.677
7	0.5642	14.404	0.7416	15.132	0.8983	10.388	10.989	0.1018	0.0819	10.632	18.086	15.362	0.4712	12.215	0.8229
8	11.922	0.4454	19.171	11.218	14.987	0.7747	15.425	14.817	16.757	19.806	0.5732	0.4832	0.5053	0.6731	0.7779

Fuente: Entrenamiento Red Neuronal, 2016
Elaborado por: Diego Romero

4.1.1.2. Pesos Sinápticos para w2

Los pesos para w2 son como siguen a continuación: de manera que éstos involucran a la capa de salida.

Tabla 30 Pesos Sinápticos De la capa oculta a la capa de salida.

#	w2	#	w2	#	w2
1	18.780	11	12.951	21	15.062
2	17.872	12	0.6023	22	12.277
3	11.004	13	11.807	23	12.830
4	11.686	14	0.5936	24	14.151
5	18.392	15	15.868	25	17.145
6	0.7550	16	0.1447	26	0.6800
7	11.844	17	0.2821	27	18.226
8	11.138	18	12.495	28	12.567
9	18.914	19	10.460	29	13.171
10	18.352	20	0.6320	30	17.427

Fuente: Entrenamiento Red Neuronal, 2016
Elaborado por: Diego Romero

4.1.1.3. Ejecución por Periodicidad

La organización puede optar por este método cuando tome en consideración realizar monitoreo continuo automático por cada cierto instante de tiempo, y actualizar el monitor de fraudes en tiempo en períodos de cada 10 minutos o el tiempo que defina la organización.

4.1.1.4. Proceso Batch

Los procesos Batch son útiles debido a que no interfieren directamente con otros procesos en horarios operativos, sino que se podrá utilizar este proceso cuando el monitoreo no sea tan exigente, y así mismo este input de transacciones servirá a la red para seguir aprendiendo de las transacciones y comportamiento de las mismas.

4.1.1.5. En línea

Este proceso es costoso y riesgoso, debido a que el tiempo de procesamiento puede afectar el tiempo de respuesta de la transacción y declinarse por time out. Para activar este proceso deben revisarse los tiempos de respuesta totales entre el módulo autorizador y respuesta de la red neuronal por cada franquicia.

4.2. Resultados obtenidos

Se tomó de un mes de transacciones, y se evaluó la efectividad del proceso actual y la red neuronal, obteniendo que:

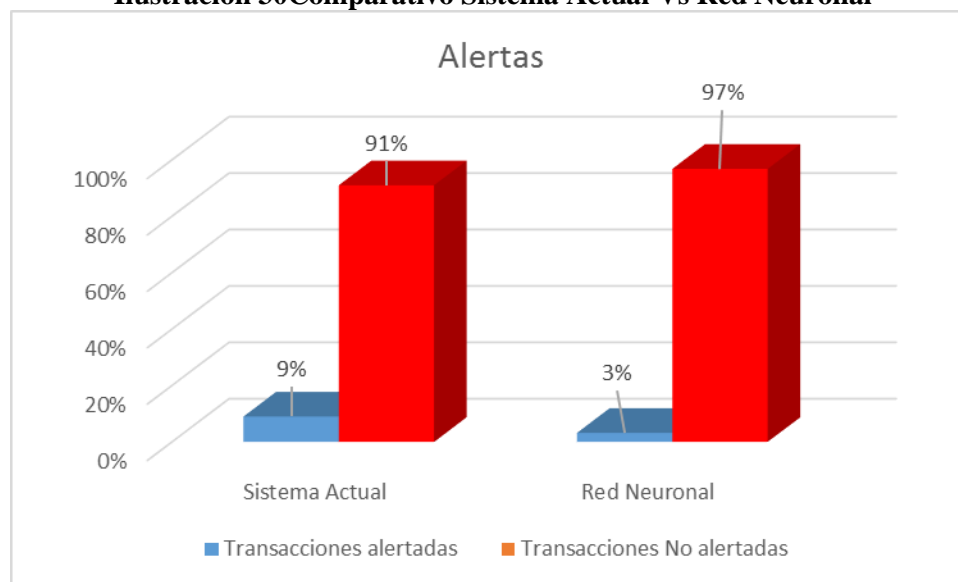
Se revisaron 500 transacciones al azar de un mes de transacciones, se obtuvieron con el sistema actual 45 alertas, y luego al pasar los datos por la red, se obtuvo 16 alertas.

Tabla 31 Comparativo Sistema Actual / Red Neuronal (Alertas)

Alertas	Sistema Actual	Red Neuronal
Transacciones alertadas	45	16
Transacciones No alertadas	455	484
Total	500	500

Fuente: Resultados Red Neuronal, 2016
Elaborado por: Diego Romero

Ilustración 30 Comparativo Sistema Actual Vs Red Neuronal



Fuente: Resultados Red Neuronal, 2016
Elaborado por: Diego Romero

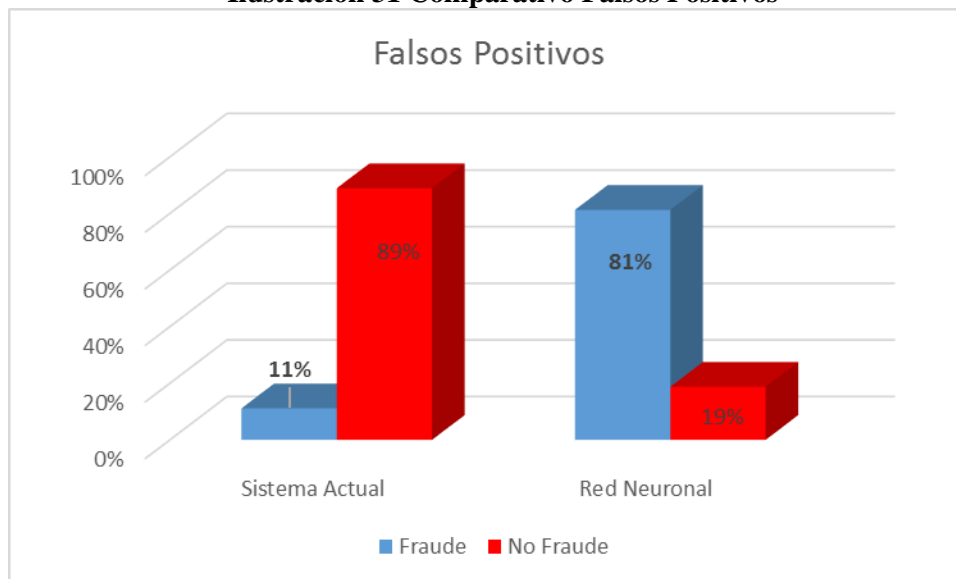
Posteriormente al validar las transacciones alertadas, se llamó al cliente a verificar la transaccionalidad y se obtuvo que de las 45 transacciones alertadas solo 5 fueron fraude, las 40 fueron solo falsos positivos, y utilizando la red neuronal, se obtiene que 13 transacciones de 16 fueron consideradas fraude y solo 3 fueron falsos positivos.

Tabla 32 Comparativo Sistema Actual / Red Neuronal (Falsos Positivos)

Falsos Positivos	Sistema Actual	Red Neuronal
Fraude	5	13
No Fraude	40	3
Total	45	16

Fuente: Resultados Red Neuronal, 2016
Elaborado por: Diego Romero

Ilustración 31 Comparativo Falsos Positivos



Fuente: Resultados Red Neuronal, 2016
Elaborado por: Diego Romero

Las características de estas transacciones inusuales son:

- Canal Internet / Velocidad de 5 transacciones en menos de 10 minutos.
- Inicialmente se evidencia que el comportamiento es realizar una transacción de \$1 varias veces e ir aumentando su monto en cada intento.

4.3. Funcionamiento de la red neuronal en tiempo real.

En este punto se hace una prueba con datos para evidenciar cómo funcionaría la red neuronal en tiempo real.

- Ingreso de la transacción sospechosa

Tabla 33 Transacción Sospechosa

Compra de monedas en casino	
Marca	MasterCard
Tipo	IP Platinum
País	Colombia
Modo	Internet
MCC	Gambling
Valor	100
Transacciones en menos de 10 minutos	5
Valor acumulado	500
Resultado	1

Fuente: Resultados Red Neuronal, 2016

Elaborado por: Diego Romero

- Ingreso de la transacción no sospechosa

Tabla 34 Transacción No Sospechosa

Compra de monedas en casino	
Marca	MasterCard
Tipo	IP Platinum
País	Colombia
Modo	Internet
MCC	Gambling
Valor	100
Transacciones en menos de 10 minutos	5
Valor acumulado	500
Resultado	0

Fuente: Resultados Red Neuronal, 2016

Elaborado por: Diego Romero

Conclusiones

1. Se cumple la hipótesis que con modelo de backpropagation, permite el entrenamiento de una red neuronal, a su vez utilizando una función de activación sigmoide, ya que su comportamiento es continuo.
2. Se logró obtener los pesos sinápticos mediante el método del descenso del gradiente, utilizando una función recursiva para el cálculo de la derivada de las entradas netas respecto a los pesos sinápticos, haciendo que el modelo minimice el error utilizando la retropropagación.
3. La carga operativa con esta herramienta disminuye considerablemente, ya que no se requiere que se haga gestión de monitoreo manual por usuario, ya que su aprendizaje es confiable y el 80% de las transacciones podrán ser candidatas a ser mitigadas.
4. Dentro de las reglas de negocio se concluye que cuando se observan cinco transacciones en 10 minutos, con el modelo de backpropagation y su entrenamiento de red neuronal, se obtiene que una alerta significativa de comportamiento inusual de un tarjetahabiente.
5. Esta herramienta no detiene ni acaba con el fraude, su objetivo es aprender de él para identificar la transaccionalidad inusual para prevenir el fraude.

Recomendaciones

1. Se recomienda que el proceso de ejecución del modelo sea con una periodicidad parametrizable a las necesidades de la empresa.
2. Fomentar el uso de las redes neuronales, con el uso de la inteligencia artificial, a fin de lograr la identificación de perfiles individualizados del tarjetahabiente, basado en el histórico de sus transaccionalidad, actualizando de forma automática su habitualidad por tipo de comercio, sector y montos promedios.
3. Implementar una política de bloqueos automáticos o manuales para los tarjetahabientes, ya sean bloqueos temporales hasta obtener la confirmación del cliente.
4. Realizar semanalmente el mantenimiento de la red neuronal, es decir ingresando nuevos patrones de transacciones fraudulentas para que su aprendizaje no supervisado se retroalimente.
5. Se recomienda la creación de la unidad de inteligencia de prevención de fraudes, para la constante implementación de alertas en relación a los resultados de las herramientas informáticas dedicadas al monitoreo de fraudes.

Anexos

Anexo 1 Codificación y Transformación de la Variable País

País	Código	T(x)
ALB	1	0,00
ARE	2	0,01
ARG	3	0,02
AUS	4	0,04
AUS	5	0,05
AUT	6	0,06
BEL	7	0,07
BGR	8	0,08
BMU	9	0,10
BOL	10	0,11
BRA	11	0,12
BUL	12	0,13
CAN	13	0,14
CAN	14	0,16
CHE	15	0,17
CHL	16	0,18
CHL	17	0,19
CHN	18	0,20
COD	19	0,22
COL	20	0,23
COL	21	0,24
CRI	22	0,25
CYP	23	0,27
CZE	24	0,28
DEU	25	0,29
DNK	26	0,30
DOM	27	0,31
ECU	28	0,33
ESP	29	0,34
FIN	30	0,35
FRA	31	0,36
FRA	32	0,37

GBR	33	0,39
País	Código	T(x)
GBRC	34	0,40
GCB	35	0,41
GIB	36	0,42
GRB	37	0,43
GRC	38	0,45
GTM	39	0,46
HDN	40	0,47
HKG	41	0,48
HND	42	0,49
IA	43	0,51
IDN	44	0,52
IND	45	0,53
IRL	46	0,54
IRL	47	0,55
ISL	48	0,57
ISR	49	0,58
ITA	50	0,59
JPN	51	0,60
LBN	52	0,61
LUX	53	0,63
MCO	54	0,64
MEX	55	0,65
MLT	56	0,66
MTL	57	0,67
MUS	58	0,69
MYS	59	0,70
NDL	60	0,71
NIC	61	0,72
NLD	62	0,73
NR	63	0,75
NZL	64	0,76
PAK	65	0,77
PAN	66	0,78
PAN	67	0,80
PER	68	0,81

PHL	69	0,82
País	Código	T(x)
POL	70	0,83
PRT	71	0,84
RUS	72	0,86
SAU	73	0,87
SGP	74	0,88
SWE	75	0,89
SX	76	0,90
SXM	77	0,92
TUR	78	0,93
TWN	79	0,94
UKR	80	0,95
URY	81	0,96
USA	82	0,98
VEN	83	0,99
ZAF	84	1,00

Anexo 2 Codificación y Transformación (Tx) de MCC

MCC	Código	T(x)	MCC	Código	T(x)
1520	1	0,00	3247	38	0,14
1711	2	0,00	3248	39	0,14
1799	3	0,01	3256	40	0,14
2741	4	0,01	3260	41	0,15
3000	5	0,01	3293	42	0,15
3001	6	0,02	3296	43	0,16
3007	7	0,02	3366	44	0,16
3008	8	0,03	3381	45	0,16
3009	9	0,03	3389	46	0,17
3010	10	0,03	3405	47	0,17
3011	11	0,04	3501	48	0,17
3026	12	0,04	3510	49	0,18
3029	13	0,04	3516	50	0,18
3030	14	0,05	3537	51	0,19
3034	15	0,05	3548	52	0,19
3039	16	0,06	3551	53	0,19
3043	17	0,06	3565	54	0,20
3047	18	0,06	3591	55	0,20
3051	19	0,07	3601	56	0,20
3052	20	0,07	3604	57	0,21
3058	21	0,07	3613	58	0,21
3063	22	0,08	3623	59	0,21
3066	23	0,08	3635	60	0,22
3072	24	0,09	3663	61	0,22
3076	25	0,09	3665	62	0,23
3102	26	0,09	3670	63	0,23
3127	27	0,10	3680	64	0,23
3136	28	0,10	3708	65	0,24
3174	29	0,10	3718	66	0,24
3211	30	0,11	3719	67	0,24
3219	31	0,11	3741	68	0,25
3221	32	0,11	3765	69	0,25
3236	33	0,12	3828	70	0,26
3238	34	0,12	4011	71	0,26
3243	35	0,13	4111	72	0,26
3245	36	0,13	4112	73	0,27
3246	37	0,13	4121	74	0,27

MCC	Código	T(x)	MCC	Código	T(x)
4131	75	0,27	5261	112	0,41
4214	76	0,28	5299	113	0,41
4215	77	0,28	5300	114	0,42
4225	78	0,29	5309	115	0,42
4411	79	0,29	5310	116	0,43
4468	80	0,29	5311	117	0,43
4511	81	0,30	5321	118	0,43
4722	82	0,30	5331	119	0,44
4784	83	0,30	5332	120	0,44
4789	84	0,31	5399	121	0,44
4812	85	0,31	5411	122	0,45
4814	86	0,31	5422	123	0,45
4816	87	0,32	5441	124	0,46
4899	88	0,32	5451	125	0,46
4900	89	0,33	5462	126	0,46
4945	90	0,33	5499	127	0,47
5013	91	0,33	5511	128	0,47
5035	92	0,34	5531	129	0,47
5039	93	0,34	5532	130	0,48
5045	94	0,34	5533	131	0,48
5046	95	0,35	5541	132	0,49
5047	96	0,35	5542	133	0,49
5065	97	0,36	5571	134	0,49
5072	98	0,36	5599	135	0,50
5074	99	0,36	5611	136	0,50
5085	100	0,37	5621	137	0,50
5094	101	0,37	5631	138	0,51
5099	102	0,37	5641	139	0,51
5122	103	0,38	5651	140	0,51
5131	104	0,38	5655	141	0,52
5139	105	0,39	5661	142	0,52
5172	106	0,39	5681	143	0,53
5192	107	0,39	5691	144	0,53
5199	108	0,40	5698	145	0,53
5200	109	0,40	5699	146	0,54
5211	110	0,40	5712	147	0,54
5251	111	0,41	5713	148	0,54

MCC	Código	T(x)	MCC	Código	T(x)
5714	149	0,55	5969	186	0,69
5718	150	0,55	5970	187	0,69
5719	151	0,56	5971	188	0,69
5722	152	0,56	5973	189	0,70
5732	153	0,56	5976	190	0,70
5733	154	0,57	5977	191	0,70
5734	155	0,57	5983	192	0,71
5735	156	0,57	5984	193	0,71
5799	157	0,58	5992	194	0,71
5812	158	0,58	5993	195	0,72
5813	159	0,59	5994	196	0,72
5814	160	0,59	5995	197	0,73
5816	161	0,59	5996	198	0,73
5851	162	0,60	5997	199	0,73
5912	163	0,60	5998	200	0,74
5921	164	0,60	5999	201	0,74
5931	165	0,61	6010	202	0,74
5933	166	0,61	6011	203	0,75
5939	167	0,61	6012	204	0,75
5940	168	0,62	6051	205	0,76
5941	169	0,62	6211	206	0,76
5942	170	0,63	6300	207	0,76
5943	171	0,63	6513	208	0,77
5944	172	0,63	6540	209	0,77
5945	173	0,64	7011	210	0,77
5946	174	0,64	7033	211	0,78
5947	175	0,64	7217	212	0,78
5948	176	0,65	7230	213	0,79
5949	177	0,65	7261	214	0,79
5950	178	0,66	7273	215	0,79
5960	179	0,66	7277	216	0,80
5962	180	0,66	7278	217	0,80
5964	181	0,67	7299	218	0,80
5965	182	0,67	7311	219	0,81
5966	183	0,67	7333	220	0,81
5967	184	0,68	7338	221	0,81
5968	185	0,68	7372	222	0,82

Diseño e implementación de un modelo de clasificación para la detección de transacciones inusuales en la conducta habitual de un tarjetahabiente en una entidad del sistema financiero

MCC	Código	T(x)	MCC	Código	T(x)
7375	223	0,82	8351	260	0,96
7379	224	0,83	8398	261	0,96
7392	225	0,83	8641	262	0,97
7393	226	0,83	8651	263	0,97
7394	227	0,84	8661	264	0,97
7399	228	0,84	8699	265	0,98
7512	229	0,84	8734	266	0,98
7523	230	0,85	8931	267	0,99
7538	231	0,85	8999	268	0,99
7542	232	0,86	9222	269	0,99
7622	233	0,86	9399	270	1,00
7629	234	0,86	9402	271	1,00
7630	235	0,87			
7631	236	0,87			
7829	237	0,87			
7832	238	0,88			
7841	239	0,88			
7922	240	0,89			
7929	241	0,89			
7941	242	0,89			
7990	243	0,90			
7991	244	0,90			
7993	245	0,90			
7994	246	0,91			
7995	247	0,91			
7996	248	0,91			
7997	249	0,92			
7999	250	0,92			
8011	251	0,93			
8043	252	0,93			
8062	253	0,93			
8071	254	0,94			
8099	255	0,94			
8111	256	0,94			
8220	257	0,95			
8244	258	0,95			
8299	259	0,96			