



**FACULTAD DE INGENIERÍA, ARQUITECTURA Y  
URBANISMO**

**ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS**

**TESIS**

**IMPLEMENTACIÓN DE UN SISTEMA DE  
PRONÓSTICO DE VENTAS UTILIZANDO REDES  
NEURONALES ARTIFICIALES PARA LA EMPRESA  
CERÁMICOS LAMBAYEQUE SAC**

**PARA OPTAR EL TÍTULO PROFESIONAL DE INGENIERO  
DE SISTEMAS**

**Autor(a) (es):**

**Bach. Benites Sernaqué José Manuel  
ORCID: <https://orcid.org/0000-0002-4876-6887>**

**Asesor(a):**

**Mg. Malca Quispe María Nicida  
ORCID: <https://orcid.org/0000-0002-2759-3569>**

**Línea de Investigación:**

**Infraestructura, Tecnología y Medio Ambiente**

**Pimentel – Perú 2021**

## **APROBACIÓN DEL JURADO**

### **IMPLEMENTACIÓN DE UN SISTEMA DE PRONÓSTICO DE VENTAS UTILIZANDO REDES NEURONALES ARTIFICIALES PARA LA EMPRESA CERÁMICOS LAMBAYEQUE SAC**

---

**Bach. Benites Sernaqué José Manuel**

**Autor**

---

**Mg. Malca Quispe Nicida**

**Asesor**

---

**Mg. Tuesta Monteza Víctor**

**Presidente de Jurado**

---

**Mg. Ramos Moscol Mario**

**Secretario de Jurado**

---

**Mg. Vidaurre Flores Miguel**

**Vocal de Jurado**

## **Dedicatorias**

A mi señora madre que desde un inicio hasta el fin de sus días me supo apoyar con sus buenos valores, esfuerzo y su confianza para seguir adelante con mis estudios.

A mi Padre, por su gran apoyo incondicional que en todo momento me supo ofrecer y que hasta el día de hoy ha sabido comprender lo difícil que ha sido para mí esta Carrera.

A mi esposa que día a día me comprende y sabe valorar mi esfuerzo, así mismo me apoya con sus ideas y sus buenos deseos para mi superación.

A mis dos pequeños hijos que son el motor de mi vida y me dan fuerzas para superar cualquier obstáculo que existe en mi camino.

## **Agradecimientos**

Agradezco infinitamente a mis asesores; Ing. Nicida Malca Quispe, Ing. Christian Abraham Dios Castillo, por sus aportes y apoyo en el desarrollo de mi informe de tesis. A mis docentes, que hicieron parte de mi formación, facilitándome los distintos instrumentos académicos que sirvieron en el proceso del desarrollo de este trabajo de grado. A todas las personas que durante mi formación académica, sumaron ideas para mi aprendizaje, los cuales se plasmaron en el presente trabajo como complemento para obtener resultados satisfactorios.

## Resumen

Cerámicos Lambayeque SAC, es una joven y sólida empresa con 10 años en el mercado local y nacional, dedicado a la actividad de la producción y comercialización de productos del rubro de construcción, tales como ladrillos maquinados en sus diferentes tipos y categorías, así como también productos de agregados para el proceso de construcción, tales como piedra chancada, arenilla y arena gruesa.

En la actualidad la empresa industrial “Cerámicos Lambayeque SAC”, factura cerca de 1 millón de soles mensuales en las ventas de sus productos de ladrillos, y cerca de 0.5 millones de soles en sus productos de agregados.

Bajo las condiciones de su rentabilidad, ésta no cuenta con una solución informática, que le sirva de herramienta para pronosticar las ventas futuras.

Con la finalidad de mejorar la toma de decisiones sobre las ventas o evaluar estrategias de ventas en la empresa Cerámicos Lambayeque SAC, se ha desarrollado un sistema de pronóstico de ventas aplicando redes neuronales, para pronosticar las ventas futuras en la entidad.

Para la implementación del sistema de pronóstico de ventas, se ha utilizado la Red Neuronal Perceptron Multicapa y la Regresión de Serie Temporal teniendo en cuenta variables de entrada, recolección y normalización de datos, la determinación de la estructura de la red, entrenamiento y validación.

Se realizó pruebas de funcionalidad del sistema de pronóstico de ventas y como resultado a estas, podemos concluir que la venta mensual diaria pronosticada utilizando la red neuronal con un ratio de aprendizaje de 0.01 fluctúa dentro de los límites de la venta diaria mensual, lo cual significa un pronóstico rentable.

**Palabras claves:** Redes Neuronales, Neurona, Perceptron, Perceptron Multicapa, Regresión Serie Temporal, Función de Activación, Entrenamiento de Red Neuronal, Pronóstico de Ventas.

## **Abstract**

Cerámicos Lambayeque SAC, is a young and solid company with 10 years in the local and national market, dedicated to the activity of the production and commercialization of construction products, such as machined bricks in its different types and categories, as well as aggregate products for the construction process, such as crushed stone, grit and coarse sand.

At present, the industrial company “Cerámicos Lambayeque SAC”, invoices about 1 million soles per month in the sales of its brick products, and about 0.5 million soles in its aggregate products.

Under the conditions of its profitability, it does not have a computer solution, which serves as a tool to forecast future sales.

In order to improve decision-making on sales or evaluate sales strategies in the company Cerámicos Lambayeque SAC, a sales forecasting system has been developed by applying neural networks, to forecast future sales in the entity.

For the implementation of the sales forecasting system, the Multilayer Perceptron Neural Network and the Time Series Regression have been used, taking into account input variables, data collection and normalization, the determination of the network structure, training and validation.

Functionality tests of the sales forecast system were carried out and as a result of these, we can conclude that the daily monthly sale predicted using the neural network with a learning ratio of 0.01 fluctuates within the limits of the monthly daily sale, which means a profitable forecast.

**Keywords:** Neural Networks, Neuron, Perceptron, Multilayer Perceptron, Time Series Regression, Activation Function, Neural Network Training, Sales Forecast.

## Índice

<b>I. INTRODUCCIÓN</b> .....	8
<b>1.1. Realidad Problemática.</b> .....	10
<b>1.2. Trabajos previos.</b> .....	11
<b>1.3. Teorías relacionadas al tema.</b> .....	16
<b>1.4. Formulación del Problema.</b> .....	36
<b>1.5. Justificación e importancia del estudio.</b> .....	36
<b>1.6. Hipótesis.</b> .....	39
<b>1.7. Objetivos.</b> .....	39
<b>1.7.1. Objetivo general.</b> .....	39
<b>1.7.2. Objetivos específicos.</b> .....	39
<b>II. MATERIAL Y MÉTODO</b> .....	39
<b>2.1. Tipo y Diseño de Investigación.</b> .....	39
<b>2.2. Población y muestra.</b> .....	40
<b>2.3. Variables, Operacionalización.</b> .....	40
<b>2.4. Técnicas e instrumentos de recolección de datos, validez y     confiabilidad.</b> .....	42
<b>2.5. Procedimiento de análisis de datos.</b> .....	44
<b>2.6. Criterios éticos.</b> .....	44
<b>2.7. Criterios de Rigor Científico.</b> .....	45
<b>III. RESULTADOS.</b> .....	45
<b>3.1. Resultados en Tablas y Figuras.</b> .....	45
<b>3.2. Discusión de resultados.</b> .....	55
<b>3.3. Aporte práctico.</b> .....	65
<b>IV. CONCLUSIONES Y RECOMENDACIONES</b> .....	121
<b>4.1. Conclusiones.</b> .....	121
<b>4.2. Recomendaciones.</b> .....	122
REFERENCIAS.....	123
ANEXOS. ....	126

## I. INTRODUCCIÓN

Las ventas históricas, datos contables, es el requisito principal para reunir condiciones que deben tratarse mediante los pilotos de las Neuronas-RN. La base de información de las ventas, y sustancialmente como resultado de los sistemas transaccionales de las empresas, suelen ser de multitud de datos en desorden, a veces incompletos y en oportunidades erróneas o manipuladas.

Frente al tema, estas cualidades de la data financiera, que revistan planificar incidentes para el procedimiento estadístico o formalización con algoritmos clásicos, son los insumos en la que las RN están proporcionando los mejores efectos, ya que garantizan de un nivel de capacidad de filtrar los sonidos que van con la información, así como el gran nivel de pasividad a fallos (Serrano & Bonifacio, 2008, pág. 156).

La rentabilidad económica de una empresa se mide en las ventas diarias mensuales obtenidas dentro un periodo.

El objetivo de esta investigación tiene como finalidad implementar un sistema que pronostique las ventas haciendo uso de las RNA para la empresa de producción y comercializadora “Cerámicos Lambayeque SAC”.

El software utiliza los datos de entrada que son: ventas diarias mensuales de los años 2016 y 2017. Se definió un almacén de ventas estático utilizando un arreglo bidimensional de 200 filas por 5 columnas para almacenar dichas ventas, las ventas del año 2016 se almacenan en el arreglo desde el ítem 0 hasta el ítem 71, las ventas del año 2017 se ubican en la posición 72 hasta la posición 143 y finalmente a partir de la posición 144 hasta la posición 199 se almacenan las ventas diarias mensuales pronosticadas para el año 2018.

Las ventas ubicadas en el arreglo se normalizan, utilizamos cada venta diaria y lo dividimos entre 1000.



Visualizamos los datos que se encuentran en el arreglo cada vez que se realice el entrenamiento de la neurona para ver las ventas pronosticadas. Inicializamos pesos, fijamos pesos y visualizamos los pesos generados por la RN después de cada preparación.

En la implementación de la neurona se ha utilizado cuatro (4) neuronas de entrada, doce (12) neuronas ocultas y una (1) neurona de salida. Definimos los atributos que nuestra red neuronal tendrá, asignamos valores a cada atributo de la red. Para esto creamos matrices para asignar los pesos sinápticos.

Finalmente se realiza el entrenamiento de la RNA asignándoles los datos resultantes y una cantidad de iteraciones que debe ejecutar la red, así como también el ratio de aprendizaje. En cada entrenamiento se obtienen y visualizan errores.

El software permite al usuario comparar las ventas de la empresa utilizando las ventas diarias y mensuales desde el año 2016 al 2017 para pronosticar las ventas diarias a partir del mes inicial de enero a julio del periodo 2018.

Para el procedimiento y desarrollo de la presente solución, se hizo uso del Lenguaje de codificación Visual C# 2013, dado que tiene la característica de ser sencillo e intuitivo, novedoso y orientado a objetos.

El material presentado en la tesis está organizado en seis capítulos. El primer capítulo comienza con un resumen breve de la situación actual del problema, y la interrogante a la misma, luego se justificó la importancia de lo investigado y finalmente culminamos el capítulo formulando los objetivos de la investigación. El capítulo dos, hace mención a las referencias de la investigación a nivel internacional, nacional, regional, institucional; en el estado del arte se describe las técnicas de reconocimiento mediante modelos de color, algoritmos y métodos de clasificación utilizados en frutos; en la bases teórico científicas se comienza con un breve reseña del tomate y los conceptos, técnicas, métodos relacionados con la visión artificial para clasificar el tomate. El capítulo tres hace

mención a la característica que se aplica a tipos y diseños de encuestas, como poblaciones y muestras, hipótesis, variables utilizadas y actividad de las variables. Métodos, métodos y herramientas de recopilación de datos utilizados; procesamiento para la obtención de datos, plan de análisis descriptivo, criterios éticos y criterios de rigor científico. En el capítulo cuatro se exponen los resultados de esta tesis y en el capítulo cinco presentamos la propuesta de la investigación y finalizamos con el capítulo seis con las conclusiones y recomendaciones.

### **1.1. Realidad Problemática.**

Existen La tendencia del rubro construcción hoy en día está elevándose de manera exitosa y la Industria ladrillera es una actividad de mucha importancia económica con enormes potenciales para el desarrollo empresarial sostenible.

Hoy en día la actividad está representada por la existencia de aproximadamente 35 empresas ladrilleras, agrupadas en asociaciones, sin embargo las mismas tienen limitaciones de tecnologías de información que les permitan elaborar y hacer uso de estrategias de ventas.

Una de las cosas que más incomoda a los empresarios de negocio en el rubro, es la falta de un modelo macroeconómico que sirva para pronosticar y analizar las ventas en un determinado periodo.

En esa línea los temas en que los fines óptimos de la Tecnología de Información son indudables, tales que resulta necesario medir los beneficios completamente módicos.

En consecuencia, sus temas monetarios van escalando. En esta ocasión la alta dirección se encuentra en un déficit; Sin embargo, se piensa que no puede permitirse hacerlo sin una herramienta seguridad y de confianza para el impacto en el rendimiento de los resultados.

En consecuencia, se requiere que las Tic se automaticen. Para esto hay que inducir hacia beneficios que mejoren la estructura de decisiones de la empresa. Dado la necesidad precedente, las empresas requieren de un seguimiento de sus operaciones, para lo cual se han creado un número de aplicaciones de software con la finalidad de recolectar datos y entregar información precisa y confiable a la alta gerencia.

Las ventas diarias mensuales, son más que un almacén de datos con información útil y necesaria para el desempeño de la Empresa. Dicha información está organizada de tal manera que facilite la búsqueda de registros y la emisión de reportes detallados, para el análisis de la alta gerencia.

Por las razones señaladas, impulsamos la creación de un sistema que pronostique las ventas a través de redes neuronales, cuya finalidad principal es la transformación de datos a información analítica importante y predictiva para su rentabilidad.

Con la creación de estos sistemas, damos un gran salto al desarrollo de múltiples software que conllevan a realizar análisis en la administración empresarial de manera estratégica, que facilite a la alta gerencia la toma de decisiones mejorada y predecir su rentabilidad a largo y corto plazo.

## **1.2. Trabajos previos.**

### **1.2.1. Nivel Internacional**

A) Según Foix, C., Weber, R. (2014), en su investigación denominada ***Pronóstico del Precio del Cobre Mediante Redes Neuronales***, tiene como objetivo la búsqueda de metodologías de pronósticos eficientes que puedan contar con resultados de mayor exactitud para su uso, tanto en la industria como en la definición de presupuesto nacional. Esto generó una evaluación del trabajo que se pudo predecir de las distintas RN de tipo multilayer perceptrón, elaboradas sobre el nivel del precio pasado del cobre y variables resultantes del mismo, tales como la última desviación del precio y

la inversión estándar de los .próximos periodos. En los efectos finales, indica que se hicieron comparaciones con los datos elaborados mediante el empleo de procesos ordinarios y las figuras modeladas de series de tiempo, estableciendo el tipo de comportamiento capaz de abstraer la data existente en los precios ya dados durante periodos anteriores. Se concluye que los procesos de modelado no lineales de pronósticos, basados en RN multilayer perceptrón, sobrepasaron la fiabilidad de los procesos de modelados lineales de series de tiempo utilizados comúnmente. De esta modo, las RN demostraron tener la capacidad de obtener mejor la información incrustada en los precios de periodos históricos considerados en el estudio (Foix & Weber, 2014).

B) Según García Díaz, C. (2014), en su trabajo de grado denominado ***Predicción del Comportamiento de la Rentabilidad de la Acción de ISAGEN, a partir de la Técnica de Redes Neuronales Artificiales para optar el grado de Magister en Economía.*** Tiene como objetivo principal establecer si el desempeño de un modelo de RNA Autorregresiva medido en términos de error de pronóstico, es más eficiente, que el desempeño de un modelo ARMA y un ARMA GARCH, verificados en el escenario de predicción de la rentabilidad de una acción del mercado ubicado en la ciudad de Colombia. Este objetivo tuvo éxito realizando el modelado de la actuación de la rentabilidad de la acción de ISAGEN, para obtener predicciones que facilitaron la decisión de inversión; para tal efecto, resulta necesario conocer algunas características del mercado en el que se indican estas acciones y del sector del rubro económico al que pertenece la empresa, en búsqueda de acciones, características o variables que permitan explicar los mencionados comportamientos. Finalmente el modelo de RNA Autorregresiva que hace uso de la tangente sigmoidea como acción para la activación en la capa oculta, estimado en el presente trabajo. Esto presentó mejor nivel de error en función de los errores de pronóstico comparado con el mejor modelo ARMA y el mejor

modelo ARMA GARCH estimados; aunque es eminente aclarar que las diferencias entre ambos modelos con resultados de errores, en conjunto, no son significantes, por tanto, se eligió por efectuar una prueba de capacidad predictiva, basada en (Diebold & Mariano, 1991), entre los dos mejores modelos: el RNA y el ARMAGARCH, y se pudo afirmar que en este caso específico no hay diferencia significativa en su capacidad de predicción (García Díaz, 2014).

C) Según Escobar R, Valdes H, & Zapata C, (2010), en su artículo científico que denominó ***Redes Neuronales Artificiales en predicción de Series de Tiempo. Una aplicación a la industria*** presentada al Departamento de Informática y Computación. Facultad de Ingeniería de la Universidad Tecnológica Metropolitana. Santiago, Chile. Tiene como objetivo tener una evaluación de la capacidad que presentan las RNA para predecir series de tiempos, así como estudiar dos series tomadas del campo de la industria: Consumo Mensual de Gas Natural y la Generación de Electricidad Mensual. Para tal efecto al objetivo se pudo lograr utilizando el tipo de RN: Perceptrón Multicapa y los algoritmos de aprendizaje Backpropagation y Resilient Propagation. Los resultados que se lograron obtener en cada serie dan manifiesto de la gran potencialización y capacidad que tienen las RN aplicadas en el proceso de la predicción de las series de tiempo, obteniéndose un error absoluto medio porcentual de predicción inferior al 3% en la serie Generación de Electricidad Mensual e inferior al 5% en la serie de Consumo Mensual de Gas Natural. Finalmente se concluye que al verificar los resultados obtenidos durante el proceso que resultó con cada configuración en ambas series; el algoritmo Resilient Propagation expone una gran ventaja sobre el algoritmo de Backpropagation, dado que resulta con mejores indicadores beneficiosos y un tiempo de procesamiento mucho menor en la mayoría de los casos estudiados del mismo tipo.

### 1.2.2. Nivel Nacional

A) Según Huayna Dueñas, Calvo Huaraz, & Huiman Sánchez, (2010), en su artículo científico denominado **“Modelo de Evaluación de Créditos Financieros basados en Redes Neuronales orientado a Edpymes** publicado en la Revista de Investigación en Informática y Sistemas, Departamento de Ingeniería de Sistemas y Ciencias de la Computación, Universidad Nacional de San Marcos. Realizó un análisis a los principales riesgos que afrontan la gran mayoría de instituciones financieras, lo cual se señala que hay probabilidad de que los que presentan deudas no puedan pagarles. Este artículo le mostrará patrones que le ayudarán a resolver el problema. Esto es para minimizar el riesgo de crédito para los clientes cuando piden prestado capital o crédito a los bancos. Este modelo se dirige a Edpymes utilizando técnicas de ARN (AI), como las redes neuronales generadas mediante algoritmos de retropropagación. Al emplearse el modelo de pronóstico, se alcanzó minimizar el indicador de moras de un 3,5% al 2,5%. Este 1% fluctúa y representa aproximadamente. Aproximadamente 900 clientes atendidos a gran escala (Huayna Dueñas, Calvo Huaraz, & Huiman Sánchez, 2010).

### 1.2.3. Nivel Regional

A) Según Vidaurre Siadén, (2012), en su tesis de grado denominado **Aplicación de las Redes Neuronales Artificiales para el Pronóstico de la Demanda de Agua Potable en la Empresa EPSEL S.A. de la ciudad de Lambayeque** para optar el título de ingeniero en informática y sistemas. Su objetivo general es desarrollar un modelo ARN para la previsión del consumo de agua a medio plazo que mejorará el proceso de previsión de la demanda de agua de EPSEL S.A.(Vidaurre Siadén, 2012). Esto se logró aplicando técnicas automáticas inclinadas a la inteligencia artificial, como las RNA, las cuales minimizan el nivel de error en la previsión de la demanda de agua potable. Se utilizó una

metodología para la implementación de redes neuronales en la predicción del agua, basada en el trabajo de (Kaastra & Boyd, 1995), esto incluye ocho etapas: selección de variables, recopilación de datos y preprocesamiento de datos. Definición de conjuntos de entrenamiento, validación y pruebas. Selección de la arquitectura de la red neuronal, criterios de evaluación, implementación del modelo de red neuronal artificial. En análisis de los resultados están enfocados a evaluar los indicadores de error de pronóstico. Se puede concluir que aplicando tecnologías de inteligencia artificial como las redes neuronales artificiales con arquitectura multinivel perceptrón, es posible predecir la cantidad de agua potable necesaria a medio plazo.

#### 1.2.4. Nivel Institucional

A) Según Malaver Elera, (2015), en su tesis de pre - grado denominado ***Aplicación de Redes Neuronales para determinar el pronóstico de las ventas en la empresa Catering & Buffets Mys ubicada en la ciudad de Piura.*** para optar el título profesional de Ingeniero de Sistemas en la Universidad Cesar Vallejo. Tiene como objetivo principal determinar el pronóstico de las ventas de los servicios que ofrece Catering & Buffets MyS con el menor porcentaje de error al aplicar las redes neuronales. Esto se logró según la metodología que planteó Kaastra y Boyd, donde se encuentra una serie de pasos para poder modelar desde cero la red neuronal para obtener el pronóstico. El tipo de RN que se señalaron, fueron las redes neuronales multicapas, ya que poseen una capa oculta donde se realiza todo el procesamiento de los datos que la capa de inicial le facilita y cuyo resultado es enviada a la capa de salida. Se realizaron tres diferentes configuraciones de red neuronal para hallar el más mínimo error, siendo la tercera red neuronal planteada, la más adecuada por presentar el menor porcentaje de error (1%), con lo que se logró comprobar la hipótesis planteada. Como conclusión a esta investigación, se

comprobó que las redes multicapas son las más adecuadas para realizar los pronósticos de las ventas, y que la herramienta de red neuronal son precisas para este tipo de aplicación.

### **1.3. Teorías relacionadas al tema.**

#### **1.3.1. Redes Neuronales**

Según Duque, (2020), refiere que una RNA es un sistema que procesa información y tiene ciertas capacidades en común con las RNA biológicas:

- a) El proceso de la información, ocurre en variedad de elementos simples que suelen llamarse neuronas.
- b) Las señales son conducidas entre neuronas mediante enlaces de conexión.
- c) Cada nodo va con un peso asociado, mediante el cual típicamente, multiplica a la señal que se transmitió.
- d) Individualmente la neurona emplea únicamente la función para su activación (constantemente no lineal) a su entrada de red (sumatoria de entradas pesadas) para el resultado de su salida.

#### **A) Estructura**

Según Serrano & Bonifacio, (2008, pág. 4), indica que una RN es conformada por un gran número de procesos elementales, que se relacionan unas con otras mediante sinapsis. La neurona viene a ser un elemento procesador muy simple (Figura 1). Sus vías o conductos de entrada de los datos se le conocen con el nombre de dendritas; el cuerpo conecta las diferentes señales de ingreso o entrada y emite una respuesta, habitualmente alcanzando una situación de trasmisión de tipo no lineal; el axón lanza una señal hacia las demás neuronas. La parte intensa de la interacción entre dos (02) neuronas es otorgado por el peso sináptico que las fusiona: Si los resultados son objetivos, las neuronas presinápticas activarán neuronas postsinápticas. Si es negativo, lo detendrá. Por lo tanto, una medida mayor o menor basada en su valor absoluto (Serrano & Bonifacio, 2008).



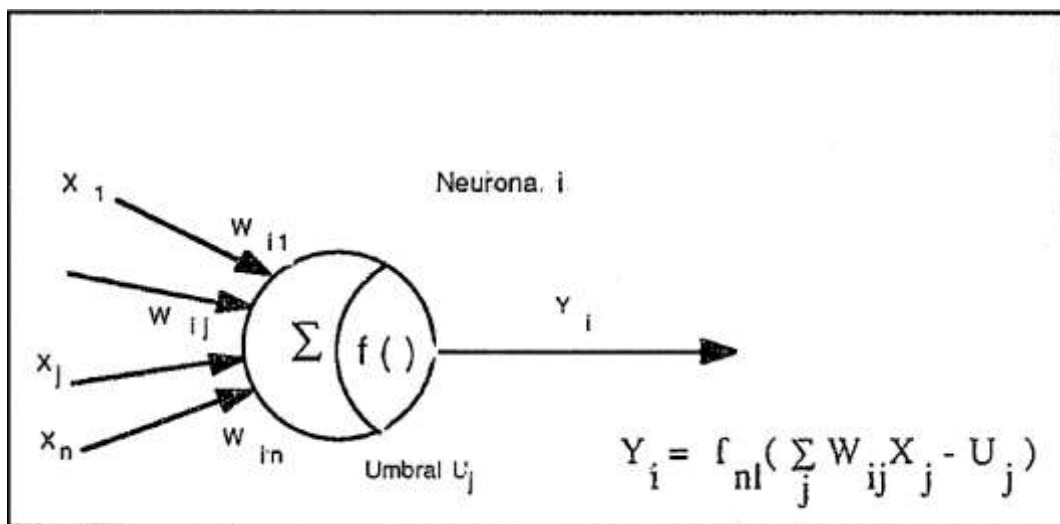


Figura 1. Modelo de Red Neuronal Artificial.

Fuente: (Serrano & Bonifacio, 2008).

## B) Operación

Según Serrano & Bonifacio, (2008, pág. 157), la RN se comporta en 2 distintos tipos de trabajo: aprendizaje o learning y recuerdo o recall. En principio, la RN está unida de forma aleatoria, el aprendizaje útil almacenado es vacío. Para que pueda resultar una acción óptima hay que entrenarla, de tal forma que se aplique una serie de pruebas, lo que se le conoce como la fase de conocimiento y aprendizaje. El aprendizaje no se transmite de forma inmediata en la RN, como sucede en varios sistemas algorítmicos de forma transaccionales, sino resulta de patrones que le son entregados, muestras adecuadas del proceso que prueba que el sistema se lleve a ejecutar. A partir de ello y a través del proceso de un algoritmo de conocimiento u aprendizaje, La RN de manera interactiva altera los valores de sus pesos sinápticos, adecuando de manera progresiva su respuesta a lo que de ella espera el codificador. En la RN, el cálculo no viene definitivo por un algoritmo, sino por la complejidad en la distribución de conexiones que durante la etapa de aprendizaje se crea entre las mismas. En las sinapsis se acumulan los datos de un modo disperso.

Si escogió un tipo de RN ventajoso a lo que se requiere para el problema y se ha elegido un grupo moderado de patrones característicos de la operación que pretendemos efectuar, la red neuronal indicará la capacidad de

generalización; Con estos ejemplos, las reglas básicas se desglosan y los nuevos modelos que no se ven a tiempo responderán a tiempo.

Al finalizar los entrenamientos, la red neuronal estará operativo en el estado de recordación, en el que indicará a un determinado proceso de entrada, sin realizar alguna modificación de su característica estructural ni en sus interconexiones, otorgando un indicador de acuerdo al entrenamiento recibido. El cálculo que ejerce la neurona en estado - modo recuerdo, da lo siguiente: sea la neurona  $i$ , sean  $X_j$  sus entradas (salidas de las neuronas pre sinápticas  $j$ ),  $W_{ij}$  los pesos sinápticos de interacción de las neuronas  $j$  con la  $i$ ,  $U_i$  su umbral de disparo y  $f(.)$  su función de traspaso. La neurona  $i$  realiza la operación siguiente: pesa cada entrada  $X_j$  que recibe de los axones de las demás de acuerdo a peso sináptico asociado,  $W_{ij}$ , suma todas estas entradas ponderadas y realiza la comparación del resultado con su umbral interno, a esto se le aplica la función de transferencia, en consecuencia una función de tipo no lineal, de tipo escalón y de tipo pseudolineal o sigmoideo

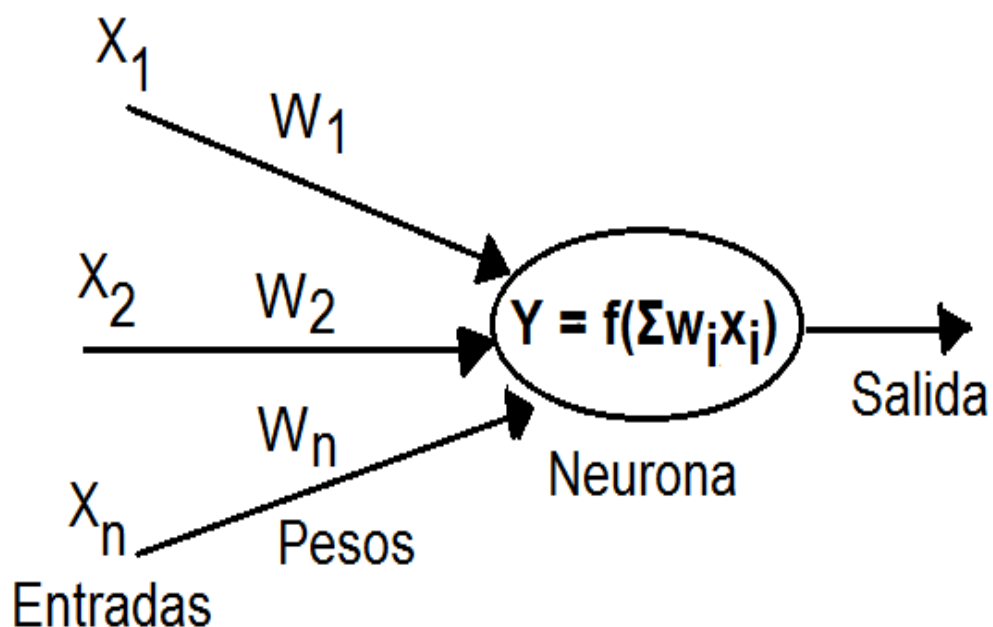


Figura 2. Operación en una neurona artificial

Fuente. (Carranza Hernández, 2014)

### C) Arquitectura

Según Serrano & Bonifacio, (2008, págs. 157,158), refiere que la característica arquitectónica de la RN es la forma en que se constituyen las neuronas. Las diferentes características arquitectónicas se pueden dividir en dos tipos básicos: feed forward, en los que la data fluye siempre hacia adelante, como en el caso de la Figura 3, desde las capas de entrada hacia las capas de salidas, y las recurrentes. Esto genera que la data en ocasiones tenga que fluir hacia atrás, a través de realimentaciones.

Dicho ello, hay varios modelos de neuronas diferentes, según su funcionalidad, el tipo de conocimiento y/o aprendizaje y su estructura arquitectónica.

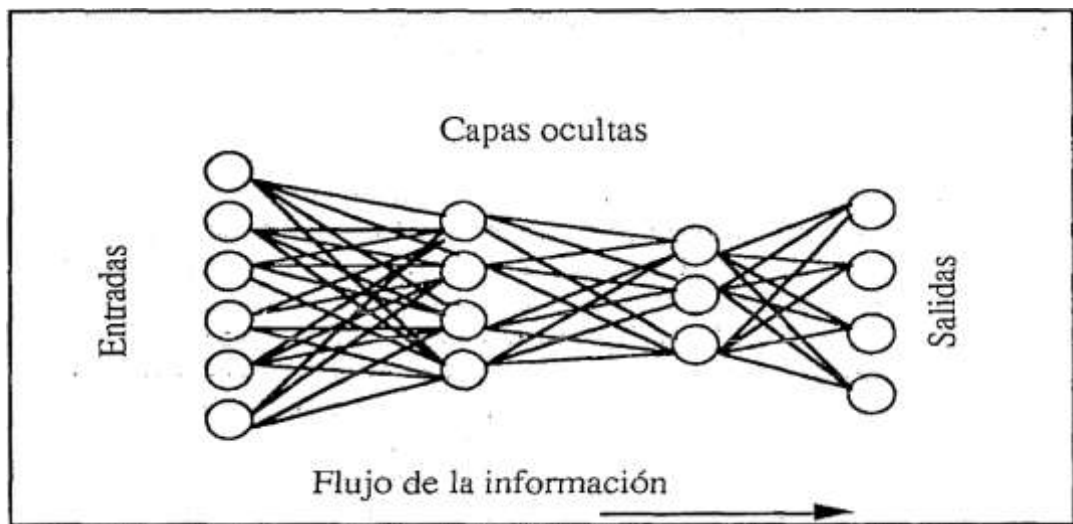


Figura 1: Arquitectura de una neurona artificial

Fuente: (Serrano & Bonifacio, 2008, págs. 157,158)

#### 1.3.2. El Perceptrón

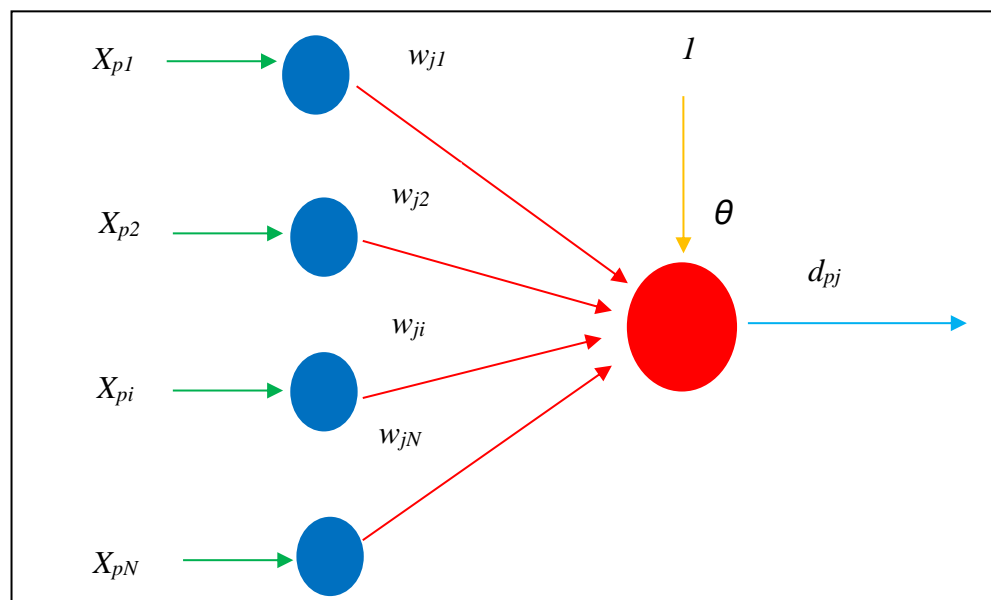
Según Caicedo Bravo & López Sotelo, (2017), El Perceptron resultó ser el primer modelo de ARN presentado en el conocimiento científico presentado en 1958 por el psicólogo Frank Rosenblatt.

La distribución estructural del Perceptrón es simple: Inicialmente, hay varias neuronas encargadas de enfocar los estímulos externos de la red, y la salida

presenta solo una capa de neuronas con funciones de activación binaria o bipolar. La salida de una neurona tiene al menos dos valores posibles (Caicedo Bravo & López Sotelo, 2017).

### A) Arquitectura y Funcionamiento.

Según Caicedo Bravo & López Sotelo, (2017), comenta que el perceptrón es una red monocapa, Esto indica que la red tiene solo una capa de procesamiento como capa de salida porque la capa de entrada original no procesa la señal de alerta y no se considera en la operación. Posee interacción total, ya que las neuronas de las capas de salidas están conectadas al igual de las neuronas de entrada, tal como se señala en la *Figura 4*. Además, se crean unidades de tendencia o umbrales para que el plano de separación no se fije en el origen del espacio n-dimensional en el que se realiza la descomposición lineal. Esto se debe a algunos problemas con la característica de que la muestra es positiva sin límite de corte. La función de activación es gestionada por la neurona de salida en binario [0, 1] o bipolar [1, 1] desplazado.



*Figura 4.* Arquitectura de una neurona artificial.

Fuente: (Caicedo Bravo & López Sotelo, 2017)

## B) Algoritmo de Aprendizaje.

Según Caicedo Bravo & López Sotelo, (2017), afirma que el algoritmo de aprendizaje de un perceptrón es el siguiente:

$N$	Cantidad de neuronas que existen en la capa de entrada.
$X_{pi}$	Es el componente $i$ -ésima de la entrada que corresponde al $p$ -ésimo patrón.
$Net_{pj}$	Entrada total de la neurona $j$ -ésima, que es la única capa de procesamiento del Perceptrón.
$W_{ji}$	Número del peso de la conexión entre la $j$ -ésima neurona de la capa de procesamiento y la $i$ -ésima neurona de la capa que entra.
$\theta_j$	Número del umbral o tendencia para la $j$ -ésima neurona de la capa que procesa.
$Y_{pj}$	Número de salida de la $j$ -ésima neurona de la capa que procesa.
$d_{pj}$	Número de salida deseado para la $j$ -ésima neurona de la capa que procesa.
$e_p$	Cantidad de error para el $p$ -ésimo patrón de aprendizaje.
$\alpha$	Valor de la Tasa de aprendizaje.

### Paso 1

Establecemos aleatoriamente valores iniciales para el peso  $W_{ji}$  y el umbral  $\theta_j$ . Como guía, estos valores deben estar entre -1 y 1.

### Paso 2

Mostramos el vector de entrada de red  $X_p$  y se especifica el vector de salida deseado  $d_{pj}$ .

$$X_p = \{X_{p1}, X_{p2}, \dots, X_{pN}\}$$

### Paso 3

Se calculan los valores netos originados de las entradas para las unidades de la capa de salida.

$$Neta_{pj} = \sum_{i=1}^N w_{ji} X_{pi} \theta_j$$

### Paso 4

Se calcula la salida de la red.

$$y_{pj} = f_j(Neta_{pj})$$

### Paso 5

Se actualizan los pesos de la capa de salida.

$$W_{ji}(t+1) = W_{ji}(t) + \alpha [d_{pj} - y_{pj}] X_{pi}$$

### Paso 6

Se calcula el error del p-ésimo patrón.

$$e_p = \frac{1}{2} \sum_{j=1}^M (d_{pj} - y_{pj})^2$$

### Paso 7

La condición es; Si el error es distinto de cero para cada muestra aprendida, vuelva al paso 2. De lo contrario, continúe.

### Limitaciones

Según Caicedo Bravo & López Sotelo, (2017), presenta las siguientes limitaciones del perceptrón:

- 1) Imposibilidad para solucionar problemas no lineales.
- 2) No es apto para solucionar problemas complejos.

Para una mejor ilustración de estas limitaciones, se señala la siguiente *Figura 5*, donde resulta que el perceptrón es capaz de

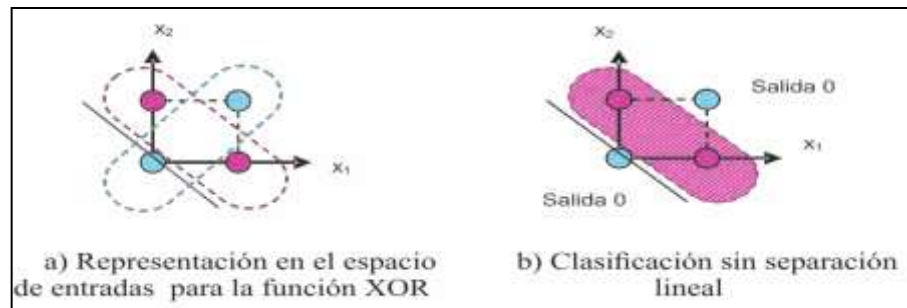
separar linealmente las salidas, por lo que no puede resolver las funciones lógicas XOR descritas en la Tabla 1. En este escenario, las limitaciones de la red neuronal se presentan en un problema relativamente simple, por lo que las sugerencias de Minsky y Pearl, puede entenderse de cierta manera.

Tabla 1.

*Función Lógica XOR*

Patrón	$X_1$	$X_2$	XOR
$X^1$	0	0	0
$X^2$	0	1	1
$X^3$	1	0	1
$X^4$	1	1	0

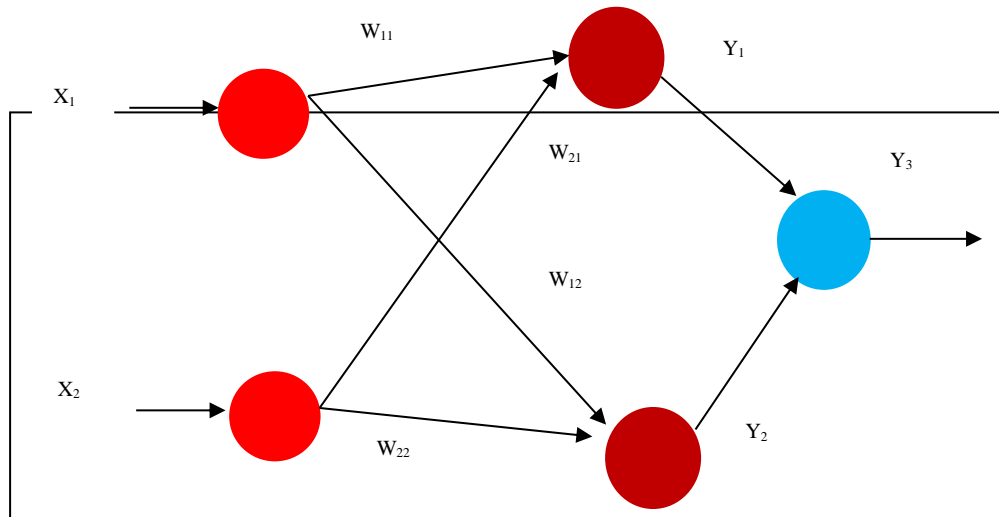
*Nota: Tomado de (Caicedo Bravo & López Sotelo, 2017)*



*Figura 5. Solución operador lógico XOR con un Perceptrón.*

*Fuente. Tomado de (Caicedo Bravo & López Sotelo, 2017)*

Un diseño de perceptrón multinivel con dos neuronas en el nivel de entrada, una capa oculta con dos neuronas y cada neurona en la capa de salida debidamente entrenada puede resolver de manera óptima el problema de la segregación no lineal causado por la función XOR. (Caicedo Bravo & López Sotelo, 2017).



*Figura 6.* Perceptrón multicapa para la solución de función XOR.

Fuente: (Caicedo Bravo & López Sotelo, 2017)

### 1.3.3. Perceptrón Multicapa

Según Serrano & Bonifacio, (2008), sobre el diseño multicapa. La primera capa es únicamente un grupo de neuronas que envían las variables de entrada hacia las capas ocultas, estas son las encargadas del procesamiento, y una capa de salida que muestra el resultado al mundo exterior.

A mitad de los años setenta, Paul Werbos en su tesis para obtener su doctorado propone: el algoritmo de retropropagación prepara el perceptrón multicapa y lo hace aplicable a una variedad de soluciones de preguntas desafiantes. (Caicedo Bravo & López Sotelo, 2017)

De acuerdo a lo señalado en la Figura 7, Excepto por tres niveles de neuronas, puede comprender la estructura del perceptrón multicapa (perceptrón multicapa MLP) en comparación con el perceptrón y la adarina: Entrada, Oculta y Salida.

En ello, se puede formular más de 1 capa oculta, lo cual no es recomendable, dado que se aumenta de manera forzada la complejidad automatizada del algoritmo de aprendizaje, MLP con una capa oculta es suficiente. Para el



MLP, seleccionar las neuronas en la capa original las asocia con todas las neuronas en la siguiente capa oculta y conecta las neuronas en la siguiente capa oculta en la capa oculta, por lo que la asociación del proceso está completa. Todas las neuronas están ubicadas en las capas anterior y posterior.

En la última capa o capa de salida, las neuronas están conectadas a neuronas en la capa previamente encapsulada. Consulte la figura 7 para una mejor comprensión. En general, las unidades de prioridad o umbral simplemente se descomponen para que el origen no oscurezca el área. En el espacio n-dimensional donde ocurre la distribución.

En consecuencia, la función de activación de procesos utilizada por neuronas en la red MLP. Normalmente se utiliza en forma lineal o en grandes conjuntos de instancias de esta firma. (Caicedo Bravo & López Sotelo, 2017)

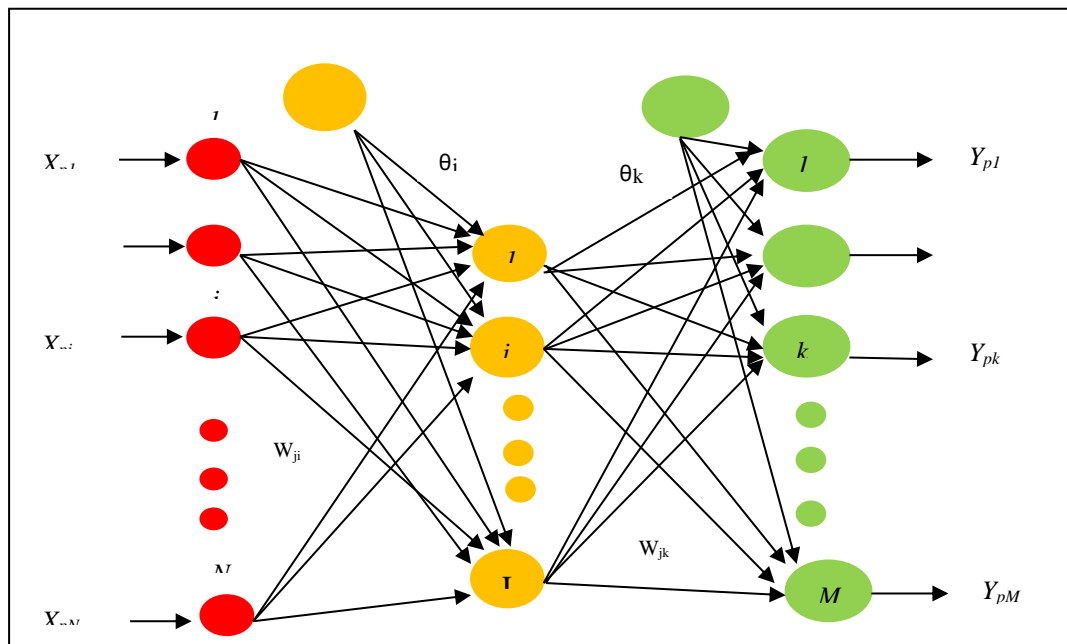


Figura 7. Arquitectura General de un MLP.

Fuente. Extraído de (Caicedo Bravo & López Sotelo, 2017)

## A) Entrenamiento de un MLP

Según Caicedo et al. (2015) y Lara Rosano (s/a de publicación). El algoritmo de aprendizaje de retropropagación se define conceptualmente de la siguiente manera:

- 1) Se selecciona un conjunto bien definido de modelos para resolver un problema. Se forma la red. Esta fase es importante porque depende de la calidad de la información utilizada para entrenar la red.
- 2) Se emplea el índice de entrada de la red y la salida de la neurona oculta se calculan y transmiten hasta que se calcula la salida final de la red.
- 3) Se busca el error entre el valor deseado y la salida de la red.
- 4) Se da el error posterior, es decir, se calcula el error en la capa oculta con base en el error de la capa saliente.
- 5) Se actualizan los pesos de las capas de salidas y ocultas, de tal manera que se debe hacer una aproximación de las variaciones en el peso  $w$  de cada clase. Esto se basa en calcular el margen de error del nivel de salida y estimar el error del nivel compactado en el procedimiento anterior.
- 6) Se verifica el estado de búsqueda del algoritmo ya sea porque el resultado del error obtenido en la salida es bajo por el problema, según estudios o por prácticas. De acuerdo a ello se ha espumado un margen determinado de iteraciones, en este caso, se supone que la solución no convergerá y deberá realizar cambios en el diseño de su red. Alternativamente, necesita aumentar el número máximo de iteraciones realizadas.

- 7) En el escenario del incumplimiento en la condición de la detención, la red se incluirá en el modelo de formación.

### B) Algoritmo Backpropagation.

Según (Caicedo Bravo & López Sotelo) en la *Figura 08* y *Figura 09*, señalan la nomenclatura y el algoritmo de aprendizaje Backpropagation.

$x_p$	Patrón o vector de entrada
$x_{pi}$	Entrada <i>i-ésima</i> del vector de entrada $x_p$
$N$	Dimensión del vector de entrada
$P$	Número de ejemplos, vectores de entrada y salidas diferentes.
$L$	Número de neuronas de la capa oculta: $h$
$M$	Número de neuronas de la capa de salida, dimensión del vector de salida
$w_{ji}^h$	Peso de interconexión entre la neurona <i>i-ésima</i> de la entrada y la <i>j-ésima</i> de la capa oculta.
$\theta_j^h$	Término de tendencia de la neurona <i>j-ésima</i> de la capa oculta.
$Neta_{pj}^h$	Entrada neta de la <i>j-ésima</i> neurona de la capa oculta
$i_{pj}^h$	Salida de la <i>j-ésima</i> neurona de la capa oculta
$f_j^h$	Función de activación de la <i>j-ésima</i> unidad oculta
$w_{kj}^o$	Peso de interconexión entre la <i>j-ésima</i> neurona de la capa oculta y la <i>k-ésima</i> neurona de la capa de salida.
$\theta_k^o$	Término de tendencia de la <i>k-ésima</i> neurona de la capa de salida.
$Neta_{pk}^o$	Entrada neta de la <i>k-ésima</i> neurona de la capa de salida.
$y_{pk}$	Salida de la <i>k-ésima</i> unidad de salida
$f_k^o$	Función de activación de la <i>k-ésima</i> unidad de salida $o \in \mathfrak{R}^m; x \in \mathfrak{R}^n$
$d_{pk}$	Valor de salida deseado para la <i>k-ésima</i> neurona de la capa de salida.
$e_p$	Valor del error para el <i>p-ésimo</i> patrón de aprendizaje.
$\alpha$	Taza o velocidad de aprendizaje
$\delta_{pk}^o$	Término de error para la <i>k-ésima</i> neurona de la capa de salida.
$\delta_{pj}^h$	Término de error para la <i>j-ésima</i> neurona de la capa oculta $h$
$f_j^{\prime h}$	Derivada de la función de activación de la <i>j-ésima</i> neurona de la capa oculta.
$f_k^{\prime o}$	Derivada de la función de activación de la <i>k-ésima</i> neurona de la capa de salida.

*Figura 8.* Notas utilizada por el algoritmo Backpropagation.

Fuente. *Extraído de* (Caicedo Bravo & López Sotelo, 2017).

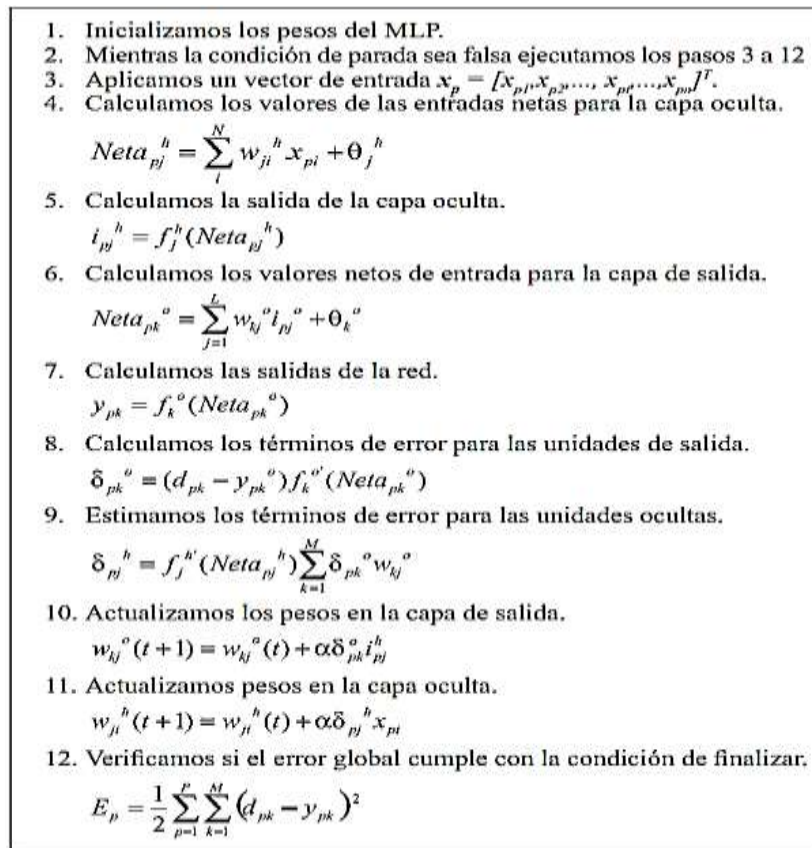


Figura 9. Algoritmo Backpropagation.

Fuente. *Extraído de* (Caicedo Bravo & López Sotelo, 2017)

El Algoritmo backpropagation, es considerada una fase de movimiento donde se presenta, ante la red entrenada, un patrón de entrada y éste se ejecuta a través de las capas sucesivas, de las neuronas hasta obtener una salida y, después, una etapa de aprendizaje donde se alteran los pesos de la red de tal forma que tenga coincidencia a la salida deseada por quien la ejecutó con la salida obtenida por el proceso de la red. (jmmarin)

### 1.3.4. Regresión de Series Temporales

#### A) Concepto.

Según Paggi Straneo, (2003), relata que una serie de temporal está conformado por un conjunto de valores que pueden ser indicados como observaciones tomadas de un determinado sistema a lo largo del tiempo; en los particular, son interesantes las creadas por los sistemas dinámicos, especialmente para las aplicaciones de las

categorías económico-financieras, son las mismas que forman la parte principal de lo que se entrega en el mundo real.

La disposición del pronóstico de una serie temporal va de la mano con factores de aprendizaje que se tenga del sistema proveniente, a la pureza de los datos para su uso y manejo que se haga de ellos.

En esta investigación, la serie proviene de las ventas diarias mensuales gestionadas por el sistema de ventas de la empresa.

## **B) Importancia.**

Según Escovar R, Valdés H, & Zapata C, (2010), la predicción de series de tiempo considera la hipótesis que dado un conjunto reservado de datos con respecto al tiempo y correspondientes a un mismo fenómeno, los valores futuros son dependientes de los valores históricos, es decir, buscando en los valores históricos de una serie de tiempo se puede predecir su comportamiento en el futuro.

Una predicción de manera óptima y precisa en las ventas diarias mensuales, permitirá mantener el stock y la cantidad de productos necesarios para vender.

Muchas series de tiempo tienen características no lineales, por lo que es necesario utilizar técnicas fuera del marco clásico para realizar sus predicciones y obtener modelos más eficientes.

## **C) Modelos**

Según Rios, (2008, pág. 4) describe que se pueden diferenciar dos tipos de modelos de series de tiempo:

- 1) Modelo definido. Estas son extrapolaciones simples y no se refieren al origen o la naturaleza de la aleatoriedad

subyacente a la secuencia. Su relativa simplicidad se asocia a menudo con una precisión reducida. Un ejemplo de modelo determinista es un modelo de media móvil en el que la predicción de una variable se calcula a partir de la media del valor anterior "n".

- 2) Modelos estocásticos: Se basa en una descripción simple del proceso aleatorio subyacente a la secuencia. En pocas palabras, se supone que las series observadas  $Y_1, Y_2, Y_T$  se extraerán de un grupo de variables aleatorias con una distribución común específica difícil de determinar, y se construirá un modelo aproximado para ayudar a generar predicciones. (Rios, 2008).

#### **D) Etapas**

Según Zapata C, Escovar R, & Valdés H, (2010), refiere que la aplicación de Redes Neuronales a la predicción de series de tiempo se debe contemplar los procesos o etapas siguientes:

- 1) Búsqueda de las variables de entrada.
- 2) Preparación del conjunto de datos, consiste en la normalización de los datos.
- 3) Composición de la red, determinación de cada elemento que agrupa la red.
- 4) Definición del algoritmo de entrenamiento y los parámetros de configuración propios.
- 5) Realización de verificación, verificación del proceso de aprendizaje en línea.
- 6) Calculo de los factores.

#### **1.3.5. Metodología Propuesta**

Para la realización de este proyecto "Implementación de un sistema de pronóstico de ventas utilizando redes neuronales artificiales para la

Empresa Cerámicos Lambayeque S.A.C”, se tuvo en cuenta la metodología desarrollada por (Kaastra & Boyd, 1995), el cual cada proceso se muestra en la siguiente figura.

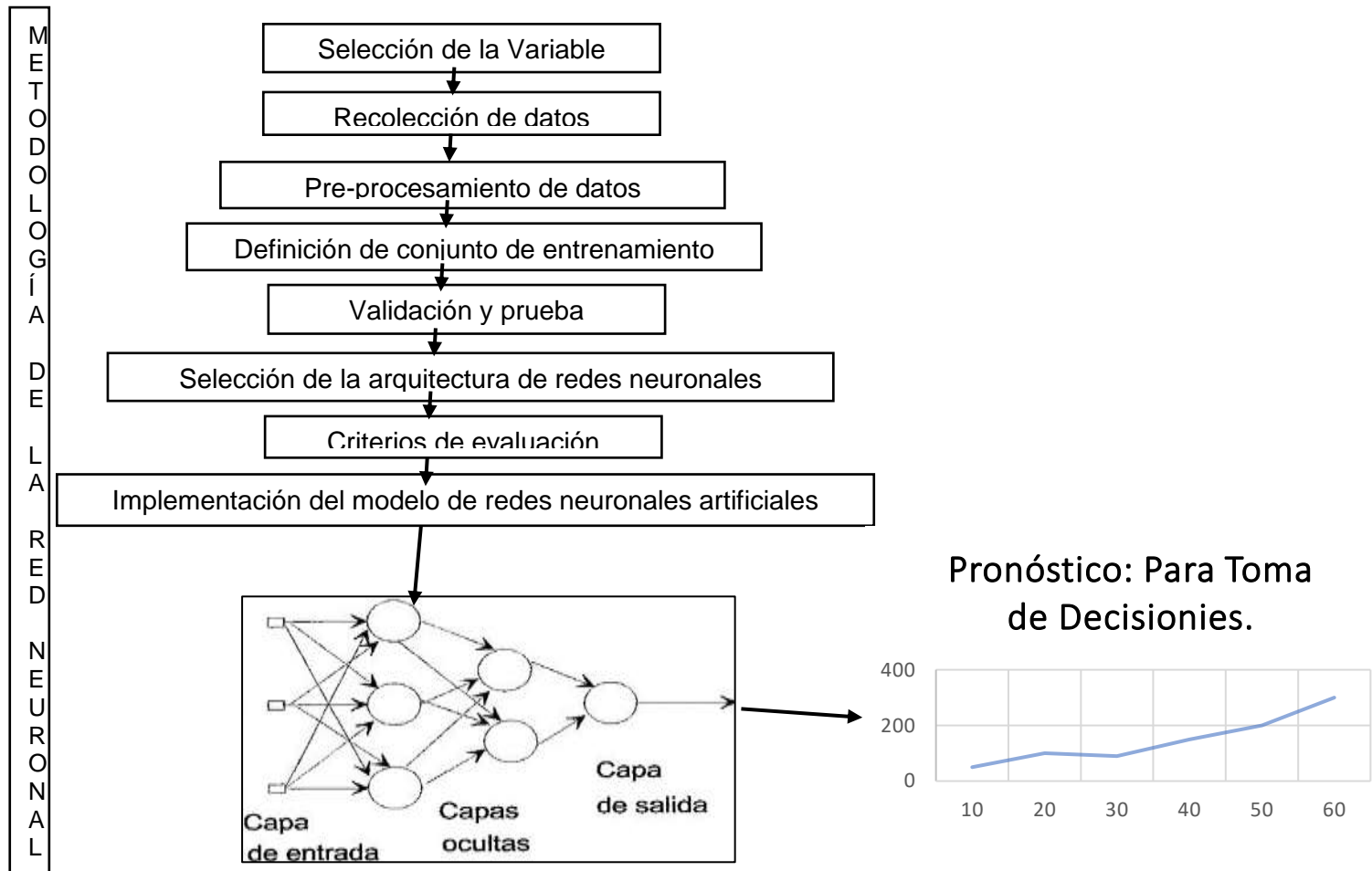


Figura 10. Método para la aplicación de las redes neuronales en el pronóstico de ventas desarrollada por (Kaastra & Boyd, 1995). Fuente. *Elaboración propia*



Los resultados están enfocados a evaluar los indicadores de error de pronóstico. Se puede concluir que la demanda de ventas a medio plazo se puede predecir aplicando tecnologías de inteligencia artificial como las redes neuronales artificiales con arquitectura de perceptrón multicapa.

### **1.3.6. Definición de Términos Básicos**

#### **1.3.6.1. El Pronóstico**

Según Galeana, (2014), en la publicación de su artículo sobre **conceptos Generales de Pronósticos**, señala que se tenga en cuenta que los pronósticos utilizan datos históricos para determinar eventos futuros. Hay varios métodos de predicción, pero un solo buen modelo es raro. Pocas empresas pueden permitirse saltarse el proceso de previsión para anticipar posibles riesgos.

La planificación eficaz depende de la previsión de la demanda de los productos de una empresa. (Galeana, 2014).

#### **1.3.6.2. El Pronóstico de Ventas.**

Según Arturo, (2010), en el artículo **Crece Negocios**, un pronóstico es una estimación de las ventas futuras (desde una perspectiva física o financiera) de uno o más productos durante un período de tiempo.

Esto le permite crear un presupuesto de ventas y, en base a eso, preparar otros presupuestos para producción, compras de consumo o mercancías, requisitos de personal, flujo de caja y más (k, 2010)

La realización de Pronósticos le permite crear un presupuesto de ventas, en base al cual puede crear otros presupuestos como producción, compra de consumibles o mercadería, requisitos de personal, flujo de efectivo y más. (k, 2010)

En otras palabras, al crear un pronóstico, puede saber cuántos productos desea producir, cuántos consumibles y productos necesita, cuántos

empleados necesita, cuánta inversión necesita, etc. De esta forma, logras una gestión empresarial más eficiente y posibilitas la planificación, coordinación y control de actividades y recursos. (k, 2010)

Asimismo, la previsión permite conocer la rentabilidad de un proyecto (restando los costes futuros de las ventas futuras) y, por tanto, la viabilidad del proyecto. Como resultado, la previsión es a menudo uno de los aspectos más importantes de un plan de negocios (k, 2010)

### **1.3.6.3. Red Neuronal.**

Según Diego Calvo, (2017), en su artículo denominado ***Definición de Red Neuronal Artificial***, indica que Las Redes neuronales artificiales (RNA). Es un paradigma de aprendizaje y procesamiento automático inspirado en el funcionamiento del sistema nervioso humano.

Una red neuronal está compuesta por un conjunto de neuronas interconectadas entre sí mediante enlaces.

Cada neurona toma como entradas las salidas de las neuronas de las capas antecesoras, cada una de esas entradas se multiplica por un peso, se agregan los resultados parciales y mediante una función de activación se calcula la salida. Esta salida es a su vez es entrada de la neurona a la que precede (Calvo, 2017).

La unión de todas estas neuronas interconectadas es lo que compone la red neuronal artificial.

Las redes neuronales no son más que redes interconectadas masivamente en paralelo de elementos simples (usualmente adaptativos) y con organización jerárquica, las cuales intentan interactuar con los objetos del mundo real del mismo modo que lo hace el sistema nervioso biológico (Calvo, 2017).

Según J.M Egea Caparrós (1994), refiere que una red neuronal es un método de computación que simula el funcionamiento neuronal durante

el aprendizaje. Uno de los retos de la inteligencia artificial es el de resolver problemas de reconocimiento de patrones, es decir, una de las capacidades en las que la (maquina inteligente» debe imitar al cerebro humano es en la capacidad de asociar un conjunto de rasgos con una idea o concepto

#### **1.3.6.4. Función de Activación.**

Según Gestal Pose, (2010, pág. 6), en su documento llamado **“Introducción a las Redes Neuronales Artificiales”**, señala que Función de activación de neuronas se encarga de asociar la entrada de la neurona con el posterior estado de activación de la neurona.

Hay dos modelos para la función de activación:

- **Modelo Acotados;** El valor de activación de una neurona puede ser cualquier valor en un rango continuo de valores. (Gestal Pose, 2010).
- **Modelo no Acotados;** no existe ningún límite para los valores de activación. (Gestal Pose, 2010)

#### **1.3.6.5. Entrenamiento de la Red Neuronal.**

En lo mencionado por el grupo de Investigación en Planeamiento de Sistemas Eléctricos e Investigación de Operaciones de la Universidad Tecnológica de Pereira (Gomez Rojas, Henao Lopez, & Harold, 2004) y el artículo denominado **“Entrenamiento de una red neuronal artificial usando el algoritmo Simulated Annealing”**, se alude que el proceso de entrenar una RNA, altera el valor de los pesos y bias asociados a cada neurona a fin de que la RNA pueda generar una salida a partir de unos datos señalados.

#### **1.3.6.6. Toma de Decisiones.**

Según Riquelme, (2020), en su artículo **“Toma de decisiones: Concepto de vital importancia en la empresa”** indica, para la sociedad la palabra toma de decisiones es uno de las características de mayor

trascendencia, dado que al momento de elegir hacer algo que pudiera traer consecuencias desfavorables genera confusiones e interrogantes (Riquelme, 2020).

Hay que considerar que al tomar la elección correcta se hace necesario basarse en las experiencias previas, además cada individuo tiene diferentes maneras de afrontar la resolución de interrogantes, de acuerdo a sus conocimientos y a su historia (Riquelme, 2020).

#### **1.3.6.7. Venta.**

La Venta es una relación la cual está estrechamente relacionada con la compra, inclusive, se puede decir que es su contra parte ya que esta consiste en la colocación en el mercado de un determinado producto o servicio con el objetivo de que sea comprado por un consumidor. La dependencia de la venta hacia un mercado prospero la convierte en un proceso en constante movimiento y expansión, ya que la presencia de un producto en el campo de interés marca el impacto de este en el consumidor. (Yirda, 2021)

El término «ventas» define varias actividades necesarias para proveer a un cliente o empresa de un servicio o producto definido a cambio de dinero. Sin embargo, para muchas empresas, las relaciones de esta palabra van mucho más allá. (ES, 2019)

### **1.4. Formulación del Problema.**

¿Cómo implementar una solución que utilice redes neuronales artificiales, para que la empresa Cerámicos Lambayeque SAC, pronostique sus ventas futuras?

### **1.5. Justificación e importancia del estudio.**

#### **1.5.1. Justificación Social.**

La Esta encuesta hace que sea más fácil para las empresas predecir las ventas futuras y es un insumo clave para estas ventas pasadas a través de

un proceso lógico que les permite predecir las ventas. Fecha mensual para el próximo período.

Los resultados de ventas anuales permiten a los altos ejecutivos de ventas tomar decisiones más informadas al ejecutar sus estrategias de ventas, a través de las cuales pueden mejorar o aumentar los ingresos de su organización.

### **1.5.2. Justificación Tecnológica.**

En la actualidad, la tecnología y el conocimiento a nivel de ingeniería e innovación ha mejorado mucho; en este ámbito, las redes neuronales han realizado importantes contribuciones en los campos de la economía, las finanzas y la banca. Principalmente relacionado con el sector bancario, se utiliza para predecir una variedad de prospectos y negocios con poco riesgo crediticio o ventaja financiera.

El proceso de pronóstico tiene mucho sentido, ya que le permite determinar cuántos productos deben estar en stock para cumplir con el pronóstico de ventas mensuales de la aplicación de miles de soles por día.

Dada la revisión de la previsión de ventas de la empresa industrial Cerámicos Lambayeque, podemos ver un porcentaje de los beneficios para el próximo período. Asimismo, permitirá a los altos ejecutivos de las empresas tomar decisiones más amplias.

### **1.5.3. Justificación Tecnológica.**

Con la solución desarrollada, puede aumentar las ganancias y mejorar los costos comerciales, dado que el alcance de la solución que se implementará, permitirá obtener un pronóstico de las ventas donde impacta de manera positiva a los ingresos de la Empresa.

#### **1.5.4. Justificación Académica.**

Lo investigado, contiene información para el conocimiento necesario y aplicación de las redes neuronales en el proceso del pronóstico de las ventas futuras para una empresa del sector productivo y comercial.

Adicional a ello, permite incluir en la práctica los conocimientos que durante el periodo universitario se obtuvieron, también respalda la investigación de nuevas herramientas y métodos utilizados para implementar soluciones utilizando redes neuronales utilizadas para pronosticar las ventas futuras por parte de las empresas de marketing y fabricación.

Por tanto, el presente documento evidencia por el alto impacto de la previsión en la resolución de problemas que se pueden utilizar para crear valor en un entorno empresarial utilizando RN, las cuales conllevan a una evaluación para medir su grado de exactitud y/o efectividad.

Con el uso de las RN, se podrá evaluar y revisar las ventas e ingresos futuros de la empresa comercial, lo que ayuda a la empresa a tomar decisiones efectivas para los empresarios y evitar perder dinero si se toman malas decisiones en la cuenta. Invertir.

#### **1.5.5. Justificación Científica.**

El presente informe, aporta la información y conocimientos necesarios que pueden ser utilizados en realizar pronósticos de ventas efectivos, eficientes y efectivos para nuevos procesos de investigación en tecnología y redes neuronales.

#### **1.5.6. Justificación Institucional.**

Los términos y conceptos utilizados en este informe aplican redes neuronales para desarrollar aplicaciones que ayuden a los usuarios a predecir las ventas futuras de la empresa, la innovación en investigación, la metodología y la tecnología útil como guías para los estudiantes.

### **1.6. Hipótesis.**

El desarrollo de la aplicación de redes neuronales, se combinará con la implementación de un Sistema a la Empresa “CERAMICOS LAMBAYEQUE SAC” para pronosticar sus ventas futuras.

### **1.7. Objetivos.**

#### **1.7.1. Objetivo general.**

Implementar un Sistema Informático utilizando redes neuronales artificiales para pronosticar las ventas en la Empresa “Cerámicos Lambayeque SAC”.

#### **1.7.2. Objetivos específicos.**

- a. Extraer y analizar la información existente de las ventas históricas, hasta dos periodos de la empresa Cerámicos Lambayeque SAC.
- b. Identificar el modelo de la Red Neuronal Artificial (RNA), para desarrollar el sistema.
- c. Implementar un algoritmo ideal para el pronóstico de las Ventas.
- d. Identificar los datos de entrada para pronosticar las ventas.
- e. Evaluar resultados.

## **II. MATERIAL Y MÉTODO**

### **2.1. Tipo y Diseño de Investigación.**

#### **2.1.1. Tipo de Investigación.**

Según la clasificación general de investigación científica (Alvites Castillo, 2000) la presente investigación se orienta al tipo tecnológica, propositiva.

Según (Hernández Sampieri, Fernández Collado, & Pilar Baptista, 1991), es de tipo tecnológica, debido a que se aplica los conocimientos de Redes Neuronales en la implementación de una herramienta software en C#.NET 2013, para generar el margen de error se utiliza la Red Neuronal Perceptrón que servirá para pronosticar las ventas. También pertenece al tipo proposicional porque da resultados basados en las tolerancias que pueden ocurrir al implementar la solución.

### **2.1.2. Diseño de Investigación.**

Como parte del desarrollo de este estudio, se ajustó para utilizar el diseño de campo, ya que es un tipo de diseño adaptativo para el tratamiento a nivel de ingeniería.

Para ello, se eligió un diseño estadístico que proporciona medidas que determinan el valor de la variable. Además, incluye estudios cuantitativos o evaluativos de datos agregados.

## **2.2. Población y muestra.**

### **2.2.1. Población.**

La población en estudio está determinada, de acuerdo a lo siguiente:

a) Ventas realizadas por la empresa de los periodos 2016 y 2017.

Esta información se encuentra en la base de datos de la empresa, con la cual se llevará a cabo el pronóstico.

### **2.2.2. Muestra.**

Según (Dieterich, 2001), para determinar la muestra, se considera población. La muestra consiste en obtener una porción representativa y adecuada de la población. Solo una parte representativa o un subconjunto de la población puede hacer una generalización válida. Se estima que el Formulario es específico si cumple con las características clave de la población (ventas) en relación con las variables esperadas a examinar.

a) Ventas diarias mensuales facturadas por la empresa en el año 2016 y 2017.

## **2.3. Variables, Operacionalización.**

### **2.3.1. Variable Independiente.**

Redes neuronales.

### **2.3.2. Variable Dependiente.**

Sistema de pronóstico de ventas.



### 2.3.3. Operacionalización.

Tabla 2.  
*Redes neuronales.*

V. I.	Dimensión	Indicador	Ítem	Técnicas e Instrumentos de Validación
Red Neuronal	Grado de Confiabilidad	Aproximación del pronóstico de ventas	GC = $\frac{\sum \text{Venta mensual pronosticada} * 100}{\sum \text{Venta mensual obtenida}}$	Observación
	Error de pronóstico acumulativo	Sumatoria de errores de predicción	$EPA = \sum_{t=1}^N e_t ;$ e <sub>t</sub> =Error del Pronóstico n = Cantidad de tiempos observados.	
	Precisión	Porcentaje de acierto	$P = \frac{VP}{VP + FP}$	

*Nota:* Elaboración propia.

Tabla 3.  
*Sistema de pronóstico de ventas.*

V. D.	Dimensión	Indicador	Ítem	Técnicas e Instrumentos de Validación
Sistema de pronóstico	Tiempo de respuesta	Tiempo utilizado para mostrar resultados	$TR$ = <i>Hora de inicio</i> – <i>Hora de finalización</i>	Observación

Consumo de memoria	Memoria consumida durante un proceso	RAM = $\frac{\text{Total de memoria}}{\left(\frac{\sum \text{Uso de memoria por proceso}}{\sum \text{N}^\circ \text{ de procesos ejecutados}}\right)}$
	Consumo de CPU durante un proceso	CPU CPU = $\frac{\text{Total de CPU}}{\left(\frac{\sum \text{Uso de CPU por proceso}}{\sum \text{N}^\circ \text{ de procesos ejecutados}}\right)}$

*Nota:* Elaboración propia.

## **2.4. Técnicas e instrumentos de recolección de datos, validez y confiabilidad.**

### **2.4.1. Métodos.**

En la elaboración de proyectos de Ingeniería de Sistemas, podemos indicar que existen métodos inductivos y deductivos. El más adecuado para la presente investigación es el inductivo, dado que induce que la información a utilizar en la muestra, son las ventas del periodo 2016 y 2017, por consiguiente, se considera deductivo porque se obtendrá como resultado el pronóstico de las ventas, el cual será evaluado para comprobar su validez.

### **2.4.2. Técnicas.**

Fundamentalmente las técnicas son relacionadas, según lo siguiente:

#### **2.4.2.1. Recolección de información.**

Esta técnica le permite seleccionar teorías, conceptos, conocimientos científicos, métodos y procedimientos para explicar y explicar objetivamente su investigación en su estado actual.

Se basaron principalmente en observación, entrevista, y documentación bibliográfica especializada en redes neuronales, contabilidad, economía e ingeniería del software. Esto nos permite la implementación de la herramienta software en C#.NET.

#### 2.4.2.2. Planteamiento de soluciones.

Son las propias técnicas de las redes neuronales implementadas en la aplicación tecnológica de la investigación, para tal efecto se optó por el lenguaje de programación C#.NET.

#### 2.4.2.3. Análisis de resultados.

Esta técnica fue utilizada en la agrupación, tabulación y ponderación de datos de la información, a través de una computadora, para posteriormente interpretarlos utilizando una hoja de cálculo del programa MS Excel, y cuyos datos se han obtenido utilizando el sistema de predicción.

#### 2.4.2.4. Descripción de instrumentos utilizados.

Tabla 4.

*Detalle instrumento-Técnica*

TÉCNICA	INSTRUMENTO
Recolección de Información	a) Ventas diarias mensuales desde el año 2016 al 2017. b) Bibliografía (libros, revistas, tesis, artículos científicos, ponencias, etc.).
Coordinaciones Observación	Coordinaciones con el Jefe de Ventas Observación de EEFF
Análisis de Sistemas Transaccionales	Herramientas de extracción de información (Se aplicó en el desarrollo del Software).
Planteamiento de Soluciones	Red neuronal perceptrón multicapa y regresión de serie temporal, utilizados para la predicción de las ventas y la rentabilidad a través del Lenguaje de programación C#.NET.
Análisis de los resultados	Resultados de tipo estadísticos
Prueba del sistema	Cronometro.

*Nota:* Elaboración Propia.

## **2.5. Procedimiento de análisis de datos.**

En la presente investigación y el marco de la recolección de datos, se hicieron uso de técnicas que se señalan en la Tabla 4.

El proceso de información (datos) resultante sigue el proceso de agregación definido previamente. Una vez que se recuperan los datos finales, se completarán en la tabla diseñada o en el desglose.

### **2.5.1. Plan de análisis estadístico de datos.**

Utilizamos herramientas informáticas de Microsoft Excel para tabular, pesar e interpretar todos los datos obtenidos. Esto nos permitió evaluar y validar el comportamiento de las variables y presentar el análisis estadístico en forma de tablas y gráficos.

## **2.6. Criterios éticos.**

Al respecto, el código “Dental” del Instituto Tecnológico del Perú se utiliza en el capítulo III, “Delitos contra la ética profesional y sanciones” y en el capítulo II, “De relaciones públicas”, en el artículo 106. Se indica:

Los ingenieros deben tener cuidado de no perjudicar sus intereses al enviar comentarios sobre su trabajo, intereses o técnicas, buscando solidificar el conocimiento general de la ingeniería y el servicio comunitario, y actuando con creciente convicción. (Colegio de Ingenieros, 2011, pág. 14).

En atención a ello, la acción de un ingeniero deberá:

- A. Someterse a una responsabilidad global en su trabajo.
- B. Proceder, de tal manera que su objetivo sea el servicio a la sociedad; promover propuestas de soluciones, A nivel público, social y ambiental.
- C. Certificar un Software y/o Sistema, en el marco del cumplimiento para el bien social.
- D. Sumar esfuerzos para contribuir con soluciones de asuntos importantes de interés social causados por el Software, su instalación, estabilización, soporte y documentación.

Ofrecer de manera voluntaria la asistencia técnica por una buena razón y contribuir en la educación pública relacionada con esta profesión.

## **2.7. Criterios de Rigor Científico.**

### **2.7.1. Credibilidad.**

Se basa en una encuesta para aumentar la probabilidad de que se acepte la información contenida en esta encuesta. Para ello se utilizan los siguientes: observación continua, análisis documental, etc.

### **2.7.2. Transferencia.**

Se refiere a la capacidad de aplicar los resultados de la investigación a otros temas o contextos. El proyecto requirió lo siguiente: Recopilar mucha información y elaborar sobre el proceso involucrado en el desarrollo de estudios analíticos y recolección de datos.

### **2.7.3. Dependencia.**

Este es el proceso de repetir los resultados con el mismo sujeto y el mismo agente. Para ello se realiza lo siguiente: Definir y describir técnicas de observación artificial. Esto establece una pista de auditoría a través de informes de investigación, revisiones de documentos y más.

### **2.7.4. Confiabilidad.**

Está asociada a la garantía de los descubrimientos de esta investigación; es decir, que no estén sesgados por incitaciones, por intereses y perspectivas del investigador.

Se ejecuta a través de una descripción de baja confianza, que busca el análisis de un documento mediante la observación.

## **III. RESULTADOS.**

### **3.1. Resultados en Tablas y Figuras.**

Para verificar el Grado de confianza y veracidad de la Red Neuronal – Tabla 5 – se hizo uso de las ventas totales por mes obtenida y pronosticada mostrado en la Tabla 10 que se encuentra en el anexo.

Tabla 5.

*Grado de confiabilidad de la red neuronal desde Enero a Julio del año 2018*

MES	VENTA DIARIA S/.	VENTA DIARIA PRONOSTICADA CON LA RED S/.	GRADO DE CONF. DE LA RNA %
ENERO	670675	668705	99,71
FEBRERO	678351	645902	95,22
MARZO	708787	663617	93,63
ABRIL	701295	659217	94,00
MAYO	701760	646789	92,17
JUNIO	692871	650631	93,90
JULIO	704171	650239	92,34

*Nota:* Elaboración propia.

En la Figura 11 se presenta una comparación total mensual obtenida – barra azul –, pronosticada – barra anaranjada – y grado de confiabilidad de la red.

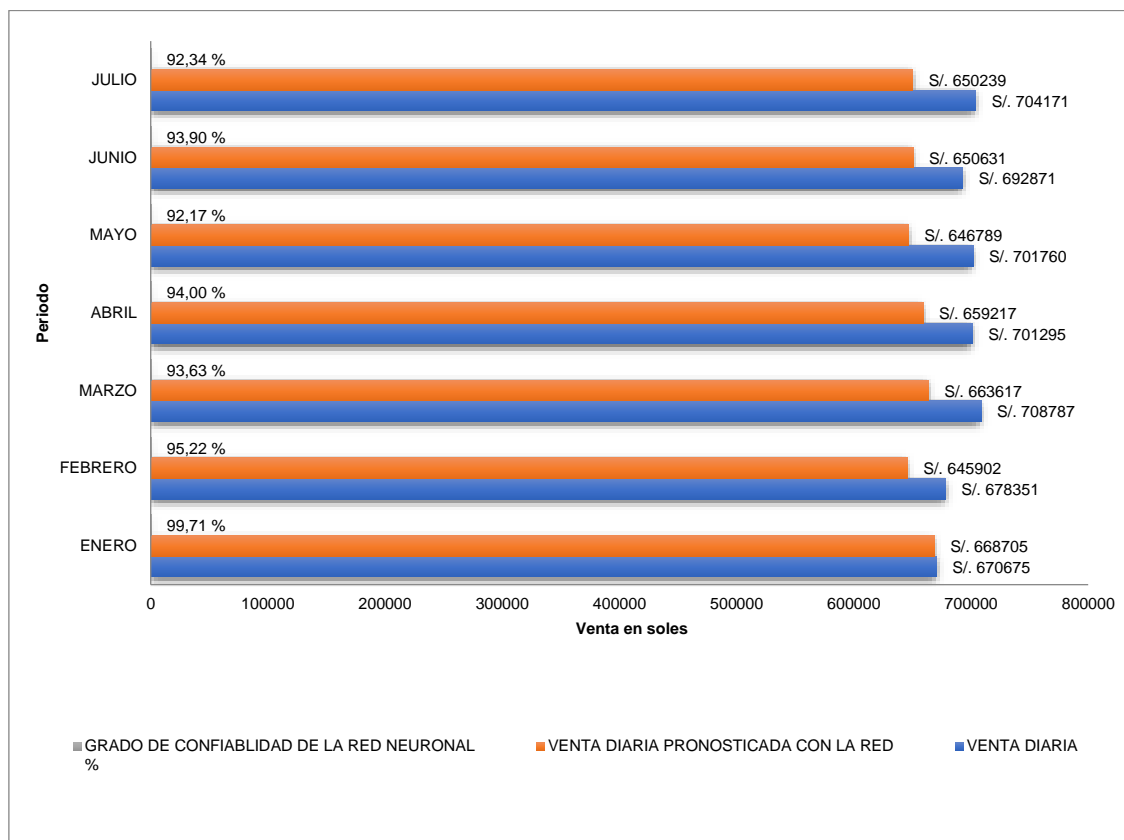


Figura 2: Venta mensual, pronosticada mensual con la red neuronal en nuevos soles y grado de confianza en % de la RNA desde Enero a Julio del año 2018.

Fuente: *Elaboración propia.*

En la estimación del tiempo consumido por la solución creada (Sistema) y en el modelo de la información de las ventas anunciadas diarias por la red neuronal en pantalla, se ha utilizado el tiempo de inicio en que se inicia la ejecución de la red neuronal y el tiempo final que representa la terminación de la ejecución de la RN. En la Figura 12 se presenta el tiempo total consumido por la red neuronal – Final tiempo de series –.

```

file:///C:/Users/jose/Desktop/clase maestra/MANUEL BENITES SERNAQUE/red/r...
[188] 21,026 22,296 22,311 21,984 16,239
[189] 21,569 22,310 22,239 22,313 16,260
[190] 15,883 22,312 21,307 22,118 22,219
[191] 15,876 22,309 21,910 22,108 22,169
[192] 15,860 22,311 22,068 22,276 22,235
[193] 15,838 22,311 21,982 22,164 22,128
[194] 15,858 22,312 21,942 21,730 22,302
[195] 15,857 22,311 21,956 21,862 22,087
[196] 15,887 22,306 21,896 22,027 21,800
[197] 15,910 22,311 21,879 21,875 22,311
[198] 15,902 22,311 22,046 22,032 22,297
[199] 15,889 22,311 22,129 22,210 0,000

Creando una 4-12-1 red neuronal
Fijamos Maxima Epoca = 10000
Fijamos Ratio de Aprendizaje = 0,01

Empezamos Entrenamiento
Epoca = 2000 Error = 5,7287
Epoca = 4000 Error = 5,6131
Epoca = 6000 Error = 6,2387
Epoca = 8000 Error = 5,6008
Ver

Final Modelo Red Neuronal Pesos y Bias:
0,186279999734510 0,3262775404468970 0,2913514548804190 0,1880227717612930
0,8639642481336170 0,7095345885377330 0,2133279922365600 0,2269696246713160
0,2685010080239730 0,1707687537961770 0,0936211764161662 0,1802609558502770
0,3635131382162710 0,2219959201994390 0,0936211764161662 0,1802609558502770
0,1803650600301230 0,0948695502926404 0,7196230380080590 0,6403611712805450
0,1171836078387290 0,1291512193563710 0,1982682542805600 0,1244825038286020
0,1603607219988670 0,0928763374099764 0,1796252229433680 0,2083828869071500
0,0973111712094879 0,1796252229433680 0,2083828869071500 0,09990162256351677
0,7071355982866500 0,5115785997417200 0,1839278787633320 0,0892548028159518
0,1220523156674710 0,1350971943981010 0,1159813023899340 0,2042494734545950
0,1938437292919450 0,1306670702902790 0,1184696542653900 0,1426680207677110
0,2064789975978130 0,1184696542653900 0,15223657797896140 0,2167255405841610
0,8335005065284810 0,5223657797896140 0,0952180312722786 0,1515156266937980
0,1852569518054610 0,0952180312722786 0,0079440307670574 0,0248509138315597
0,0040006884261995 0,0079440307670574 0,0045867197246290 0,0078860773515621
0,0084534076721111 0,0045867197246290 0,0062067459133920 0,0089479769528995
0,0050701133989831 0,0062067459133920 0,008379603592104 1,7035711436986900
0,0089479769528995 0,008379603592104 1,6677656861399500 1,7256088643744500
1,7034963138971100 1,6677656861399500 1,5870438520840800 1,6992042308750800
1,6430302913563200 1,5870438520840800 1,6770440150472500 2,0702020931634100
1,7256601652895200 1,6770440150472500 2,0702020931634100

Precision Modelo (+/- 30) en Datos de Entrenamiento = 0,2250
Prediccion Pasajera por Mes:
2215

Final Tiempo Series
22,1473482697952 4

Final Tiempo Series
00:24:37.2842552
Presione Enter Para Visualizar Grafico.

```

Figura 3: Tiempo consumido por el sistema en el pronóstico de las ventas.

Fuente: *Elaboración propia.*

El consumo de memoria realizado por el proceso encargado de ejecutar el sistema, se calculó de la siguiente manera: Se tomó el tamaño de la memoria privada – Megabytes – asignado por el sistema operativo a dicho proceso y luego se dividió sucesivamente entre 1024 para obtener la cantidad de Gigabytes de memoria consumida. En la Figura 13 mostramos el consumo de la memoria.

```

file:///C:/Users/jose/Desktop/clase maestra/MANUEL BENITES SERNAQUE/red/r... -  x
[192] 0,000 0,000 0,000 0,000 0,000
[193] 0,000 0,000 0,000 0,000 0,000
[194] 0,000 0,000 0,000 0,000 0,000
[195] 0,000 0,000 0,000 0,000 0,000
[196] 0,000 0,000 0,000 0,000 0,000
[197] 0,000 0,000 0,000 0,000 0,000
[198] 0,000 0,000 0,000 0,000 0,000
[199] 0,000 0,000 0,000 0,000 0,000

Creando una 4-12-1 red neuronal
Fijamos Maxima Epoca = 10000
Fijamos Ratio de Aprendizaje = 0,01
Empezamos Entrenamiento
Epoca = 2000 Error = 4,4462
Epoca = 4000 Error = 5,6793
Epoca = 6000 Error = 4,2880
Epoca = 8000 Error = 5,0402
Ver

Final Modelo Red Neuronal Pesos y Bias:
0,2242220534394100 1,7550660873234400 -0,0115206028036467 -0,3409227491973760
1,7301370368435700 0,0142850925232853 -0,4874479731019420 0,0081190069142861
0,6180814699035800 0,0349705740321986
0,2288193055998150 0,0065825569400972 0,4502824116400700 1,9013608679851300
0,2261934094940140 -0,1580430222449830 1,8767299656051600 0,2268059637477410
-0,2986463755829220 0,1972628716720270
0,8082736265248600 0,1980202555212910 0,4561607301027950 0,1958658361562480
0,1839480191906930 1,8576904835244600 -0,0507298381389055 -0,3657383202776830
1,8311373476196600 -0,0249994975874111
-0,5077569233343220 -0,0531401176997630 0,5774082710289770 -0,0033716586064677
0,1893364517114010 -0,0535510535248363 0,5128486075889970 2,2826530998775600
0,3096454474041140 0,0179246577420522
2,2536453335464600 0,2925134490284990 -0,1201973042538530 0,2762898697820460
0,8444954781529400 0,2525166955271530 0,5181924221416690 0,2745455992460320
-0,1179016731077700 -0,0351493445685667
0,1344014932266950 0,0052846826141091 -0,0332588639670334 -0,1308271810514050
0,0522603178903202 -0,1263084952286360 -0,1042256157279230 -0,1208138326972850
-0,1238358704683350 -0,1238961602994180
2,2345151286980300 2,1384466020888100 2,2709256187794700 0,6925853417009100
2,1262889123256900 2,2618648377004200 1,5768478898180400 2,2342110526990900
2,2564008303110700 2,2106909873431100
2,2708462632226000 2,2201286535526000 2,4186634117207700

Precision Modelo (+/- 30) en Datos de Entrenamiento = 0,3600

Prediccion Pasajera por Mes:
2192

Final Tiempo Series
21,9213354397033 4

TIEMPO FINAL EJECUCION DE LA RED NEURONAL
00:00:26.4078524

MEMORIA UTILIZADA POR EL PROCESO ACTUAL PARA EJECUTAR LA RED NEURONAL ES:
0,0165023803710938 GB

Presione Enter Para Visualizar Grafico.

```

Figura 4: Consumo de memoria por el sistema.

Fuente: *Elaboración propia.*



En el Error de Pronóstico Acumulativo, las ventas diarias pronosticadas mediante la red neuronal, son comparadas con las ventas diarias reales y con las ventas pronosticadas por la empresa, las cuales se detallan en la Tabla 6.

En la tabla 6 se muestran los errores de predicción generados por la empresa y la red neuronal propuesta. El error de pronóstico de la empresa se calcula de la siguiente manera:

$$\text{Error pronóstico empresa} = \text{Pronóstico empresa} - \text{Venta diaria}$$

El error de la red se obtuvo aplicando la fórmula matemática siguiente:

$$\text{Error de red} = \text{Venta diaria pronosticada con la red} - \text{Venta diaria}$$

Tabla 6.

*Error de pronóstico de la empresa y de la red neuronal desde enero a julio del año 2018.*

MES	DIA	VENTA DIARIA	PRONOSTICO EMPRESA	ERROR PRONOSTICO EMPRESA	VENTA DIARIA PRONOSTICADA CON LA RED	ERROR DE LA RED
	1	20361	22676	2315	22380	2019
	2	23291	22677	-614	22294	-997
	3	23327	22678	-649	22349	-978
	4	22868	22679	-189	22349	-519
	5	23630	22679	-951	21921	-1709
	6	24428	22680	-1748	22448	-1980
	7	22375	22681	306	22437	62
	8	20396	22682	2286	21881	1485
	9	20138	22682	2544	22452	2314
	10	23943	22683	-1260	22193	-1750
	11	24940	22684	-2256	22595	-2345
	12	20401	22684	2283	22444	2043
	13	22777	22685	-92	22498	-279
	14	24489	22686	-1803	22454	-2035

ENERO	15	22926	22687	-239	22264	-662
	16	22978	22687	-291	22168	-810
	17	23756	22688	-1068	22513	-1243
	18	21116	22689	1573	22538	1422
	19	21211	22690	1479	22357	1146
	20	21368	22690	1322	21662	294
	21	24958	22691	-2267	22391	-2567
	22	20706	22692	1986	22432	1726
	23	20980	22693	1713	21747	767
	24	22770	22693	-77	22553	-217
	25	22964	22694	-270	21435	-1529
	26	22869	22695	-174	22256	-613
	27	20122	22695	2573	22425	2303
	28	20166	22696	2530	22489	2323
	29	22832	22697	-135	22433	-399
30	21589	22698	1109	22347	758	
<b>TOTAL</b>		<b>670675</b>	<b>680611</b>		<b>668705</b>	
	31	20422	22698	2276	22301	1879
	32	22488	22699	211	22439	-49
	33	24400	22700	-1700	22318	-2082
	34	22568	22701	133	22477	-91
	35	23924	22701	-1223	22446	-1478
	36	20341	22702	2361	22405	2064
	37	21628	22703	1075	22441	813
	38	24559	22704	-1855	21725	-2834
	39	24677	22704	-1973	22521	-2156
	40	24109	22705	-1404	21936	-2173
	41	24849	22706	-2143	22191	-2658
	42	21067	22706	1639	22442	1375
	43	20997	22707	1710	21832	835
	44	23223	22708	-515	22546	-677
	45	23102	22709	-393	21905	-1197
FEBRERO	46	21370	22709	1339	22448	1078
	47	20290	22710	2420	22446	2156
	48	22317	22711	394	22511	194

49	22271	22712	441	22584	313	
50	23736	22712	-1024	21800	-1936	
51	20299	22713	2414	22090	1791	
52	21170	22714	1544	22413	1243	
53	24903	22714	-2189	22447	-2456	
54	22313	22715	402	22463	150	
55	23426	22716	-710	22187	-1239	
56	22281	22717	436	22278	-3	
57	22977	22717	-260	22440	-537	
58	23389	22718	-671	21887	-1502	
59	23404	22719	-685	22520	-884	
60	21851	22720	869	21764	-87	
<b>TOTAL</b>	<b>678351</b>	<b>681270</b>		<b>645902</b>		
61	22884	22720	-164	22375	-509	
62	24206	22721	-1485	22504	-1702	
63	22934	22722	-212	21565	-1369	
64	23439	22723	-716	22510	-929	
65	23598	22723	-875	19735	-3863	
66	23136	22724	-412	22298	-838	
67	23872	22725	-1147	22442	-1430	
68	22647	22725	78	22501	-146	
69	24592	22726	-1866	22409	-2183	
70	21205	22727	1522	21571	366	
71	23256	22728	-528	22164	-1092	
72	24896	22728	-2168	22441	-2455	
73	23835	22729	-1106	22387	-1448	
74	20789	22730	1941	22350	1561	
MARZO	75	20899	22731	1832	22160	1261
	76	24126	22731	-1395	22030	-2096
	77	20428	22732	2304	22138	1710
	78	24657	22733	-1924	22146	-2511
	79	22265	22734	469	22214	-51
	80	24194	22734	-1460	21399	-2795
	81	21754	22735	981	21940	186
	82	21302	22736	1434	22433	1131

83	22193	22736	543	22490	297	
84	21737	22737	1000	22473	736	
85	20373	22738	2365	21801	1428	
86	24986	22739	-2247	22338	-2648	
87	22978	22739	-239	22443	-535	
88	23222	22740	-482	22501	-721	
89	23758	22741	-1017	22546	-1212	
90	22029	22742	713	21345	-684	
91	22597	22742	145	22343	-254	
<b>TOTAL</b>	<b>708787</b>	<b>704671</b>		<b>663617</b>		
92	20674	22743	2069	22433	1759	
93	24507	22744	-1763	22512	-1995	
94	24580	22745	-1835	22506	-2074	
95	22271	22745	474	22036	-235	
96	23499	22746	-753	22139	-1360	
97	24877	22747	-2130	22509	-2368	
98	24439	22747	-1692	22355	-2084	
99	20867	22748	1881	22460	1593	
100	20923	22749	1826	21907	984	
101	23133	22750	-383	22460	-673	
102	23654	22750	-904	22391	-1263	
103	21238	22751	1513	22388	1150	
104	24022	22752	-1270	22336	-1686	
105	24288	22753	-1535	22493	-1795	
106	20419	22753	2334	22327	1908	
107	24326	22754	-1572	22508	-1818	
ABRIL	108	24962	22755	-2207	22460	-2502
	109	22382	22755	373	22500	118
	110	20866	22756	1890	22473	1607
	111	23869	22757	-1112	22205	-1664
	112	22389	22758	369	22506	117
	113	20398	22758	2360	21824	1426
	114	22071	22759	688	22509	438
	115	22085	22760	675	16130	-5955
	116	22715	22761	46	22389	-326

	117	20298	22761	2463	22507	2209
	118	22409	22762	353	22498	89
	119	21752	22763	1011	22516	764
	120	23102	22764	-338	16476	-6626
	121	20098	22764	2666	22512	2414
	122	24182	22765	-1417	22385	-1797
	<b>TOTAL</b>	<b>701295</b>	<b>705375</b>		<b>659217</b>	
	123	23572	22766	-806	21876	-1696
	124	21864	22766	902	22466	602
	125	23574	22767	-807	15804	-7770
	126	23073	22768	-305	22511	-562
	127	23812	22691	-1121	22507	-1305
	128	24345	22769	-1576	22003	-2342
	129	23262	22770	-492	21909	-1353
	130	24809	22771	-2038	16006	-8803
	131	21440	22772	1332	22508	1068
	132	20083	22772	2689	22504	2421
	133	23466	22773	-693	22461	-1005
	134	21427	22774	1347	22493	1066
	135	22471	22775	304	16266	-6205
	136	22133	22775	642	15898	-6235
MAYO	137	21473	22776	1303	22511	1038
	138	23080	22777	-303	22120	-960
	139	22093	22777	684	21972	-121
	140	20194	22778	2584	22435	2241
	141	21806	22779	973	15879	-5927
	142	20862	22780	1918	22511	1649
	143	24787	22780	-2007	22158	-2629
	144	24606	22781	-1825	22453	-2153
	145	23823	22782	-1041	22259	-1564
	146	20860	22783	1923	15902	-4958
	147	23596	22783	-813	22511	-1085
	148	22458	22784	326	21612	-846
	149	24833	22785	-2048	22350	-2483
	150	23335	22786	-549	22406	-929

	151	23245	22786	-459	15890	-7355
	152	20867	22787	1920	22511	1644
	153	20511	22788	2277	22097	1586
<b>TOTAL</b>		<b>701760</b>		<b>706001</b>	<b>646789</b>	
	154	24914	22788	-2126	22263	-2651
	155	21808	22789	981	22364	556
	156	20989	22790	1801	15896	-5093
	157	23822	22791	-1031	22510	-1312
	158	23197	22791	-406	22073	-1124
	159	24462	22792	-1670	22510	-1952
	160	24685	22793	-1892	22290	-2395
	161	20067	22794	2727	15880	-4187
	162	22101	22794	693	21955	-146
	163	20484	22795	2311	21658	1174
	164	22311	22796	485	21137	-1174
	165	21684	22797	1113	21935	251
	166	23735	22797	-938	15907	-7828
	167	20321	22798	2477	22401	2080
	168	22158	22799	641	22085	-73
JUNIO	169	22399	22799	400	22385	-14
	170	22182	22800	618	22208	26
	171	22133	22801	668	15920	-6213
	172	22172	22802	630	22481	309
	173	23215	22802	-413	22143	-1072
	174	20619	22803	2184	22504	1885
	175	21287	22804	1517	22210	923
	176	20327	22805	2478	15920	-4407
	177	22825	22805	-20	22489	-336
	178	22081	22806	725	22124	43
	179	21146	22807	1661	22045	899
	180	22955	22807	-148	22322	-633
	181	22613	22808	195	15922	-6691
	182	23200	22809	-391	22489	-711
	183	22856	22810	-46	22125	-731
	184	24123	22810	-1313	22480	-1643

TOTAL	692871		706782	650631		
	185	21234	22811	1577	22167	933
	186	22205	22812	607	15936	-6269
	187	21852	22813	961	22024	172
	188	23300	22813	-487	22450	-850
	189	24718	22814	-1904	22132	-2586
	190	22888	22815	-73	22284	-604
	191	24513	22816	-1697	15936	-8577
	192	21602	22816	1214	22489	887
	193	21831	22817	986	22076	245
	194	22747	22818	71	22315	-432
	195	21831	22818	987	22318	487
	196	20003	22819	2816	15945	-4058
	197	24958	22820	-2138	22464	-2494
	198	21115	22821	1706	22016	901
	199	23424	22821	-603	22446	-978
	200	24349	22822	-1527	22035	-2314
JULIO	201	21404	22823	1419	15971	-5433
	202	22389	22824	435	22306	-83
	203	24659	22824	-1835	21530	-3129
	204	21367	22825	1458	22293	926
	205	23831	22826	-1005	21878	-1953
	206	23341	22827	-514	15989	-7352
	207	22137	22827	690	22481	344
	208	21802	22828	1026	22071	269
	209	21469	22829	1360	22083	614
	210	24102	22829	-1273	22464	-1638
	211	20128	22830	2702	15971	-4157
	212	24741	22831	-1910	22322	-2419
	213	24752	22832	-1920	21991	-2761
	214	24546	22832	-1714	21878	-2668
	215	20933	22833	1900	21978	1045
TOTAL		<b>704171</b>	<b>707486</b>		<b>650239</b>	

Nota: *Elaboración propia*

Tabla 7.

*Datos de verdaderos positivos, falsos positivos y precisión del pronóstico realizado por la empresa y por la red neuronal.*

DIA	VENTA REAL	PRONOSTICO EMPRESA	V. P.	F. P.	PRONOSTICO		
					RED NEURONAL	V. P.	F. P.
1	20361	22676	20361	2315	22380	20361	2019
2	23291	22677	22677	0	22294	22294	0
3	23327	22678	22678	0	22349	22349	0
4	22868	22679	22679	0	22349	22349	0
5	23630	22679	22679	0	21921	21921	0
6	24428	22680	22680	0	22448	22448	0
7	22375	22681	22375	306	22437	22375	62
8	20396	22682	20396	2286	21881	20396	1485
9	20138	22682	20138	2544	22452	20138	2314
10	23943	22683	22683	0	22193	22193	0
11	24940	22684	22684	0	22595	22595	0
12	20401	22684	20401	2283	22444	20401	2043
13	22777	22685	22685	0	22498	22498	0
14	24489	22686	22686	0	22454	22454	0
15	22926	22687	22687	0	22264	22264	0
16	22978	22687	22687	0	22168	22168	0
17	23756	22688	22688	0	22513	22513	0
18	21116	22689	21116	1573	22538	21116	1422
19	21211	22690	21211	1479	22357	21211	1146
20	21368	22690	21368	1322	21662	21368	294
21	24958	22691	22691	0	22391	22391	0
22	20706	22692	20706	1986	22432	20706	1726
23	20980	22693	20980	1713	21747	20980	767
24	22770	22693	22693	0	22553	22553	0
25	22964	22694	22694	0	21435	21435	0
26	22869	22695	22695	0	22256	22256	0
27	20122	22695	20122	2573	22425	20122	2303
28	20166	22696	20166	2530	22489	20166	2323
29	22832	22697	22697	0	22433	22433	0



30	21589	22698	21589	1109	22347	21589	758
31	20422	22698	20422	2276	22301	20422	1879
32	22488	22699	22488	211	22439	22439	0
33	24400	22700	22700	0	22318	22318	0
34	22568	22701	22568	133	22477	22477	0
35	23924	22701	22701	0	22446	22446	0
36	20341	22702	20341	2361	22405	20341	2064
37	21628	22703	21628	1075	22441	21628	813
38	24559	22704	22704	0	21725	21725	0
39	24677	22704	22704	0	22521	22521	0
40	24109	22705	22705	0	21936	21936	0
41	24849	22706	22706	0	22191	22191	0
42	21067	22706	21067	1639	22442	21067	1375
43	20997	22707	20997	1710	21832	20997	835
44	23223	22708	22708	0	22546	22546	0
45	23102	22709	22709	0	21905	21905	0
46		22709	0	22709	22448	0	22448
47	20290	22710	20290	2420	22446	20290	2156
48	22317	22711	22317	394	22511	22317	194
49	22271	22712	22271	441	22584	22271	313
50	23736	22712	22712	0	21800	21800	0
51	20299	22713	20299	2414	22090	20299	1791
52	21170	22714	21170	1544	22413	21170	1243
53	24903	22714	22714	0	22447	22447	0
54	22313	22715	22313	402	22463	22313	150
55	23426	22716	22716	0	22187	22187	0
56	22281	22717	22281	436	22278	22278	0
57	22977	22717	22717	0	22440	22440	0
58	23389	22718	22718	0	21887	21887	0
59	23404	22719	22719	0	22520	22520	0
60	21851	22720	21851	869	21764	21764	0
61	22884	22720	22720	0	22375	22375	0
62	24206	22721	22721	0	22504	22504	0
63	22934	22722	22722	0	21565	21565	0
64	23439	22723	22723	0	22510	22510	0
65	23598	22723	22723	0	19735	19735	0

66	23136	22724	22724	0	22298	22298	0
67	23872	22725	22725	0	22442	22442	0
68	22647	22725	22647	78	22501	22501	0
69	24592	22726	22726	0	22409	22409	0
70	21205	22727	21205	1522	21571	21205	366
71	23256	22728	22728	0	22164	22164	0
72	24896	22728	22728	0	22441	22441	0
73	23835	22729	22729	0	22387	22387	0
74	20789	22730	20789	1941	22350	20789	1561
75	23912	22731	22731	0	22160	22160	0
76	24126	22731	22731	0	22030	22030	0
77	20428	22732	20428	2304	22138	20428	1710
78	24657	22733	22733	0	22146	22146	0
79	22265	22734	22265	469	22214	22214	0
80	24194	22734	22734	0	21399	21399	0
81	21754	22735	21754	981	21940	21754	186
82	21302	22736	21302	1434	22433	21302	1131
83	22193	22736	22193	543	22490	22193	297
84	21737	22737	21737	1000	22473	21737	736
85	20373	22738	20373	2365	21801	20373	1428
86	24986	22739	22739	0	22338	22338	0
87	22978	22739	22739	0	22443	22443	0
88	23222	22740	22740	0	22501	22501	0
89	23758	22741	22741	0	22546	22546	0
90	22029	22742	22029	713	21345	21345	0
91	22597	22742	22597	145	22343	22343	0
92	20674	22743	20674	2069	22433	20674	1759
93	24507	22744	22744	0	22512	22512	0
94	24580	22745	22745	0	22506	22506	0
95	22271	22745	22271	474	22036	22036	0
96	23499	22746	22746	0	22139	22139	0
97	24877	22747	22747	0	22509	22509	0
98	24439	22747	22747	0	22355	22355	0
99	20867	22748	20867	1881	22460	20867	1593
100	20923	22749	20923	1826	21907	20923	984
101	23133	22750	22750	0	22460	22460	0

102	23654	22750	22750	0	22391	22391	0
103	21238	22751	21238	1513	22388	21238	1150
104	24022	22752	22752	0	22336	22336	0
105	24288	22753	22753	0	22493	22493	0
106	20419	22753	20419	2334	22327	20419	1908
107	24326	22754	22754	0	22508	22508	0
108	24962	22755	22755	0	22460	22460	0
109	22382	22755	22382	373	22500	22382	118
110	20866	22756	20866	1890	22473	20866	1607
111	23869	22757	22757	0	22205	22205	0
112	22389	22758	22389	369	22506	22389	117
113	20398	22758	20398	2360	21824	20398	1426
114	22071	22759	22071	688	22509	22071	438
115	22085	22760	22085	675	16130	16130	0
116	22715	22761	22715	46	22389	22389	0
117	20298	22761	20298	2463	22507	20298	2209
118	22409	22762	22409	353	22498	22409	89
119	21752	22763	21752	1011	22516	21752	764
120	23102	22764	22764	0	16476	16476	0
121	20098	22764	20098	2666	22512	20098	2414
122	24182	22765	22765	0	22385	22385	0
123	23572	22766	22766	0	21876	21876	0
124	21864	22766	21864	902	22466	21864	602
125	23574	22767	22767	0	15804	15804	0
126	23073	22768	22768	0	22511	22511	0
127	23812	22691	22691	0	22507	22507	0
128	24345	22769	22769	0	22003	22003	0
129	23262	22770	22770	0	21909	21909	0
130	24809	22771	22771	0	16006	16006	0
131	21440	22772	21440	1332	22508	21440	1068
132	20083	22772	20083	2689	22504	20083	2421
133	23466	22773	22773	0	22461	22461	0
134	21427	22774	21427	1347	22493	21427	1066
135	22471	22775	22471	304	16266	16266	0
136	22133	22775	22133	642	15898	15898	0
137	21473	22776	21473	1303	22511	21473	1038

138	23080	22777	22777	0	22120	22120	0
139	22093	22777	22093	684	21972	21972	0
140	20194	22778	20194	2584	22435	20194	2241
141	21806	22779	21806	973	15879	15879	0
142	20862	22780	20862	1918	22511	20862	1649
143	24787	22780	22780	0	22158	22158	0
144	24606	22781	22781	0	22453	22453	0
145	23823	22782	22782	0	22259	22259	0
146	20860	22783	20860	1923	15902	15902	0
147	23596	22783	22783	0	22511	22511	0
148	22458	22784	22458	326	21612	21612	0
149	24833	22785	22785	0	22350	22350	0
150	23335	22786	22786	0	22406	22406	0
151	23245	22786	22786	0	15890	15890	0
152	20867	22787	20867	1920	22511	20867	1644
153	20511	22788	20511	2277	22097	20511	1586
154	24914	22788	22788	0	22263	22263	0
155	21808	22789	21808	981	22364	21808	556
156	20989	22790	20989	1801	15896	15896	0
157	23822	22791	22791	0	22510	22510	0
158	23197	22791	22791	0	22073	22073	0
159	24462	22792	22792	0	22510	22510	0
160	24685	22793	22793	0	22290	22290	0
161	20067	22794	20067	2727	15880	15880	0
162	22101	22794	22101	693	21955	21955	0
163	20484	22795	20484	2311	21658	20484	1174
164	22311	22796	22311	485	21137	21137	0
165	21684	22797	21684	1113	21935	21684	251
166	23735	22797	22797	0	15907	15907	0
167	20321	22798	20321	2477	22401	20321	2080
168	22158	22799	22158	641	22085	22085	0
169	22399	22799	22399	400	22385	22385	0
170	22182	22800	22182	618	22208	22182	26
171	22133	22801	22133	668	15920	15920	0
172	22172	22802	22172	630	22481	22172	309
173	23215	22802	22802	0	22143	22143	0

174	20619	22803	20619	2184	22504	20619	1885
175	21287	22804	21287	1517	22210	21287	923
176	20327	22805	20327	2478	15920	15920	0
177	22825	22805	22805	0	22489	22489	0
178	22081	22806	22081	725	22124	22081	43
179	21146	22807	21146	1661	22045	21146	899
180	22955	22807	22807	0	22322	22322	0
181	22613	22808	22613	195	15922	15922	0
182	23200	22809	22809	0	22489	22489	0
183	22856	22810	22810	0	22125	22125	0
184	24123	22810	22810	0	22480	22480	0
185	21234	22811	21234	1577	22167	21234	933
186	22205	22812	22205	607	15936	15936	0
187	21852	22813	21852	961	22024	21852	172
188	23300	22813	22813	0	22450	22450	0
189	24718	22814	22814	0	22132	22132	0
190	22888	22815	22815	0	22284	22284	0
191	24513	22816	22816	0	15936	15936	0
192	21602	22816	21602	1214	22489	21602	887
193	21831	22817	21831	986	22076	21831	245
194	22747	22818	22747	71	22315	22315	0
195	21831	22818	21831	987	22318	21831	487
196	20003	22819	20003	2816	15945	15945	0
197	24958	22820	22820	0	22464	22464	0
198	21115	22821	21115	1706	22016	21115	901
199	23424	22821	22821	0	22446	22446	0
200	24349	22822	22822	0	22035	22035	0
201	21404	22823	21404	1419	15971	15971	0
202	22389	22824	22389	435	22306	22306	0
203	24659	22824	22824	0	21530	21530	0
204	21367	22825	21367	1458	22293	21367	926
205	23831	22826	22826	0	21878	21878	0
206	23341	22827	22827	0	15989	15989	0
207	22137	22827	22137	690	22481	22137	344
208	21802	22828	21802	1026	22071	21802	269
209	21469	22829	21469	1360	22083	21469	614

210	24102	22829	22829	0	22464	22464	0
211	20128	22830	20128	2702	15971	15971	0
212	24741	22831	22831	0	22322	22322	0
213	24752	22832	22832	0	21991	21991	0
214	24546	22832	22832	0	21878	21878	0
215	20933	22833	20933	1900	21978	20933	1045
PRECISIÓN		96.54%			97.71%		

Nota: *Elaboración propia*

### 3.2. Discusión de resultados.

Aquí analizaremos e interpretaremos los resultados del sistema de información o la solución desarrollada por la empresa industrial "Cerámicos Lambayeque SAC". El análisis tiene como objetivo evaluar los indicadores descritos en el Capítulo III. Ellos son:

- Grado de Confiabilidad de la Red Neuronal.
- Tiempo Consumido por el Sistema.
- Consumo de memoria por el sistema.
- Error de Pronóstico Acumulativo.

**Para la evaluación del grado de confiabilidad de la red neuronal**, se analizaron los datos de este indicador, la Venta Diaria Real, con el pronóstico de la Venta Diaria, utilizando RNA.

En la Tabla 6 y Figura 11 podemos ver que en el mes de Enero la unidad logró en la venta diaria real el importe de S/. 670,675 soles y la venta diaria pronosticada con la red neuronal obtuvo el importe de S/. 668,705 soles. El nivel de confianza RNA de este mes es del 99,71%. Esto significa que el RN está cerca de las ventas diarias reales, ya que la diferencia en el enfoque de las ventas diarias reales es solo del 0,29%.

La empresa en el mes de Mayo alcanzó a la venta diaria real en un importe de S/. 701,760 soles, y la venta diaria pronosticada con la RN se obtuvo un

importe de S/. 646,789 soles. El grado de confianza de la RN para este mes es de 92,17 %, esto revela que la RN trató de aproximarse más a la venta diaria real, mostrando la diferencia de tan solo en el 7,83 % en el acercamiento a la venta diaria real.

Los resultados obtenidos en el pronóstico aplicando redes neuronales artificiales-RNA mejoran la toma de decisiones en el área de compras y ventas de la empresa.

En la Figura 11 se puede ver de manera más clara e intuitiva que la previsión de enero a julio de 2018 con la Red Neural Artificial (RNA) aplicada está mucho más cerca del valor real de las ventas diarias totales de la empresa

El Tiempo Consumido por el Sistema se ha definido de la siguiente manera:

Tiempo de inicio

DateTime ti;

ti = DateTime.Now;

Inicio de ejecución de la red neuronal dentro de un bucle de repetición finito para predecir la ventas diarias desde el mes de Enero a Julio del 2018.

Tiempo final

DateTime tf;

tf = DateTime.Now;

Finalización de la ejecución de la red neuronal y presentación de las ventas pronosticadas en pantalla.

Tiempo acumulado

DateTime tf;

tf = DateTime.Now;

Tiempo que utiliza la RN para pronosticar las ventas diarias mensuales, este tiempo incrementa a 24 min. con 37 seg., Esto significa que el RN usa las ventas encontradas en la clasificación de la matriz para predecir las ventas

futuras, las coloca en la disposición de la matriz de ventas utilizada en el próximo pronóstico y repite el ciclo en este período hasta que se completa el pronóstico completo. El tiempo utilizado para ejecutar el RN depende del componente (procesador), etc. que estemos utilizando.

El Consumo de Memoria por el Sistema se realizó de la siguiente manera:

```
Process proc=Process.GetCurrentProcess ();
```

Se obtiene el proceso asignado por el sistema operativo para ejecutar el sistema.

```
Long TamañoMemoriaPrivada = proc.PrivateMemorySize64;
```

Se obtuvo la memoria privada consumida por el proceso. Esta memoria la convertimos de Megabytes a Gigabytes.

```
MemoriaConsumida = convertMBaGB(TamañoMemoriaPrivada)
```

```
private static double convertMBaGB(long MB)
```

```
{
```

```
    return (MB / (double)1024 / (double)1024 / (double)1024);
```

```
}
```

La memoria que utiliza el sistema para el proceso de tiempo de ejecución de RN es de 0,0165328 gigabytes. Esta memoria es el espacio de memoria total acumulado que ocupan los objetos (int, double, etc.) utilizados para desarrollar y codificar la red neuronal. -. La programación de RN debe optimizarse para reducir el consumo de memoria.

**En el Error de Pronóstico Acumulativo - Tabla 6 -**, al aplicar RN, puede ver los resultados obtenidos en el pronóstico. También mejora los resultados obtenidos con el método de mínimos cuadrados utilizado por las empresas, que es un modelo de pronóstico tradicional. La Tabla 6 muestra claramente que el pronóstico para el período 2018 (enero-julio) con la red neuronal



aplicada está mucho más cerca del valor real que el pronóstico de la compañía.

En la Tabla 6, puede ver qué tan grande es el error en las predicciones comerciales, pero qué tan pequeño es el error en las predicciones de redes neuronales.

### **3.3. Aporte práctico.**

#### **3.3.1. Objetivo.**

El objetivo de este informe de investigación fue el desarrollo de un sistema automatizado aplicando redes neuronales, para realizar el pronóstico de ventas en la empresa Industrial Cerámicos Lambayeque SAC.

Teniendo en cuenta los datos de ventas diarias (de los últimos 2 años), para pronosticar las ventas de los meses siguientes. Para predecir las ventas diarias, se debe realizar un análisis de regresión de series de tiempo utilizando información de desplazamiento histórica combinada con una red neuronal.

En efecto se muestra el esquema de la red neuronal que se aplicó para el proceso del pronóstico de las ventas.

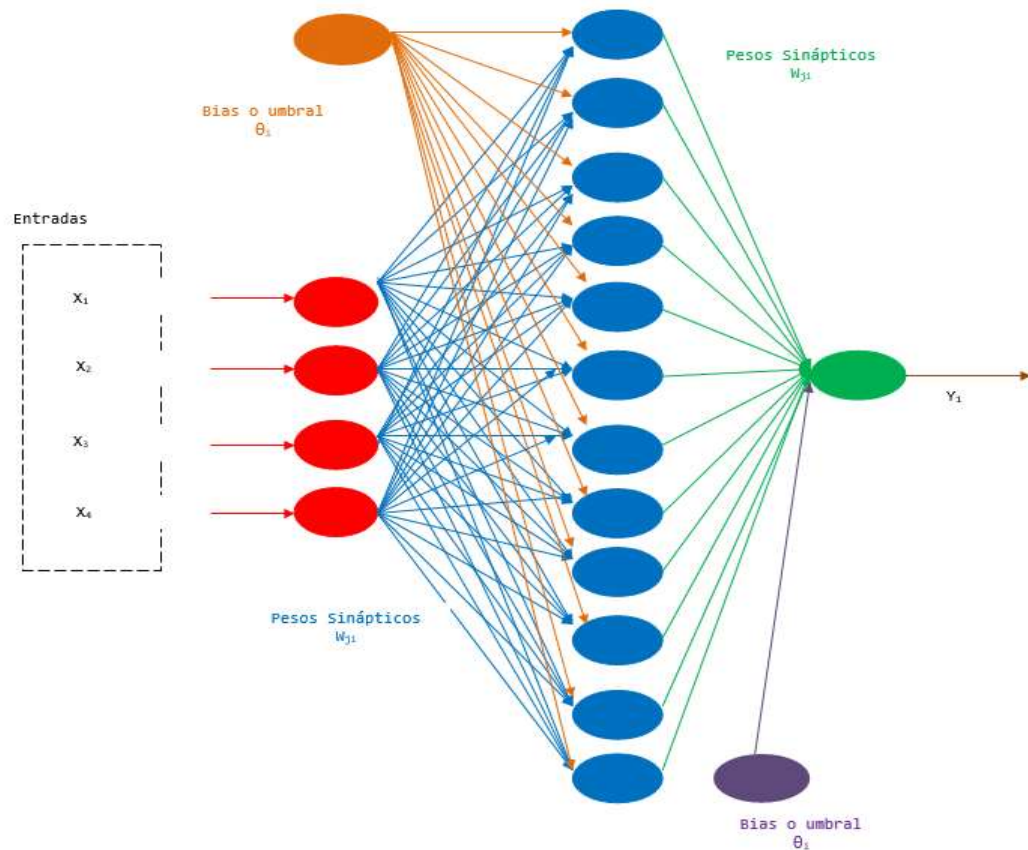


Figura 14. Esquema de la Red Neuronal para la solución del pronóstico de las ventas. Fuente. *Elaboración propia.*

En la Figura 14, se puede observar que existen 4 neuronas de ingreso o entrada, 12 neuronas encapsuladas u ocultas y una sola neurona que sale, la cual se analizará para el pronóstico de las ventas futuras.

Adicional a ello podemos apreciar que cada neurona de entrada se relaciona con una neurona oculta, generando 48 pesos sinápticos, así mismo existe dos (02) neuronas de tipo umbral encontrándose en la dirección de entrada y la otra en dirección de salida, que se relaciona con cada neurona oculta (entrada-salida), generando 13 pesos sinápticos, y cada neurona oculta genera una sola salida atrayendo así 12 pesos sinápticos más.

En conclusión se tiene 4 neuronas de entrada, 12 neuronas ocultas, 73 pesos sinápticos y una neurona de salida.

### 3.3.2. Desarrollo del módulo de interfaz.

Para programar la aplicación se utilizó Microsoft Visual C# 2013, se empezó por crear una “Blank Solution” y se le asignó el nombre de “rn”, después creamos una carpeta a la cual llamamos “aplicación” en esta carpeta procedemos a crear un nuevo proyecto y lo llamamos “red neuronal”, luego se procedió a crear una nueva aplicación de consola C# y la llamamos “PredecirVentasMensual”. En la figura se observa cómo quedaría la aplicación.

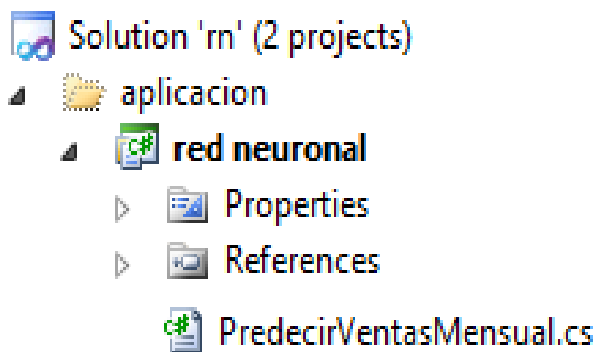


Figura 5: Solución de la Red Neuronal Predictiva.

Fuente: *Elaboración Propia*

### 3.3.3. Lectura de la codificación del sistema.

Así mismos, se detallan paso a paso la codificación del sistema para el funcionamiento y el proceso del pronóstico de las ventas.

#### - PASO N°1: Definimos un Almacén de Ventas

Almacenamos las ventas diarias mensuales en un arreglo bidimensional de 200 filas, en cada fila se almacena un arreglo con 5 columnas de ventas. Las ventas están almacenadas hasta la fila 143, estas ventas servirán para el entrenamiento y pronóstico que debe realizar la red neuronal. En la fila 144 hasta la fila 199 se almacenara las ventas pronosticadas. La función me retorna una matriz bidimensional con todas las ventas almacenadas.

```

staticdouble[ ][ ] ObtenerDatosVenta()
{
double[ ][ ] VentaMes = newdouble[200][ ];
//primer año
//Aquí se ingresara la matriz de las ventas mensuales en el periodo
2017, las mismas que son //extraídas de la información de la empresa.

//VENTAS POR DIA 1ER MES
VentaMes[0] = newdouble[] { 20784, 20308, 22495, 23666, 22949 };
VentaMes[1] = newdouble[] { 23527, 22121, 21198, 22972, 24070 };
VentaMes[2] = newdouble[] { 20970, 22829, 23985, 23324, 23537 };
VentaMes[3] = newdouble[] { 24210, 20348, 22531, 22060, 20132 };
VentaMes[4] = newdouble[] { 20627, 24004, 22551, 24529, 23218 };
VentaMes[5] = newdouble[] { 24469, 20859, 22327, 20735, 20316 };
//VENTAS POR DIA 2DO MES
VentaMes[6] = newdouble[] { 20864, 24994, 23013, 25288, 23665 };
VentaMes[7] = newdouble[] { 24783, 21141, 23918, 24326, 20715 };
VentaMes[8] = newdouble[] { 23977, 23078, 21049, 20641, 25060 };
VentaMes[9] = newdouble[] { 21155, 22253, 22526, 22051, 21058 };
VentaMes[10] = newdouble[] { 24464, 22002, 21119, 20612, 23132 };

VentaMes[11] = newdouble[] { 20841, 21684, 25206, 22779, 22263 };
//VENTAS POR DIA 3ER MES
VentaMes[12] = newdouble[] { 21793, 24110,23809, 23036, 23263 };
VentaMes[13] = newdouble[] { 21828, 24179,23932, 20928, 25026 };
VentaMes[14] = newdouble[] { 20441, 21375, 24392, 24949, 24087 };
VentaMes[15] = newdouble[] { 21345, 24639,24735, 23567, 23057 };
VentaMes[16] = newdouble[] { 20598, 20289,20499, 25092, 24312 };
VentaMes[17] = newdouble[] { 22816, 20964,24467, 22242, 24199 };
//VENTAS POR DIA 4TO MES
VentaMes[18] = newdouble[] { 23399, 23688,22874, 25384, 22736 };
VentaMes[19] = newdouble[] { 20551, 21190,21020, 22216, 25246 };
VentaMes[20] = newdouble[] { 23429, 24879,20154, 23999, 24905 };

```

VentaMes[21] = newdouble[] { 23273, 24360,24165, 24339, 24571 };  
 VentaMes[22] = newdouble[] { 25182, 20671,25075, 24367, 21890 };  
 VentaMes[23] = newdouble[] { 22550, 22144,24189, 24689, 20469 };  
 //VENTAS POR DIA 5TO MES  
 VentaMes[24] = newdouble[] { 22364, 21688,24252, 20718, 22076 };  
 VentaMes[25] = newdouble[] { 21943, 20900,24741, 20205, 21996 };  
 VentaMes[26] = newdouble[] { 21750, 21072,20694, 25033, 20844 };  
 VentaMes[27] = newdouble[] { 23941, 20888,22433, 21663, 23584 };  
 VentaMes[28] = newdouble[] { 20231, 20737,25088, 21371, 23592 };  
 VentaMes[29] = newdouble[] { 23372, 21487,24015, 21045, 20854 };  
 //VENTAS POR DIA 6TO MES  
 VentaMes[30] = newdouble[] { 23545, 23873,22393, 20609, 20463 };  
 VentaMes[31] = newdouble[] { 23508, 23761,20601, 24885, 22008 };  
 VentaMes[32] = newdouble[] { 21640, 24052,23221, 22930, 22334 };  
 VentaMes[33] = newdouble[] { 21604, 23164,21967, 24747, 24837 };  
 VentaMes[34] = newdouble[] { 21204, 22345,20796, 20605, 22955 };  
 VentaMes[35] = newdouble[] { 21911, 20676,23686, 24795, 23114 };  
 //VENTAS POR DIA 7MO MES  
 VentaMes[36] = newdouble[] { 20161, 21949,20647, 20790, 20707 };  
 VentaMes[37] = newdouble[] { 23340, 25240,24822, 21592, 23711 };  
 VentaMes[38] = newdouble[] { 20717, 24230,24858, 24714, 22643 };  
 VentaMes[39] = newdouble[] { 23589, 22009,20150, 22399, 20325 };  
 VentaMes[40] = newdouble[] { 20687, 25313,22951, 20880, 22113 };  
 VentaMes[41] = newdouble[] { 20171, 23696,24726, 21583, 23589 };  
 //VENTAS POR DIA 8TAVO MES  
 VentaMes[42] = newdouble[] { 21611, 23223,20250, 20847, 20702 };  
 VentaMes[43] = newdouble[] { 20016, 24650,20947, 20067, 21782 };  
 VentaMes[44] = newdouble[] { 22688, 22839,25100, 20739, 22429 };  
 VentaMes[45] = newdouble[] { 24528, 20158,23485, 24946, 20399 };  
 VentaMes[46] = newdouble[] { 24449, 22190,24146, 20042, 21057 };  
 VentaMes[47] = newdouble[] { 22873, 23719,22529, 20228, 25269 };  
 //VENTAS POR DIA 9VENO MES  
 VentaMes[48] = newdouble[] { 21929, 24491,22696,23933,24676 };

```

VentaMes[49] = newdouble[] { 22827, 24523,21267,23400,25378 };
VentaMes[50] = newdouble[] { 20638, 20814,24371,22085,22588 };
VentaMes[51] = newdouble[] { 25125, 25307,22718,24269,21019 };
VentaMes[52] = newdouble[] { 23016, 23205,23891,24984,23018 };
VentaMes[53] = newdouble[] { 21788, 22097,24323,25151,23656 };
//VENTAS POR DIA 10MO MES
VentaMes[54] = newdouble[] { 23752, 20270,20098,24424,24661 };
VentaMes[55] = newdouble[] { 25233, 25022,21056,22556,21503 };
VentaMes[56] = newdouble[] { 22137, 24389,20683,21522,24075 };
VentaMes[57] = newdouble[] { 22729, 24283,20513,20654,24054 };
VentaMes[58] = newdouble[] { 24932, 23770,21347,22953,21884 };
VentaMes[59] = newdouble[] { 20344, 20018,21535,25364,21798 };
//VENTAS POR DIA 11AVO MES
VentaMes[60] = newdouble[] { 21546, 20580,21941,23650,20934 };
VentaMes[61] = newdouble[] { 20034, 21878,25387,23047,20544 };
VentaMes[62] = newdouble[] { 23091, 23525,24643,22390,20566 };
VentaMes[63] = newdouble[] { 22741, 23240,21788,22671,20712 };
VentaMes[64] = newdouble[] { 21450, 23933,23996,21444,22300 };
VentaMes[65] = newdouble[] { 25305, 24421,21395,20679,25375 };
//VENTAS POR DIA 12AVO MES
VentaMes[66] = newdouble[] { 20308, 22515,21200,25072,24471 };
VentaMes[67] = newdouble[] { 20944, 22277,22180,23601,24314 };
VentaMes[68] = newdouble[] { 21322, 25299,20511,20072,23311 };
VentaMes[69] = newdouble[] { 21104, 22942,22270,24490,23459 };
VentaMes[70] = newdouble[] { 20609, 21213,21199,20441,22956 };
VentaMes[71] = newdouble[] { 20519, 24422,24133,20914,23383 };
//2do año
//VENTAS POR DIA 1ER MES
VentaMes[72] = newdouble[] { 22593,24146,21622,25215,20078 };
VentaMes[73] = newdouble[] { 21083,22874,23639,21826,22484 };
VentaMes[74] = newdouble[] { 20816,22456,20394,20215,20661 };
VentaMes[75] = newdouble[] { 20561,25299,20265,24973,21889 };
VentaMes[76] = newdouble[] { 22802,23468,20363,25139,23185 };

```

VentaMes[77] = newdouble[] { 22361,23562,24905,24389,21636 };  
 //VENTAS POR DIA 2DO MES  
 VentaMes[78] = newdouble[] { 24675,20879,24848,21969,24232 };  
 VentaMes[79] = newdouble[] { 20457,22105,20400,24391,22728 };  
 VentaMes[80] = newdouble[] { 20397,23542,21535,20160,22864 };  
 VentaMes[81] = newdouble[] { 23128,23003,25169,21404,20706 };  
 VentaMes[82] = newdouble[] { 21188,24001,20205,24916,22006 };  
 VentaMes[83] = newdouble[] { 22396,21003,23321,24735,20302 };  
 //VENTAS POR DIA 3ER MES  
 VentaMes[84] = newdouble[] { 21051,20815,20363,24504,23794 };  
 VentaMes[85] = newdouble[] { 21964,20468,24011,22127,22112 };  
 VentaMes[86] = newdouble[] { 25038,20401,20105,21315,20824 };  
 VentaMes[87] = newdouble[] { 21876,23143,21543,22115,20918 };  
 VentaMes[88] = newdouble[] { 24471,20531,21687,21884,25335 };  
 VentaMes[89] = newdouble[] { 25334,24821,20464,20468,25227 };  
 //VENTAS POR DIA 4TO MES  
 VentaMes[90] = newdouble[] { 22671,24122,21474,20586,22073 };  
 VentaMes[91] = newdouble[] { 24827,22931,22234,21598,23237 };  
 VentaMes[92] = newdouble[] { 21855,23980,24177,23869,20436 };  
 VentaMes[93] = newdouble[] { 24918,21761,25208,21619,21847};  
 VentaMes[94] = newdouble[] { 24886,20499,21060,24402,21458 };  
 VentaMes[95] = newdouble[] { 22910,20176,23343,22748,24282 };  
 //VENTAS POR DIA5TO MES  
 VentaMes[96] = newdouble[] { 20869,23118,20434,20422,23288 };  
 VentaMes[97] = newdouble[] { 22682,22315,24710,22937,20295 };  
 VentaMes[98] = newdouble[] { 20900,25169,22601,21246,23881 };  
 VentaMes[99] = newdouble[] { 24587,24117,20186,20949,22148 };  
 VentaMes[100] = newdouble[] { 21717,22621,25293,23631,25358 };  
 VentaMes[101] = newdouble[] { 23137,23716,25045,23732,20176 };  
 //VENTAS POR DIA 6TO MES  
 VentaMes[102] = newdouble[] { 21936,23201,24903,25098,24619 };  
 VentaMes[103] = newdouble[] { 23973,21969,20630,23618,21764 };  
 VentaMes[104] = newdouble[] { 24921,23223,23840,20226,22404 };

VentaMes[105] = newdouble[] { 23106,24726,22366,22865,21370 };  
 VentaMes[106] = newdouble[] { 25221,23932,23150,20973,24400 };  
 VentaMes[107] = newdouble[] { 20804,21352,22559,22865,22645 };  
 //VENTAS POR DIA 7TIMO MES  
 VentaMes[108] = newdouble[] { 22248,22177,24546,23816,20746 };  
 VentaMes[109] = newdouble[] { 21391,21607,21369,24622,24329 };  
 VentaMes[110] = newdouble[] { 24655,22234,20450,22369,21178 };  
 VentaMes[111] = newdouble[] { 23955,20161,24844,23186,24422 };  
 VentaMes[112] = newdouble[] { 20417,24199,24611,21297,21163 };  
 VentaMes[113] = newdouble[] { 21172,21330,22211,21101,21885 };  
 //VENTAS POR DIA 8AVO MES  
 VentaMes[114] = newdouble[] { 21630,25162,20724,23512,21064 };  
 VentaMes[115] = newdouble[] { 20115,21924,20078,21153,21472 };  
 VentaMes[116] = newdouble[] { 24539,24261,24888,23551,20564 };  
 VentaMes[117] = newdouble[] { 23620,21817,23724,21074,22502 };  
 VentaMes[118] = newdouble[] { 25215,21452,24607,21419,20459 };  
 VentaMes[119] = newdouble[] { 23356,20555,23476,21199,23293 };  
 //VENTAS POR DIA 9VENO MES  
 VentaMes[120] = newdouble[] { 24637,21883,22217,21123,22627 };  
 VentaMes[121] = newdouble[] { 24865,23241,22429,25364,24059 };  
 VentaMes[122] = newdouble[] { 23127,20242,21981,21414,24617 };  
 VentaMes[123] = newdouble[] { 20589,20830,22681,24531,24221 };  
 VentaMes[124] = newdouble[] { 21558,24721,23515,24495,23952 };  
 VentaMes[125] = newdouble[] { 20489,20413,20115,23088,22448 };  
 //VENTAS POR DIA 10CIMO MES  
 VentaMes[126] = newdouble[] { 24860,21480,21978,23621,23443 };  
 VentaMes[127] = newdouble[] { 22003,21177,25097,23879,25127 };  
 VentaMes[128] = newdouble[] { 21804,22541,25023,22372,20070 };  
 VentaMes[129] = newdouble[] { 22249,22465,23199,24409,21283 };  
 VentaMes[130] = newdouble[] { 20872,21551,23042,20736,20429 };  
 VentaMes[131] = newdouble[] { 24563,20294,23966,23886,23019 };  
 //VENTAS POR DIA 11AVO MES  
 VentaMes[132] = newdouble[] { 21027,21252,21388,20233,23968 };



```

VentaMes[133] = newdouble[] { 20784,24872,22250,24160,21819 };
VentaMes[134] = newdouble[] { 24589,21240,20793,21585,24407 };
VentaMes[135] = newdouble[] { 22818,21844,22198,20756,20705 };
VentaMes[136] = newdouble[] { 23146,22752,21833,23162,21965 };
VentaMes[137] = newdouble[] { 21206,21524,21414,22523,22081 };
//VENTAS POR DIA 12AVO MES
VentaMes[138] = newdouble[] { 24493,20196,20011,20988,20847 };
VentaMes[139] = newdouble[] { 23057,22313,22370,20538,20993 };
VentaMes[140] = newdouble[] { 23167,21049,23461,24769,23887 };
VentaMes[141] = newdouble[] { 20634,23186,21066,24700,22968 };
VentaMes[142] = newdouble[] { 24663,22738,24009,22026,24617 };
VentaMes[143] = newdouble[] { 20810,23768,21952,21478,23427 };
//3er año
//Desde aquí empieza el pronóstico, se ejecuta el algoritmo para la
predicción de las ventas //futuras.
//VENTAS POR DIA 1ER MES
VentaMes[144] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[145] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[146] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[147] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[148] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[149] = newdouble[] { 0, 0, 0, 0, 0 };
//VENTAS POR DIA 2DO MES
VentaMes[150] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[151] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[152] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[153] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[154] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[155] = newdouble[] { 0, 0, 0, 0, 0 };
//VENTAS POR DIA 3ER MES
VentaMes[156] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[157] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[158] = newdouble[] { 0, 0, 0, 0, 0 };

```

```
VentaMes[159] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[160] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[161] = newdouble[] { 0, 0, 0, 0, 0 };
//VENTAS POR DIA 4TO MES
VentaMes[162] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[163] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[164] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[165] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[166] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[167] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[168] = newdouble[] { 0, 0, 0, 0, 0 };
//VENTAS POR DIA 5TO MES
VentaMes[169] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[170] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[171] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[172] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[173] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[174] = newdouble[] { 0, 0, 0, 0, 0 };
//VENTAS POR DIA6TO MES
VentaMes[175] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[176] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[177] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[178] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[179] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[180] = newdouble[] { 0, 0, 0, 0, 0 };
//VENTAS POR DIA 7TIMO MES
VentaMes[181] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[182] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[183] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[184] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[185] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[186] = newdouble[] { 0, 0, 0, 0, 0 };
//VENTAS POR DIA 8AVO MES
```

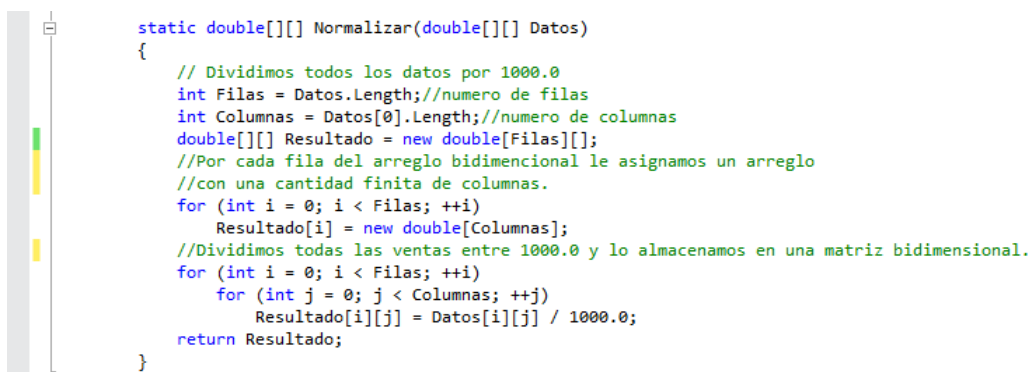
```

VentaMes[187] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[188] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[189] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[190] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[191] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[192] = newdouble[] { 0, 0, 0, 0, 0 };
//VENTAS POR DIA 9VENO MES
VentaMes[193] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[194] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[195] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[196] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[197] = newdouble[] { 0, 0, 0, 0, 0 };
VentaMes[198] = newdouble[] { 0, 0, 0, 0, 0 };
//VENTAS POR DIA 10CIMO MES
VentaMes[199] = newdouble[] { 0, 0, 0, 0, 0 };
return VentaMes;
    }

```

## - PASO N°2: Normalización de Datos

Se normaliza las ventas diarias dividiendo cada venta entre 1000. Por ejemplo, dividimos 20784 / 1000 y se obtuvo 20,784. La función retorna un arreglo bidimensional con todas las ventas normalizadas.



```

static double[][] Normalizar(double[][] Datos)
{
    // Dividimos todos los datos por 1000.0
    int Filas = Datos.Length;//numero de filas
    int Columnas = Datos[0].Length;//numero de columnas
    double[][] Resultado = new double[Filas][];
    //Por cada fila del arreglo bidimensional le asignamos un arreglo
    //con una cantidad finita de columnas.
    for (int i = 0; i < Filas; ++i)
        Resultado[i] = new double[Columnas];
    //Dividimos todas las ventas entre 1000.0 y lo almacenamos en una matriz bidimensional.
    for (int i = 0; i < Filas; ++i)
        for (int j = 0; j < Columnas; ++j)
            Resultado[i][j] = Datos[i][j] / 1000.0;
    return Resultado;
}

```

*Figura 66:* Parte del código fuente que indica el método para la normalización de los datos. *Fuente:* *Elaboración Propia*

- **PASO N° 3: Visualizar ventas por cada entrenamiento de la Red**  
La subrutina permite visualizar las primeras n filas o todas las filas, los datos pueden visualizarse con 2 o n decimales, visualizamos índices.

```

static void Visualizar_Matriz(double[][] Matriz, int Numero_Filas, int Numero_Decimales, bool Indices)
{
    int Longitud = Matriz.Length.ToString().Length; // Longitud de la matriz

    string msg = "";
    // Mostramos una o mas filas
    for (int i = 0; i < Numero_Filas; ++i)
    {
        if (Indices == true)
            Console.WriteLine("[" + i.ToString().PadLeft(Longitud) + "] "); msg += "[" + i.ToString().PadLeft(Longitud) + "] ";
        for (int j = 0; j < Matriz[i].Length; ++j)
        {
            double v = Matriz[i][j];
            if (v >= 0.0)
                Console.WriteLine(" "); msg += " "; // " "
            Console.WriteLine(v.ToString("F" + Numero_Decimales) + " "); msg += v.ToString("F" + Numero_Decimales) + " ";
        }
        Console.WriteLine(""); msg += "\n";

        sw.WriteLine(msg); msg = "";
    }
    // Despues, mostramos la ultima fila
    if (Numero_Filas < Matriz.Length)
    {
        Console.WriteLine(".....");
        sw.WriteLine(".....");
        int UltimaFila = Matriz.Length - 1;
        if (Indices == true)
            Console.WriteLine("[" + UltimaFila.ToString().PadLeft(Longitud) + "] "); msg += "[" + UltimaFila.ToString().PadLeft(Longitud) + "] ";
        for (int j = 0; j < Matriz[UltimaFila].Length; ++j)
        {
            double v = Matriz[UltimaFila][j];
            if (v >= 0.0)
                Console.WriteLine(" "); msg += " "; // " "
            Console.WriteLine(v.ToString("F" + Numero_Decimales) + " "); msg += v.ToString("F" + Numero_Decimales) + " ";
        }
        Console.WriteLine("\n"); msg += "\n";

        sw.WriteLine(msg); msg = "";
    }
}

```

*Figura 77: Parte del código fuente que indica la función para visualizar la matriz de datos. Fuente: Elaboración Propia*

Quando se ejecuta la función esta permite visualizar las n primeras filas o todas las filas, los datos pueden visualizarse con 2 o n decimales, visualizamos índices. Como podemos ver en la figura.

```

Inicio red neuronal serie de tiempos.
El objetivo es predecir la rentabilidad de las ventas sobre un tiempo determinado.
Datos desde el mes de x del año y hasta el mes de x del año y.

Normalizar Entrenamiento Datos:
[ 0] 20,784 20,308 22,495 23,666 22,949
[ 1] 23,527 22,121 21,198 22,972 24,070
[ 2] 20,970 22,829 23,985 23,324 23,537
[ 3] 24,210 20,348 22,531 22,060 20,132
[ 4] 20,627 24,004 22,551 24,529 23,218
[ 5] 24,469 20,859 22,327 20,735 20,316
[ 6] 20,864 24,994 23,013 25,288 23,665
[ 7] 24,783 21,141 23,918 24,326 20,715
[ 8] 23,977 23,078 21,049 20,641 25,060
[ 9] 21,155 22,253 22,526 22,051 21,058
[10] 24,464 22,002 21,119 20,612 23,132
[11] 20,841 21,684 25,206 22,779 22,263
[12] 21,793 24,110 23,809 23,036 23,263
[13] 21,828 24,179 23,932 20,928 25,026
[14] 20,441 21,375 24,392 24,949 24,087
[15] 21,345 24,639 24,735 23,567 23,057
[16] 20,598 20,289 20,499 25,092 24,312
[17] 22,816 20,964 24,467 22,242 24,199
[18] 23,399 23,688 22,874 25,384 22,736
[19] 20,551 21,190 21,020 22,216 25,246
[20] 23,429 24,879 20,154 23,999 24,905
[21] 23,273 24,360 24,165 24,339 24,571
[22] 25,182 20,671 25,075 24,367 21,890
[23] 22,550 22,144 24,189 24,689 20,469
[24] 22,364 21,688 24,252 20,718 22,076
[25] 21,943 20,900 24,741 20,205 21,996
[26] 21,750 21,072 20,694 25,033 20,844
[27] 23,941 20,888 22,433 21,663 23,584
[28] 20,231 20,737 25,088 21,371 23,592
[29] 23,372 21,487 24,015 21,045 20,854
[30] 23,545 23,873 22,393 20,609 20,463
[31] 23,508 23,761 20,601 24,885 22,008
[32] 21,640 24,052 23,221 22,930 22,334
[33] 21,604 23,164 21,967 24,747 24,837
[34] 21,204 22,345 20,796 20,605 22,955
[35] 21,911 20,676 23,686 24,795 23,114
[36] 20,161 21,949 20,647 20,790 20,707
[37] 23,340 25,240 24,822 21,592 23,711
[38] 20,717 24,230 24,858 24,714 22,643
[39] 23,589 22,009 20,150 22,399 20,325
[40] 20,687 25,313 22,951 20,880 22,113
[41] 20,171 23,696 24,726 21,583 23,589
[42] 21,611 23,223 20,250 20,847 20,702
[43] 20,016 24,650 20,947 20,067 21,782
[44] 22,688 22,839 25,100 20,739 22,429
[45] 24,528 20,158 23,485 24,946 20,399
[46] 24,449 22,190 24,146 20,042 21,057
[47] 22,873 23,719 22,529 20,228 25,269
[48] 21,929 24,491 22,696 23,933 24,676

```

Figura 88: Muestra el entrenamiento de la red en modo consola.

Fuente: Elaboración Propia

- **PASO N° 4: Visualizar Pesos**

La subrutina permite visualizar los pesos que ha generado la red neuronal después del entrenamiento.

```

static void VisualizarVector(double[] Vector, int NumeroDecimales, int LongitudLinea, bool NuevaLinea)
{
    string msg = "";

    for (int i = 0; i < Vector.Length; ++i)
    {
        //Si i es mayor a 0 y i es par
        if (i > 0 && i % LongitudLinea == 0) Console.WriteLine(""); msg += "\n";
        if (Vector[i] >= 0) Console.Write(" "); msg += " ";
        Console.Write(Vector[i].ToString("F" + NumeroDecimales) + " ");
        msg += Vector[i].ToString("F" + NumeroDecimales) + " ";
    }
    if (NuevaLinea == true)
        Console.WriteLine(""); msg += "\n";

    sw.WriteLine(msg); msg = "";
}

```

Figura 99: Parte del código fuente que indica la función para visualizar el vector de datos. Fuente: Elaboración Propia

## - PASO N° 5: Inicialización de Pesos

Empezamos por crear un arreglo con 73 elementos para almacenar 73 números aleatorios.

```

private void Inicializa_Pesos(StreamWriter sw) // Inicializamos pesos
{
    // Inicializamos pesos y bias con valores pequeños aleatorios
    int Numero_Pesos = (Numero_Entrada * Numero_Oculto) +
        (Numero_Oculto * Numero_Salida) + Numero_Oculto + Numero_Salida;
    /*= (4 * 12) + (12*1) + 12 + 1= 73 */
    double[] Inicio_Pesos = new double[Numero_Pesos];
    for (int i = 0; i < Inicio_Pesos.Length; ++i)
    {
        Inicio_Pesos[i] = (0.001 - 0.0001) * Aleatorio.NextDouble() + 0.0001;
        //=====
        //Imprimir pesos
        //=====
        //sw.WriteLine(Inicio_Pesos[i]);
    }
    this.Fijar_Pesos(Inicio_Pesos, sw);
}

```

Figura 20: Parte del código fuente que indica la función para iniciar los pesos. Fuente: Elaboración Propia

Al ejecutarse la red por primera vez, se ejecuta la subrutina “Inicializa\_Pesos” para generar los siguientes pesos:

---

0,000753618942971164	0,000442904056209560
0,000835592823631872	0,000408827660190327
0,000791220420455197	0,000961709649097971
0,000602345072292883	0,000554616291473907
0,000285429838618929	0,000744375261126261
0,000602996315156574	0,000207061956220801
0,000915424359410733	0,000346106335123119
0,000497960085979644	0,000916388155061932
0,000979794777827242	0,000815289034468722
0,000346334011920883	0,000403444305343295
0,000362715656292958	0,000511487908992678
0,000520583230313185	0,000232142536263979
0,000669393165535011	0,000299183265585072
0,000522560690586716	0,000469065951634695
0,000983936127826542	0,000746854149478885
0,000127330291656465	0,000657847255262475
0,000876133138442474	0,000539171247249083
0,000995812373094173	0,000296566105166714
0,000709463034295227	0,000905826195145876
0,000383132613721831	0,000906795822925305
0,000835217117813983	0,000177921895160303
0,000863246604783110	0,000854721189595163
0,000992711957820091	0,000253128551157717
0,000129362678541505	0,000679703764328595
0,000729947785395173	0,000275422093074500
0,000573655728378173	0,000890372338188054
0,000940616793018122	0,000842880865811781
0,000718858254244019	0,000761857741913692
0,000592133890833768	0,000872397286897710
0,000172998954901937	0,000711774110240757
0,000268412116760580	0,000662379522138452
0,000507994466697795	0,000844132964985507
0,000367454679155468	0,000297268898411314
0,000989689401206416	0,000999975428031746
0,000678427725601209	0,000533229082232913
0,000786667229415228	0,000686984712903846

---

- **PASO N° 6: Fijar Pesos**

En este paso se fijan los 73 pesos generados en el paso anterior, la subrutina recibe un arreglo con los pesos generados por la subrutina "Inicializa\_Pesos".

```

public void Fijar_Pesos(double[] Pesos, StreamWriter sw)
{
    // Copia serializada de pesos y favoritos en Pesos[] array
    // para i-h Entrada_Oculto_Pesos, i-h Oculto_Favorito, h-o Oculto_Salida_Pesos, h-o Salida_Favoritos
    int Numero_Pesos = (Numero_Entrada * Numero_Oculto) +
        (Numero_Oculto * Numero_Salida) + Numero_Oculto + Numero_Salida;
    // = (4 * 12) + (12*1) + 12 + 1 = 73
    // =====
    // Imprimir numero de pesos.
    // =====
    if (Pesos.Length != Numero_Pesos)
        throw new Exception("Error al tratar de fijar pesos en el array");
    int k = 0; // Puntos ingresar parametro pesos
    // tenemos 4 entradas en la neurona
    for (int i = 0; i < Numero_Entrada; ++i)
    {
        // tenemos 12 entradas ocultas en la neurona.
        for (int j = 0; j < Numero_Oculto; ++j)
        {
            Entrada_Oculto_Pesos[i][j] = Pesos[k++];
        }
    }
}

```

Figura 21: Parte del código fuente que indica la función para fijar los pesos. Fuente: *Elaboración Propia*

Comparamos la cantidad de elementos del arreglo con el número de pesos obtenidos. En el caso que sea diferente se activara una excepción dando a conocer el error.

Se asigna los pesos sinápticos a cada neurona oculta por cada entrada en la neurona. Luego al ejecutar el siguiente el bucle anidado se obtuvo.

---

00,000753618942971164	20,000940616793018122
00,000835592823631872	20,000718858254244019
00,000791220420455197	20,000592133890833768
00,000602345072292883	20,000172998954901937
00,000285429838618929	20,000268412116760580
00,000602996315156574	20,000507994466697795
00,000915424359410733	20,000367454679155468
00,000497960085979644	20,000989689401206416
00,000979794777827242	20,000678427725601209
00,000346334011920883	20,000786667229415228
00,000362715656292958	30,000127354863624719
00,000520583230313185	30,000442904056209560
10,000669393165535011	30,000408827660190327
10,000522560690586716	30,000961709649097971
10,000983936127826542	30,000554616291473907
10,000127330291656465	30,000744375261126261
10,000876133138442474	30,000207061956220801
10,000995812373094173	30,000346106335123119
10,000709463034295227	30,000916388155061932
10,000383132613721831	30,000815289034468722
10,000835217117813983	30,000403444305343295
20,000729947785395173	30,000511487908992678
20,000573655728378173	

---



En la siguiente imagen se muestra la ejecución de las Bais o umbral para la neurona oculta.

```

for (int i = 0; i < Numero_Oculto; ++i)
{
    Oculito_Favorito[i] = Pesos[k++];
}

```

Figura 22: Parte del código fuente que indica la ejecución de la Bais o umbral de la neurona oculta. Fuente: Elaboración Propia

Al ejecutar la el código de la figura 22, se obtiene el siguiente resultado.

00,000232142536263979	60,0002754220930745000
10,000299183265585072	70,0008903723381880540
20,000469065951634695	80,0008428808658117810
30,000746854149478885	90,0007618577419136920
40,000657847255262475	100,000872397286897710
50,000539171247249083	110,000711774110240757

En la siguiente imagen se muestra la ejecución para el cálculo de la salida de los pesos de las neuronas ocultas y salidas favoritas o resultado esperado.

```

for (int i = 0; i < Numero_Oculto; ++i)
{
    for (int j = 0; j < Numero_Salida; ++j)
    {
        Oculito_Salida_Pesos[i][j] = Pesos[k++];
    }
}

for (int i = 0; i < Numero_Salida; ++i)
{
    Salida_Favoritos[i] = Pesos[k++];
}

```

Figura 23: Parte del código fuente que indica los pesos ocultos de las neuronas y la salida favorita de los mismos. Fuente: Elaboración Propia

Se asigna los pesos sinápticos a cada neurona oculta de salida por cada neurona oculta de entrada, como resultado lo siguiente:

00,0006623795221384520	60,0002531285511577170
10,0002965661051667140	70,0006797037643285950
20,0009058261951458760	80,0008441329649855070
30,0009067958229253050	90,0002972688984113140
40,0001779218951603030	100,000999975428031746
50,0008547211895951630	110,000533229082232913

Al calcular las salidas favoritas se obtiene el siguiente resultado:  
0,000686984712903846

- **PASO N°7: Implementación de la Red**

Se procedió a crear la siguiente clase:

```
Public class Red_Neuronal {...}
```

En esta clase se implementara todo lo necesario para predecir las ventas diarias mensuales.

- **PASO N° 8: Definición de atributos de la Clase**

- a. Definir variable número de nodos de entrada que tendrá la red.

```
Private int Numero_Entrada;
```

- b. Definir variable número de nodos oculto que tendrá la red.

```
Private int Numero_Oculto;
```

- c. Definir variable número de nodos salida que tendrá la red.

```
Private int Numero_Salida;
```

- d. Definir como variable una matriz unidimensional para almacenar las entradas de la red.

Sintaxis:  $x_i$

```
Private double[] Entrada_Nodos;
```

- e. Definir como variable una matriz bidimensional para almacenar los pesos sinápticos que tendrá la entrada oculta la red.

Sintaxis:  $w_{ji}$

```
Private double[][] Entrada_Oculto_Pesos;
```

- f. Definir como variable una matriz unidimensional para almacenar el umbral (bias) que tendrá como entrada oculta la red.

Sintaxis:  $\theta_i$

```
Private double[] Oculto_Favorito;
```

- g. Definir como variable una matriz unidimensional para almacenar las entradas ocultas de la red.

```
Private double[] Oculto_Nodos;
```

- h. Definir como variable una matriz bidimensional para almacenar los pesos sinápticos que tendrá la salida oculta la red.

```
Private double[][] Oculto_Salida_Pesos;
```

- i. Definir como variable una matriz unidimensional para almacenar el umbral (bias) que tendrá como salida oculta la red.

```
Private double[] Salida_Favoritos;
```

- j. Definir como variable una matriz unidimensional para almacenar la salida de la red.

```
Private double[] Salida_Nodos;
```

#### - **PASO N° 9: Crear constructor Clase**

El constructor de la clase tiene 3 parámetros del tipo entero:

- a) En el primer parámetro se asigna el número de entradas que tendrá la red neuronal.
- b) En el segundo parámetro se especifica el número de neuronas ocultas que tendrá la red neuronal.
- c) En el tercer parámetro se asigna el número de salidas que tendrá la red neuronal.

```
Public Red_Neuronal (int _NumeroEntrada, int _NumeroOculto, int  
_NumeroSalida, StreamWriter sw)
```

- **PASO N° 10: Asignar valores a los atributos de la clase utilizando el constructor**

a. Neuronas de entrada.

`this.Numero_Entrada = _NumeroEntrada;`

Asignamos 4 neuronas de entrada.

b. Neuronas ocultas

`this.Numero_Oculto = _NumeroOculto;`

Asignamos 12 neuronas ocultas.

c. Neuronas de salida: `this.Numero_Salida = _NumeroSalida;`

Asignamos 1 neurona de salida.

d. Crear una matriz bidimensional con todos sus valores en 0.0.

Pesos sinápticos entrada oculta red neuronal.

`this.Entrada_Oculto_Pesos = CrearMatriz(_NumeroEntrada, _NumeroOculto, 0.0);`

La matriz queda especificada de la siguiente manera:

$$\begin{array}{c}
 \text{Entrada\_Oculto\_Pesos} \\
 = \begin{bmatrix}
 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\
 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\
 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\
 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0
 \end{bmatrix}
 \end{array}$$

`this.Oculto_Favorito = new double[_NumeroOculto];`

La matriz queda creada de la siguiente manera:

Oculto\_Favorito = 

--	--	--	--	--	--	--	--	--	--	--	--	--

`this.Oculto_Nodos = newdouble[_NumeroOculto];`

La matriz queda creada de la siguiente manera:

Oculto\_Favorito = 

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

- e. Crear una matriz bidimensional con todos sus valores en 0.0. Pesos sinápticos salida oculta red neuronal.  
`this.Oculto_Salida_Pesos = CrearMatriz (_NumeroOculto, _NumeroSalida, 0.0);`

La matriz queda especificada de la siguiente manera:

Oculto\_Salida\_Pesos =  $\begin{pmatrix} 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \end{pmatrix}$

```
this.Salida_Favoritos = newdouble[_NumeroSalida];
this.Salida_Nodos = newdouble[_NumeroSalida];
this.Aleatorio = newRandom(0);
```

- f. Inicializar pesos y bias. `this.Inicializa_Pesos(sw);`

**PASO N° 11: Crear matrices para asignación de pesos sinápticos**

La siguiente función crea una matriz bidimensional de acuerdo a las filas y columnas especificadas, y le asigna un valor. La función retorna una matriz bidimensional.

```

private static double[][] CrearMatriz(int Filas, int Columnas, double Valor)
// Creamos una matriz para el entrenamiento
{
//Matriz bidimensional con una cantidad finita de filas por n columnas.
double[][] Resultado = new double[Filas][Columnas];
//Por cada fila de la matriz le asignamos una cantidad finita de columnas.
for (int r = 0; r < Resultado.Length; ++r)
Resultado[r] = new double[Columnas];
//Asinamos un valor a la matriz en la posicion que le corresponde.
for (int i = 0; i < Filas; ++i)
for (int j = 0; j < Columnas; ++j)
Resultado[i][j] = Valor;
//Retornamos la matriz bidimensional.
return Resultado;
}

```

Figura 24: Parte del código fuente que indica la creación de la Matriz para el entrenamiento de la neurona. Fuente: Elaboración Propia

- **PASO N° 12: Obtener pesos sinápticos Generados en el entrenamiento de la Red.**

La función permite obtener una matriz unidimensional con los pesos sinápticos y bias obtenidos una vez finalizado el entrenamiento de la red.

```

public double[] Obtener_Pesos(StreamWriter sw)
{
int Numero_Pesos = (Numero_Entrada * Numero_Oculto) +
(Numero_Oculto * Numero_Salida) + Numero_Oculto + Numero_Salida;
double[] Resultado = new double[Numero_Pesos];
int k = 0;
for (int i = 0; i < Entrada_Ocultas.Length; ++i)
{
for (int j = 0; j < Entrada_Ocultas[i].Length; ++j)
{
Resultado[k++] = Entrada_Ocultas[i][j];
}
}
for (int i = 0; i < Oculto_Favorito.Length; ++i)
{
Resultado[k++] = Oculto_Favorito[i];
}
for (int i = 0; i < Oculto_Salida_Pesos.Length; ++i)
{
for (int j = 0; j < Oculto_Salida_Pesos[i].Length; ++j)
{
Resultado[k++] = Oculto_Salida_Pesos[i][j];
}
}
for (int i = 0; i < Salida_Favoritos.Length; ++i)
Resultado[k++] = Salida_Favoritos[i];
return Resultado;
}

```

Figura 25: Parte del código fuente para obtener los pesos y crear la matriz para almacenar los pesos sinápticos generados por la red neuronal. Fuente: Elaboración Propia

Tabla 8.

Pesos sinápticos y bias generados por cada iteración de entrenamiento.

	Iteración 01	Iteración 02	Iteración 03	Iteración 04	Iteración 05	... Iteración N
	0,04610962871958	-2,1051079349	0,553803683137	1,0312489760460	0,2242220534394	
	1,5706653774998	-1,4941511827	-4,491631588246	0,5753736629884	1,7550660873234	
	0,4974834178267	-1,1061735237	-0,313174206979	-0,393003382954	-0,0115206028036	
	0,0856372246972	-2,4584808811	2,461969749680	0,2359228923068	-0,3409227491973	
	0,0022159215175	-1,4474463130	-4,327635544364	-1,877878734072	1,73013703684357	
	0,1506643846065	-1,9633502804	-0,988068035141	-0,198563847094	0,01428509252328	
	0,3064068314005	-1,9432754443	1,770802045512	0,2922963373627	-0,4874479731019	
	0,0570085126910	-3,1536558047	0,378219252098	0,3868968273313	0,00811900691428	
	0,0797324340966	-1,0414621211	-0,260099747487	0,2888315340421	0,61808146990358	
	0,0544228594646	-0,6363419449	-2,459688080300	-0,446822604034	0,03497057403219	
	0,7460455114326	-2,4559711543	1,111675382826	0,2139901132396	0,22881930559981	
	0,0767621133051	-2,6355138260	-4,440683101073	-0,160032918498	0,00658255694009	
	0,2356219100017	0,06952988996	0,529434152929	1,2355718943839	0,45028241164007	
Pesos Sinápticos Entrada Oculta	1,9618964020017	0,07359652329	-4,471387515096	0,5614595185411	1,90136086798513	
	0,6635493071136	0,08447567596	-0,307335966557	-0,382785296079	0,22619340949401	
	0,2619388313461	0,06795188739	2,345633897578	0,258246863817670	-0,1580430222449	
	0,2104537241018	0,07419050339	-4,3070385268886	-1,86876615921750	1,8767299656051600	
	0,3469470806240	0,07287824916	-0,9822810854609	-0,19227170974319	0,2268059637477410	
	0,5074089441269	0,06736959613	1,6875494647900	0,319255247116	-0,2986463755829	
	0,2652956201633	0,06243137699	0,42099211620943	0,3535559931572	0,19726287167202	
	0,2460637517477	0,08496997039	-0,2552579497447	0,2846769782989	0,80827362652486	
	0,2034415729160	0,05794083244	-2,4452321707937	-0,439856139937	0,19802025552129	
	1,0257411681938	0,06971142240	1,06125356283384	0,2727882250276	0,45616073010279	
	0,2486279563067	0,06355542021	-4,4268033417629	-0,157591741038	0,19586583615624	

	0,0115644631079	0,06875842353	0,73369227116412	1,2503794723177	0,18394801919069
	1,6766593692547	0,07294005699	0,09155818218971	0,6268637124117	1,85769048352446
	0,4621147191711	0,08341895220	0,08328624218866	-0,384770481822	-0,0507298381389
	0,0577305554365	0,06761275853	2,90777409755249	0,2380596269129	-0,3657383202776
	-0,039045551273	0,07324313270	0,08970790878768	-1,863731144271	1,83113734761966
	0,1289492719525	0,07113839716	0,12956693433922	-0,195892628690	-0,0249994975874
	0,3093326814148	0,06626353922	2,11979658839531	0,2934422198273	-0,5077569233343
	0,0122213439468	0,06156260821	0,95832069459933	0,3843991710129	-0,0531401176997
	0,0308576882889	0,08350723831	0,06260325779889	0,2729110719912	0,57740827102897
	0,0615004876074	0,05715739419	0,09571085231148	-0,441367007401	-0,0033716586064
	0,7766755005633	0,06843031527	1,37213508008067	0,2725116912571	0,18933645171140
	0,0367742825231	0,06340190900	0,22172568964354	-0,157407176701	-0,0535510535248
	0,2703605738738	0,06850865805	0,61150006529035	1,2386372681483	0,51284860758899
	1,8052062973006	0,07327318245	0,08946720393236	0,6117186672046	2,28265309987756
	0,6190974902985	0,08334505940	0,08316933956253	0,1137744017987	0,30964544740411
	0,3218794225826	0,06814476243	2,38753222950633	0,3522536457223	0,01792465774205
	0,2965794469754	0,07369047251	0,08829360137994	0,1668918514848	2,25364533354646
	0,3772808540127	0,07207145380	0,11977030170551	0,0877550390100	0,29251344902849
	0,5756043691105	0,06659229767	1,74360769223460	0,4205398352723	-0,1201973042538
	0,3260162820225	0,06139372043	1,12026105020459	0,3904454256777	0,27628986978204
	0,3454280471675	0,08452822766	0,06135323920288	0,5275771978050	0,84449547815294
	0,2336429389132	0,05398574669	0,09624177896517	0,1077920866459	0,25251669552715
	0,9623763531367	0,06844542028	1,13307432969282	0,4052349643989	0,51819242214166
	0,3007640105010	0,06335530782	0,19255365939065	0,0780518850315	0,27454559924603
	-0,102740166095	0,01312843957	-0,2845302185102	-0,126252586377	-0,1179016731077
Pesos	-0,018983215367	0,36563825987	-0,1288880108854	-0,145654574150	-0,0351493445685
Sinápticos	-0,091415199572	0,13715660022	0,06003723312553	0,0011441676801	-0,1344014932266
Oculto	-0,106771223296	-0,2149843650	-0,2005822101801	-0,164200401773	0,00528468261410
Favorito	-0,089708196520	0,43473032390	-0,1190536027625	-0,117929252396	-0,0332588639670
	-0,102934854235	-0,0539947192	0,03047384956279	0,0024598424852	-0,1308271810514



	-0,077081360319	0,28716548324	-0,2251297329019	-0,156320248397	-0,0522603178903
	-0,100095523912	-0,2778928589	-0,2938030731264	-0,157604872662	-0,1263084952286
	-0,107545561739	0,18589789727	0,06335857445967	-0,185194311795	-0,1042256157279
	-0,094378991279	0,52674925597	-0,0200766862384	0,0161763295005	-0,1208138326972
	-0,076659866169	-0,2583794910	-0,2626955987019	-0,168723346818	-0,1238358704683
	-0,096621282600	-0,0572234060	-0,1265097512084	0,0030243888377	-0,1238961602994
	1,735552869130	-1,5825719952	4,02288956553829	3,2479217073532	2,23451512869803
	1,649759228200	-1,9333822714	-0,4120763268692	3,3680009694524	2,1384466020881
	1,761579830947	-1,6971574303	-0,3929206359674	0,5985123983596	2,27092561877947
Pesos	1,761227438842	-1,3540765625	4,00642021878630	3,4375523911244	0,69258534170091
Sinápticos	1,679737473932	-2,0059414003	-0,4281727445480	2,1789066935581	2,12628891232569
Salida	1,756167342374	-1,5139445931	-0,3962919527232	0,0380151861417	2,26186483770042
Oculto	1,679560837809	-1,8564923493	3,94902420303891	3,3627598448375	1,57684788981804
	1,730944242085	-1,2943073633	1,95682936382797	3,4055017994087	2,23421105269909
	1,762099562020	-1,7453097862	-0,3983276174364	2,3679647022901	2,25640083031107
	1,703176722409	-2,0986078764	2,53270100233244	1,3976237178248	2,21069098734311
	1,755242765412	-1,3085612841	4,04819362828154	3,2091692503068	2,27084626322260
	1,716914863299	-1,5168037617	-0,5792597103533	-0,475742729647	2,22012865355260
Pesos	1,839230881732	2,46639231166	4,31584401297627	3,7119935731750	2,41866341172070
Sinápticos					
Salida					
Favorito					

*Nota: Fuente elaboración propia.*

- **PASO N° 13: Entrenamiento de la Red Neuronal.**

Se realiza el entrenamiento de la red neuronal, realizando las declaraciones de máxima época para el ratio de aprendizaje la cual se muestra a continuación.

1) Crear un arreglo bidimensional salida oculta gradiente.

$$\text{OcultoSalidaGradiente} = \begin{pmatrix} 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \end{pmatrix}$$

2) Crear un arreglo unidimensional salida bias gradiente.

3) Crear un arreglo bidimensional entrada oculta gradiente.

$$\text{EntradaOcultaGradientess} = \begin{bmatrix} 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \end{bmatrix}$$

4) Crear un arreglo unidimensional bias oculto gradiente.

5) Crear un arreglo unidimensional señal salida.

6) Crear un arreglo unidimensional señal oculta.

7) Definimos la siguiente variable para contabilizar la cantidad de iteraciones que realizara la red neuronal.

```
int Epoca = 0;
```

8) Crear un arreglo unidimensional para almacenar las entradas.

9) Crear un arreglo unidimensional para almacenar las salidas.

10) Se crea la secuencia de entrenamiento

$$Secuencia = \begin{bmatrix} 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ \dots \\ 199 \end{bmatrix}$$

11) Chequear error intervalo. Dividimos 10000 entre 5.

$$\begin{aligned} \text{ErrorIntervalo} &= \frac{\text{MaximaEpoca}}{5} \\ \frac{10000}{5} &= 2000 \end{aligned}$$

12) Creamos un bucle de entrenamiento.

Iniciamos la ejecución del bucle mientras la época sea menor a 10000.

13) Evaluamos época e incrementamos.

La variable época tomara los siguientes valores hasta llegar a la máxima época.

$$Epoca = 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad \dots \quad 1000$$

14) Verificamos si existe un error en el intervalo obtenido. Dividimos la Época entre el Error Intervalo y chequeamos si el resto es 0, y que la Época es menor a Máxima Época.

```

if (Epoca % ErrorIntervalo == 0 && Epoca < MaximaEpoca)
{
    double EntrenamientoError = Error(EntrenamientoDatos, sw);
    //24022019
    Console.WriteLine("Epoca = " + Epoca + " Error = " +
        EntrenamientoError.ToString("F4")); sw.WriteLine("Epoca = "
        + Epoca + " Error = " +
        EntrenamientoError.ToString("F4"));
}

```

Figura 26: Parte del código fuente para el entrenamiento de la red neuronal. Fuente: Elaboración Propia

---

Entrenamiento 1

Epoca = 2000 Error = 2,0502

Epoca = 4000 Error = 1,7472

Epoca = 6000 Error = 1,7577

Epoca = 8000 Error = 1,6933

Entrenamiento 2

Epoca = 2000 Error = 4,4463

Epoca = 4000 Error = 5,6789

Epoca = 6000 Error = 4,2880

Epoca = 8000 Error = 5,0403

Entrenamiento 3

Epoca = 2000 Error = 4,4462

Epoca = 4000 Error = 5,6793

Epoca = 6000 Error = 4,2880

Epoca = 8000 Error = 5,0403

Entrenamiento 4

Epoca = 2000 Error = 4,4462

Epoca = 4000 Error = 5,6793

Epoca = 6000 Error = 4,2880

Epoca = 8000 Error = 5,0403

Entrenamiento 5

Epoca = 2000 Error = 4,4462

Epoca = 4000 Error = 5,6793

Epoca = 6000 Error = 4,2880

Epoca = 8000 Error = 5,0402

Entrenamiento n

...

}

15) Barajar datos de manera aleatoria. Visitar cada dato de las ventas en orden aleatorio.

Barajar (Secuencia);

La subrutina realiza lo siguiente:

Índice Posición elemento venta

0 143	38 55	76 98
1 162	39 11	77 151
2 21	40 192	78 108
3 146	41 119	79 199
4 144	42 65	80 148
5 29	43 149	81 88
6 181	44 132	82 19
7 43	45 79	83 6
8 99	46 141	84 7
9 112	47 47	85 159
10 71	48 102	86 4
11 137	49 177	87 196
12 100	50 59	88 75
13 168	51 92	89 74
14 164	52 16	90 107
15 161	53 193	91 179
16 156	54 198	92 15
17 188	55 111	93 154
18 10	56 131	94 152
19 113	57 97	95 66
20 139	58 129	96 42
21 80	59 133	97 128
22 73	60 67	98 104
23 13	61 8	99 35
24 63	62 130	100 2
25 57	63 178	101
26 37	64 28	122
27 135	65 45	102
28 147	66 165	109
29 136	67 120	103
30 187	68 191	125
31 68	69 86	104
32 17	70 160	118
33 44	71 150	105 64
34 183	72 24	106 12
35 3	73 70	107 78
36 51	74 145	108 0
37 61	75 96	109 34

110 25  
111 40  
112  
114  
113  
157  
114 87  
115 81  
116  
117  
117  
171  
118  
140  
119  
197  
120  
167  
121  
182  
122  
105  
123  
155  
124 54  
125 31  
126 41  
127 89  
128 39  
129 38  
130  
142  
131  
103  
132  
169  
133  
185  
134 60  
135 82  
136  
115  
137 20  
138  
180

139 26  
140 77  
141  
170  
142  
175  
143 27  
144 36  
145  
158  
146 32  
147 93  
148 91  
149 49  
150  
121  
151  
134  
152  
153  
153 90  
154  
194  
155  
186  
156  
195  
157 50  
158 23  
159 33  
160 72  
161  
127  
162 52  
163 83  
164  
123  
165  
176  
166 22  
167 62  
168 5  
169 30  
170 46  
171 69

172  
124  
173 84  
174  
101  
175  
173  
176  
138  
177  
166  
178 95  
179  
174  
180 18  
181 9  
182 53  
183 58  
184 85  
185  
126  
186  
172  
187 48  
188 14  
189 94  
190  
106  
191  
190  
192  
110  
193 1  
194  
189  
195  
116  
196  
184  
197 56  
198 76  
199  
163

16) Visitamos cada dato para el entrenamiento en orden aleatorio.

- a) Se obtuvieron los números aleatorios que se encuentran almacenados en la secuencia.

Posiciones aleatorias obtenidas:

143

162

21

146

144

29

181

43

99

112

71 ...

- b) Copiamos cada elemento de la matriz EntrenamientoDatos a la matriz xvalores indicando que se van a copiar 4 valores.

- c) Computar salidas.

17) Se computó señales en el nodo de salida y Restamos los valores que se encuentran en la matriz t Valores menos los valores de la matriz Salida\_Nodos.

Se multiplicó el error obtenido en la señal por la activación de la neurona.

18) Se computó la neurona oculta para salida gradientes peso usando señales de salida.

19) Se computó salidos gradientes favoritos usando señales de salida.

20) Se computó señales de los nodos ocultos y se calculó la tangente hiperbólica.

21) Se computó la entrada oculta gradientes peso.

22) Se computó el nodo oculto gradientes favoritos.

- 23) Guardamos pesos y favoritos
  - a) Guardamos pesos ocultos entrada neurona.
  - b) Guardamos favoritos oculto.
  - c) Guardamos pesos ocultos salida neurona.
  - d) Guardamos nodo salida favoritos.
- 24) Se obtuvieron los pesos.
- 25) Se retornaron los pesos.

- **PASO N° 14:** Se obtuvo el error entrenamiento de la Red Neuronal.  
La función recibe las ventas diarias mensuales y retorna el error.

```

private double Error(double[][] EntrenamientoDatos)
{
    // Error medio cuadrado por cada época de entrenamiento.
    double sSumaCuadradoError = 0.0;
    //Primer número de entrada de valores en los datos de entrenamiento
    double[] xValores = new double[NumeroEntradas];
    //Ultimo número de entrada de valores.
    double[] tValores = new double[NumeroSalidas];
    //Recorremos cada una de las ventas diarias mensuales.
    for (int i = 0; i < EntrenamientoDatos.Length; ++i)
    {
        //Copiamos un rango de ventas diarias mensuales empezando desde el
        //primer elemento a otro arreglo empezando desde el primer elemento.
        Array.Copy(EntrenamientoDatos[i], xValores, NumeroEntradas);
        // obtener valores apuntados
        Array.Copy(EntrenamientoDatos[i], NumeroEntradas, tValores, 0, NumeroSalidas);
        // usamos salidas actual de pesos
        double[] yValues = this.ComputarSalidas(xValores);
        for (int j = 0; j < NumeroSalidas; ++j)
        {
            //Restamos las ventas diarias.
            double err = tValores[j] - yValues[j];
            //Sumamos el error elevado al cuadrado
            sSumaCuadradoError += err * err;
        }
    }
    //Dividimos la suma total entre la cantidad de elementos de la
    //matriz de entrenamiento
    return sSumaCuadradoError / EntrenamientoDatos.Length;
}

```

*Figura 27: Parte del código que indica los errores en cada iteración*  
Fuente: Elaboración Propia



La función realiza lo siguiente dependiendo la cantidad de iteraciones de la red:

---

Iteración Nro 01

22,949 - 21,8907410218949 = 1,05825897810507

24,07 - 21,8908178640268 = 2,17918213597322

23,537 - 21,8908649799954 = 1,64613502000459

20,132 - 21,8907677465347 = -1,75876774653474

23,218 - 21,8908981190527 = 1,3271018809473

20,316 - 21,8907239521207 = -1,57472395212073

...

0 - 0,00535548637867267 = -0,00535548637867267

0 - 0,00535548637867267 = -0,00535548637867267

0 - 0,00535548637867267 = -0,00535548637867267

0 - 0,00535548637867267 = -0,00535548637867267

0 - 0,00535548637867267 = -0,00535548637867267

Suma cuadrática del error = 410,040466667633

Error= 410,040466667633 / 200

Error= 2,050202333338165

Epoca = 2000

Iteración Nro 02

22,949 - 22,311217345267 = 0,637782654733048

24,07 - 22,3112179248765 = 1,75878207512346

23,537 - 22,3112181147717 = 1,22578188522831

20,132 - 22,3112162885951 = -2,17921628859511

23,218 - 22,311218534983 = 0,90678146501703

20,316 - 22,3112148572168 = -1,99521485721678

...

0 - 0,000383364774829253 = -0,000383364774829253

0 - 0,000383364774829253 = -0,000383364774829253

0 - 0,000383364774829253 = -0,000383364774829253

0 - 0,000383364774829253 = -0,000383364774829253

---

---

0 - 0,000383364774829253 = -0,000383364774829253

0 - 0,000383364774829253 = -0,000383364774829253

Suma cuadrática del error = 349,447895948342

Error= 349,447895948342 / 200

Error= 1,74723947974171

Epoca = 4000

Iteración Nro 03

22,949 - 22,2872861494055 = 0,661713850594499

24,07 - 22,2872866640235 = 1,7827133359765

23,537 - 22,2872868844155 = 1,24971311558449

20,132 - 22,2872865175514 = -2,15528651755137

23,218 - 22,2872869721063 = 0,93071302789372

20,316 - 22,2872863417306 = -1,97128634173061

...

0 - 0,00185036152603435 = -0,00185036152603435

0 - 0,00185036152603435 = -0,00185036152603435

0 - 0,00185036152603435 = -0,00185036152603435

0 - 0,00185036152603435 = -0,00185036152603435

0 - 0,00185036152603435 = -0,00185036152603435

0 - 0,00185036152603435 = -0,00185036152603435

Suma cuadrática del error = 351,531045408005

Error= 351,531045408005/ 200

Error= 1,757655227040025

Epoca = 6000

Iteración Nro 04

22,949 - 22,5049639014628 = 0,444036098537243

24,07 - 22,5049661782108 = 1,56503382178918

23,537 - 22,5049672570015 = 1,03203274299851

20,132 - 22,5049654402549 = -2,37296544025487

23,218 - 22,5049677251193 = 0,713032274880746

20,316 - 22,5049645854239 = -2,18896458542393

---

---

...

$$0 - 0,00144691034822064 = -0,00144691034822064$$

$$0 - 0,00144691034822064 = -0,00144691034822064$$

$$0 - 0,00144691034822064 = -0,00144691034822064$$

$$0 - 0,00144691034822064 = -0,00144691034822064$$

$$0 - 0,00144691034822064 = -0,00144691034822064$$

$$0 - 0,00144691034822064 = -0,00144691034822064$$

Suma cuadrática del error = 338,657459936171

$$\text{Error} = 338,657459936171 / 200$$

$$\text{Error} = 1,693287299680855$$

Epoca = 8000

Iteración Nro 01

$$22,949 - 21,8911457845377 = 1,05785421546226$$

$$24,07 - 21,8911460825985 = 2,17885391740152$$

$$23,537 - 21,8911456889428 = 1,64585431105724$$

$$20,132 - 21,8911460888793 = -1,75914608887932$$

$$23,218 - 21,8911453896237 = 1,32685461037629$$

$$20,316 - 21,8911460901415 = -1,57514609014153$$

...

$$0 - 0,0367445598127962 = -0,0367445598127962$$

$$0 - 0,0367445598127962 = -0,0367445598127962$$

$$0 - 0,0367445598127962 = -0,0367445598127962$$

$$0 - 0,0367445598127962 = -0,0367445598127962$$

$$0 - 0,0367445598127962 = -0,0367445598127962$$

$$0 - 0,0367445598127962 = -0,0367445598127962$$

Suma cuadrática del error = 889,269931584164

$$\text{Error} = 889,269931584164 / 200$$

$$\text{Error} = 4,44634965792082$$

Epoca = 2000

Iteración Nro 02

$$22,949 - 21,028571028676 = 1,92042897132395$$

---

---

24,07 - 21,0285711945279 = 3,04142880547208  
23,537 - 21,0285709831466 = 2,5084290168534  
20,132 - 21,0285711979344 = -0,896571197934378  
23,218 - 21,0285708249436 = 2,1894291750564  
20,316 - 21,0285711986168 = -0,712571198616754  
...  
0 - 0,00683306216287294 = -0,00683306216287294  
0 - 0,00683306216287294 = -0,00683306216287294  
0 - 0,00683306216287294 = -0,00683306216287294  
0 - 0,00683306216287294 = -0,00683306216287294  
0 - 0,00683306216287294 = -0,00683306216287294  
0 - 0,00683306216287294 = -0,00683306216287294  
Suma cuadrática del error = 1135,77821070236  
Error= 1135,77821070236/ 200  
Error= 5,6788910535118  
Epoca = 4000

#### Iteración Nro 03

22,949 - 22,1464020019903 = 0,802597998009723  
24,07 - 22,1464021183869 = 1,92359788161309  
23,537 - 22,146401973261 = 1,39059802673903  
20,132 - 22,1464021207401 = -2,01440212074008  
23,218 - 22,1464018657091 = 1,07159813429088  
20,316 - 22,1464021212104 = -1,83040212121044  
...  
0 - 0,000662796644802288 = -0,000662796644802288  
0 - 0,000662796644802288 = -0,000662796644802288  
0 - 0,000662796644802288 = -0,000662796644802288  
0 - 0,000662796644802288 = -0,000662796644802288  
0 - 0,000662796644802288 = -0,000662796644802288  
0 - 0,000662796644802288 = -0,000662796644802288  
Suma cuadrática del error = 857,600198755952  
Error= 857,600198755952/ 200

---

---

Error= 4,28800099377976

Epoca = 6000

Iteración Nro 04

22,949 - 21,383936080806 = 1,56506391919401

24,07 - 21,3839361637334 = 2,68606383626656

23,537 - 21,3839360618929 = 2,15306393810715

20,132 - 21,3839361653918 = -1,25193616539175

23,218 - 21,3839359869329 = 1,83406401306711

20,316 - 21,3839361657227 = -1,06793616572271

...

0 - 0,00239968013910019 = -0,00239968013910019

0 - 0,00239968013910019 = -0,00239968013910019

0 - 0,00239968013910019 = -0,00239968013910019

0 - 0,00239968013910019 = -0,00239968013910019

0 - 0,00239968013910019 = -0,00239968013910019

0 - 0,00239968013910019 = -0,00239968013910019

Suma cuadrática del error = 1008,05405385301

Error= 1008,05405385301/ 200

Error= 5,04027026926505

Epoca = 8000

---

- **PASO N° 15: Computar Salidas.**

Se ejecuta la neurona para el pronóstico de las ventas futuras, se suman los nodos ocultos del arreglo de marca y las sumas de los nodos de salida de la neurona, asimismo se copian los valores X para la entrada, y se computan los i ocultos para sumar de pesos y entradas.

Para la ejecución de la Red se ha desarrollado el algoritmo que a continuación se muestra en la siguiente figura:

```

//VENTA PRONOSTICADA
public double[] Computar_Salidas(double[] xValores, StreamWriter sw)
{
    double[] Oculito_Sumas = new double[Numero_Oculito]; // Suma de nodos oculito arreglo de marca
    double[] Salida_Sumas = new double[Numero_Salida]; // Sumas nodo salida

    for (int i = 0; i < xValores.Length; ++i)
    { // Copiar x-valores para entradas
        this.Entrada_Nodos[i] = xValores[i];
        //sw.WriteLine(i + " " + this.Entrada_Nodos[i]);
    }

    for (int j = 0; j < Numero_Oculito; ++j) // Computar i-oculito suma de pesos * entradas
        for (int i = 0; i < Numero_Entrada; ++i)
            Oculito_Sumas[j] += this.Entrada_Nodos[i] * this.Entrada_Oculito_Pesos[i][j];

    for (int i = 0; i < Numero_Oculito; ++i) // Adicionamos favoritismo para sumas oculita
        Oculito_Sumas[i] += this.Oculito_Favorito[i];

    for (int i = 0; i < Numero_Oculito; ++i) // Aplicamos activacion
        this.Oculito_Nodos[i] = Tangente_Hyperbolica(Oculito_Sumas[i]);

    for (int j = 0; j < Numero_Salida; ++j) // Computa h-salida suma de pesos * oculito salidas
        for (int i = 0; i < Numero_Oculito; ++i)
            Salida_Sumas[j] += Oculito_Nodos[i] * Oculito_Salida_Pesos[i][j];

    for (int i = 0; i < Numero_Salida; ++i) // Adicionamos favoritismos para sumas salida
        Salida_Sumas[i] += Salida_Favoritos[i];

    Array.Copy(Salida_Sumas, this.Salida_Nodos, Salida_Sumas.Length); // Solo de verdad valor 1

    double[] Retener_Resultado = new double[Numero_Salida]; // Se puede definir una captura de entradas
    Array.Copy(this.Salida_Nodos, Retener_Resultado, Retener_Resultado.Length);
    return Retener_Resultado;
}

```

Figura 28: Parte del código que indica la función para el pronóstico de la venta. Fuente: Elaboración Propia

## Resultados de la ejecución de la red neuronal

Inicio red neuronal serie de tiempos.

El objetivo es pronosticar las ventas sobre un tiempo determinado.

Datos desde el mes de x del año y hasta el mes de x del año y.

Normalizar Entrenamiento Datos:

---

[ 0]	20,784	20,308	22,495	23,666	22,949
[ 1]	23,527	22,121	21,198	22,972	24,070
[ 2]	20,970	22,829	23,985	23,324	23,537
[ 3]	24,210	20,348	22,531	22,060	20,132
[ 4]	20,627	24,004	22,551	24,529	23,218
[ 5]	24,469	20,859	22,327	20,735	20,316
[ 6]	20,864	24,994	23,013	25,288	23,665
[ 7]	24,783	21,141	23,918	24,326	20,715

---

---

[ 8]	23,977	23,078	21,049	20,641	25,060
[ 9]	21,155	22,253	22,526	22,051	21,058
[10]	24,464	22,002	21,119	20,612	23,132
[11]	20,841	21,684	25,206	22,779	22,263
[12]	21,793	24,110	23,809	23,036	23,263
[13]	21,828	24,179	23,932	20,928	25,026
[14]	20,441	21,375	24,392	24,949	24,087
[15]	21,345	24,639	24,735	23,567	23,057
[16]	20,598	20,289	20,499	25,092	24,312
[17]	22,816	20,964	24,467	22,242	24,199
[18]	23,399	23,688	22,874	25,384	22,736
[19]	20,551	21,190	21,020	22,216	25,246
[20]	23,429	24,879	20,154	23,999	24,905
[21]	23,273	24,360	24,165	24,339	24,571
[22]	25,182	20,671	25,075	24,367	21,890
[23]	22,550	22,144	24,189	24,689	20,469
[24]	22,364	21,688	24,252	20,718	22,076
[25]	21,943	20,900	24,741	20,205	21,996
[26]	21,750	21,072	20,694	25,033	20,844
[27]	23,941	20,888	22,433	21,663	23,584
[28]	20,231	20,737	25,088	21,371	23,592
[29]	23,372	21,487	24,015	21,045	20,854
[30]	23,545	23,873	22,393	20,609	20,463
[31]	23,508	23,761	20,601	24,885	22,008
[32]	21,640	24,052	23,221	22,930	22,334
[33]	21,604	23,164	21,967	24,747	24,837
[34]	21,204	22,345	20,796	20,605	22,955
[35]	21,911	20,676	23,686	24,795	23,114
[36]	20,161	21,949	20,647	20,790	20,707
[37]	23,340	25,240	24,822	21,592	23,711
[38]	20,717	24,230	24,858	24,714	22,643
[39]	23,589	22,009	20,150	22,399	20,325
[40]	20,687	25,313	22,951	20,880	22,113
[41]	20,171	23,696	24,726	21,583	23,589
[42]	21,611	23,223	20,250	20,847	20,702
[43]	20,016	24,650	20,947	20,067	21,782
[44]	22,688	22,839	25,100	20,739	22,429
[45]	24,528	20,158	23,485	24,946	20,399
[46]	24,449	22,190	24,146	20,042	21,057
[47]	22,873	23,719	22,529	20,228	25,269
[48]	21,929	24,491	22,696	23,933	24,676
[49]	22,827	24,523	21,267	23,400	25,378
[50]	20,638	20,814	24,371	22,085	22,588
[51]	25,125	25,307	22,718	24,269	21,019
[52]	23,016	23,205	23,891	24,984	23,018
[53]	21,788	22,097	24,323	25,151	23,656
[54]	23,752	20,270	20,098	24,424	24,661
[55]	25,233	25,022	21,056	22,556	21,503
[56]	22,137	24,389	20,683	21,522	24,075

---

---

[ 57]	22,729	24,283	20,513	20,654	24,054
[ 58]	24,932	23,770	21,347	22,953	21,884
[ 59]	20,344	20,018	21,535	25,364	21,798
[ 60]	21,546	20,580	21,941	23,650	20,934
[ 61]	20,034	21,878	25,387	23,047	20,544
[ 62]	23,091	23,525	24,643	22,390	20,566
[ 63]	22,741	23,240	21,788	22,671	20,712
[ 64]	21,450	23,933	23,996	21,444	22,300
[ 65]	25,305	24,421	21,395	20,679	25,375
[ 66]	20,308	22,515	21,200	25,072	24,471
[ 67]	20,944	22,277	22,180	23,601	24,314
[ 68]	21,322	25,299	20,511	20,072	23,311
[ 69]	21,104	22,942	22,270	24,490	23,459
[ 70]	20,609	21,213	21,199	20,441	22,956
[ 71]	20,519	24,422	24,133	20,914	23,383
[ 72]	22,593	24,146	21,622	25,215	20,078
[ 73]	21,083	22,874	23,639	21,826	22,484
[ 74]	20,816	22,456	20,394	20,215	20,661
[ 75]	20,561	25,299	20,265	24,973	21,889
[ 76]	22,802	23,468	20,363	25,139	23,185
[ 77]	22,361	23,562	24,905	24,389	21,636
[ 78]	24,675	20,879	24,848	21,969	24,232
[ 79]	20,457	22,105	20,400	24,391	22,728
[ 80]	20,397	23,542	21,535	20,160	22,864
[ 81]	23,128	23,003	25,169	21,404	20,706
[ 82]	21,188	24,001	20,205	24,916	22,006
[ 83]	22,396	21,003	23,321	24,735	20,302
[ 84]	21,051	20,815	20,363	24,504	23,794
[ 85]	21,964	20,468	24,011	22,127	22,112
[ 86]	25,038	20,401	20,105	21,315	20,824
[ 87]	21,876	23,143	21,543	22,115	20,918
[ 88]	24,471	20,531	21,687	21,884	25,335
[ 89]	25,334	24,821	20,464	20,468	25,227
[ 90]	22,671	24,122	21,474	20,586	22,073
[ 91]	24,827	22,931	22,234	21,598	23,237
[ 92]	21,855	23,980	24,177	23,869	20,436
[ 93]	24,918	21,761	25,208	21,619	21,847
[ 94]	24,886	20,499	21,060	24,402	21,458
[ 95]	22,910	20,176	23,343	22,748	24,282
[ 96]	20,869	23,118	20,434	20,422	23,288
[ 97]	22,682	22,315	24,710	22,937	20,295
[ 98]	20,900	25,169	22,601	21,246	23,881
[ 99]	24,587	24,117	20,186	20,949	22,148
[100]	21,717	22,621	25,293	23,631	25,358
[101]	23,137	23,716	25,045	23,732	20,176
[102]	21,936	23,201	24,903	25,098	24,619
[103]	23,973	21,969	20,630	23,618	21,764
[104]	24,921	23,223	23,840	20,226	22,404
[105]	23,106	24,726	22,366	22,865	21,370

---



---

[106]	25,221	23,932	23,150	20,973	24,400
[107]	20,804	21,352	22,559	22,865	22,645
[108]	22,248	22,177	24,546	23,816	20,746
[109]	21,391	21,607	21,369	24,622	24,329
[110]	24,655	22,234	20,450	22,369	21,178
[111]	23,955	20,161	24,844	23,186	24,422
[112]	20,417	24,199	24,611	21,297	21,163
[113]	21,172	21,330	22,211	21,101	21,885
[114]	21,630	25,162	20,724	23,512	21,064
[115]	20,115	21,924	20,078	21,153	21,472
[116]	24,539	24,261	24,888	23,551	20,564
[117]	23,620	21,817	23,724	21,074	22,502
[118]	25,215	21,452	24,607	21,419	20,459
[119]	23,356	20,555	23,476	21,199	23,293
[120]	24,637	21,883	22,217	21,123	22,627
[121]	24,865	23,241	22,429	25,364	24,059
[122]	23,127	20,242	21,981	21,414	24,617
[123]	20,589	20,830	22,681	24,531	24,221
[124]	21,558	24,721	23,515	24,495	23,952
[125]	20,489	20,413	20,115	23,088	22,448
[126]	24,860	21,480	21,978	23,621	23,443
[127]	22,003	21,177	25,097	23,879	25,127
[128]	21,804	22,541	25,023	22,372	20,070
[129]	22,249	22,465	23,199	24,409	21,283
[130]	20,872	21,551	23,042	20,736	20,429
[131]	24,563	20,294	23,966	23,886	23,019
[132]	21,027	21,252	21,388	20,233	23,968
[133]	20,784	24,872	22,250	24,160	21,819
[134]	24,589	21,240	20,793	21,585	24,407
[135]	22,818	21,844	22,198	20,756	20,705
[136]	23,146	22,752	21,833	23,162	21,965
[137]	21,206	21,524	21,414	22,523	22,081
[138]	24,493	20,196	20,011	20,988	20,847
[139]	23,057	22,313	22,370	20,538	20,993
[140]	23,167	21,049	23,461	24,769	23,887
[141]	20,634	23,186	21,066	24,700	22,968
[142]	24,663	22,738	24,009	22,026	24,617
[143]	20,810	23,768	21,952	21,478	23,427
[144]	0,000	0,000	0,000	0,000	0,000
[145]	0,000	0,000	0,000	0,000	0,000
[146]	0,000	0,000	0,000	0,000	0,000
[147]	0,000	0,000	0,000	0,000	0,000
[148]	0,000	0,000	0,000	0,000	0,000
[149]	0,000	0,000	0,000	0,000	0,000
[150]	0,000	0,000	0,000	0,000	0,000
[151]	0,000	0,000	0,000	0,000	0,000
[152]	0,000	0,000	0,000	0,000	0,000
[153]	0,000	0,000	0,000	0,000	0,000
[154]	0,000	0,000	0,000	0,000	0,000

---

---

[155]	0,000	0,000	0,000	0,000	0,000
[156]	0,000	0,000	0,000	0,000	0,000
[157]	0,000	0,000	0,000	0,000	0,000
[158]	0,000	0,000	0,000	0,000	0,000
[159]	0,000	0,000	0,000	0,000	0,000
[160]	0,000	0,000	0,000	0,000	0,000
[161]	0,000	0,000	0,000	0,000	0,000
[162]	0,000	0,000	0,000	0,000	0,000
[163]	0,000	0,000	0,000	0,000	0,000
[164]	0,000	0,000	0,000	0,000	0,000
[165]	0,000	0,000	0,000	0,000	0,000
[166]	0,000	0,000	0,000	0,000	0,000
[167]	0,000	0,000	0,000	0,000	0,000
[168]	0,000	0,000	0,000	0,000	0,000
[169]	0,000	0,000	0,000	0,000	0,000
[170]	0,000	0,000	0,000	0,000	0,000
[171]	0,000	0,000	0,000	0,000	0,000
[172]	0,000	0,000	0,000	0,000	0,000
[173]	0,000	0,000	0,000	0,000	0,000
[174]	0,000	0,000	0,000	0,000	0,000
[175]	0,000	0,000	0,000	0,000	0,000
[176]	0,000	0,000	0,000	0,000	0,000
[177]	0,000	0,000	0,000	0,000	0,000
[178]	0,000	0,000	0,000	0,000	0,000
[179]	0,000	0,000	0,000	0,000	0,000
[180]	0,000	0,000	0,000	0,000	0,000
[181]	0,000	0,000	0,000	0,000	0,000
[182]	0,000	0,000	0,000	0,000	0,000
[183]	0,000	0,000	0,000	0,000	0,000
[184]	0,000	0,000	0,000	0,000	0,000
[185]	0,000	0,000	0,000	0,000	0,000
[186]	0,000	0,000	0,000	0,000	0,000

---

Creando una 4-12-1 red neuronal

Fijamos Maxima Epoca = 10000

Fijamos Ratio de Aprendizaje = 0,01

Empezamos Entrenamiento

Epoca = 2000 Error = 2,0502

Epoca = 4000 Error = 1,7472

Epoca = 6000 Error = 1,7577

Epoca = 8000 Error = 1,6933

Ver

Final Modelo Red Neuronal Pesos y Bias:

---

0,0461096287195837	0,2703605738738640
1,5706653774998000	1,8052062973006200
0,4974834178267580	0,6190974902985530
0,0856372246972597	0,3218794225826330
0,0022159215175112	0,2965794469754730
0,1506643846065010	0,3772808540127120
0,3064068314005360	0,5756043691105970
0,0570085126910750	0,3260162820225290
0,0797324340966527	0,3454280471675420
0,0544228594646140	0,2336429389132000
0,7460455114326060	0,9623763531367600
0,0767621133051983	0,3007640105010700
0,2356219100017080	-0,1027401660952860
1,9618964020017900	-0,0189832153676310
0,6635493071136250	-0,0914151995725199
0,2619388313461330	-0,1067712232966040
0,2104537241018990	-0,0897081965208239
0,3469470806240290	-0,1029348542353530
0,5074089441269300	-0,0770813603191073
0,2652956201633960	-0,1000955239122600
0,2460637517477150	-0,1075455617393020
0,2034415729160010	-0,0943789912790329
1,0257411681938100	-0,0766598661698165
0,2486279563067500	-0,0966212826004157
0,0115644631079411	1,7355528691308100
1,6766593692547300	1,6497592282606000
0,4621147191711550	1,7615798309477100
0,0577305554365701	1,7612274388428500
-0,0390455512734092	1,6797374739322500
0,1289492719525390	1,7561673423746200
0,3093326814148590	1,6795608378098500
0,0122213439468744	1,7309442420851700
0,0308576882889032	1,7620995620200600
0,0615004876074101	1,7031767224096800
0,7766755005633250	1,7552427654123800
0,0367742825231242	1,7169148632990100
	1,8392308817325400

---

Precisión Modelo (+/- 30) en Datos de Entrenamiento = 0,3500

Predicción Pasajera por Mes:

2238

Final Tiempo Series

22,3801504430766 0

Normalizar Entrenamiento Datos:

[ 0]	20,784	20,308	22,495	23,666	22,949	[101]	23,137	23,716	25,045	23,732	20,176
[ 1]	23,527	22,121	21,198	22,972	24,070	[102]	21,936	23,201	24,903	25,098	24,619
[ 2]	20,970	22,829	23,985	23,324	23,537	[103]	23,973	21,969	20,630	23,618	21,764
[ 3]	24,210	20,348	22,531	22,060	20,132	[104]	24,921	23,223	23,840	20,226	22,404
[ 4]	20,627	24,004	22,551	24,529	23,218	[105]	23,106	24,726	22,366	22,865	21,370
[ 5]	24,469	20,859	22,327	20,735	20,316	[106]	25,221	23,932	23,150	20,973	24,400
[ 6]	20,864	24,994	23,013	25,288	23,665	[107]	20,804	21,352	22,559	22,865	22,645
[ 7]	24,783	21,141	23,918	24,326	20,715	[108]	22,248	22,177	24,546	23,816	20,746
[ 8]	23,977	23,078	21,049	20,641	25,060	[109]	21,391	21,607	21,369	24,622	24,329
[ 9]	21,155	22,253	22,526	22,051	21,058	[110]	24,655	22,234	20,450	22,369	21,178
[10]	24,464	22,002	21,119	20,612	23,132	[111]	23,955	20,161	24,844	23,186	24,422
[11]	20,841	21,684	25,206	22,779	22,263	[112]	20,417	24,199	24,611	21,297	21,163
[12]	21,793	24,110	23,809	23,036	23,263	[113]	21,172	21,330	22,211	21,101	21,885
[13]	21,828	24,179	23,932	20,928	25,026	[114]	21,630	25,162	20,724	23,512	21,064
[14]	20,441	21,375	24,392	24,949	24,087	[115]	20,115	21,924	20,078	21,153	21,472
[15]	21,345	24,639	24,735	23,567	23,057	[116]	24,539	24,261	24,888	23,551	20,564
[16]	20,598	20,289	20,499	25,092	24,312	[117]	23,620	21,817	23,724	21,074	22,502
[17]	22,816	20,964	24,467	22,242	24,199	[118]	25,215	21,452	24,607	21,419	20,459
[18]	23,399	23,688	22,874	25,384	22,736	[119]	23,356	20,555	23,476	21,199	23,293
[19]	20,551	21,190	21,020	22,216	25,246	[120]	24,637	21,883	22,217	21,123	22,627
[20]	23,429	24,879	20,154	23,999	24,905	[121]	24,865	23,241	22,429	25,364	24,059
[21]	23,273	24,360	24,165	24,339	24,571	[122]	23,127	20,242	21,981	21,414	24,617
[22]	25,182	20,671	25,075	24,367	21,890	[123]	20,589	20,830	22,681	24,531	24,221
[23]	22,550	22,144	24,189	24,689	20,469	[124]	21,558	24,721	23,515	24,495	23,952
[24]	22,364	21,688	24,252	20,718	22,076	[125]	20,489	20,413	20,115	23,088	22,448
[25]	21,943	20,900	24,741	20,205	21,996	[126]	24,860	21,480	21,978	23,621	23,443
[26]	21,750	21,072	20,694	25,033	20,844	[127]	22,003	21,177	25,097	23,879	25,127
[27]	23,941	20,888	22,433	21,663	23,584	[128]	21,804	22,541	25,023	22,372	20,070
[28]	20,231	20,737	25,088	21,371	23,592	[129]	22,249	22,465	23,199	24,409	21,283
[29]	23,372	21,487	24,015	21,045	20,854	[130]	20,872	21,551	23,042	20,736	20,429
[30]	23,545	23,873	22,393	20,609	20,463	[131]	24,563	20,294	23,966	23,886	23,019
[31]	23,508	23,761	20,601	24,885	22,008	[132]	21,027	21,252	21,388	20,233	23,968
[32]	21,640	24,052	23,221	22,930	22,334	[133]	20,784	24,872	22,250	24,160	21,819
[33]	21,604	23,164	21,967	24,747	24,837	[134]	24,589	21,240	20,793	21,585	24,407
[34]	21,204	22,345	20,796	20,605	22,955	[135]	22,818	21,844	22,198	20,756	20,705
[35]	21,911	20,676	23,686	24,795	23,114	[136]	23,146	22,752	21,833	23,162	21,965
[36]	20,161	21,949	20,647	20,790	20,707	[137]	21,206	21,524	21,414	22,523	22,081
[37]	23,340	25,240	24,822	21,592	23,711	[138]	24,493	20,196	20,011	20,988	20,847
[38]	20,717	24,230	24,858	24,714	22,643	[139]	23,057	22,313	22,370	20,538	20,993
[39]	23,589	22,009	20,150	22,399	20,325	[140]	23,167	21,049	23,461	24,769	23,887
[40]	20,687	25,313	22,951	20,880	22,113	[141]	20,634	23,186	21,066	24,700	22,968
[41]	20,171	23,696	24,726	21,583	23,589	[142]	24,663	22,738	24,009	22,026	24,617
[42]	21,611	23,223	20,250	20,847	20,702	[143]	20,810	23,768	21,952	21,478	23,427
[43]	20,016	24,650	20,947	20,067	21,782	[144]	22,380	0,000	0,000	0,000	0,000
[44]	22,688	22,839	25,100	20,739	22,429	[145]	0,000	0,000	0,000	0,000	0,000
[45]	24,528	20,158	23,485	24,946	20,399	[146]	0,000	0,000	0,000	0,000	0,000
[46]	24,449	22,190	24,146	20,042	21,057	[147]	0,000	0,000	0,000	0,000	0,000
[47]	22,873	23,719	22,529	20,228	25,269	[148]	0,000	0,000	0,000	0,000	0,000
[48]	21,929	24,491	22,696	23,933	24,676	[149]	0,000	0,000	0,000	0,000	0,000
[49]	22,827	24,523	21,267	23,400	25,378	[150]	0,000	0,000	0,000	0,000	0,000
[50]	20,638	20,814	24,371	22,085	22,588	[151]	0,000	0,000	0,000	0,000	0,000
[51]	25,125	25,307	22,718	24,269	21,019	[152]	0,000	0,000	0,000	0,000	0,000
[52]	23,016	23,205	23,891	24,984	23,018	[153]	0,000	0,000	0,000	0,000	0,000
[53]	21,788	22,097	24,323	25,151	23,656	[154]	0,000	0,000	0,000	0,000	0,000

[ 54]	23,752	20,270	20,098	24,424	24,661	[155]	0,000	0,000	0,000	0,000	0,000
[ 55]	25,233	25,022	21,056	22,556	21,503	[156]	0,000	0,000	0,000	0,000	0,000
[ 56]	22,137	24,389	20,683	21,522	24,075	[157]	0,000	0,000	0,000	0,000	0,000
[ 57]	22,729	24,283	20,513	20,654	24,054	[158]	0,000	0,000	0,000	0,000	0,000
[ 58]	24,932	23,770	21,347	22,953	21,884	[159]	0,000	0,000	0,000	0,000	0,000
[ 59]	20,344	20,018	21,535	25,364	21,798	[160]	0,000	0,000	0,000	0,000	0,000
[ 60]	21,546	20,580	21,941	23,650	20,934	[161]	0,000	0,000	0,000	0,000	0,000
[ 61]	20,034	21,878	25,387	23,047	20,544	[162]	0,000	0,000	0,000	0,000	0,000
[ 62]	23,091	23,525	24,643	22,390	20,566	[163]	0,000	0,000	0,000	0,000	0,000
[ 63]	22,741	23,240	21,788	22,671	20,712	[164]	0,000	0,000	0,000	0,000	0,000
[ 64]	21,450	23,933	23,996	21,444	22,300	[165]	0,000	0,000	0,000	0,000	0,000
[ 65]	25,305	24,421	21,395	20,679	25,375	[166]	0,000	0,000	0,000	0,000	0,000
[ 66]	20,308	22,515	21,200	25,072	24,471	[167]	0,000	0,000	0,000	0,000	0,000
[ 67]	20,944	22,277	22,180	23,601	24,314	[168]	0,000	0,000	0,000	0,000	0,000
[ 68]	21,322	25,299	20,511	20,072	23,311	[169]	0,000	0,000	0,000	0,000	0,000
[ 69]	21,104	22,942	22,270	24,490	23,459	[170]	0,000	0,000	0,000	0,000	0,000
[ 70]	20,609	21,213	21,199	20,441	22,956	[171]	0,000	0,000	0,000	0,000	0,000
[ 71]	20,519	24,422	24,133	20,914	23,383	[172]	0,000	0,000	0,000	0,000	0,000
[ 72]	22,593	24,146	21,622	25,215	20,078	[173]	0,000	0,000	0,000	0,000	0,000
[ 73]	21,083	22,874	23,639	21,826	22,484	[174]	0,000	0,000	0,000	0,000	0,000
[ 74]	20,816	22,456	20,394	20,215	20,661	[175]	0,000	0,000	0,000	0,000	0,000
[ 75]	20,561	25,299	20,265	24,973	21,889	[176]	0,000	0,000	0,000	0,000	0,000
[ 76]	22,802	23,468	20,363	25,139	23,185	[177]	0,000	0,000	0,000	0,000	0,000
[ 77]	22,361	23,562	24,905	24,389	21,636	[178]	0,000	0,000	0,000	0,000	0,000
[ 78]	24,675	20,879	24,848	21,969	24,232	[179]	0,000	0,000	0,000	0,000	0,000
[ 79]	20,457	22,105	20,400	24,391	22,728	[180]	0,000	0,000	0,000	0,000	0,000
[ 80]	20,397	23,542	21,535	20,160	22,864	[181]	0,000	0,000	0,000	0,000	0,000
[ 81]	23,128	23,003	25,169	21,404	20,706	[182]	0,000	0,000	0,000	0,000	0,000
[ 82]	21,188	24,001	20,205	24,916	22,006	[183]	0,000	0,000	0,000	0,000	0,000
[ 83]	22,396	21,003	23,321	24,735	20,302	[184]	0,000	0,000	0,000	0,000	0,000
[ 84]	21,051	20,815	20,363	24,504	23,794	[185]	0,000	0,000	0,000	0,000	0,000
[ 85]	21,964	20,468	24,011	22,127	22,112	[186]	0,000	0,000	0,000	0,000	0,000
[ 86]	25,038	20,401	20,105	21,315	20,824	[187]	0,000	0,000	0,000	0,000	0,000
[ 87]	21,876	23,143	21,543	22,115	20,918	[188]	0,000	0,000	0,000	0,000	0,000
[ 88]	24,471	20,531	21,687	21,884	25,335	[189]	0,000	0,000	0,000	0,000	0,000
[ 89]	25,334	24,821	20,464	20,468	25,227	[190]	0,000	0,000	0,000	0,000	0,000
[ 90]	22,671	24,122	21,474	20,586	22,073	[191]	0,000	0,000	0,000	0,000	0,000
[ 91]	24,827	22,931	22,234	21,598	23,237	[192]	0,000	0,000	0,000	0,000	0,000
[ 92]	21,855	23,980	24,177	23,869	20,436	[193]	0,000	0,000	0,000	0,000	0,000
[ 93]	24,918	21,761	25,208	21,619	21,847	[194]	0,000	0,000	0,000	0,000	0,000
[ 94]	24,886	20,499	21,060	24,402	21,458	[195]	0,000	0,000	0,000	0,000	0,000
[ 95]	22,910	20,176	23,343	22,748	24,282	[196]	0,000	0,000	0,000	0,000	0,000
[ 96]	20,869	23,118	20,434	20,422	23,288	[197]	0,000	0,000	0,000	0,000	0,000
[ 97]	22,682	22,315	24,710	22,937	20,295	[198]	0,000	0,000	0,000	0,000	0,000
[ 98]	20,900	25,169	22,601	21,246	23,881	[199]	0,000	0,000	0,000	0,000	0,000
[ 99]	24,587	24,117	20,186	20,949	22,148						
[100]	21,717	22,621	25,293	23,631							
	25,358										

Creando una 4-12-1 red neuronal

Fijamos Maxima Epoca = 10000

Fijamos Ratio de Aprendizaje = 0,01

Empezamos Entrenamiento

Epoca = 2000 Error = 4,4463

Epoca = 4000 Error = 5,6789

Epoca = 6000 Error = 4,2880

Epoca = 8000 Error = 5,0403

Ver

Final Modelo Red Neuronal Pesos y Bias:

---

-2,1051079349356000	0,0732731824549982
-1,4941511827987900	0,0833450594025279
-1,1061735237682100	0,0681447624316121
-2,4584808811047200	0,0736904725158530
-1,4474463130478000	0,0720714538095534
-1,9633502804588600	0,0665922976726581
-1,9432754443759200	0,0613937204359983
-3,1536558047134100	0,0845282276619879
-1,0414621211143900	0,0539857466984566
-0,6363419449093740	0,0684454202857999
-2,4559711543036700	0,0633553078274614
-2,6355138260591900	0,0131284395718799
0,0695298899637291	0,3656382598702660
0,0735965232954498	0,1371566002207880
0,0844756759646985	-0,2149843650066490
0,0679518873996883	0,4347303239045360
0,0741905033967864	-0,0539947192600959
0,0728782491620749	0,2871654832433670
0,0673695961345250	-0,2778928589975720
0,0624313769923803	0,1858978972750940
0,0849699703934965	0,5267492559790230
0,0579408324495435	-0,2583794910456760
0,0697114224042797	-0,0572234060605274
0,0635554202157232	-1,5825719952154800
0,0687584235315322	-1,9333822714896600
0,0729400569983514	-1,6971574303136300
0,0834189522060213	-1,3540765625180100
0,0676127585340497	-2,0059414003915000
0,0732431327030759	-1,5139445931235500
0,0711383971606407	-1,8564923493211700
0,0662635392296037	-1,2943073633685500
0,0615626082174701	-1,7453097862364400
0,0835072383188765	-2,0986078764635900
0,0571573941960348	-1,3085612841152700
0,0684303152720717	-1,5168037617450300
0,0634019090030278	2,4663923116681800
0,0685086580587125	

---

Precisión Modelo (+/- 30) en Datos de Entrenamiento = 0,3600

Predicción Pasajera por Mes:

2229

Final Tiempo Series

22,2937741930667 1

Normalizar Entrenamiento Datos:

[101]	23,13	23,71	25,04	23,73	20,17	[149]	0,00	0,00	0,00	0,00	0,00
[102]	21,93	23,20	24,90	25,09	24,61	[150]	0,00	0,00	0,00	0,00	0,00
[103]	23,97	21,96	20,63	23,61	21,76	[151]	0,00	0,00	0,00	0,00	0,00
[104]	24,92	23,22	23,84	20,22	22,40	[152]	0,00	0,00	0,00	0,00	0,00
[105]	23,10	24,72	22,36	22,86	21,37	[158]	0,00	0,00	0,00	0,00	0,00
[106]	25,22	23,93	23,15	20,97	24,40	[159]	0,00	0,00	0,00	0,00	0,00
[107]	20,80	21,35	22,55	22,86	22,64	[160]	0,00	0,00	0,00	0,00	0,00
[108]	22,24	22,17	24,54	23,81	20,74	[161]	0,00	0,00	0,00	0,00	0,00
[109]	21,39	21,60	21,36	24,62	24,32	[153]	0,00	0,00	0,00	0,00	0,00
[110]	24,65	22,23	20,45	22,36	21,17	[154]	0,00	0,00	0,00	0,00	0,00
[111]	23,95	20,16	24,84	23,18	24,42	[155]	0,00	0,00	0,00	0,00	0,00
[112]	20,41	24,19	24,61	21,29	21,16	[156]	0,00	0,00	0,00	0,00	0,00
[113]	21,17	21,33	22,21	21,10	21,88	[157]	0,00	0,00	0,00	0,00	0,00
[114]	21,63	25,16	20,72	23,51	21,06	[162]	0,00	0,00	0,00	0,00	0,00
[115]	20,11	21,92	20,07	21,15	21,47	[163]	0,00	0,00	0,00	0,00	0,00
[116]	24,53	24,26	24,88	23,55	20,56	[164]	0,00	0,00	0,00	0,00	0,00
[117]	23,62	21,81	23,72	21,07	22,50	[165]	0,00	0,00	0,00	0,00	0,00
[118]	25,21	21,45	24,60	21,41	20,45	[166]	0,00	0,00	0,00	0,00	0,00
[119]	23,35	20,55	23,47	21,19	23,29	[167]	0,00	0,00	0,00	0,00	0,00
[120]	24,63	21,88	22,21	21,12	22,62	[168]	0,00	0,00	0,00	0,00	0,00
[121]	24,86	23,24	22,42	25,36	24,05	[169]	0,00	0,00	0,00	0,00	0,00
[122]	23,12	20,24	21,98	21,41	24,61	[170]	0,00	0,00	0,00	0,00	0,00
[123]	20,58	20,83	22,68	24,53	24,22	[171]	0,00	0,00	0,00	0,00	0,00
[124]	21,55	24,72	23,51	24,49	23,95	[172]	0,00	0,00	0,00	0,00	0,00
[125]	20,48	20,41	20,11	23,08	22,44	[173]	0,00	0,00	0,00	0,00	0,00
[126]	24,86	21,48	21,97	23,62	23,44	[174]	0,00	0,00	0,00	0,00	0,00
[127]	22,00	21,17	25,09	23,87	25,12	[175]	0,00	0,00	0,00	0,00	0,00
[128]	21,80	22,54	25,02	22,37	20,07	[176]	0,00	0,00	0,00	0,00	0,00
[129]	22,24	22,46	23,19	24,40	21,28	[177]	0,00	0,00	0,00	0,00	0,00
[130]	20,87	21,55	23,04	20,73	20,42	[178]	0,00	0,00	0,00	0,00	0,00
[131]	24,56	20,29	23,96	23,88	23,01	[179]	0,00	0,00	0,00	0,00	0,00
[132]	21,02	21,25	21,38	20,23	23,96	[180]	0,00	0,00	0,00	0,00	0,00
[133]	20,78	24,87	22,25	24,16	21,81	[181]	0,00	0,00	0,00	0,00	0,00
[134]	24,58	21,24	20,79	21,58	24,40	[182]	0,00	0,00	0,00	0,00	0,00
[135]	22,81	21,84	22,19	20,75	20,70	[183]	0,00	0,00	0,00	0,00	0,00
[136]	23,14	22,75	21,83	23,16	21,96	[184]	0,00	0,00	0,00	0,00	0,00
[137]	21,20	21,52	21,41	22,52	22,08	[185]	0,00	0,00	0,00	0,00	0,00
[138]	24,49	20,19	20,01	20,98	20,84	[186]	0,00	0,00	0,00	0,00	0,00
[139]	23,05	22,31	22,37	20,53	20,99	[187]	0,00	0,00	0,00	0,00	0,00
[140]	23,16	21,04	23,46	24,76	23,88	[188]	0,00	0,00	0,00	0,00	0,00
[141]	20,63	23,18	21,06	24,70	22,96	[189]	0,00	0,00	0,00	0,00	0,00
[142]	24,66	22,73	24,00	22,02	24,61	[190]	0,00	0,00	0,00	0,00	0,00

[143]	20,81	23,76	21,95	21,47	23,42	[191]	0,00	0,00	0,00	0,00	0,00
[144]	22,38	22,29	0,00	0,00	0,00	[192]	0,00	0,00	0,00	0,00	0,00
[145]	0,00	0,00	0,00	0,00	0,00	[193]	0,00	0,00	0,00	0,00	0,00
[146]	0,00	0,00	0,00	0,00	0,00	[194]	0,00	0,00	0,00	0,00	0,00
[147]	0,00	0,00	0,00	0,00	0,00	[195]	0,00	0,00	0,00	0,00	0,00
[148]	0,00	0,00	0,00	0,00	0,00	[196]	0,00	0,00	0,00	0,00	0,00
[197]	0,00	0,00	0,00	0,00	0,00						
[198]	0,00	0,00	0,00	0,00	0,00						
[199]	0,00	0,00	0,00	0,00	0,00						

Creando una 4-12-1 red neuronal

Fijamos Maxima Epoca = 10000

Fijamos Ratio de Aprendizaje = 0,01

Empezamos Entrenamiento

Epoca = 2000 Error = 4,4462

Epoca = 4000 Error = 5,6793

Epoca = 6000 Error = 4,2880

Epoca = 8000 Error = 5,0403

Ver

Final Modelo Red Neuronal Pesos y Bias:

0,5538036831376600	0,0894672039323634
-4,4916315882469200	0,0831693395625379
-0,3131742069793670	2,3875322295063300
2,4619697496803700	0,0882936013799404
-4,3276355443648300	0,1197703017055130
-0,9880680351414960	1,7436076922346000
1,7708020455127500	1,1202610502045900
0,3782192520986670	0,0613532392028805
-0,2600997474878820	0,0962417789651725
-2,4596880803006400	1,1330743296928200
1,1116753828267300	0,1925536593906510
-4,4406831010737900	-0,2845302185102320
0,5294341529298060	-0,1288880108854340
-4,4713875150966700	0,0600372331255330
-0,3073359665575650	-0,2005822101801700
2,3456338975789200	-0,1190536027625730
-4,3070385268886400	0,0304738495627940
-0,9822810854609810	-0,2251297329019160
1,6875494647531900	-0,2938030731264180
0,4209921162094370	0,0633585744596723



---

-0,2552579497447200	-0,0200766862384515
-2,4452321707937900	-0,2626955987019670
1,0612535628338400	-0,1265097512084320
-4,4268033417629400	4,0228895655382900
0,7336922711641260	-0,4120763268692040
0,0915581821897191	-0,3929206359674090
0,0832862421886634	4,0064202187863000
2,9077740975524900	-0,4281727445480350
0,0897079087876870	-0,3962919527232780
0,1295669343392240	3,9490242030389100
2,1197965883953100	1,9568293638279700
0,9583206945993360	-0,3983276174364650
0,0626032577988963	2,5327010023324400
0,0957108523114868	4,0481936282815400
1,3721350800806700	-0,5792597103533000
0,2217256896435460	4,3158440129762700
0,6115000652903580	

---

Precision Modelo (+/- 30) en Datos de Entrenamiento = 0,3600

Prediccion Pasajera por Mes:

2235

Final Tiempo Series

22,3485606168274 2

Normalizar Entrenamiento Datos:

---

[ 0]	20,78	20,30	22,49	23,66	22,94	[100]	21,71	22,62	25,29	23,63	25,35
[ 1]	23,52	22,12	21,19	22,97	24,07	[101]	23,13	23,71	25,04	23,73	20,17
[ 2]	20,97	22,82	23,98	23,32	23,53	[102]	21,93	23,20	24,90	25,09	24,61
[ 3]	24,21	20,34	22,53	22,06	20,13	[103]	23,97	21,96	20,63	23,61	21,76
[ 4]	20,62	24,00	22,55	24,52	23,21	[104]	24,92	23,22	23,84	20,22	22,40
[ 5]	24,46	20,85	22,32	20,73	20,31	[105]	23,10	24,72	22,36	22,86	21,37
[ 6]	20,86	24,99	23,01	25,28	23,66	[106]	25,22	23,93	23,15	20,97	24,40
[ 7]	24,78	21,14	23,91	24,32	20,71	[107]	20,80	21,35	22,55	22,86	22,64
[ 8]	23,97	23,07	21,04	20,64	25,06	[108]	22,24	22,17	24,54	23,81	20,74
[ 9]	21,15	22,25	22,52	22,05	21,05	[109]	21,39	21,60	21,36	24,62	24,32
[10]	24,46	22,00	21,11	20,61	23,13	[110]	24,65	22,23	20,45	22,36	21,17
[11]	20,84	21,68	25,20	22,77	22,26	[111]	23,95	20,16	24,84	23,18	24,42
[12]	21,79	24,11	23,80	23,03	23,26	[112]	20,41	24,19	24,61	21,29	21,16
[13]	21,82	24,17	23,93	20,92	25,02	[113]	21,17	21,33	22,21	21,10	21,88
[14]	20,44	21,37	24,39	24,94	24,08	[114]	21,63	25,16	20,72	23,51	21,06
[15]	21,34	24,63	24,73	23,56	23,05	[115]	20,11	21,92	20,07	21,15	21,47
[16]	20,59	20,28	20,49	25,09	24,31	[116]	24,53	24,26	24,88	23,55	20,56
[17]	22,81	20,96	24,46	22,24	24,19	[117]	23,62	21,81	23,72	21,07	22,50

---

[ 18]	23,39	23,68	22,87	25,38	22,73	[122]	23,12	20,24	21,98	21,41	24,61
[ 19]	20,55	21,19	21,02	22,21	25,24	[123]	20,58	20,83	22,68	24,53	24,22
[ 20]	23,42	24,87	20,15	23,99	24,90	[124]	21,55	24,72	23,51	24,49	23,95
[ 21]	23,27	24,36	24,16	24,33	24,57	[125]	20,48	20,41	20,11	23,08	22,44
[ 22]	25,18	20,67	25,07	24,36	21,89	[126]	24,86	21,48	21,97	23,62	23,44
[ 23]	22,55	22,14	24,18	24,68	20,46	[127]	22,00	21,17	25,09	23,87	25,12
[ 24]	22,36	21,68	24,25	20,71	22,07	[128]	21,80	22,54	25,02	22,37	20,07
[ 25]	21,94	20,90	24,74	20,20	21,99	[129]	22,24	22,46	23,19	24,40	21,28
[ 26]	21,75	21,07	20,69	25,03	20,84	[130]	20,87	21,55	23,04	20,73	20,42
[ 27]	23,94	20,88	22,43	21,66	23,58	[131]	24,56	20,29	23,96	23,88	23,01
[ 28]	20,23	20,73	25,08	21,37	23,59	[132]	21,02	21,25	21,38	20,23	23,96
[ 29]	23,37	21,48	24,01	21,04	20,85	[133]	20,78	24,87	22,25	24,16	21,81
[ 30]	23,54	23,87	22,39	20,60	20,46	[134]	24,58	21,24	20,79	21,58	24,40
[ 31]	23,50	23,76	20,60	24,88	22,00	[135]	22,81	21,84	22,19	20,75	20,70
[ 32]	21,64	24,05	23,22	22,93	22,33	[136]	23,14	22,75	21,83	23,16	21,96
[ 33]	21,60	23,16	21,96	24,74	24,83	[137]	21,20	21,52	21,41	22,52	22,08
[ 34]	21,20	22,34	20,79	20,60	22,95	[138]	24,49	20,19	20,01	20,98	20,84
[ 35]	21,91	20,67	23,68	24,79	23,11	[139]	23,05	22,31	22,37	20,53	20,99
[ 36]	20,16	21,94	20,64	20,79	20,70	[140]	23,16	21,04	23,46	24,76	23,88
[ 37]	23,34	25,24	24,82	21,59	23,71	[141]	20,63	23,18	21,06	24,70	22,96
[ 38]	20,71	24,23	24,85	24,71	22,64	[142]	24,66	22,73	24,00	22,02	24,61
[ 39]	23,58	22,00	20,15	22,39	20,32	[143]	20,81	3,76	21,95	21,478	2,42
[ 40]	20,68	25,31	22,95	20,88	22,11	[144]	22,38	22,29	22,34	0,00	0,00
[ 41]	20,17	23,69	24,72	21,58	23,58	[145]	0,00	0,00	0,00	0,00	0,00
[ 42]	21,61	23,22	20,25	20,84	20,70	[146]	0,00	0,00	0,00	0,00	0,00
[ 43]	20,01	24,65	20,94	20,06	21,78	[147]	0,00	0,00	0,00	0,00	0,00
[ 44]	22,68	22,83	25,10	20,73	22,42	[148]	0,00	0,00	0,00	0,00	0,00
[ 45]	24,52	20,15	23,48	24,94	20,39	[149]	0,00	0,00	0,00	0,00	0,00
[ 46]	24,44	22,19	24,14	20,04	21,05	[150]	0,00	0,00	0,00	0,00	0,00
[ 47]	22,87	23,71	22,52	20,22	25,26	[151]	0,00	0,00	0,00	0,00	0,00
[ 48]	21,92	24,49	22,69	23,93	24,67	[152]	0,00	0,00	0,00	0,00	0,00
[ 49]	22,82	24,52	21,26	23,40	25,37	[153]	0,00	0,00	0,00	0,00	0,00
[ 50]	20,63	20,81	24,37	22,08	22,58	[154]	0,00	0,00	0,00	0,00	0,00
[ 51]	25,12	25,30	22,71	24,26	21,01	[155]	0,00	0,00	0,00	0,00	0,00
[ 52]	23,01	23,20	23,89	24,98	23,01	[156]	0,00	0,00	0,00	0,00	0,00
[ 53]	21,78	22,09	24,32	25,15	23,65	[157]	0,00	0,00	0,00	0,00	0,00
[ 54]	23,75	20,27	20,09	24,42	24,66	[158]	0,00	0,00	0,00	0,00	0,00
[ 55]	25,23	25,02	21,05	22,55	21,50	[159]	0,00	0,00	0,00	0,00	0,00
[ 56]	22,13	24,38	20,68	21,52	24,07	[160]	0,00	0,00	0,00	0,00	0,00
[ 57]	22,72	24,28	20,51	20,65	24,05	[161]	0,00	0,00	0,00	0,00	0,00
[ 58]	24,93	23,77	21,34	22,95	21,88	[162]	0,00	0,00	0,00	0,00	0,00
[ 59]	20,34	20,01	21,53	25,36	21,79	[163]	0,00	0,00	0,00	0,00	0,00
[ 60]	21,54	20,58	21,94	23,65	20,93	[164]	0,00	0,00	0,00	0,00	0,00
[ 61]	20,03	21,878	25,38	23,04	20,54	[165]	0,00	0,00	0,00	0,00	0,00
[ 62]	23,09	23,52	24,64	22,39	20,56	[166]	0,00	0,00	0,00	0,00	0,00
[ 63]	22,74	23,24	21,78	22,67	20,71	[167]	0,00	0,00	0,00	0,00	0,00
[ 64]	21,45	23,93	23,99	21,44	22,30	[168]	0,00	0,00	0,00	0,00	0,00
[ 65]	25,30	24,42	21,39	20,67	25,37	[169]	0,00	0,00	0,00	0,00	0,00
[ 66]	20,30	22,51	21,20	25,07	24,47	[170]	0,00	0,00	0,00	0,00	0,00

---

[ 67]	20,94	22,27	22,18	23,60	24,31	[171]	0,00	0,00	0,00	0,00	0,00
[ 68]	21,32	25,29	20,51	20,07	23,31	[172]	0,00	0,00	0,00	0,00	0,00
[ 69]	21,10	22,94	22,27	24,49	23,45	[173]	0,00	0,00	0,00	0,00	0,00
[ 70]	20,60	21,21	21,19	20,44	22,95	[174]	0,00	0,00	0,00	0,00	0,00
[ 71]	20,51	24,42	24,13	20,91	23,38	[175]	0,00	0,00	0,00	0,00	0,00
[ 72]	22,59	24,14	21,62	25,21	20,07	[176]	0,00	0,00	0,00	0,00	0,00
[ 73]	21,08	22,87	23,63	21,82	22,48	[177]	0,00	0,00	0,00	0,00	0,00
[ 74]	20,81	22,45	20,39	20,21	20,66	[178]	0,00	0,00	0,00	0,00	0,00
[ 75]	20,56	25,29	20,26	24,97	21,88	[179]	0,00	0,00	0,00	0,00	0,00
[ 76]	22,80	23,46	20,36	25,13	23,18	[180]	0,00	0,00	0,00	0,00	0,00
[ 77]	22,36	23,56	24,90	24,38	21,63	[181]	0,00	0,00	0,00	0,00	0,00
[ 78]	24,67	20,87	24,84	21,96	24,23	[182]	0,00	0,00	0,00	0,00	0,00
[ 79]	20,45	22,10	20,40	24,39	22,72	[183]	0,00	0,00	0,00	0,00	0,00
[ 80]	20,39	23,54	21,53	20,16	22,86	[184]	0,00	0,00	0,00	0,00	0,00
[ 81]	23,12	23,00	25,16	21,40	20,70	[185]	0,00	0,00	0,00	0,00	0,00
[ 82]	21,18	24,00	20,20	24,91	22,00	[186]	0,00	0,00	0,00	0,00	0,00
[ 83]	22,39	21,00	23,32	24,73	20,30	[187]	0,00	0,00	0,00	0,00	0,00
[ 84]	21,05	20,81	20,36	24,50	23,79	[188]	0,00	0,00	0,00	0,00	0,00
[ 85]	21,96	20,46	24,01	22,12	22,11	[189]	0,00	0,00	0,00	0,00	0,00
[ 86]	25,03	20,40	20,10	21,31	20,82	[190]	0,00	0,00	0,00	0,00	0,00
[ 87]	21,87	23,14	21,54	22,11	20,91	[191]	0,00	0,00	0,00	0,00	0,00
[ 88]	24,47	20,53	21,68	21,88	25,33	[192]	0,00	0,00	0,00	0,00	0,00
[ 89]	25,33	24,82	20,46	20,46	25,22	[193]	0,00	0,00	0,00	0,00	0,00
[ 90]	22,67	24,12	21,47	20,58	22,07	[194]	0,00	0,00	0,00	0,00	0,00
[ 91]	24,82	22,93	22,23	21,59	23,23	[195]	0,00	0,00	0,00	0,00	0,00
[ 92]	21,85	23,98	24,17	23,869	20,43	[196]	0,00	0,00	0,00	0,00	0,00
[ 93]	24,91	21,76	25,20	21,61	21,84	[197]	0,00	0,00	0,00	0,00	0,00
[ 94]	24,88	20,49	21,06	24,40	21,45	[198]	0,00	0,00	0,00	0,00	0,00
[ 95]	22,91	20,17	23,34	22,74	24,28	[199]	0,00	0,00	0,00	0,00	0,00
[ 96]	20,86	23,11	20,43	20,422	23,28						
[ 97]	22,68	22,31	24,71	22,937	20,29						
[ 98]	20,90	25,16	22,60	21,246	23,88						
[ 99]	24,58	24,11	20,18	20,949	22,14						

---

Creando una 4-12-1 red neuronal

Fijamos Maxima Epoca = 10000

Fijamos Ratio de Aprendizaje = 0,01

Empezamos Entrenamiento

Epoca = 2000 Error = 4,4462

Epoca = 4000 Error = 5,6793

Epoca = 6000 Error = 4,2880

Epoca = 8000 Error = 5,0403

Ver

Final Modelo Red Neuronal Pesos y Bias:

---

1,0312489760460900	0,6117186672046870
0,5753736629884970	0,1137744017987040
-0,3930033829544040	0,3522536457223710
0,2359228923068560	0,1668918514848600
-1,8778787340726900	0,0877550390100484
-0,1985638470948850	0,4205398352723530
0,2922963373627020	0,3904454256777050
0,3868968273313750	0,5275771978050490
0,2888315340421640	0,1077920866459800
-0,4468226040345430	0,4052349643989700
0,2139901132396330	0,0780518850315274
-0,1600329184987240	-0,1262525863772640
1,2355718943839500	-0,1456545741506070
0,5614595185411950	0,0011441676801098
-0,3827852960799160	-0,1642004017732320
0,2582468638176170	-0,1179292523964580
-1,8687661592175000	0,0024598424852657
-0,1922717097431950	-0,1563202483972450
0,3192552471162110	-0,1576048726621880
0,3535559931572220	-0,1851943117952260
0,2846769782989240	0,0161763295005288
-0,4398561399374310	-0,1687233468187470
0,2727882250276200	0,0030243888377827
-0,1575917410389110	3,2479217073532700
1,2503794723177000	3,3680009694524600
0,6268637124117990	0,5985123983596340
-0,3847704818225290	3,4375523911244600
0,2380596269129800	2,1789066935581700
-1,8637311442713700	0,0380151861417673
-0,1958926286904990	3,3627598448375200
0,2934422198273520	3,4055017994087100
0,3843991710129580	2,3679647022901900
0,2729110719912990	1,3976237178248900
-0,4413670074019880	3,2091692503068100
0,2725116912571360	-0,4757427296474250
-0,1574071767016810	3,7119935731750900
1,2386372681483500	

---

Precision Modelo (+/- 30) en Datos de Entrenamiento = 0,3600

Prediccion Pasajera por Mes:

2235

Final Tiempo Series

22,3487955098142 3

Normalizar Entrenamiento Datos:

[0]	20,78	20,30	22,49	23,66	22,94	[101]	23,13	23,71	25,04	23,73	20,17
[1]	23,52	22,12	21,19	22,97	24,07	[102]	21,93	23,20	24,90	25,09	24,61
[2]	20,97	22,82	23,98	23,32	23,53	[103]	23,97	21,96	20,63	23,61	21,76
[3]	24,21	20,34	22,53	22,06	20,13	[104]	24,92	23,22	23,84	20,22	22,40
[4]	20,62	24,00	22,55	24,52	23,21	[105]	23,10	24,72	22,36	22,86	21,37
[5]	24,46	20,85	22,32	20,73	20,31	[106]	25,22	23,93	23,15	20,97	24,40
[6]	20,86	24,99	23,01	25,28	23,66	[107]	20,80	21,35	22,55	22,86	22,64
[7]	24,78	21,14	23,91	24,32	20,71	[108]	22,24	22,17	24,54	23,81	20,74
[8]	23,97	23,07	21,04	20,64	25,06	[109]	21,39	21,60	21,36	24,62	24,32
[9]	21,15	22,25	22,52	22,05	21,05	[110]	24,65	22,23	20,45	22,36	21,17
[10]	24,46	22,00	21,11	20,61	23,13	[111]	23,95	20,16	24,84	23,18	24,42
[11]	20,84	21,68	25,20	22,77	22,26	[112]	20,41	24,19	24,61	21,29	21,16
[12]	21,79	24,11	23,80	23,03	23,26	[113]	21,17	21,33	22,21	21,10	21,88
[13]	21,82	24,17	23,93	20,92	25,02	[114]	21,63	25,16	20,72	23,51	21,06
[14]	20,44	21,37	24,39	24,94	24,08	[115]	20,11	21,92	20,07	21,15	21,47
[15]	21,34	24,63	24,73	23,56	23,05	[116]	24,53	24,26	24,88	23,55	20,56
[16]	20,59	20,28	20,49	25,09	24,31	[117]	23,62	21,81	23,72	21,07	22,50
[17]	22,81	20,96	24,46	22,24	24,19	[118]	25,21	21,45	24,60	21,41	20,45
[18]	23,39	23,68	22,87	25,38	22,73	[119]	23,35	20,55	23,47	21,19	23,29
[19]	20,55	21,19	21,02	22,21	25,24	[120]	24,63	21,88	22,21	21,12	22,62
[20]	23,42	24,87	20,15	23,99	24,90	[121]	24,86	23,24	22,42	25,36	24,05
[21]	23,27	24,36	24,16	24,33	24,57	[122]	23,12	20,24	21,98	21,41	24,61
[22]	25,18	20,67	25,07	24,36	21,89	[123]	20,58	20,83	22,68	24,53	24,22
[23]	22,55	22,14	24,18	24,68	20,46	[124]	21,55	24,72	23,51	24,49	23,95
[24]	22,36	21,68	24,25	20,71	22,07	[125]	20,48	20,41	20,11	23,08	22,44
[25]	21,94	20,90	24,74	20,20	21,99	[126]	24,86	21,48	21,97	23,62	23,44
[26]	21,75	21,07	20,69	25,03	20,84	[127]	22,00	21,17	25,09	23,87	25,12
[27]	23,94	20,88	22,43	21,66	23,58	[128]	21,80	22,54	25,02	22,37	20,07
[28]	20,23	20,73	25,08	21,37	23,59	[129]	22,24	22,46	23,19	24,40	21,28
[29]	23,37	21,48	24,01	21,04	20,85	[130]	20,87	21,55	23,04	20,73	20,42
[30]	23,54	23,87	22,39	20,60	20,46	[131]	24,56	20,29	23,96	23,88	23,01
[31]	23,50	23,76	20,60	24,88	22,00	[132]	21,02	21,25	21,38	20,23	23,96
[32]	21,64	24,05	23,22	22,93	22,33	[133]	20,78	24,87	22,25	24,16	21,81
[33]	21,60	23,16	21,96	24,74	24,83	[134]	24,58	21,24	20,79	21,58	24,40
[34]	21,20	22,34	20,79	20,60	22,95	[135]	22,81	21,84	22,19	20,75	20,70
[35]	21,91	20,67	23,68	24,79	23,11	[136]	23,14	22,75	21,83	23,16	21,96
[36]	20,16	21,94	20,64	20,79	20,70	[137]	21,20	21,52	21,41	22,52	22,08
[37]	23,34	25,24	24,82	21,59	23,71	[138]	24,49	20,19	20,01	20,98	20,84
[38]	20,71	24,23	24,85	24,71	22,64	[139]	23,05	22,31	22,37	20,53	20,99
[39]	23,58	22,00	20,15	22,39	20,32	[140]	23,16	21,04	23,46	24,76	23,88
[40]	20,68	25,31	22,95	20,88	22,11	[141]	20,63	23,18	21,06	24,70	22,96
[41]	20,17	23,69	24,72	21,58	23,58	[142]	24,66	22,73	24,00	22,02	24,61
[42]	21,61	23,22	20,25	20,84	20,70	[143]	20,81	23,76	21,95	21,47	23,42
[43]	20,01	24,65	20,94	20,06	21,78	[144]	22,38	22,29	22,34	22,34	0,00
[44]	22,68	22,83	25,10	20,73	22,42	[145]	0,00	0,00	0,00	0,00	0,00

---

[45]	24,52	20,15	23,48	24,94	20,39	[146]	0,00	0,00	0,00	0,00	0,00
[46]	24,44	22,19	24,14	20,04	21,05	[147]	0,00	0,00	0,00	0,00	0,00
[47]	22,87	23,71	22,52	20,22	25,26	[148]	0,00	0,00	0,00	0,00	0,00
[48]	21,92	24,49	22,69	23,93	24,67	[149]	0,00	0,00	0,00	0,00	0,00
[49]	22,82	24,52	21,26	23,40	25,37	[150]	0,00	0,00	0,00	0,00	0,00
[50]	20,63	20,81	24,37	22,08	22,58	[151]	0,00	0,00	0,00	0,00	0,00
[51]	25,12	25,30	22,71	24,26	21,01	[152]	0,00	0,00	0,00	0,00	0,00
[52]	23,01	23,20	23,89	24,98	23,01	[153]	0,00	0,00	0,00	0,00	0,00
[53]	21,78	22,09	24,32	25,15	23,65	[154]	0,00	0,00	0,00	0,00	0,00
[54]	23,75	20,27	20,09	24,42	24,66	[155]	0,00	0,00	0,00	0,00	0,00
[55]	25,23	25,02	21,05	22,55	21,50	[156]	0,00	0,00	0,00	0,00	0,00
[56]	22,13	24,38	20,68	21,52	24,07	[157]	0,00	0,00	0,00	0,00	0,00
[57]	22,72	24,28	20,51	20,65	24,05	[158]	0,00	0,00	0,00	0,00	0,00
[58]	24,93	23,77	21,34	22,95	21,88	[159]	0,00	0,00	0,00	0,00	0,00
[59]	20,34	20,01	21,53	25,36	21,79	[160]	0,00	0,00	0,00	0,00	0,00
[60]	21,54	20,58	21,94	23,65	20,93	[161]	0,00	0,00	0,00	0,00	0,00
[61]	20,03	21,87	25,38	23,04	20,54	[162]	0,00	0,00	0,00	0,00	0,00
[62]	23,09	23,52	24,64	22,39	20,56	[163]	0,00	0,00	0,00	0,00	0,00
[63]	22,74	23,24	21,78	22,67	20,71	[164]	0,00	0,00	0,00	0,00	0,00
[64]	21,45	23,93	23,99	21,44	22,30	[165]	0,00	0,00	0,00	0,00	0,00
[65]	25,30	24,42	21,39	20,67	25,37	[166]	0,00	0,00	0,00	0,00	0,00
[66]	20,30	22,51	21,20	25,07	24,47	[167]	0,00	0,00	0,00	0,00	0,00
[67]	20,94	22,27	22,18	23,60	24,31	[168]	0,00	0,00	0,00	0,00	0,00
[68]	21,32	25,29	20,51	20,07	23,31	[169]	0,00	0,00	0,00	0,00	0,00
[69]	21,10	22,94	22,27	24,49	23,45	[170]	0,00	0,00	0,00	0,00	0,00
[70]	20,60	21,21	21,19	20,44	22,95	[171]	0,00	0,00	0,00	0,00	0,00
[71]	20,51	24,42	24,13	20,91	23,38	[172]	0,00	0,00	0,00	0,00	0,00
[72]	22,59	24,14	21,62	25,21	20,07	[173]	0,00	0,00	0,00	0,00	0,00
[73]	21,08	22,87	23,63	21,82	22,48	[174]	0,00	0,00	0,00	0,00	0,00
[74]	20,81	22,45	20,39	20,21	20,66	[175]	0,00	0,00	0,00	0,00	0,00
[75]	20,56	25,29	20,26	24,97	21,88	[176]	0,00	0,00	0,00	0,00	0,00
[76]	22,80	23,46	20,36	25,13	23,18	[177]	0,00	0,00	0,00	0,00	0,00
[77]	22,36	23,56	24,90	24,38	21,63	[178]	0,00	0,00	0,00	0,00	0,00
[78]	24,67	20,87	24,84	21,96	24,23	[179]	0,00	0,00	0,00	0,00	0,00
[79]	20,45	22,10	20,40	24,39	22,72	[180]	0,00	0,00	0,00	0,00	0,00
[80]	20,39	23,54	21,53	20,16	22,86	[181]	0,00	0,00	0,00	0,00	0,00
[81]	23,12	23,00	25,16	21,40	20,70	[182]	0,00	0,00	0,00	0,00	0,00
[82]	21,18	24,00	20,20	24,91	22,00	[183]	0,00	0,00	0,00	0,00	0,00
[83]	22,39	21,00	23,32	24,73	20,30	[184]	0,00	0,00	0,00	0,00	0,00
[84]	21,05	20,81	20,36	24,50	23,79	[185]	0,00	0,00	0,00	0,00	0,00
[85]	21,96	20,46	24,01	22,12	22,11	[186]	0,00	0,00	0,00	0,00	0,00
[86]	25,03	20,40	20,10	21,31	20,82	[187]	0,00	0,00	0,00	0,00	0,00
[87]	21,87	23,14	21,54	22,11	20,91	[188]	0,00	0,00	0,00	0,00	0,00
[88]	24,47	20,53	21,68	21,88	25,33	[189]	0,00	0,00	0,00	0,00	0,00
[89]	25,33	24,82	20,46	20,46	25,22	[190]	0,00	0,00	0,00	0,00	0,00
[90]	22,67	24,12	21,47	20,58	22,07	[191]	0,00	0,00	0,00	0,00	0,00
[91]	24,82	22,93	22,23	21,59	23,23	[192]	0,00	0,00	0,00	0,00	0,00
[92]	21,85	23,98	24,17	23,86	20,43	[193]	0,00	0,00	0,00	0,00	0,00
[93]	24,91	21,76	25,20	21,61	21,84	[194]	0,00	0,00	0,00	0,00	0,00

---

---

[94]	24,88	20,49	21,06	24,40	21,45	[195]	0,00	0,00	0,00	0,00	0,00
[95]	22,91	20,17	23,34	22,74	24,28	[196]	0,00	0,00	0,00	0,00	0,00
[96]	20,86	23,11	20,43	20,42	23,28	[197]	0,00	0,00	0,00	0,00	0,00
[97]	22,68	22,31	24,71	22,93	20,29	[198]	0,00	0,00	0,00	0,00	0,00
[98]	20,90	25,16	22,60	21,24	23,88	[199]	0,00	0,00	0,00	0,00	0,00
[99]	24,58	24,11	20,18	20,94	22,14						
[100]	21,77	22,62	25,29	23,63							
	25,35										

---

Creando una 4-12-1 red neuronal

Fijamos Maxima Epoca = 10000

Fijamos Ratio de Aprendizaje = 0,01

Empezamos Entrenamiento

Epoca = 2000 Error = 4,4462

Epoca = 4000 Error = 5,6793

Epoca = 6000 Error = 4,2880

Epoca = 8000 Error = 5,0402

Ver

Final Modelo Red Neuronal Pesos y Bias:

---

0,2242220534394100	2,2826530998775600
1,7550660873234400-	0,3096454474041140
0,0115206028036467	0,0179246577420522
-0,3409227491973760	2,2536453335464600
1,7301370368435700	0,2925134490284990
0,0142850925232853	-0,1201973042538530
-0,4874479731019420	0,2762898697820460
0,0081190069142861	0,8444954781529400
0,6180814699035800	0,2525166955271530
0,0349705740321986	0,5181924221416690
0,2288193055998150	0,2745455992460320
0,0065825569400972	-0,1179016731077700
0,4502824116400700	-0,0351493445685667
1,9013608679851300	-0,1344014932266950
0,2261934094940140	0,0052846826141091
-0,1580430222449830	-0,0332588639670334
1,8767299656051600	-0,1308271810514050
0,2268059637477410	-0,0522603178903202
-0,2986463755829220	-0,1263084952286360
0,1972628716720270	-0,1042256157279230
0,8082736265248600	-0,1208138326972850
0,1980202555212910	-0,1238358704683350
0,4561607301027950	-0,1238961602994180
0,1958658361562480	2,2345151286980300

---

---

0,1839480191906930	2,1384466020888100
1,8576904835244600	2,2709256187794700
-0,0507298381389055	0,6925853417009100
-0,3657383202776830	2,1262889123256900
1,8311373476196600	2,2618648377004200
-0,0249994975874111	1,5768478898180400
-0,5077569233343220	2,2342110526990900
-0,0531401176997630	2,2564008303110700
0,5774082710289770	2,2106909873431100
-0,0033716586064677	2,2708462632226000
0,1893364517114010	2,2201286535526000
-0,0535510535248363	2,4186634117207700
0,5128486075889970	

---

Precision Modelo (+/- 30) en Datos de Entrenamiento = 0,3600

Prediccion Pasajera por Mes:

2192

Final Tiempo Series

21,9213354397033



## IV. CONCLUSIONES Y RECOMENDACIONES

### 4.1. Conclusiones.

- a. Se consiguió extraer y analizar la información de las ventas diarias mensuales del año 2016 y 2017. Las ventas se utilizaron para realizar el pronóstico de las ventas diarias para el año 2018.
- b. Se logró identificar el modelo de la red neuronal que fue aplicado para pronosticar las ventas futuras.
- c. Se implementó el algoritmo ideal y se utilizó el análisis de regresión de serie temporal mediante datos de ventana con desplazamiento combinados para la red neuronal perceptrón multicapa, de tal manera que pueda pronosticar las ventas futuras.
- d. Se implementó la red neuronal con 4 nodos de entrada, 12 nodos de procesamiento ocultos y un único nodo de salida, además se generaron 73 pesos sinápticos.
- e. En el entrenamiento de la red para la predicción de las ventas se utilizó el algoritmo de retro propagación con un índice de aprendizaje de 0.01 y 10.000 iteraciones.
- f. Se logró desarrollar la herramienta Informática – software – capaz de realizar el pronóstico de las ventas.
- g. En el análisis de los resultados obtenidos, se aprecia que la venta diaria mensual pronosticada con la red neuronal fluctúa dentro de los límites de la venta diaria mensual promedio, esto indica que el entrenamiento de la red se aproxima al valor deseado.
- h. El tiempo consumido por el sistema para ejecutar la red neuronal asciende a 24 minutos con 37 segundos.
- i. La memoria consumida por el sistema para ejecutar la red neuronal es de 0.0165328 Gigabytes.
- j. La magnitud del error de la predicción hecha por la empresa es grande, en cambio la predicción de la red neuronal tiene un nivel de error muy bajo.

#### **4.2. Recomendaciones.**

- a. Debe profundizarse en el estudio de nuevos algoritmos, técnicas, métodos que permitan evaluar la rentabilidad económica de una empresa.
- b. Seguir con la implementación del software, para ir agregándole nuevos requisitos funcionales.
- c. Optimizar la programación de la red neuronal de forma dinámica para agilizar el trabajo de automatización.

## REFERENCIAS

- Alvites Castillo, V. (2000). *Metodo Cientifico. Planificación de la Investigación. Segunda Edicion*. Chiclayo: Ciencia.
- Caicedo Bravo, E. F., & López Sotelo, J. A. (2017). *Una aproximación práctica a las redes neuronales artificiales*. Colombia: Programa editorial Univalle [236].
- Calvo, D. (12 de Julio de 2017). *Diego Calvo*. Obtenido de Definición de red neuronal artificial: <https://www.diegocalvo.es/definicion-de-red-neuronal/>
- Caparrós, J. E. (1994). Redes neuronales: concepto, fundamentos y aplicaciones. En J. E. Caparros, *Redes neuronales: concepto, fundamentos y aplicaciones* (págs. 221-228). España, Murcia: QUIMICA CLINICA.
- Carranza Hernández, S. N. (2014). *Implementación de un Sistema de Información para el Reconocimiento de Caracteres Basado en la Red Neuronal Perceptron*. Lima: Pontificia Universidad Catolica del Peru.
- Colegio de Ingenieros, L. (2011). *Código Deontológico*. Obtenido de [http://www.cip.org.pe/publicaciones/2018/CODIGO\\_DEONTOLOGICO2012.pdf](http://www.cip.org.pe/publicaciones/2018/CODIGO_DEONTOLOGICO2012.pdf)
- Diebold, F. X., & Mariano, R. S. (1991). *Comparing Predictive Accuracy I; An Asymptotic Test*. Estados Unidos: Revista Journal of Business and Economic Statistics, vol. 13, no. 2, 1993, pp. 253-265.
- Dieterich, H. (2001). *Nueva Guía para la Investigación Científica*. Mexico, España y America Latina: Heinz Dieterich Steffan.
- ES, P. R. (18 de Noviembre de 2019). *¿Qué son las ventas? Guía Práctica*. Obtenido de <https://blog.hubspot.es/sales/que-son-las-ventas>
- Escobar R, L., Valdes H, J., & Zapata C, S. (2010). *Redes Neuronales Artificiales en predicción de Series de Tiempo*. Chile: C&T - Universidad de Palermo.
- Foix, C., & Weber, R. (2014). Pronóstico del Precio del Cobre mediante Redes Neuronales. *Revista Ingeniería de Sistemas*, 60-63.
- Galeana, C. (20 de 05 de 2014). *Conceptos Generales de Pronósticos*. Obtenido de prezi.com: <https://prezi.com/7inu5kijozy8/conceptos-generales-de-pronosticos/>

- García Díaz, C. M. (2014). *Predicción del comportamiento de la rentabilidad de la acción de Isagen, apartir de la técnica de redes neuronales artificiales*. Bogota: Pontifica Universidad Javeriana.
- Gestal Pose, M. (2010). *Introducción a la Redes Neuronales Artificiales*. España: Universidad da Coruña. Obtenido de Introducción a la Redes Neuronales Artificiales.
- Gomez Rojas, G. A., Henao Lopez, J. C., & Harold, S. (2004). Entrenamiento de una Red Neuronal Artificial usando el Algortimo Simulated Annealing. *researchgate*, 20-25.
- Hernández Sampieri, R., Fernández Collado, C., & Pilar Baptista, L. (1991). *Metodología de la Investigación, Cuarta Edición*. Mexico: McGRAWHILLIINTERAMERICMA EDITORES, SA DE C.V.
- Huayna Dueñas, A. M., Calvo Huaraz, V., & Huiman Sánchez, J. C. (2010). Modelo de Evaluación de Créditos Financieros basados en Redes Neuronales orientado a Edpymes. *Revista de Investigación UNMSM*, 23-31.
- jmmarin. (s.f.). *Introducción a las redes neuronales aplicadas*. Madrid. Obtenido de uc3m:  
<http://halweb.uc3m.es/esp/Personal/personas/jmmarin/esp/DM/tema3dm.pdf>
- k, A. (5 de Marzo de 2010). *CreceNegocios*. Obtenido de El pronóstico de ventas:  
<https://www.crecenegocios.com/el-pronostico-de-ventas/>
- Kaastra, I., & Boyd, M. (23 de Marzo de 1995). *Designing a neural network for forecasting Financial an Economic Time Series*. Obtenido de citeseerx:  
<https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.15.8688&rep=rep1&type=pdf>
- Malaver Elera, M. A. (2015). *Aplicación de redes neuronales para determinar el pronóstico de las ventas en la empresa Catering & Buffets MYS Ubicada en la Ciudad de Piura*. Piura: Universidad Cesar Vallejo, Repositorio Digital Institucional.
- Paggi Straneo, H. (2003). *Predicción de series temporales usando redes neuronales*. Uruguay: UR. FI-INCO.
- Rios, G. (2008). *Series de Tiempo*. Chile: Universidad De Chile, Facultad De Ciencias Físicas y Matemáticas.

- Riquelme, M. (12 de Julio de 2020). *Toma de decisiones: Concepto de vital importancia en la empresa*. Obtenido de webyempresas.com: <https://www.webyempresas.com/toma-de-decisiones/>
- Serrano, C., & Bonifacio, M. (2008). Predicción de la quiebra bancaria mediante el empleo de redes neuronales artificiales. *Revista Española de Financiación y Contabilidad*, 4-5.
- Turmeno, P. (02 de Diciembre de 2020). *Introducción a las redes neuronales y su aplicación a la investigación astrofísica*. Obtenido de monografias.com: <https://www.monografias.com/trabajos105/introduccion-redes-neuronales-y-su-aplicacion-investigacion-astrofisica/introduccion-redes-neuronales-y-su-aplicacion-investigacion-astrofisica.shtml>
- Vargas, C. A. (2015). *Nomenclatura del Algoritmo Backpropagation*. Bolivia: Universidad del Valle.
- Vidaurre Siadén, Y. A. (2012). *Aplicación de las redes neuronales artificiales para el pronóstico de la demanda de agua potable en la Empresa Epsel S.A. de la ciudad de Lambayeque*. Lambayeque: USAT.
- Yirda, A. (25 de 01 de 2021). *ConceptoDefinicion*. Obtenido de Definicion de Venta: <https://conceptodefinicion.de/venta/>
- Zapata C, S., Escovar R, L., & Valdés H, J. (2010). *Redes Neuronales Artificiales en predicción de Series de Tiempo. Una Aplicación a la Industria*. Santiago: C&T - Universidad de Palermo.

## ANEXOS.

### Anexo 1. Resolución de aprobación del trabajo de investigación



FACULTAD DE INGENIERÍA, ARQUITECTURA Y URBANISMO

RESOLUCIÓN N° 0678-2019/FIAU-USS

Chiclayo, 18 de junio de 2019

#### VISTO:

El Acta de Reunión N° S/N de fecha 18 de junio de 2019, del Comité Evaluador de la Escuela Académico Profesional de INGENIERÍA DE SISTEMAS, donde se propone la aprobación de la modificación de título de la tesis presentada por el(los) tesista(s) BENITES SERNAQUE JOSE MANUEL; y

#### CONSIDERANDO:

Que, de conformidad con la Ley Universitaria N° 30220 en su artículo 48° que a la letra dice: *"La investigación constituye una función esencial y obligatoria de la universidad, que la fomenta y realiza, respondiendo a través de la producción de conocimiento y desarrollo de tecnologías a las necesidades de la sociedad, con especial énfasis en la realidad nacional. Los docente, estudiantes y graduados participan en la actividad investigadora en su propia institución o en redes de investigación nacional e internacional, creadas por las instituciones universitarias públicas o privadas."*

Que, mediante Resolución de Facultad N° 2120-2014/FIAU-USS de fecha 31 de diciembre de 2014 se aprobó el Proyecto de Tesis titulado "IMPLEMENTACIÓN DE UNA SOLUCIÓN DE INTELIGENCIA DE NEGOCIOS HACIENDO USO DE DATA MINING Y REDES NEURONALES PARA PREDECIR LA RENTABILIDAD ECONÓMICA DE LA EMPRESA INDUSTRIAL CERÁMICOS LAMBAYEQUE SAC", presentado por el(los) tesista(s) BENITES SERNAQUE JOSE MANUEL;

Que, es necesario facilitar el adecuado desarrollo de las Tesis aprobadas con la finalidad de dar continuidad al proceso de investigación; y

Estando a lo expuesto, en uso de las atribuciones conferidas y de conformidad con las normas y reglamentos vigentes;

#### SE RESUELVE:

**ARTÍCULO 1°:** APROBAR la modificación de título de la Tesis denominada: *"IMPLEMENTACIÓN DE UNA SOLUCIÓN DE INTELIGENCIA DE NEGOCIOS HACIENDO USO DE DATA MINING Y REDES NEURONALES PARA PREDECIR LA RENTABILIDAD ECONÓMICA DE LA EMPRESA INDUSTRIAL CERÁMICOS LAMBAYEQUE SAC"*, por el siguiente: *"IMPLEMENTACIÓN DE UN SISTEMA DE PRONÓSTICO DE VENTAS UTILIZANDO REDES NEURONALES ARTIFICIALES PARA LA EMPRESA CERÁMICOS LAMBAYEQUE SAC"*, presentada por el(los) tesista(s) BENITES SERNAQUE JOSE MANUEL, de la Escuela Académico Profesional de INGENIERÍA DE SISTEMAS.

**ARTÍCULO 2°:** MODIFICAR la Resolución de Asesor Especialista N° 2118-2014/FIAU-USS de fecha 31 de diciembre de 2014, Resolución de Jurado de Tesis N° 0810-A-2018/FIAU-USS de fecha 11 de setiembre de 2018 y Resolución de Aprobación de Proyecto N° 2120-2014/FIAU-USS de fecha 31 de diciembre de 2014, en el extremo que dice: *"IMPLEMENTACIÓN DE UNA SOLUCIÓN DE INTELIGENCIA DE NEGOCIOS HACIENDO USO DE DATA MINING Y REDES NEURONALES PARA PREDECIR LA RENTABILIDAD ECONÓMICA DE LA EMPRESA INDUSTRIAL CERÁMICOS LAMBAYEQUE SAC"*, por lo siguiente: *"IMPLEMENTACIÓN DE UN SISTEMA DE PRONÓSTICO DE VENTAS UTILIZANDO REDES NEURONALES ARTIFICIALES PARA LA EMPRESA CERÁMICOS LAMBAYEQUE SAC"*.

REGÍSTRESE, COMUNÍQUESE Y ARCHÍVESE

USS UNIVERSIDAD SEÑOR DE SIPÁN S.A.  
Dr. Andrés Alberto Ruiz Gómez  
DECANO DE LA FACULTAD DE INGENIERÍA  
ARQUITECTURA Y URBANISMO



UNIVERSIDAD SEÑOR DE SIPÁN S.A.C.  
Mg. Luis Roberto Caza Colchada  
DECANO FACULTAD DE INGENIERÍA  
ARQUITECTURA Y URBANISMO

ADMISIÓN E INFORMES

074 481610 - 074 481632

CAMPUS USS

Cc: CPDPM *Administración* *Académico* *Instrumental*

Chiclayo, Perú

www.uss.edu.pe

Anexo 2. Carta de aceptación de la institución para la recolección de datos.



**Cerámicos Lambayeque**  
www.ceramicoslambayeque.com.pe

**Cerámicos Lambayeque**  
...perfección de ladrillo

**CARTA DE AUTORIZACION**

Lambayeque, 03 de Noviembre 2014

CERAMICOS LAMBAYEQUE SAC

Autorización:

Yo, **Cesar A. Rodríguez Gómez**, representante legal de la empresa Cerámicos Lambayeque Sac con ruc número: 20480127791, autorizo a don **José Manuel Benites Sernaqué** con documento de identidad número 44274000 para que realice su trabajo de investigación y pueda obtener información de mi representada para su desarrollo de tesis, en mi nombre, recolecte información que sea de su alcance.

Atentamente



**Sector Culpón s/n - José L. Ortiz - Chiclayo**  
(Espalda Grito Morí - Carretera Lambayeque)  
Cel. 978959739 Rpm \*393857

Anexo 3. Instrumentos de recolección de datos, con su respectiva validación de los instrumentos.

		Ventas Reales	RED NEURONAL					
			Pronóstico	Error de Pronóstico	Grado de Confiabilidad	Verdaderos Positivos	Falsos Positivos	Precisión
DATABASE DE IMÁGENES	Dia 1							
	Dia 2							
	Dia 3							
	Dia 4							
	Dia 5							
	Dia 6							
	Dia 7							
	Dia 8							
	Dia 9							
	Dia 10							
	Dia 11							
	Dia 12							
	Dia 13							
	Dia 14							
	Dia 15							
	Dia 16							
	Dia 17							
	Dia 18							
	Dia 19							
	Dia 20							
	Dia 21							
	Dia 22							
.								
.								
.								
Dia 215								



	Ventas Reales	CERÁMICOS LAMBAYEQUE					
		Pronóstico	Error de Pronóstico	Grado de Confiabilidad	Verdaderos Positivos	Falsos Positivos	Precisión
Dia 1							
Dia 2							
Dia 3							
Dia 4							
Dia 5							
Dia 6							
Dia 7							
Dia 8							
Dia 9							
Dia 10							
Dia 11							
Dia 12							
Dia 13							
Dia 14							
Dia 15							
Dia 16							
Dia 17							
Dia 18							
Dia 19							
Dia 20							
Dia 21							
Dia 22							
.							
.							
.							
Dia 215							

DATABASE DE IMÁGENES

Tabla 8.

*Detalle instrumento-Técnica*

<b>TÉCNICA</b>	<b>INSTRUMENTO</b>
Recolección de Información	a) Ventas diarias mensuales desde el año 2016 al 2017. b) Bibliografía (libros, revistas, tesis, artículos científicos, ponencias, etc.).
Coordinaciones	Coordinaciones con el Jefe de Ventas
Observación	Observación de EEFF
Análisis de Sistemas Transaccionales	Herramientas de extracción de información (Se aplicó en el desarrollo del Software).
Planteamiento de Soluciones	Red neuronal perceptrón multicapa y regresión de serie temporal, utilizados para la predicción de las ventas y la rentabilidad a través del Lenguaje de programación C#.NET.
Análisis de los resultados	Resultados de tipo estadísticos.
Prueba del sistema	Cronometro.

*Nota:* Elaboración Propia.

En la Tabla 9 se presenta las ventas diarias por mes obtenidos por la empresa en el año 2016.

Tabla 9.

*Ventas diarias por mes del año 2016*

<b>Mes</b>	<b>Día</b>	<b>Venta Diaria</b>					
ENERO	01-02	20,784	20,308				
	04-09	22,495	23,666	22,949	23,527	22,121	21,198
	11-16	22,972	24,070	20,970	22,829	23,985	23,324
	18-23	23,537	24,210	20,348	22,531	22,060	20,132
	25-30	20,627	24,004	22,551	24,529	23,218	24,469
FEBRERO	01-06	20,859	22,327	20,735	20,316	20,864	24,994

	08-13	23,013	25,288	23,665	24,783	21,141	23,918
	15-20	24,326	20,715	23,977	23,078	21,049	20,641
	22-27	25,060	21,155	22,253	22,526	22,051	21,058
	29	24,464					
MARZO	01-05	22,002	21,119	20,612	23,132	20,841	
	07-12	21,684	25,206	22,779	22,263	21,793	24,110
	14-19	23,809	23,036	23,263	21,828	24,179	23,932
	21-26	20,928	25,026	20,441	21,375	24,392	24,949
	28-31	24,087	21,345	24,639	24,735		
ABRIL	01-02	23,567	23,057				
	04-09	20,598	20,289	20,499	25,092	24,312	22,816
	11-16	20,964	24,467	22,242	24,199	23,399	23,688
	18-23	22,874	25,384	22,736	20,551	21,190	21,020
	25-30	22,216	25,246	23,429	24,879	20,154	23,999
MAYO	02-07	24,905	23,273	24,360	24,165	24,339	24,571
	09-14	25,182	20,671	25,075	24,367	21,890	22,550
	16-21	22,144	24,189	24,689	20,469	22,364	21,688
	23-28	24,252	20,718	22,076	21,943	20,900	24,741
	30-31	20,205	21,996	21,750	21,072	20,694	25,033
JUNIO	01-04	20,844	23,941	20,888	22,433		
	06-11	21,663	23,584	20,231	20,737	25,088	21,371
	13-18	23,592	23,372	21,487	24,015	21,045	20,854
	20-25	23,545	23,873	22,393	20,609	20,463	23,508
	27-30	23,761	20,601	24,885	22,008		
JULIO	01-02	21,640	24,052				
	04-09	23,221	22,930	22,334	21,604	23,164	21,967
	11-16	24,747	24,837	21,204	22,345	20,796	20,605
	18-23	22,955	21,911	20,676	23,686	24,795	23,114
	25-30	20,161	21,949	20,647	20,790	20,707	23,340
AGOSTO	01-06	25,240	24,822	21,592	23,711	20,717	24,230

	08-13	24,858	24,714	22,643	23,589	22,009	20,150
	15-20	22,399	20,325	20,687	25,313	22,951	20,880
	22-27	22,113	20,171	23,696	24,726	21,583	23,589
	29-31	21,611	23,223	20,250			
SEPTIEMBRE	01-03	20,847	20,702	20,016			
	05-10	24,650	20,947	20,067	21,782	22,688	22,839
	12-17	25,100	20,739	22,429	24,528	20,158	23,485
	19-24	24,946	20,399	24,449	22,190	24,146	20,042
	26-30	21,057	22,873	23,719	22,529	20,228	
OCTUBRE	01	25,269					
	03-08	21,929	24,491	22,696	23,933	24,676	22,827
	10-15	24,523	21,267	23,400	25,378	20,638	20,814
	17-22	24,371	22,085	22,588	25,125	25,307	22,718
	24-29	24,269	21,019	23,016	23,205	23,891	24,984
	31	23,018					
NOVIEMBRE	01-05	21,788	22,097	24,323	25,151	23,656	
	07-12	23,752	20,270	20,098	24,424	24,661	25,233
	14-19	25,022	21,056	22,556	21,503	22,137	24,389
	21-26	20,683	21,522	24,075	22,729	24,283	20,513
	28-30	20,654	24,054	24,932			
DICIEMBRE	01-03	23,770	21,347	22,953			
	05-10	21,884	20,344	20,018	21,535	25,364	21,798
	12-17	21,546	20,580	21,941	23,650	20,934	20,034
	19-23	21,878	25,387	23,047	20,544	23,091	
	26-31	23,525	24,643	22,390	20,566	22,741	23,240

Nota: Fuente de la Empresa.

En la Tabla 10 se presenta las ventas diarias por mes obtenidos por la empresa en el año 2017.

Tabla 10.

*Ventas diarias del periodo 2017*

<b>Mes</b>	<b>Día</b>	<b>Venta Diaria</b>					
ENERO	02-07	21,788	22,671	20,712	21,450	23,933	23,996
	09-14	21,444	22,300	25,305	24,421	21,395	20,679
	16-21	25,375	20,308	22,515	21,200	25,072	24,471
	23-28	20,944	22,277	22,180	23,601	24,314	21,322
	30-31	25,299	20,511				
FEBRERO	01-04	20,072	23,311	21,104	22,942		
	06-11	22,270	24,490	23,459	20,609	21,213	21,199
	13-18	20,441	22,956	20,519	24,422	24,133	20,914
	20-25	23,383	22,593	24,146	21,622	25,215	20,078
	27-28	21,083	22,874				
MARZO	01-04	23,639	21,826	22,484	20,816		
	06-11	22,456	20,394	20,215	20,661	20,561	25,299
	13-18	20,265	24,973	21,889	22,802	23,468	20,363
	20-25	25,139	23,185	22,361	23,562	24,905	24,389
	27-31	21,636	24,675	20,879	24,848	21,969	
ABRIL	01	24,232					
	03-08	20,457	22,105	20,400	24,391	22,728	20,397
	10-15	23,542	21,535	20,160	22,864	23,128	23,003
	17-22	25,169	21,404	20,706	21,188	24,001	20,205
	24-29	24,916	22,006	22,396	21,003	23,321	24,735
MAYO	01-06	20,302	21,051	20,815	20,363	24,504	23,794
	08-13	21,964	20,468	24,011	22,127	22,112	25,038
	15-20	20,401	20,105	21,315	20,824	21,876	23,143
	22-27	21,543	22,115	20,918	24,471	20,531	21,687
	29-31	21,884	25,335	25,334			
JUNIO	01-03	24,821	20,464	20,468			
	05-10	25,227	22,671	24,122	21,474	20,586	22,073

	12-17	24,827 22,931 22,234 21,598 23,237 21,855
	19-24	23,980 24,177 23,869 20,436 24,918 21,761
	26-30	25,208 21,619 21,847 24,886 20,499
JULIO	01	21,060
	03-08	24,402 21,458 22,910 20,176 23,343 22,748
	10-15	24,282 20,869 23,118 20,434 20,422 23,288
	17-22	22,682 22,315 24,710 22,937 20,295 20,900
	24-29	25,169 22,601 21,246 23,881 24,587 24,117
	31	20,186
AGOSTO	01-05	20,949 22,148 21,717 22,621 25,293
	07-12	23,631 25,358 23,137 23,716 25,045 23,732
	14-19	20,176 21,936 23,201 24,903 25,098 24,619
	22-26	23,973 21,969 20,630 23,618 21,764 24,921
	28-31	23,223 23,840 20,226 22,404
SEPTIEMBRE	01-02	23,106 24,726
	04-09	22,366 22,865 21,370 25,221 23,932 23,150
	11-16	20,973 24,400 20,804 21,352 22,559 22,865
	18-23	22,645 22,248 22,177 24,546 23,816 20,746
	25-30	21,391 21,607 21,369 24,622 24,329 24,655
OCTUBRE	02-07	22,234 20,450 22,369 21,178 23,955 20,161
	09-14	24,844 23,186 24,422 20,417 24,199 24,611
	16-21	21,297 21,163 21,172 21,330 22,211 21,101
	23-28	21,885 21,630 25,162 20,724 23,512 21,064
	30-31	20,115 21,924
NOVIEMBRE	01-04	20,078 21,153 21,472 24,539
	06-11	24,261 24,888 23,551 20,564 23,620 21,817
	13-18	23,724 21,074 22,502 25,215 21,452 24,607
	20-25	21,419 20,459 23,356 20,555 23,476 21,199
	27-30	23,293 24,637 21,883 22,217
DICIEMBRE	01-02	21,123 22,627

	04-09	24,865	23,241	22,429	25,364	24,059	23,127
	11-16	20,242	21,981	21,414	24,617	20,589	20,830
	18-23	22,681	24,531	24,221	21,558	24,721	23,515
	25-30	24,495	23,952	20,489	20,413	20,115	23,088

Nota: *Información de la empresa.*

Los datos mostrados en la Tabla 8 y Tabla 9 servirán para realizar el pronóstico de las ventas diarias por mes para el año 2018.

La venta diaria y pronosticada con la red neuronal en miles de soles para el año 2018 mensualmente por día se presenta en la Tabla 10, estos datos se obtuvieron al ejecutar la aplicación consola.

Tabla 91.

*Venta diaria y pronosticada con la red neuronal en miles de soles desde Enero a Julio del año 2018*

MES	DIA	VENTA DIARIA	VENTA DIARIA PRONOSTICADA CON LA RED
ENERO	1	20361	22380
	2	23291	22294
	3	23327	22349
	4	22868	22349
	5	23630	21921
	6	24428	22448
	7	22375	22437
	8	20396	21881
	9	20138	22452
	10	23943	22193
	11	24940	22595
	12	20401	22444
	13	22777	22498
	14	24489	22454
	15	22926	22264
	16	22978	22168
	17	23756	22513
	18	21116	22538
	19	21211	22357

	20	21368	21662
	21	24958	22391
	22	20706	22432
	23	20980	21747
	24	22770	22553
	25	22964	21435
	26	22869	22256
	27	20122	22425
	28	20166	22489
	29	22832	22433
	30	21589	22347
<b>TOTAL</b>		<b>670675</b>	<b>668705</b>
FEBRERO	31	20422	22301
	32	22488	22439
	33	24400	22318
	34	22568	22477
	35	23924	22446
	36	20341	22405
	37	21628	22441
	38	24559	21725
	39	24677	22521
	40	24109	21936
	41	24849	22191
	42	21067	22442
	43	20997	21832
	44	23223	22546
	45	23102	21905
	46	21370	22448
	47	20290	22446
	48	22317	22511
	49	22271	22584
	50	23736	21800
	51	20299	22090
	52	21170	22413
	53	24903	22447
	54	22313	22463
	55	23426	22187
	56	22281	22278
	57	22977	22440
	58	23389	21887
	59	23404	22520
	60	21851	21764
	<b>TOTAL</b>		<b>678351</b>
MARZO	61	22884	22375
	62	24206	22504
	63	22934	21565



	64	23439	22510
	65	23598	19735
	66	23136	22298
	67	23872	22442
	68	22647	22501
	69	24592	22409
	70	21205	21571
	71	23256	22164
	72	24896	22441
	73	23835	22387
	74	20789	22350
	75	20899	22160
	76	24126	22030
	77	20428	22138
	78	24657	22146
	79	22265	22214
	80	24194	21399
	81	21754	21940
	82	21302	22433
	83	22193	22490
	84	21737	22473
	85	20373	21801
	86	24986	22338
	87	22978	22443
	88	23222	22501
	89	23758	22546
	90	22029	21345
	91	22597	22343
<b>TOTAL</b>		<b>708787</b>	<b>663617</b>
ABRIL	92	20674	22433
	93	24507	22512
	94	24580	22506
	95	22271	22036
	96	23499	22139
	97	24877	22509
	98	24439	22355
	99	20867	22460
	100	20923	21907
	101	23133	22460
	102	23654	22391
	103	21238	22388
	104	24022	22336
	105	24288	22493
	106	20419	22327
	107	24326	22508
	108	24962	22460

	109	22382	22500
	110	20866	22473
	111	23869	22205
	112	22389	22506
	113	20398	21824
	114	22071	22509
	115	22085	16130
	116	22715	22389
	117	20298	22507
	118	22409	22498
	119	21752	22516
	120	23102	16476
	121	20098	22512
	122	24182	22385
<b>TOTAL</b>		<b>701295</b>	<b>659217</b>
	123	23572	21876
	124	21864	22466
	125	23574	15804
	126	23073	22511
	127	23812	22507
	128	24345	22003
	129	23262	21909
	130	24809	16006
	131	21440	22508
	132	20083	22504
	133	23466	22461
	134	21427	22493
	135	22471	16266
	136	22133	15898
	137	21473	22511
	138	23080	22120
	139	22093	21972
	140	20194	22435
	141	21806	15879
	142	20862	22511
	143	24787	22158
	144	24606	22453
	145	23823	22259
	146	20860	15902
	147	23596	22511
	148	22458	21612
	149	24833	22350
	150	23335	22406
	151	23245	15890
	152	20867	22511
	153	20511	22097
<b>MAYO</b>			

<b>TOTAL</b>		<b>701760</b>	<b>646789</b>
JUNIO	154	24914	22263
	155	21808	22364
	156	20989	15896
	157	23822	22510
	158	23197	22073
	159	24462	22510
	160	24685	22290
	161	20067	15880
	162	22101	21955
	163	20484	21658
	164	22311	21137
	165	21684	21935
	166	23735	15907
	167	20321	22401
	168	22158	22085
	169	22399	22385
	170	22182	22208
	171	22133	15920
	172	22172	22481
	173	23215	22143
	174	20619	22504
	175	21287	22210
	176	20327	15920
	177	22825	22489
	178	22081	22124
	179	21146	22045
	180	22955	22322
	181	22613	15922
	182	23200	22489
	183	22856	22125
	184	24123	22480
<b>TOTAL</b>		<b>692871</b>	<b>650631</b>
JULIO	185	21234	22167
	186	22205	15936
	187	21852	22024
	188	23300	22450
	189	24718	22132
	190	22888	22284
	191	24513	15936
	192	21602	22489
	193	21831	22076
	194	22747	22315
	195	21831	22318
	196	20003	15945
	197	24958	22464

	198	21115	22016
	199	23424	22446
	200	24349	22035
	201	21404	15971
	202	22389	22306
	203	24659	21530
	204	21367	22293
	205	23831	21878
	206	23341	15989
	207	22137	22481
	208	21802	22071
	209	21469	22083
	210	24102	22464
	211	20128	15971
	212	24741	22322
	213	24752	21991
	214	24546	21878
	215	20933	21978
<b>TOTAL</b>		<b>704171</b>	<b>650239</b>

Nota: *Elaboración propia.*

La venta diaria y pronosticada se grafica en la Fig. 17 en miles de soles por día, se utiliza la venta diaria que la empresa obtuvo desde Enero a Julio en línea anaranjada, y la venta pronosticada con la red neuronal presentada con una línea ploma.

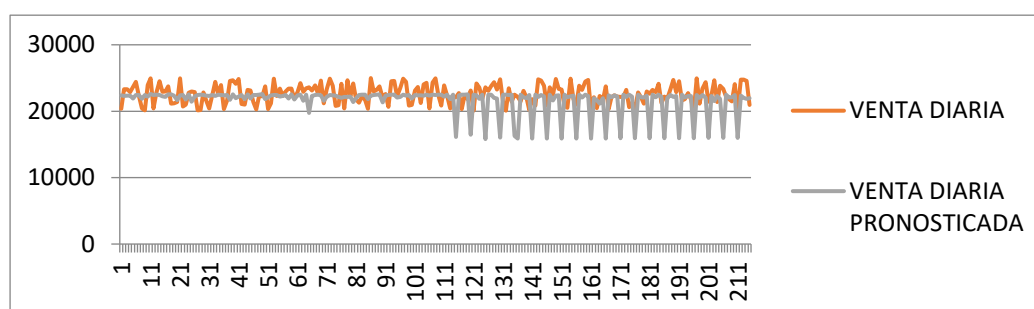


Figura 29: Venta diaria y pronosticada por día con la red neuronal graficada utilizando Hoja de cálculo de Microsoft Office Excel 2010.

Fuente: *Elaboración propia.*

En la Fig. 29 y la Fig. 30 se presenta la venta diaria y pronosticada en miles de soles por día, los gráficos se generan una vez acabada de ejecutarse la red

neuronal, se utilizó la venta diaria que la empresa obtuvo desde Enero a Julio en línea roja, y la venta pronosticada con la red neuronal presentada con una línea negra.

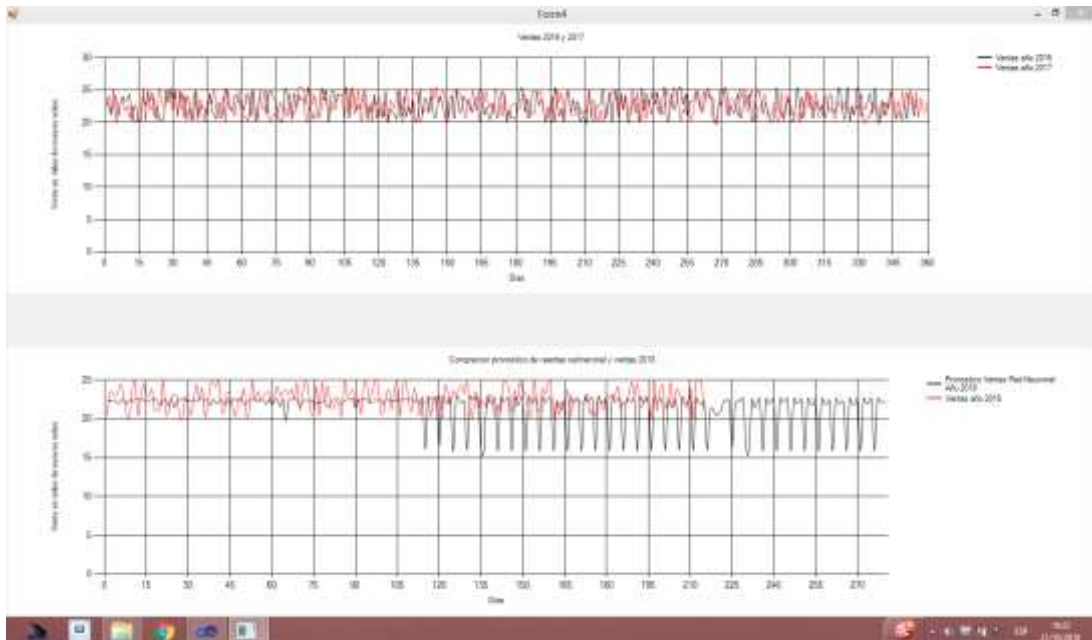


Figura 30: Venta diaria y pronosticada por día con la red neurona con un ratio de aprendizaje de 0.01 utilizando la aplicación.

Fuente: *Elaboración propia.*

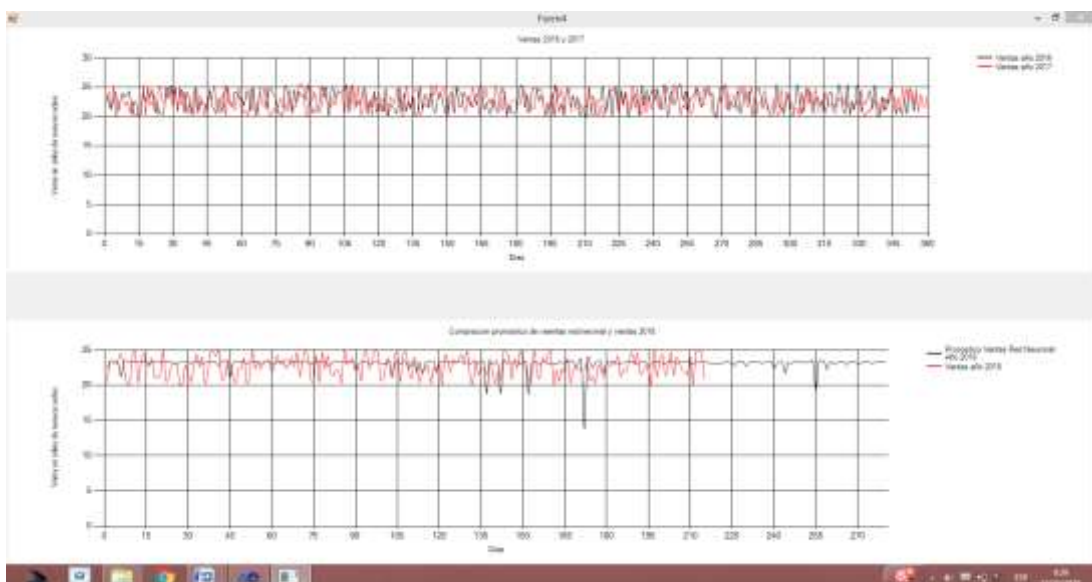


Figura 31: Venta diaria y pronosticada por día con la red neuronal con un ratio de aprendizaje de 0.1 utilizando la aplicación.

Fuente: *Elaboración propia.*

Los datos de las ventas diarias por mes mostrados en la Tabla 9 y 10 proceden de las ventas de la empresa. Estos datos sirven de aprendizaje para la red neuronal y se almacenan en un arreglo bidimensional de la siguiente manera siguiente:

20,784	20,308	22,495	23,666	22,949
23,527	22,121	21,198	22,972	24,070
20,970	22,829	23,985	23,324	23,537
24,210	20,348	22,531	22,060	20,132
20,627	24,004	22,551	24,529	23,218
....	....	....	....	....
22,627	24,865	23,241	22,429	25,364
24,059	23,127	20,242	21,981	21,414
24,617	20,589	20,830	22,681	24,531
24,221	21,558	24,721	23,515	24,495
23,952	20,489	20,413	20,115	23,088

El aprendizaje de la red neuronal se realiza tomando de 5 en 5 los datos para predecir las ventas diarias desde enero hasta julio del 2018 como se presenta en la Tabla. La venta diaria por mes y venta diaria mensual pronosticada con la red se totaliza en venta mensual promedio y venta mensual pronosticada con la red como se aprecia en la Tabla 12.

Tabla 12.

*Venta mensual promedio y venta mensual pronosticada con la red neuronal en miles de soles desde Enero a Julio para el año 2018.*

MES	VENTA MENSUAL PROMEDIO	VENTA MENSUAL PRONOSTICADA CON LA RED
<b>ENERO</b>	670675	668705
<b>FEBRERO</b>	678351	645902
<b>MARZO</b>	708787	663617
<b>ABRIL</b>	701295	659217
<b>MAYO</b>	701760	646789
<b>JUNIO</b>	692871	650631
<b>JULIO</b>	704171	650239

Nota: *Elaboración propia.*

La venta mensual pronosticada con la red se aproxima al valor deseado. Tal como se observan en la Tabla 12.

Tabla 13.

*Pronóstico de la empresa tomando las ventas diarias del año 2017 para realizar el pronóstico del año 2018 desde Enero a Julio.*

<b>AÑO</b>	<b>MES</b>	<b>DÍA</b>	<b>VENTAS</b>
2017	ENERO	1	21,788
		2	22,671
		3	20,712
		4	21,45
		5	23,933
		6	23,996
		7	21,444
		8	22,3
		9	25,305
		10	24,421
		11	21,395
		12	20,679
		13	25,375
		14	20,308
		15	22,515
		16	21,2
		17	25,072
		18	24,471
		19	20,944
		20	22,277
		21	22,18
		22	23,601
		23	24,314
		24	21,322
		25	25,299
		26	20,511
	FEBRERO	27	20,072
		28	23,311
		29	21,104
		30	22,942
		31	22,27
		32	24,49
		33	23,459
		34	20,609
		35	21,213
		36	21,199
		37	20,441

	38	22,956
	39	20,519
	40	24,422
	41	24,133
	42	20,914
	43	23,383
	44	22,593
	45	24,146
	46	21,622
	47	25,215
	48	20,078
	49	21,083
	50	22,874
MARZO	51	23,639
	52	21,826
	53	22,484
	54	20,816
	55	22,456
	56	20,394
	57	20,215
	58	20,661
	59	20,561
	60	25,299
	61	20,265
	62	24,973
	63	21,889
	64	22,802
	65	23,468
	66	20,363
	67	25,139
	68	23,185
	69	22,361
	70	23,562
	71	24,905
	72	24,389
	73	21,636
	74	24,675
	75	20,879
	76	24,848
	77	21,969
	78	24,232
ABRIL	79	20,457
	80	22,105
	81	20,4



	82	24,391
	83	22,728
	84	20,397
	85	23,542
	86	21,535
	87	20,16
	88	22,864
	89	23,128
	90	23,003
	91	25,169
	92	21,404
	93	20,706
	94	21,188
	95	24,001
	96	20,205
	97	24,916
	98	22,006
	99	22,396
	100	21,003
	101	23,321
	102	24,735
	103	20,302
	104	21,051
	105	20,815
	106	20,363
	107	24,504
	108	23,794
	109	21,964
	110	20,468
	111	24,011
	112	22,127
	113	22,112
MAYO	114	25,038
	115	20,401
	116	20,105
	117	21,315
	118	20,824
	119	21,876
	120	23,143
	121	21,543
	122	22,115
	123	20,918
	124	24,471
	125	20,531

	126	21,687
	127	21,884
	128	25,335
	129	25,334
JUNIO	130	24,821
	131	20,464
	132	20,468
	133	25,227
	134	22,671
	135	24,122
	136	21,474
	137	20,586
	138	22,073
	139	24,827
	140	22,931
	141	22,234
	142	21,598
	143	23,237
	144	21,855
	145	23,98
	146	24,177
	147	23,869
	148	20,436
	149	24,918
	150	21,761
	151	25,208
	152	21,619
	153	21,847
	154	24,886
	155	20,499
JULIO	156	21,06
	157	24,402
	158	21,458
	159	22,91
	160	20,176
	161	23,343
	162	22,748
	163	24,282
	164	20,869
	165	23,118
	166	20,434
	167	20,422
	168	23,288
	169	22,682

	170	22,315
	171	24,71
	172	22,937
	173	20,295
	174	20,9
	175	25,169
	176	22,601
	177	21,246
	178	23,881
	179	24,587
	180	24,117
	181	20,186
	182	20,949
	183	22,148
	184	21,717
	185	22,621
	186	25,293
	187	23,631
	188	25,358
	189	23,137
	190	23,716
	191	25,045
	192	23,732
	193	20,176
	194	21,936
AGOSTO	195	23,201
	196	24,903
	197	25,098
	198	24,619
	199	23,973
	200	21,969
	201	20,63
	202	23,618
	203	21,764
	204	24,921
	205	23,223
	206	23,84
	207	20,226
	208	22,404
	209	23,106
	210	24,726
SEPTIEMBRE	211	22,366
	212	22,865
	213	21,37

	214	25,221
	215	23,932
	216	23,15
	217	20,973
	218	24,4
	219	20,804
	220	21,352
	221	22,559
	222	22,865
	223	22,645
	224	22,248
	225	22,177
	226	24,546
	227	23,816
	228	20,746
	229	21,391
	230	21,607
	231	21,369
	232	24,622
	233	24,329
	234	24,655
	235	22,234
	236	20,45
	237	22,369
	238	21,178
	239	23,955
	240	20,161
	241	24,844
	242	23,186
	243	24,422
	244	20,417
	245	24,199
OCTUBRE	246	24,611
	247	21,297
	248	21,163
	249	21,172
	250	21,33
	251	22,211
	252	21,101
	253	21,885
	254	21,63
	255	25,162
	256	20,724
	257	23,512

	258	21,064
	259	20,115
	260	21,924
NOVIEMBRE	261	20,078
	262	21,153
	263	21,472
	264	24,539
	265	24,261
	266	24,888
	267	23,551
	268	20,564
	269	23,62
	270	21,817
	271	23,724
	272	21,074
	273	22,502
	274	25,215
	275	21,452
	276	24,607
	277	21,419
	278	20,459
	279	23,356
	280	20,555
	281	23,476
	282	21,199
	283	23,293
	284	24,637
285	21,883	
286	22,217	
DICIEMBRE	287	21,123
	288	22,627
	289	24,865
	290	23,241
	291	22,429
	292	25,364
	293	24,059
	294	23,127
	295	20,242
	296	21,981
	297	21,414
	298	24,617
	299	20,589
	300	20,83
	301	22,681

		302	24,531	
		303	24,221	
		304	21,558	
		305	24,721	
		306	23,515	
		307	24,495	
		308	23,952	
		309	20,489	
		310	20,413	
		311	20,115	
		312	23,088	
2018	ENERO	<b>313</b>	<b>22,6769102</b>	
		314	22,6776425	
		315	22,6783748	
		316	22,6791071	
		317	22,6798394	
		318	22,6805717	
		319	22,681304	
		320	22,6820363	
		321	22,6827686	
		322	22,6835009	
		323	22,6842332	
		324	22,6849655	
		325	22,6856979	
		326	22,6864302	
		327	22,6871625	
		328	22,6878948	
		329	22,6886271	
		330	22,6893594	
		331	22,6900917	
		332	22,690824	
		333	22,6915563	
		334	22,6922886	
		335	22,6930209	
		336	22,6937532	
		337	22,6944855	
		338	22,6952178	
		339	22,6959501	
		340	22,6966824	
		341	22,6974147	
		342	22,698147	
		FBRERO	343	22,6988793
			344	22,6996116
	345		22,7003439	

	346	22,7010763
	347	22,7018086
	348	22,7025409
	349	22,7032732
	350	22,7040055
	351	22,7047378
	352	22,7054701
	353	22,7062024
	354	22,7069347
	355	22,707667
	356	22,7083993
	357	22,7091316
	358	22,7098639
	359	22,7105962
	360	22,7113285
	361	22,7120608
	362	22,7127931
	363	22,7135254
	364	22,7142577
	365	22,71499
	366	22,7157224
	367	22,7164547
	368	22,717187
	369	22,7179193
	370	22,7186516
	371	22,7193839
	372	22,7201162
	373	22,7208485
	374	22,7215808
	375	22,7223131
	376	22,7230454
	377	22,7237777
	378	22,72451
	379	22,7252423
	380	22,7259746
MARZO	381	22,7267069
	382	22,7274392
	383	22,7281715
	384	22,7289038
	385	22,7296361
	386	22,7303684
	387	22,7311008
	388	22,7318331
	389	22,7325654

	390	22,7332977
	391	22,73403
	392	22,7347623
	393	22,7354946
	394	22,7362269
	395	22,7369592
	396	22,7376915
	397	22,7384238
	398	22,7391561
	399	22,7398884
	400	22,7406207
	401	22,741353
	402	22,7420853
	403	22,7428176
	404	22,7435499
	405	22,7442822
	406	22,7450145
	407	22,7457468
	408	22,7464792
	409	22,7472115
	410	22,7479438
	411	22,7486761
	412	22,7494084
	413	22,7501407
	414	22,750873
	415	22,7516053
	416	22,7523376
	417	22,7530699
ABRIL	418	22,7538022
	419	22,7545345
	420	22,7552668
	421	22,7559991
	422	22,7567314
	423	22,7574637
	424	22,758196
	425	22,7589283
	426	22,7596606
	427	22,7603929
	428	22,7611253
	429	22,7618576
	430	22,7625899
	431	22,7633222
	432	22,7640545
MAYO	433	22,7647868



	434	22,7655191
	435	22,7662514
	436	22,7669837
	437	22,767716
	438	22,7684483
	439	22,7691806
	440	22,7699129
	441	22,7706452
	442	22,7713775
	443	22,7721098
	444	22,7728421
	445	22,7735744
	446	22,7743067
	447	22,775039
	448	22,7757713
	449	22,7765037
	450	22,777236
	451	22,7779683
	452	22,7787006
	453	22,7794329
	454	22,7801652
	455	22,7808975
	456	22,7816298
	457	22,7823621
	458	22,7830944
	459	22,7838267
	460	22,784559
	461	22,7852913
	462	22,7860236
	463	22,7867559
	464	22,7874882
	465	22,7882205
	466	22,7889528
	467	22,7896851
	468	22,7904174
	469	22,7911497
JUNIO	470	22,7918821
	471	22,7926144
	472	22,7933467
	473	22,794079
	474	22,7948113
	475	22,7955436
	476	22,7962759
	477	22,7970082

	478	22,7977405
	479	22,7984728
	480	22,7992051
	481	22,7999374
	482	22,8006697
	483	22,801402
	484	22,8021343
	485	22,8028666
	486	22,8035989
	487	22,8043312
	488	22,8050635
	489	22,8057958
	490	22,8065282
	491	22,8072605
	492	22,8079928
	493	22,8087251
	494	22,8094574
	495	22,8101897
	496	22,810922
	497	22,8116543
	498	22,8123866
	499	22,8131189
	500	22,8138512
	501	22,8145835
	502	22,8153158
	503	22,8160481
	504	22,8167804
	505	22,8175127
	506	22,818245
JULIO	507	22,8189773
	508	22,8197096
	509	22,8204419
	510	22,8211742
	511	22,8219066
	512	22,8226389
	513	22,8233712
	514	22,8241035
	515	22,8248358
	516	22,8255681
	517	22,8263004
	518	22,8270327
	519	22,827765
	520	22,8284973
	521	22,8292296

		522	22,8299619
		523	22,8306942
		524	22,8314265
		525	22,8321588
		526	22,8328911
		527	22,8336234

Nota: *Información de la Empresa.*

#### Anexo 4.

Dado que el que suscribe el siguiente informe, trabajó en dicha entidad, se tuvo la coordinación con el Jefe de Ventas de la Empresa Cerámicos Lambayeque SAC, Sr. Boris Coiffmman Miranda, quien facilitó la información del Anexo 3