



UNIVERSIDAD DE CARABOBO
FACULTAD DE INGENIERÍA
ÁREA DE ESTUDIOS DE POSTGRADO
MAESTRÍA EN MATEMÁTICA Y COMPUTACIÓN



SISTEMA DE REDES NEURONALES PARA LA
EVALUACIÓN DE PROGRAMAS DE POSTGRADO

Autor: Ing. Eduardo Sthory.
C.I.10.234.088.
Tutor: PhD. Demetrio Rey Lago.
C.I. 7.127.552

Naguanagua, Marzo del 2019.



UNIVERSIDAD DE CARABOBO
FACULTAD DE INGENIERÍA
ÁREA DE ESTUDIOS DE POSTGRADO
MAESTRÍA EN MATEMÁTICA Y COMPUTACIÓN



SISTEMA DE REDES NEURONALES PARA LA
EVALUACIÓN DE PROGRAMAS DE POSTGRADO

Autor: Ing. Eduardo Sthory.
C.I.10.234.088.
Trabajo presentado ante el Área de
estudios de Postgrado de la Universidad
de Carabobo para optar al Título de
Magíster en Matemática y Computación.

Naguanagua, Marzo del 2019.



**UNIVERSIDAD DE CARABOBO
FACULTAD DE INGENIERÍA
ÁREA DE ESTUDIOS DE POSTGRADO
MAESTRÍA EN MATEMÁTICA Y COMPUTACIÓN**



**SISTEMA DE REDES NEURONALES PARA LA
EVALUACIÓN DE PROGRAMAS DE POSTGRADO**

**Autor: Ing. Eduardo Sthory.
C.I.10.234.088.**

**Aprobado en el Área de Estudios de Postgrado de la Universidad de
Carabobo por Miembros de la Comisión Coordinadora del Programa:**

Naguanagua, Marzo del 2019.



UNIVERSIDAD DE CARABOBO
FACULTAD DE INGENIERIA
DIRECCIÓN DE ESTUDIOS PARA GRADUADOS
SECCIÓN DE GRADO



ACTA DE DISCUSIÓN DE TRABAJO DE GRADO

En atención a lo dispuesto en los Artículos 137, 138 y 139 del Reglamento de Estudios de Postgrado de la Universidad de Carabobo, quienes suscribimos como Jurado designado por el Consejo de Postgrado de la Facultad de Ingeniería, de acuerdo a lo previsto en el Artículo 135 del citado Reglamento, para estudiar el Trabajo de Grado titulado:

“SISTEMA DE REDES NEURALES PARA LA EVALUACIÓN DE PROGRAMAS DE POSTGRADO”

*Presentado para optar al grado de **MAGÍSTER EN MATEMÁTICA Y COMPUTACIÓN** por el (la) aspirante:*

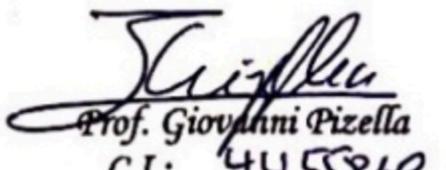
EDUARDO STHORY
V.- 10.234.088

*Habiendo examinado el Trabajo presentado, decidimos que el mismo está **APROBADO**.*

En Valencia, a los catorce (14) día del mes de Junio del año dos mil diecinueve.


Prof. Joan Ordoñez
C.I.: 12278470
Fecha: 14/06/2019


Prof. Alejandro Centeno
C.I.: 11378441
Fecha: 14/06/2019


Prof. Giovanni Pizella
C.I.: 4455859
Fecha: 14/06/2019



UNIVERSIDAD DE CARABOBO
FACULTAD DE INGENIERÍA
ÁREA DE ESTUDIOS DE POSTGRADO
MAESTRÍA EN MATEMÁTICA Y COMPUTACIÓN



VEREDICTO

Nosotros, Miembros del Jurado designado para la evaluación del Trabajo de Grado titulado: SISTEMA DE REDES NEURONALES PARA LA EVALUACIÓN DE PROGRAMAS DE POSTGRADO, presentado por: Ing. Eduardo Sthory para optar al título de Magíster en Matemática y Computación, estimamos que el mismo reúne los requisitos para ser considerado como:

Presidente

Jurado

Jurado

Dedico este trabajo a todas las personas convencidas de la educación como principal arma para escapar de la ignorancia que secuestra, somete y mantiene en las sombras al ser humano, como dijo el gran maestro "... y la verdad os hará libres".

RECONOCIMIENTOS

Al Prof. PhD. Demetrio Rey Lago por haber aceptado la tutoría de este trabajo, su dedicación y valiosas asesorías.

Al Prof. MSc. Yoan Ordoñez miembro de la Comisión Coordinadora de la Maestría de Matemática y Computación, por su invaluable atención y asesoría sin descanso.

A la insigne Alma Mater Universidad de Carabobo, particularmente al honorable Posgrado de la Escuela de Ingeniería, al cual le debo gran parte de mi desarrollo académico y profesional.

ÍNDICE GENERAL

ÍNDICE DE FIGURAS	xiv
ÍNDICE DE GRÁFICOS	xvii
ÍNDICE DE CUADROS	xviii
RESUMEN	xix
INTRODUCCION	21
EL PROBLEMA	
Planteamiento del Problema	22
Objetivos de la Investigación	26
Objetivo General	26
Objetivos Específicos	26
Justificación	27
MARCO TEÓRICO	
Antecedentes de la Investigación	29
Bases Teóricas	32
Inteligencia Artificial	32
Redes Neuronales Artificiales	34

Neurona Biológica	36
Neurona Artificial	36
Características de las Neuronas Artificiales	37
Elementos básicos de una Neurona Artificial	38
Tipos de Neuronas Artificiales	42
Perceptrón	42
Pesos de la Neurona	42
Activación de la Neurona	43
Red Neuronal Artificial (RNA)	44
Elementos de una Red Neuronal Artificial	45
Capa de entrada o visibles	45
Capas ocultas	46
Capa de salida	46
Entrenamiento de las RNA	47
Preparación de Datos	47
Normalización	48
Pendiente del Gradiente Estocástico	48
Épocas	49
Actualizaciones de Peso	49

Tasa de Aprendizaje	49
Momentum	50
Tasa de decaimiento de Aprendizaje	50
Predicción	51
Clasificación de las Redes Neuronales Artificiales (RNA)	52
Según su Arquitectura	52
Según su aprendizaje	55
Parámetros de una red neuronal	56
RNA Tipo Perceptrón Multicapa	56
Algoritmo Backpropagation	58
Máquina de Aprendizaje (Machine Learning)	61
Enfoques para definir el aprendizaje de la máquina	61
Clasificación según el estilo de Aprendizaje	62
Aprendizaje reforzado (Reinforcement Learning)	64
Saturación en la Red Neuronal Artificial	64
Regularización de Abandono	65
Topologías de redes neuronales	65
Redes de propagación hacia adelante	66
Redes Recurrentes	66

Aprendizaje Profundo (Deep Learning) y	
Redes Neuronales Profundas (DNN)	67
GPU contra CPU	70
Factores que Influyen en el Rendimiento del Aprendizaje	
de la Red	70
Términos Básicos	72

MARCO METODOLÓGICO

Tipo de Investigación	74
Modalidad de Estudio	75
Diseño de Investigación	76
Fases del Proyecto	76
Diagnóstico	76
Estudio de Factibilidad	77
Diseño	77
Población y Muestra	78
Técnicas de Recolección de Datos	79
Instrumentos de Recolección de Datos	80
Técnicas de Análisis de los Datos	80

RESULTADOS	81
LA PROPUESTA	
Encuesta al Estudiante	85
Preparación de la data	85
Muestra para el entrenamiento de la RNA	88
Generación de datos de entrada a partir de la muestra Seleccionada	89
Representación Funcional entre las RNA y sus Datos de Entrada y Salida	89
Almacenaje de los Vectores Lineales en una Tabla	91
Arquitectura de la RNA	92
Tipo de Red Neuronal	92
Número de Neuronas en la Capa de Entrada	93
Número de Neuronas en la Capa de Salida	93
Número de Neuronas en las capas ocultas	95
Funciones de activación para las neuronas	95
Momentum	96
Función de Pérdida	97

Generación de la base de entrenamiento	98
Representación en una Tabla el Conocimiento de la Evaluación de las Encuestas	98
Generación de la Base de Entrenamiento	99
Generación de los datos de salida de la RNA a partir de la muestra seleccionada	100
Entrenamiento de la RNA en la predicción de las encuestas	101
Ingreso de Parámetros	102
Justificación de los Hiper-Parámetros Ingresados	103
Porcentaje de Datos para la Validación	103
Número de muestras a propagarse	103
Épocas	103
Resultado del entrenamiento	103
Resumen de ajuste de Hiperparámetros utilizados	104
Análisis de la ejecución del Entrenamiento	105
Eficiencia de la RNA entrenada	115
Gráficos del funcionamiento de los Nodos de Tensor Flow	116
Interpretación de los Datos de Salida	119
Resultado de la Validación (Predicción)	121

CONCLUSIONES Y RECOMENDACIONES	122
BIBLIOGRAFIA	125
ANEXOS	127
ANEXO A-1 - ENCUESTA A LOS ESTUDIANTES DE	
SATISFACCIÓN	128

ÍNDICE DE FIGURAS

Figura

1	Red Neuronal Biológica y Red Neuronal Artificial	35
2	Neurona biológica	36
3	Neurona artificial	37
4	Representación de una neurona artificial	38
5	Representación de los tipos de funciones de activación 1	40
6	Representación de los tipos de funciones de activación 2	41
7	Red Neuronal Artificial	44
8	Elementos de una red Neuronal Artificial	45
9	RNA con Conexiones hacia Adelante Mono-Capa	53
10	RNA con Conexiones hacia Adelante Multicapa	53
11	RNA con Conexiones hacia Atrás	54
12	Aprendizaje Supervisado de una RNA	55
13	Modelo de un Perceptrón	57
14	Modelo de un Perceptrón Multicapa	58
15	Backpropagation hacia Adelante	59
16	Backpropagation hacia Atrás	60

17	Red con propagación hacia adelante	66
18	Red Recurrente	67
19	Efecto del Número de Muestras de Aprendizaje	70
20	Efecto del Número de Unidades Ocultas	71
21	Efecto del Número de Iteraciones	71
22	Efecto de Sobre-Entrenamiento	72
23	Vista parcial del archivo base de la data “cruda” en formato Excel	86
24	Vista de la data de entrada del modelo neuronal	87
25	Vista de la data de salida del modelo neuronal	87
26	Representación funcional RNA y los datos (entrada/salida)	89
27	Dependencia Funcional entre Encuesta/Evaluación	90
28	Relación entre los Elementos de la Muestra y la Evaluación	90
29	Arquitectura de la RNA para la Predicción de las encuestas	94
30	Representación Funcional RNA/Datos/Entrada/Salida	94
31	Base de Entrenamiento de la RNA	100
32	Ingreso de Parámetros en RNA	102
33	Estructura principal de los nodos creados por TensorFlow	118
34	Estructura principal 2 de los nodos creados por TensorFlow	118
35	Nodos auxiliares creados por TensorFlow	119

36	Datos de Entrada y Salida del Proceso de Reconocimiento	120
37	Resultados del entrenamiento de la RNA	121

ÍNDICE DE GRÁFICOS

Gráfico

1	Función de activación ReLu en las capas ocultas	105
2	Función de Gradiente en la Primera capa oculta de la RNA	106
3	Función Softmax en las neuronas de la capa de salida	107
4	Función de Gradiente de acuerdo al learning-rate 1	108
5	Función de Gradiente de acuerdo al learning-rate 2	109
6	Función de Gradiente en la capa oculta 2 de la RNA	110
7	Función de Gradiente en la capa oculta 1 de la RNA	111
8	Función de Momentum Nesterov Capa Oculta 1 de la RNA	112
9	Función de Momentum Nesterov Capa Oculta 2 de la RNA	113
10	Función de Momentum Nesterov Capa Oculta 1 y 2	114
11	Progreso de la Exactitud de la predicción de la RNA	115
12	Progreso de la Exactitud vs Validación de la RNA	116

ÍNDICE DE CUADROS

Cuadro

1	Personal que labora en el área de Posgrado de la Facultad de Ingeniería Universidad de Carabobo	78
2	Respuesta ponderada numéricamente	88
3	Matriz de datos de entrada de la RNA	91
4	Matriz de datos de salida de la RNA	92
5	Representación del Conocimiento	99

UNIVERSIDAD DE CARABOBO
FACULTAD DE INGENIERÍA
ÁREA DE ESTUDIOS DE POSTGRADO
MAESTRÍA EN MATEMÁTICA Y COMPUTACIÓN

**SISTEMA DE REDES NEURONALES PARA LA
EVALUACIÓN DE PROGRAMAS DE POSTGRADO**

Autor: Ing. Eduardo Sthory

Tutor: PhD. Demetrio Rey Lago

Marzo del 2019

RESUMEN

Esta investigación utiliza tecnología de Redes Neuronales y Deep Learning para diseñar un sistema de aprendizaje que procese encuestas de satisfacción hechas a los estudiantes de postgrado. El sistema se encarga de procesar la información suministrada y es capaz de aprender cuales son las calificaciones adecuadas para cada tipo de encuesta, de esta manera puede realizar pronósticos una vez finalizada la fase de aprendizaje. Utilizando las herramientas Python y Tensorflow se realiza la programación del sistema para posteriormente realizar el entrenamiento y su posterior validación. La metodología utilizada es de tipo descriptivo. Se quiere medir de manera independiente el impacto del uso de un sistema basado en redes neuronales en las encuestas de satisfacción de los estudiantes de postgrado. La modalidad de la investigación se enmarca en los lineamientos de proyecto factible. El diseño de la investigación es de campo, los datos de interés se recogen directamente de la realidad, mediante el trabajo directo y concreto del investigador y su equipo. Técnicas de Recolección de Datos: Se utiliza la observación directa y simple, y la entrevista no estructurada. La población de la investigación es la Coordinación de los programas de Postgrado de Ingeniería. La muestra que se toma para la evaluación de los programas de postgrado de Ingeniería es censal, es decir todos los programas de postgrado de Ingeniería de la Universidad de Carabobo y de esta forma se comprueba su funcionamiento.

Las fases del proyecto comienzan con un diagnóstico que consta de un estudio de campo basado en una investigación documental y bibliográfica, la cual hará posible respaldar el estudio y la propuesta. Luego se desarrollan entrevistas con el experto en el área para conseguir información básica acerca del problema de investigación. Análisis de los Resultados: Para el reporte de los resultados de la investigación se procedió en primera instancia a la recolección de los datos mediante el empleo de la observación directa y la entrevista no estructurada, dichas técnicas arrojaron resultados importantes para el desarrollo de la investigación y la propuesta de un sistema computacional de redes neuronales para la evaluación de los programas de postgrado de la Facultad de Ingeniería de la Universidad de Carabobo. Resultados: Se realiza el diseño del sistema de redes neuronales de acuerdo a la metodología de cascada. Luego de un proceso de aprendizaje el sistema logró obtener más del 95% de aciertos en comparación con las calificaciones humanas. Conclusiones: El sistema basado en redes neuronales logró obtener un porcentaje aceptable de aciertos (+95%), se concluye que se puede utilizar para automatizar el proceso de evaluación de las encuestas de manera aceptable o como una ayuda para las personas encargadas del proceso.

INTRODUCCION

El presente trabajo esta diseñado en capítulos, el primer capítulo, El Problema, plantea la problemática de forma general hasta llegar a lo particular, en el contexto del Postgrado de la Universidad de Carabobo Facultad de Ingeniería, luego, se formulan los objetivos y la justificación de la investigación.

El segundo capítulo, Marco Teórico, muestra los antecedentes de la investigación, las bases teóricas, los diversos conceptos relativos a la teoría de Redes Neuronales, Machine Learning y algunos tópicos de Inteligencia Artificial.

El tercer capítulo, Marco Metodológico, delinea los procedimientos de orden metodológico, a través de los cuales se intenta dar respuestas a las interrogantes objeto de investigación.

El cuarto capítulo da cuenta de los resultados, importantes para el desarrollo de la investigación y la propuesta de un sistema computacional de redes neuronales.

En el quinto capítulo, La Propuesta, se trata lo concerniente al sistema de redes neuronales planteado, diseño, entrenamiento, ejecución, eficacia y resultados del mismo.

El sexto capítulo, comprende las Conclusiones, Recomendaciones y la Bibliografía citada en el trabajo de investigación.

Finalmente se encuentran los anexos.

CAPÍTULO I

EL PROBLEMA

Planteamiento del Problema

Al pasar de los años se ha afianzado dentro de las ciencias de la computación un enfoque en los sistemas computacionales el cual tiene la finalidad de resolver problema que abarca un conjunto de metodologías caracterizadas por su inspiración es relacionados con el mundo real (reconocimiento de formas, toma de decisiones, realización de movimientos, aplicación de pruebas y su resolución, entre otros), de fácil ejecución y brindando respuestas rápidas en la soluciones de problemas, dicho conjunto de metodologías comprende una serie de elementos como lo son: la lógica borrosa, las redes neuronales, el razonamiento aproximado, los algoritmos genéticos, la teoría del caos y la teoría del aprendizaje, lo que quiere señalar que viene a facilitar el trabajo de grandes instituciones sobre todo desde el campo educativo

Dentro de estas metodologías, las Redes Neuronales son las que en la actualidad están ganando una mayor atención ello debido al gran impacto, en relación a su extraordinaria aplicabilidad práctica. Debido a ello dicha tecnología atrajo la atención de muchos profesionales, principalmente los profesionales dedicados a la estadística y al análisis de datos, que han hecho uso de las redes neuronales como herramientas orientadas a la clasificación de patrones y la estimación de variables continuas.

De allí que, con el advenimiento de computadoras más potentes, otras tareas de mayor complejidad e importancia les son asignadas a estas

máquinas, desde la elaboración de documentos hasta el control de artefactos como robots y cohetes espaciales, ello gracias al aprendizaje de las máquinas. Por consiguiente, dicha eventualidad se ha vuelto una de las mayores tendencias en el mundo de la informática, pues, dichos sistemas tienen una profunda capacidad de proporcionar información importante sobre una amplia variedad de aplicaciones, desde investigación médica innovadora hasta descubrimiento de aspectos físicos fundamentales del universo.

En ese mismo orden de ideas, resulta importante resaltar que en la década de los 40 fue creado un modelo para redes neuronales que se basó en algoritmos y matemáticas, denominado lógica de umbral. De allí que, las redes neuronales obtuvieron dos formas de enfoque, uno que se ajusta a procesos biológicos en el cerebro y otro en la aplicación para la inteligencia artificial.

Por consiguiente, uno de los mayores intereses en las redes neuronales es su posibilidad de aprender y resolver problemas como lo haría un humano, sin embargo, el uso de dichas funciones no es tan sencillo, esto depende de factores como la elección del modelo y el algoritmo de aprendizaje, entre otros. Por otra parte, esa habilidad de aprender y solucionar problemas como una persona es la que permite realizar tareas como la Evaluación de los Programas de Postgrado (EPP), el cual requiere un análisis integral de todas las variables estadísticas y resultados de los instrumentos que actualmente realiza evaluando los resultados de las encuestas de los EPP por una comisión conformada por un grupo de docentes.

De la misma manera, otra de las labores más apremiantes tienen que ver con el tiempo que dichas personas emplean, en evaluar dichas encuestas, lo cual origina retrasos e insuficiencia de tiempo para la

elaboración de la EPP, su complejidad y la falta de eficiencia en los medios electrónicos y manuales que actualmente recaban los datos que se introducen a los instrumentos de evaluación rápida, los cuales conforman el escenario de problemas que no permite que se realice adecuadamente este proceso.

En consecuencia, los Docentes que realizan la evaluación de los EPP en efecto tienen problemas para efectuar dicho trabajo y es posible que un sistema de redes neuronales podría analizar los instrumentos de los EPP y crear información que facilite dicho proceso, ahorrando tiempo y aportando un valioso diagnóstico dada sus capacidades ya mencionadas con anterioridad. De igual manera, los medios electrónicos y manuales existentes, se podrían adecuar a dicho sistema para hacer más práctica y expedita la obtención de los datos requeridos para alimentar el proceso de los EPP, redundando todo en una agilización y mejora del proceso en general, ahorrando exceso de trabajo y tiempo que actualmente se genera con el sistema mixto (manual y computacional).

De allí que, existe la necesidad de desarrollar estrategias que mejoren la obtención de los datos e integren la utilización de las redes neuronales en el análisis de los instrumentos y así poder satisfacer la necesidades que se requieren para evaluar los EPP tales como las horas de trabajo que se dedican al proceso y la inmediatez que se podría obtener con una automatización. Mediante la incorporación de dichas estrategias, se espera dar solución a los problemas mencionados anteriormente y proporcionar a la comisión coordinadora una herramienta que ayude a la realización de su labor respecto a este tema.

Formulación del Problema

¿Cuál será el modelo computacional a utilizar de Redes Neuronales para mejorar la evaluación de los programas de postgrado de la Facultad de Ingeniería de la Universidad de Carabobo?

Objetivos de la Investigación

Objetivo General

Proponer un sistema computacional de redes neuronales para la evaluación de los programas de postgrado de la Facultad de Ingeniería de la Universidad de Carabobo.

Objetivos Específicos

Diagnosticar la situación de los procesos de evaluación de programas de postgrado de Ingeniería de la Universidad de Carabobo.

Estudiar la factibilidad de incorporar un sistema de redes neuronales para la evaluación de los Programas de postgrado de Ingeniería de la Universidad de Carabobo.

Diseñar un sistema de redes neuronales para la evaluación de los programas de postgrado de Ingeniería de la Universidad de Carabobo.

Justificación del Estudio

La presente investigación se justifica desde el punto de vista académico para el área de Postgrado de Ingeniería de la Universidad de Carabobo, ya que busca minimizar el tiempo que requieren la comisión de Posgrado para realizar los EPP y el lapso prolongado que transcurre hasta su completa preparación, ya que, por medio de dicha investigación se busca implementar un sistema computacional bajo el modelo de redes neuronales que realice el proceso en forma automática con rapidez y eficiencia, dando respuesta a la problemática planteada en el menor tiempo posible ayudando a los docentes a realizar otras actividades que requieren de su presencia física y minimizando el tiempo de espera de los estudiantes.

Así mismo, con el presente trabajo se busca obtener la evaluación automática de los programas de postgrado que ofrezca una orientación en dicho proceso. Mediante el aporte de un valioso diagnóstico que pueden suministrar las redes neuronales dada las capacidades de estas para resolver tareas a través de un conjunto de observaciones que constituyen su aprendizaje, se permitiría la implantación y utilización del sistema que proporcione una herramienta que agilice y brinde ayuda para obtener calidad y eficiencia en las Evaluaciones de los Programas de Postgrado de la Facultad de Ingeniería de la Universidad de Carabobo, mejorando de esta forma el desempeño de la comisión coordinadora que lleva a cabo dicha tarea.

Por otra parte, la presente investigación presenta un interés científico y profesional ya que, por medio del mismo se implementaran técnicas novedosas y complejas de la inteligencia artificial como lo son el Aprendizaje de Máquina (Machine Learning en Inglés), que permite una introducción a

este mundo, lo que supone una innovación en esta materia y abre una vía a futuros trabajos relacionados con la materia, siendo este un paso necesario para la adecuación de los sistemas y personas a futuras tecnologías de información.

Por otra parte, una vez demostrada la validez y confiabilidad de la investigación, podrá ser utilizada en otros trabajos de investigación e instituciones que requieran el uso de tecnologías de información basadas en inteligencia artificial que se valen de técnicas como las redes neuronales y el aprendizaje de máquinas.

CAPÍTULO II

MARCO TEÓRICO

Para tener un conocimiento previo de la investigación se procederá en este capítulo a presentar una serie de trabajos relacionados con el tema a tratar, el cual ayudará a enmarcar la investigación con ello también se persigue ampliar la descripción del problema a partir de experiencias anteriores que pueden servir como guía para el desarrollo de la investigación, así mismo se desarrollaran teorías y postulados que darán la estructura teórica que sustentaran la investigación desde el punto de vista de diferentes autores e investigadores que han desarrollado trabajos de alto nivel que guardan una relación directa con el trabajo planteado.

En relación a lo antes expuesto Palella y Martins (2012) la fundamentación teórica “es el soporte principal del estudio, en él se amplía la descripción del problema, permite integrar la teoría con la investigación y establecer sus interacciones” (p.62)

Antecedentes de la Investigación

A nivel internacional, se tiene a los investigadores Esparza, Fuentes y del Real, (2017), con su trabajo de Grado Titulado *“Un modelo basado en el Clasificador Naïve Bayes para la evaluación del desempeño docente”*, presentado ante la Universidad Politécnica de Aguascalientes México, en el cual se describe el desarrollo y evaluación de un Modelo Computacional denominado Social Mining, basado en el algoritmo Naïve Bayes, para apoyar el análisis de las opiniones de los estudiantes en el proceso de la evaluación del desempeño docente, llevada a cabo mediante dispositivos móviles. Esta

propuesta considera el uso de dispositivos móviles para la recopilación de datos aprovechando su aceptación por parte de los estudiantes en el proceso de educación y aprendizaje.

Asimismo, se describe el desarrollo de corpus de subjetividad, el cual consta de un conjunto de términos afectivos relevantes de la evaluación docente para apoyar al algoritmo Naïve Bayes en la clasificación de las opiniones de los estudiantes dentro de las clases: positivo, negativo y neutral. Para medir el desempeño del proceso de la clasificación del Modelo Computacional Social Mining, se utilizan métricas como la matriz de confusión, precisión y la curva de ROC. Ya que la presente investigación trata sobre encuestas sobre el desempeño del personal docente y las asignaturas cursadas en general, este trabajo muestra procedimientos y técnicas adecuadas para el manejo de las mismas, de igual modo el manejo estadístico puede servir de base para esta investigación.

De igual forma, se tiene a Burgos y Escalona (2015), con su Trabajo de Grado Titulado *“Redes Neuronales Artificiales para predecir variables Antropométricas de trabajadores que laboran en postura decente”* presentado ante la Universidad de Carabobo. El objetivo de esta investigación, es el de modelar una RNA para predecir las variables antropométricas de trabajadores que laboran sentados, mediante el desarrollo de un algoritmo supervisado de propagación inversa (back propagation) de un software residente del Matlab 2012a. Se tomaron dos variables de entrada: estatura y anchura de hombros; y siete variables de salida: de una muestra de 300, este trabajo sería de utilidad para la presente investigación ya que utilizan y modelan Redes Neuronales Artificiales, las cuales también serán empleadas aquí, de igual manera el algoritmo de

propagación inversa (back propagation) es objeto de estudio para su posible implementación en esta investigación.

Igualmente se tomó como antecedente a la investigación el Trabajo de Grado de Santamaría y Rivero (2015), Titulado “Detección de la dirección de arribo usando técnicas de Redes Neuronales” presentado ante el Departamento de Electromagnetismo y Radiación de la Escuela de Ingeniería de Telecomunicaciones de la Facultad de Ingeniería en la Universidad de Carabobo, el cual tuvo como objetivo general: Desarrollar un modelo matemático que estime la dirección de arribo de las señales incidentes en un sistema de antenas inteligentes empleando una red neuronal con el fin de proporcionar un mejor desempeño del sistema de antenas inteligentes.

Dicha investigación se sustentó en una metodología con modalidad de proyecto factible, bajo un diseño no experimental, de tipo campo y nivel exploratorio y descriptivo, se empleó para la recolección de información la técnica de la entrevista con el cuestionario como instrumento de tipo cerrado validado por expertos y con una confiabilidad de 0,92, para el análisis se emplearon cuadras y gráficos con sus respectivos análisis

Las conclusiones a las que llegó el investigador señalaron que la aplicación de un modelo matemático empleando una red neuronal con el fin de proporcionar un mejor desempeño del sistema de antenas inteligentes, fue positivo en cuanto a los resultados arrojados ya que se estimó correctamente la dirección del arribo de las señales de las antenas inteligentes.

Así mismo, Cazorla y Sander (2015) con su Trabajo de Grado Titulado “Reconocimiento de patrones mediante redes neuronales para la determinación del nivel de contenido calcio en la Zanahoria” presentado ante

la Universidad Gastón Dachary, Argentina, la cual como objetivo tiene el establecer un método alternativo para determinar el nivel de contenido de calcio en la Zanahoria empleando el uso de redes neuronales (NRA). Dicha investigación fue de tipo descriptiva y exploratoria, los datos tomados fueron de diferentes cultivos de dicho rubro a los cuales se les aplico el análisis y luego se procedió a la aplicación de las NRA.

El análisis realizado llevo a la determinación que el reconocimiento con el uso de las NRA era más rápido y se usaba menor cantidad de horas hombres, lo que ayudo a bajar la cantidad de mano de obra a su cuarta parte trayendo como resultado un aumento en la generación de divisas para la empresa encargada de seleccionar los rubros para el uso doméstico y en restaurantes de la ciudad.

Santoyo y González (2013), con su Artículo “Comparación de predicción basada en redes neuronales contra métodos estadísticos en el pronóstico de ventas” de Escuela de Ingeniería Industrial. Universidad de Carabobo, el cual tuvo como objetivo realizar la comparación y selección de un método para pronosticar las ventas de forma eficiente y que beneficie a organizaciones que ofrecen sus productos al mercado ya que los pronósticos de ventas son datos de entrada a diferentes áreas de la empresa y de ser imprecisos pueden generar gastos para la organización.

Bases Teóricas

Inteligencia Artificial

A finales de los años 50, en los inicios de la Inteligencia Artificial (IA), se comenzaron a desatar pasiones y controversias sobre su utilidad, alcances y

límites; durante décadas dicho termino, ha estado presente en las ciencias de la computación, pasando por lo que se podrían denominar tres grandes estallidos, en los cuales han existido ciclos activos e inactivos siendo el actual el tercero y último con más auge hasta los momentos.

Sin embargo, el auge más reciente tiene una diferencia significativa con los pasados, el aprendizaje profundo ha logrado lo que las técnicas pasadas no podían lograr: que la máquina en sí sea capaz de averiguar las características de los datos dados, y aprender, con tal logro, se puede ver la gran posibilidad de que la IA se convierta en realidad, porque hasta ahora, no podía entender un nuevo concepto por sí misma, el ser humano necesitaba introducir previamente una cantidad de características y para ello utilizaba técnicas creadas en el pasado.

Para Kaplan y Haenlein (2009) la inteligencia artificial (IA) "es la capacidad de un sistema para interpretar correctamente datos externos, para aprender de dichos datos y emplear esos conocimientos para lograr tareas y metas concretas a través de la adaptación flexible" (p.15), es decir, la IA es la manera correcta de como una maquina puede emplear correctamente una información y transformarla en algo útil para quien la solicita, para ello el computador debe haber recibido una serie de datos y haber sido entrenada por medio de una serie de data con anterioridad.

Por consiguiente, la técnica mediante la cual se realiza este aprendizaje se logra utilizando una estructura llamada Red Neuronal Artificial (RNA).

Redes Neuronales Artificiales

Una red neuronal tiene que ver con el empleo de técnicas computacionales que se asemejen a las redes neuronales de los seres humanos, en relación a ello Vega (2011) señala:

Una RNA es un modelo computacional inspirado en redes neuronales biológicas que puede ser considerada como un sistema de procesamiento de información, es una estructura distribuida, de procesamiento paralelo, formada de neuronas artificiales (llamados también elementos de procesamiento), interconectados por un gran número de conexiones (sinapsis), los cuales son usados para almacenar conocimiento que está disponible para poder ser usado, intenta emular las funciones computacionales elementales de la red nerviosa del cerebro humano, en base a la interconexión de multitud de elementos de procesamiento, cada uno de los cuales presenta un comportamiento completamente local. (p.15)

En tal sentido, una red neuronal es una imitación redes neuronales biológicas, el cual consiste en un sistema de unidades de procesamientos que se encuentran en una red computacional, con el empleo de códigos que se encuentran estrechamente en conexión entre si y cada uno con una función definida, teniendo en consideración que dicha red se encuentra conformada por un modelo matemático constituido que tiene un conjunto de unidades de procesamiento denominados Neuronas, células o nodos, interconectados entre sí por varias ligaduras de comunicación directa, con el fin de recoger caracteres de entrada (señales), procesarlas y emitir caracteres (señales) de salida.

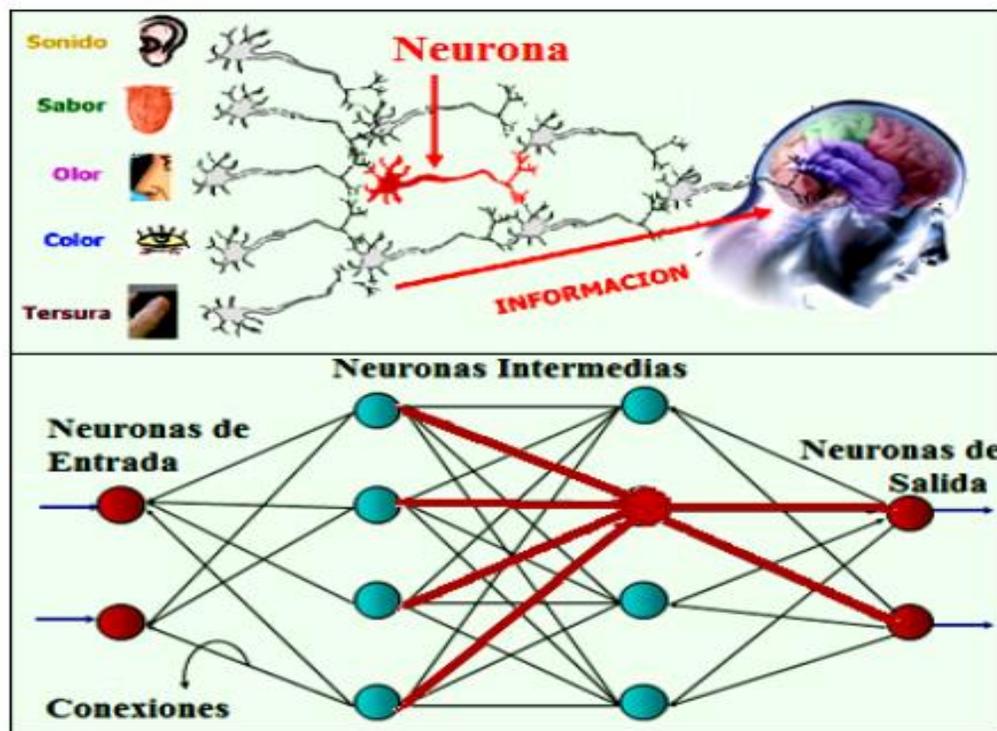


Figura 1. Red Neuronal Biológica y Red Neuronal Artificial tomado de *Diseño y construcción de una red neuronal artificial de propósito general*.

Por consiguiente para Vega (Ob.Cit) “El conocimiento en una Red Neuronal Artificial RNA, está distribuido a lo largo de todo el sistema, debido a esto, se utilizan muchas interconexiones para obtener la solución de un problema en particular” (p.17), es decir, a consideración del autor para que una red neuronal artificial tenga éxito se requiere de un sistema muy completo, el cual deberá estar configurado y conformado por variadas conexiones interna que puedan generar la respuesta que se busca en función de un determinado problema.

Neurona Biológica

Para López (2007) “Una neurona es una célula que tiene dendritas que la conectan a las fuentes de entrada y a otras neuronas, recibiendo estímulo de múltiples fuentes a través de las dendritas y se encuentra estructurada de cuatro partes” (p.63)

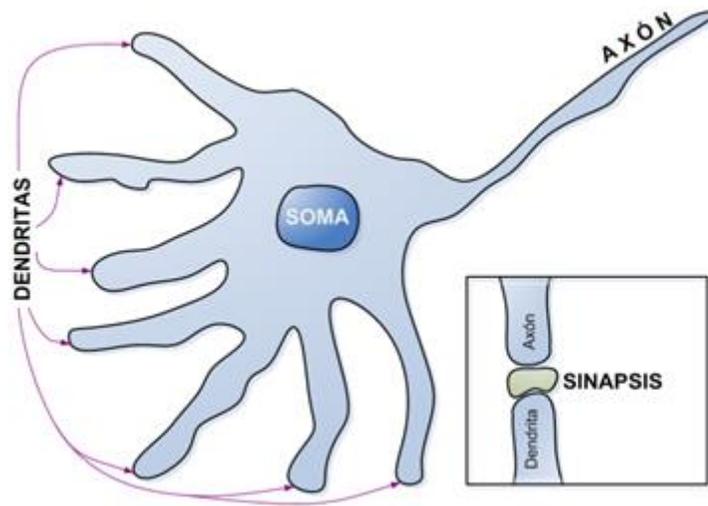


Figura 2. Neurona biológica. Tomado de Diseño y construcción de una red neuronal artificial de propósito general

Neurona Artificial (NA)

Una neurona artificial es un nodo que tiene una o más entradas y una sola salida. Cada entrada tiene un peso asociado con ella. Mediante la ponderación de entradas, podemos amplificar o des-amplificar una entrada.

Neurona Artificial



Figura 3. Neurona artificial. Tomado de Diseño y construcción de una red neuronal artificial de propósito general

Por consiguiente, las redes neuronales son elaboradas con la capacidad de imitar el funcionamiento del cerebro humano, ya que, su capacidad de aprender también pudiese ser imitada, en teoría.

Características de las Neuronas Artificiales

Las neuronas artificiales podrían poseer diferentes entradas relacionadas a propiedades distintas.

Las entradas se dividen en: excitadoras, inhibitorias, de ganancia, de disparo fortuito o de amortiguamiento.

Las entradas se ponderan, para ello se utiliza un factor multiplicativo de peso que le da la importancia a cada entrada.

En cada NA existe un valor de activación, este podría depender también de la entrada del valor anterior de activación.

Luego de calcularse el valor de activación, se establece un valor de salida empleando la función de salida que activa la NA.

Elementos básicos de una Neurona Artificial

Según Freeman, (1991), citado por Vega (2011) los elementos de una neurona son:

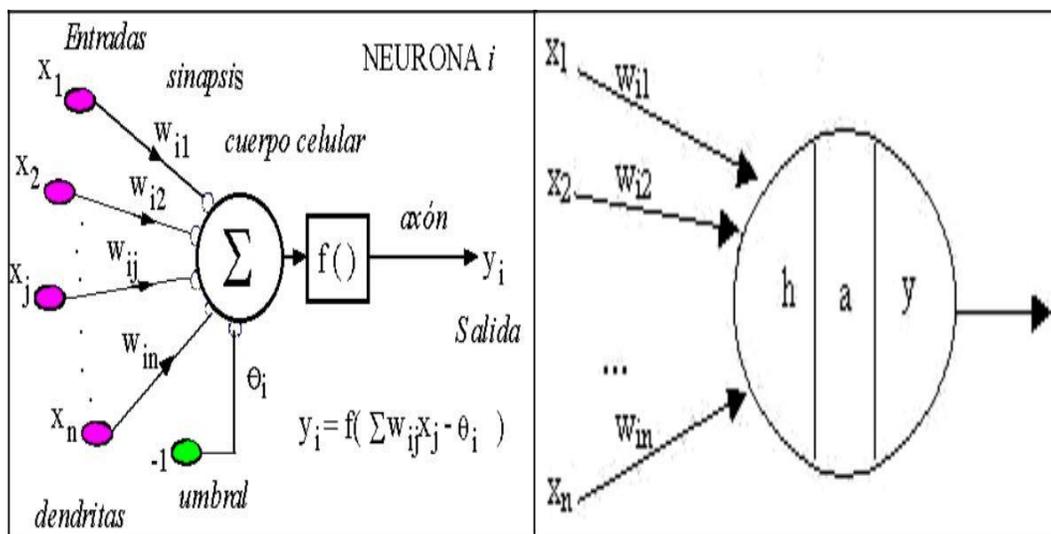


Figura 4. Representación de una neurona artificial Tomado de Reconocimiento de patrones mediante redes neuronales para la determinación del nivel de contenido calcio en la Zanahoria

- Conjunto de entradas o vector de entradas \mathbf{x} , de n componentes.
- Conjunto de pesos sinápticos w_{ij} , representan la interacción entre la neurona pre sináptica j y la post sináptica i.

- c. Regla de propagación $d(w_{ij}, x_j(t))$, proporciona el potencial post sináptico, $h_i(t)$.
- d. Función de activación $a_i(t) = f(a_i(t-1), h_i(t))$: proporciona el estado de activación de la neurona en función del estado anterior y del valor post sináptico.
- e. Función de salida $F_i(t)$: proporciona la salida $y_i(t)$, en función del estado de activación.

Las señales de entrada y salida pueden ser señales binarias (0,1 – neuronas de McCulloch y Pitts), bipolares (-1,1), números enteros o continuos, variables borrosas, entre otras.

La regla de propagación suele ser una suma ponderada del producto escalar del vector de entrada y el vector de pesos:

$$h_i(t) = \sum w_{ij} \cdot x_j$$

También se usa a menudo la distancia euclidiana entre ambos vectores:

$$h_i(t) = \sum (x_j - w_{ij})^2$$

La función de activación no suele tener en cuenta el estado anterior de la neurona, sino sólo el potencial $h_i(t)$. Suele ser una función determinista y casi siempre continua y monótona creciente. Las más comunes son la función signo (+1 si $h_i(t) > 0$, -1 en caso contrario), la función semilineal y las funciones sigmoides:

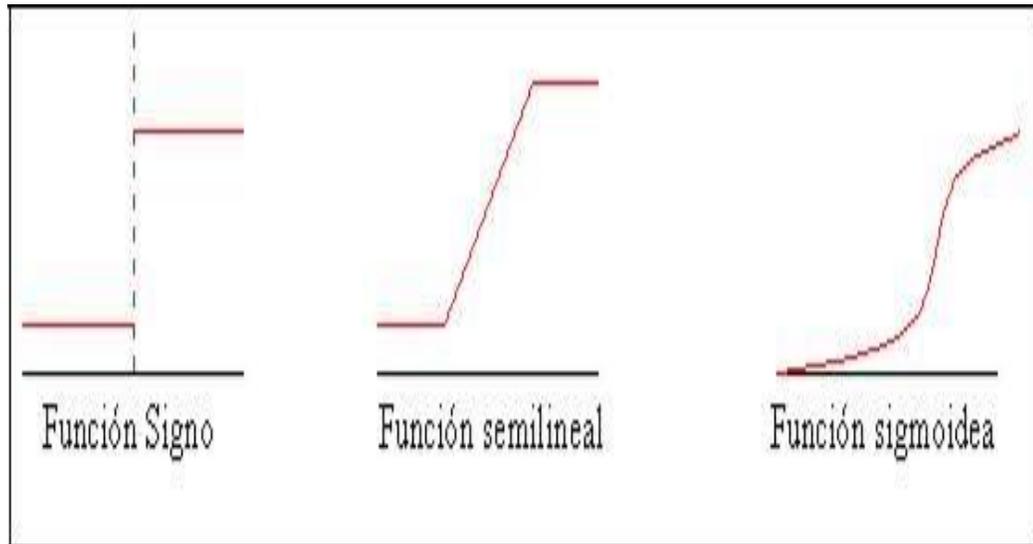


Figura 5. Representación de los tipos de funciones de activación 1. Tomado de Reconocimiento de patrones mediante redes neuronales para la determinación del nivel de contenido calcio en la Zanahoria.

La función de salida suele ser la identidad. En algunos casos es un valor umbral (la neurona no se activa hasta que su estado supera un determinado valor). Para el aprendizaje de la RNA se combinan modelos matemáticos de las células nerviosas y modelos de arquitecturas que describen las interconexiones que existen entre estas células. Modelo de funcionamiento de una neurona real:

Concebida como un sistema de entrada/salida:

$$y = f\left(\sum_{i=1}^n \omega_i x_i - \theta\right)$$

Actividad interna de la célula se define como:

$$a = \sum_{i=1}^n \omega_i X_i - \theta$$

De esta forma la variable 'y' de salida es la intensidad de la señal que emite la célula, que es el valor que toma en la actividad la función de transferencia 'f'. El umbral 'q' se interpreta como un peso sináptico 'w0' que se aplica a una entrada que vale permanentemente -1.

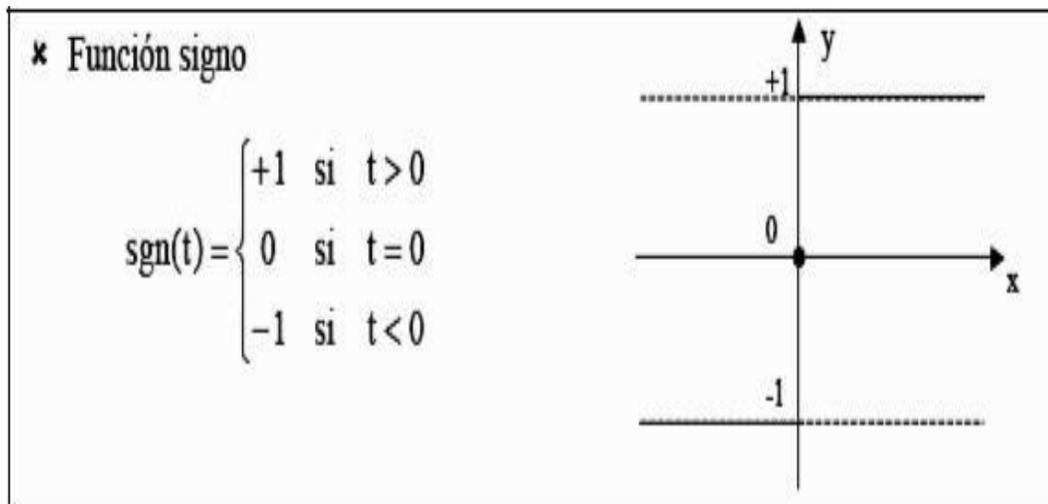


Figura 6. Representación de los tipos de funciones de activación 2. Tomado de Reconocimiento de patrones mediante redes neuronales para la determinación del nivel de contenido calcio en la Zanahoria

Las neuronas de la red se agrupan en capas, recibiendo la misma información que procesan de manera independiente.

Tipos de Neuronas Artificiales.

Las neuronas artificiales se pueden clasificar de acuerdo a los valores que pueden tomar. Por ahora es suficiente distinguir entre dos tipos principales:

- a- Neuronas binarias.
- b- Neuronas reales.

Las neuronas binarias solamente pueden tomar valores dentro del intervalo $\{0, 1\}$ o $\{-1, 1\}$, mientras que las neuronas reales pueden hacerlo dentro del rango $[0, 1]$ o $[-1, 1]$. Los pesos normalmente no están restringidos a un cierto intervalo, aunque para aplicaciones específicas puede ser esto necesario.

Perceptrón

Según López (2007) al referirse a las redes neuronales artificiales, se hace referencia a los Perceptrones Multicapa, este es un modelo de neurona única que fue precursor de redes las neuronales más grandes, quizás porque es el tipo más útil de red neuronal, es un campo de estudio que investiga cómo los modelos simples de cerebros biológicos pueden ser utilizados para resolver tareas computacionales difíciles, como ejemplo el modelado predictivo que se utiliza en el aprendizaje automático.

Pesos de la Neurona

Los pesos representan la memoria al largo plazo en las redes neuronales artificiales, para Rojas y otros (2011)

Cada neurona tiene un sesgo que se puede entender como una entrada que siempre tiene el valor 1-0 y se debe ponderar. Una neurona que tenga dos entradas requiere de tres pesos, uno para cada entrada y otro para el sesgo. Los pesos con frecuencia se inicializan a valores aleatorios pequeños, pero también se pueden usar esquemas de inicialización más complejos, pesos más grandes indican mayor complejidad y fragilidad del modelo. Es deseable mantener los pesos en la red y se pueden usar técnicas de regularización. (p.18)

Activación de la Neurona

Según Brownlee (2016) “las entradas ponderadas se suman y pasan a través de una función de activación, a veces llamada función de transferencia” (p.12). Una función de activación es un simple mapeo de la entrada ponderada sumada a la salida de la neurona. Se denomina función de activación porque regula el umbral en el que se activa la neurona y la intensidad de la señal de salida. Tradicionalmente, se usaron funciones de activación de pasos simples, donde si la entrada suma estaba por encima de un umbral, por ejemplo 0,5, entonces la neurona daría un valor de 1,0, de lo contrario daría un valor de 0,0.

Tradicionalmente se utilizan funciones de activación no lineales. Esto permite que la red combine las entradas de maneras más complejas y la vez proporcionan una capacidad más rica en las funciones que puede modelar. Funciones no lineales como la llamada sigmoide que emiten un valor entre 0 y 1 con una distribución en forma de s, la tangente hiperbólica llamada también Tanh que da salida a la misma distribución en el intervalo -1 a +1.

Más recientemente, se ha demostrado que la función de activación de rectificación proporciona mejores resultados.

Red Neuronal Artificial (RNA)

Una RNA está conformada por neuronas. Las RNA, como su nombre indican y se basan en algoritmos que intentan imitar la forma en que las neuronas funcionan en el cerebro, a través de ellas lo que se intenta no es crear modelos realistas del cerebro, sino desarrollar algoritmos y estructuras de datos que se puedan usar para modelar problemas muy complicados. Para Rojas y otros (2011) “Las neuronas se organizan en redes de neuronas. Una fila de neuronas se llama capa y una red puede tener múltiples capas. La arquitectura de las neuronas en la red se denomina a menudo topología de red, de acuerdo a ella, se conocen varios tipos de RNA” (p.21). La siguiente imagen es un ejemplo de red neuronal en su forma más simple, consta de una capa de entrada, otra oculta y la de salida.

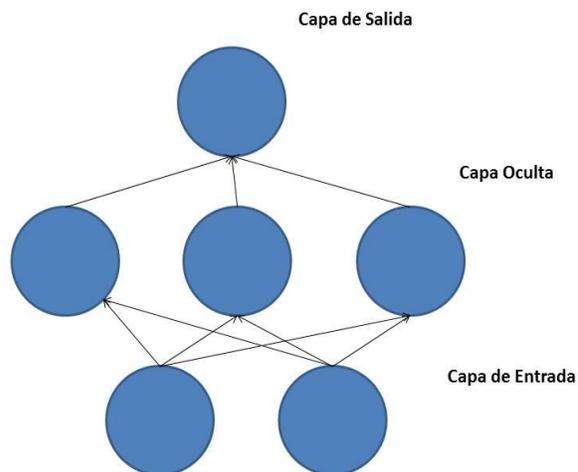


Figura 7, Red Neuronal Artificial Tomado de Implementación de una Red Neuronal Multicapa en un Microcontrolador PIC para la Identificación de Color

Elementos de una Red Neuronal Artificial

Los elementos básicos de una Red Neuronal Artificial según Martin (2007) citado por Vega (2011) son los siguientes:

Neuronas de entrada (capa de entrada).

Neuronas de salida (capa de salida).

Neuronas intermedias (capas ocultas)

Conjunto de conexiones o pesos sinápticos entre las neuronas

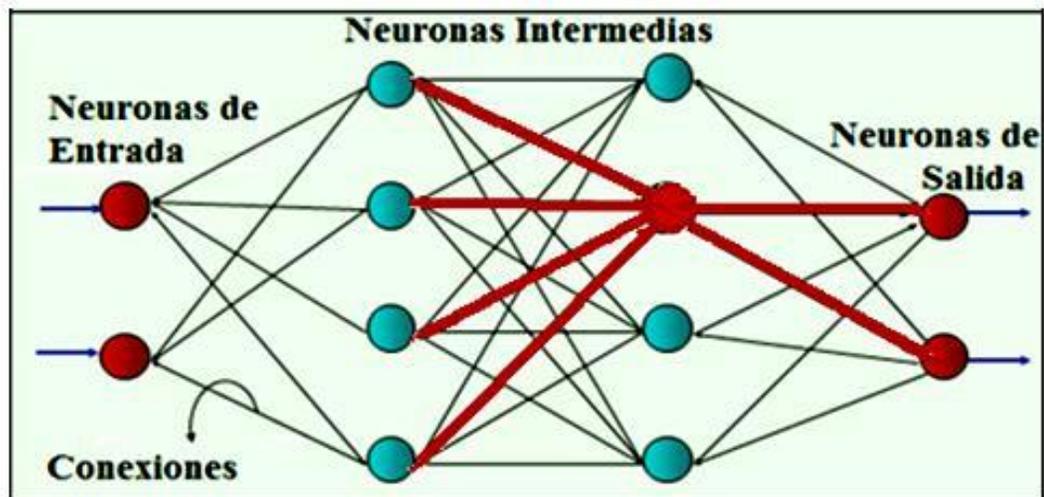


Figura 8. Elementos de una red Neuronal Artificial Tomado de *Reconocimiento de patrones mediante redes neuronales para la determinación del nivel de contenido calcio en la Zanahoria*

Capas de entrada o visibles

La capa inferior que toma la entrada de su conjunto de datos se denomina capa visible, ya que es la parte expuesta de la red. Con frecuencia una red neuronal se estructura con una capa visible con una neurona por

valor de entrada o columna en su conjunto de datos. Estas no son realmente neuronas, solo pasan el valor de entrada a la siguiente capa.

Capas ocultas

Son las capas que se encuentran después de la capa de entrada, ocultas porque no están directamente expuestas a la entrada. Debido a los aumentos en el poder de cálculo y en las eficientes bibliotecas de funciones, se pueden construir redes neuronales muy profundas, es decir, con muchas capas. La técnica del aprendizaje profundo también puede referirse a tener muchas capas ocultas en su red neuronal. Se denominan profundas porque habrían sido muy lentas para entrenar con los recursos disponibles hace algunos años, ahora pueden tomar segundos o minutos para entrenar usando técnicas modernas y hardware actualizado.

Capa de salida

Se denomina Capa de salida a la última oculta y es responsable de emitir un valor o vector de valores que corresponden al formato requerido para el problema. La función de activación en la capa de salida está condicionada por el tipo de problema que se está modelando. Para comprenderlo mejor se puede ver como un problema de regresión puede tener una única neurona de salida y la neurona puede no tener ninguna función de activación.

En cambio, un problema de clasificación binaria con una neurona de salida, puede utilizar una función de activación sigmoide para generar un valor entre 0 y 1, esto representaría la probabilidad de predecir un valor.

Mediante umbrales, esto se puede convertir en valores de clase nítido para los estén hasta 0,5 y convertirse en ajustes para los valores por debajo de 0. Finalmente 1 sería el otro valor asignado para cualquier otro resultado.

Otro problema de clasificación es el de multiclases, con múltiples neuronas en la capa de salida, una para cada clase. Para estos casos, se podría utilizar otra función de activación que emita la probabilidad de predecir cada uno de los valores de clase. La salida con mayor probabilidad se puede seleccionar para producir un valor de clasificación de clase nítido.

Entrenamiento de las RNA

Existen muchos tipos de entrenamientos de redes neuronales, después de configurada, la RNA necesita ser entrenada en su conjunto de datos necesarios para el funcionamiento, para lo cual se debe tener en consideración los datos con los cuales se va a entrenar, por lo que es importante que exista una equidad en los daos, ya que no todos deberán ser 100% negativos ni 100% positivos, pues el resultado será el leído de manera directa por la red, por lo que se considera que se debe entrenar a la red de manera equitativa para que esta realice un buen resultado de la actividad.

Preparación de datos

El primer paso es preparar los datos para el entrenamiento de la red neuronal, estos deben ser numéricos, en caso de datos no reales se asignan valores para categorizarlos en caliente, en el mismo proceso.

Esta codificación en caliente se puede utilizar también para las variables de salida en problemas de clasificación con más de una clase. Se crearía

entonces un vector binario de una sola columna para comparar directamente con la salida de la neurona en la capa de salida de la red, que produciría un valor para cada clase.

Normalización

Las RNA requieren que la entrada sea escalada de una manera consistente. Se puede cambiar la escala al rango entre 0 y 1, este proceso se denomina normalización.

Otra forma de normalizar de modo que la distribución de cada columna tenga la media de cero y la desviación estándar de 1, es aplicar esta escala a los píxeles de imagen, así como los datos tales como palabras se pueden convertir en números enteros, por ejemplo, para el rango de frecuencia de la palabra en el conjunto de datos y otras técnicas de codificación.

Pendiente del Gradiente Estocástico

El algoritmo clásico para el entrenamiento de las redes neuronales se denomina descenso de gradiente estocástico, consiste en exponer una fila de datos a la red en un momento como entrada.

La red procesa la entrada activando las neuronas hacia arriba lo que va a producir finalmente un valor de salida. Esto se denomina paso adelante en la red y se realiza después de estar capacitada para hacer predicciones sobre nuevos datos.

La salida de la red se compara con la salida esperada y se calcula el error. Este error se propaga entonces a través de la red, una capa a la vez, y los pesos se actualizan según la cantidad que hayan contribuido al error. Este

algoritmo se llama Back Propagation (propagación hacia atrás). Este procedimiento se repite para todos los datos del entrenamiento.

Épocas

Una sola ronda de actualización de la red para todo el conjunto de datos de formación en el entrenamiento se llama época. La red puede ser entrenada para un número ilimitado de épocas, todo dependerá del tiempo y recursos asignados para dicho proceso.

Actualizaciones de Peso

Los pesos de la red se actualizan a partir de los errores calculados para cada ejemplo de formación, a esto se le denomina aprendizaje en línea. Los cambios pueden ser rápidos, esto puede ser anárquico en la red. Una alternativa, es guardar los errores en los ejemplos de formación y la red se actualiza al final, este proceso se denomina aprendizaje por lotes y con frecuencia es más estable. El número de ejemplos para la actualización se reduce debido al gran conjunto de datos y la eficiencia de las computadoras, pueden ser decenas o cientos de ejemplos.

Tasa de Aprendizaje

Se refiere al parámetro de configuración que controla la cantidad de pesos que se actualizan, también es llamado tamaño de paso y controla el paso o cambio realizado en los pesos de la red para un error dado. Con frecuencia se las tasas de aprendizaje usadas son pequeñas, como 0.1, 0.01,

0.001 o menos. Se utiliza una ecuación de actualización que puede completarse con términos de configuración adicionales.

Momentum

Es cuando las propiedades de la actualización de peso anterior se incorporan para lograr que estos sigan cambiando en la misma dirección sin importar lo pequeño que sea el margen de error.

Tasa de decaimiento de aprendizaje

El decaimiento de peso es un término adicional en la regla de actualización de peso que hace que los pesos disminuyan exponencialmente a cero, si no se programa otra actualización, en relación a ello, Colmejo (2011) señala “La tasa de decaimiento de aprendizaje se utiliza para disminuir la tasa de aprendizaje durante épocas, esto permite que la red haga cambios importantes en los pesos al principio, pero más pequeños en épocas futuras para la afinación” (p.10) de manera práctica sería:

$$w_i \leftarrow w_i - n \frac{\partial E}{\partial w_i}$$

Donde

$E(w)$: es la función de costo o error.

∂ : El Gradiente descendente modificará el valor de w en dirección del descenso más pronunciado en E .

n : es la tasa de aprendizaje y si es grande tendrá una modificación correspondientemente grande de los pesos w_i (en general, no debe ser

demasiado grande, de lo contrario superará el mínimo local en su función de coste).

Para limitar eficazmente el número de parámetros libres en el modelo y evitar el exceso de ajuste, se puede regularizar la función de coste. Una forma de hacerlo es introducir una prioridad gaussiana media cero sobre los pesos, equivalente a cambiar la función de error a:

$$E(w) = E(w) + \frac{\lambda}{2} w^2$$

En la práctica, esto penaliza los pesos grandes y limita efectivamente la libertad en su modelo. El parámetro λ de regularización determina la forma en que se intercambia el coste original E , con la penalización de grandes pesos.

Aplicando el gradiente descendiente a esta nueva función de costo o error se obtiene:

$$w_i \leftarrow w_i - n \frac{\partial E}{\partial w_i} - n\lambda w_i$$

El nuevo término $-n\lambda w_i$ hace que el peso se desintegre en proporción a su tamaño.

Predicción

Según Brownlee (2016) “luego de entrenada, la red neuronal se puede usar para hacer predicciones” (p.24). Puede hacer predicciones en los datos de prueba o validación para estimar la habilidad del modelo en datos no vistos. También se puede implementar de manera operacional para hacer predicciones continuamente. La topología de red y el conjunto final de pesos

es lo que se necesita guardar del modelo, que significa que ya ha sido entrenada.

Las predicciones se realizan proporcionando la entrada a la red y realizando un paso hacia delante que le permite generar una salida que puede utilizar como predicción.

Clasificación de las Redes Neuronales Artificiales (RNA)

Las Redes Neuronales Artificiales se pueden clasificar según su arquitectura, el tipo de aprendizaje y según sus aplicaciones.

Según su arquitectura, las neuronas de una RNA están distribuidas en niveles o capas que están unidas entre sí por conexiones llamadas sinapsis; dependiendo del sentido de sus conexiones pueden ser recurrentes y no recurrentes, las no recurrentes pueden clasificarse en:

Según su arquitectura

Redes Neuronales Monocapa.

Sólo tienen una capa de neuronas que proyectan las entradas a una capa de neuronas de salida, donde se realizan diferentes cálculos.

Entre las más conocidas con esta arquitectura se encuentran: el Monolayer Perceptrón y el ADALINE.

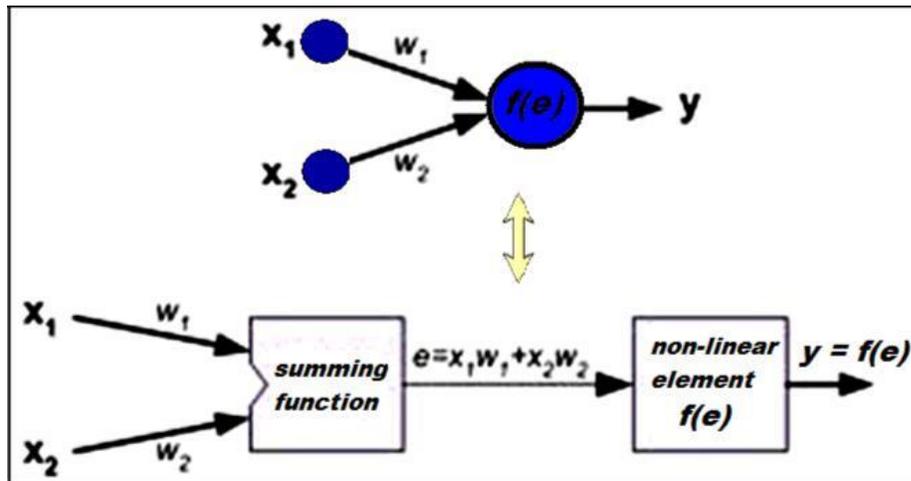


Figura 9. RNA con Conexiones hacia Adelante Mono-Capa Tomado de Reconocimiento de patrones mediante redes neuronales para la determinación del nivel de contenido calcio en la Zanahoria

Redes Neuronales Multicapa.

Es una generalización de la anterior existiendo un conjunto de capas intermedias entre la entrada y la salida.

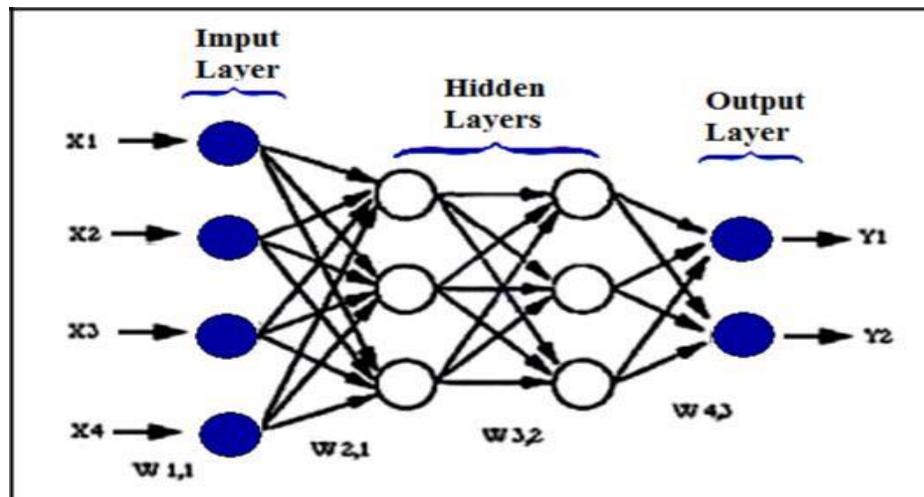


Figura 10. RNA con Conexiones hacia Adelante Multicapa. Tomado de Reconocimiento de patrones mediante redes neuronales para la determinación del nivel de contenido calcio en la Zanahoria.

Entre las RNA más conocidas con esta arquitectura se encuentran el Multilayer Perceptrón y el MADALINE.

Las recurrentes pueden clasificarse en:

Conexiones hacia Atrás (Recurrentes o Feedback)

Se caracterizan por la existencia de conexiones de retorno entre neuronas de una determinada capa con otras de capas anteriores, conexiones entre neuronas de una misma capa o conexiones de una neurona consigo misma.

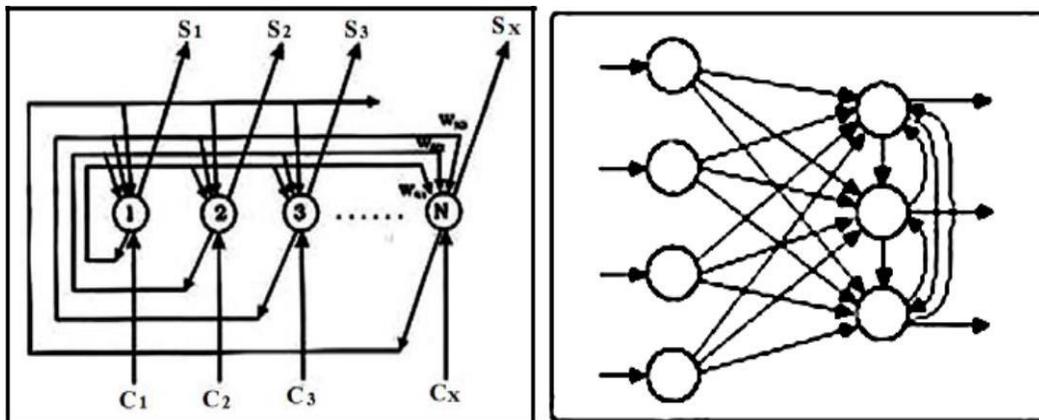


Figura 11. RNA con Conexiones hacia Atrás Tomado de Diseño y construcción de una red neuronal artificial de propósito general

Entre las más conocidas con esta arquitectura se encuentran las SOM de Kohonen, Hopfield y las redes ART.

Según su aprendizaje

Aprendizaje Supervisado (con Maestro)

Una RNA realiza un aprendizaje supervisado cuando tiene un supervisor en el entrenamiento de la red, significa que existe un parámetro de referencia ya conocido (maestro), el cual es comparado con las salidas de la red, el aprendizaje supervisado se caracteriza por el entrenamiento controlado de un agente externo que determina la respuesta que debería generar la red a partir de una entrada determinada. El supervisor comprueba la salida de la red y en el caso de que ésta no coincida con la deseada, se procederá a modificar los pesos de las conexiones, con el fin de conseguir que la salida obtenida se aproxime a la deseada. Un aprendizaje supervisado requiere disponer de un conjunto de patrones de los cuales se conoce su clase cierta (ver figura 11)

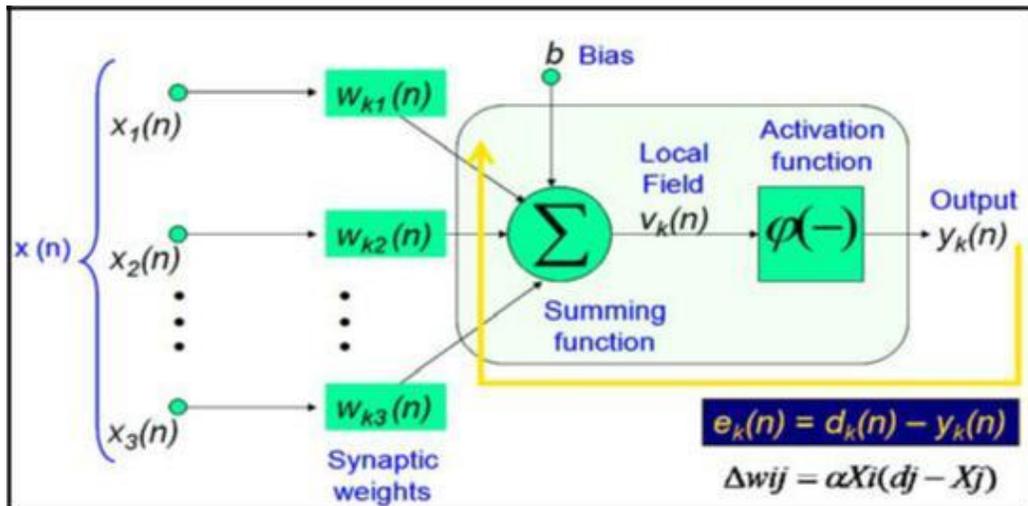


Figura 12. **Aprendizaje Supervisado de una RNA** Tomado de *Reconocimiento de patrones mediante redes neuronales para la determinación del nivel de contenido calcio en la Zanahoria*

Parámetros de una red neuronal

Según Vega (2011), se consideran ciertos parámetros para la creación de una RNA. A continuación, se mencionan las más importantes:

N: Número de neuronas en la capa de entrada.

Y: Número de neuronas en la capa de salida.

O: Número de neuronas en la capa oculta.

W_{ij}: Peso sináptico de la neurona i hacia la neurona j.

RNA Tipo Perceptrón Multicapa

Según Vega (2011), el Perceptrón intenta modelar el comportamiento de la neurona biológica, trabaja con funciones de activación, el cuerpo de la neurona es representado como un sumador lineal de estímulos Z_j , seguida por una función no lineal $f(Z_j)$ que es la función de activación, esta utiliza la suma de estímulos para determinar la salida de la neurona.

En el Perceptrón cada entrada es multiplicada por un factor de peso W correspondiente, los resultados son sumados y luego evaluados contra el valor de umbral, θ si el valor es mayor al máximo el Perceptrón se activa.

La primera capa actúa como un grupo de sensores que detectan una señal específica con la importancia correspondiente. La segunda capa toma salidas de los detectores y clasifica el patrón de entrada dado. El aprendizaje empieza ajustando la fuerza en las conexiones (pesos W_i) y el valor de umbral θ . La función g_i es fija y esta predeterminada.

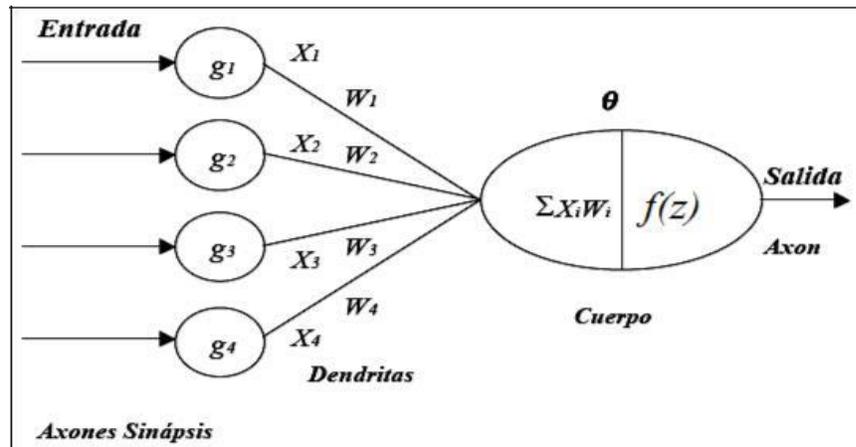


Figura 13. **Modelo de un Perceptrón** Tomado de *Reconocimiento de patrones mediante redes neuronales para la determinación del nivel de contenido calcio en la Zanahoria*

El entrenamiento de un Perceptrón se hace utilizando una función de aprendizaje. Si la respuesta dada por el Perceptrón es incorrecta, se modifican todos los pesos de las conexiones de acuerdo a la función.

$$\Delta w_i = \eta t_i x_i$$

Donde η es la ratio de aprendizaje, t_i es la salida deseada y x es la entrada del Perceptrón. La arquitectura típica de una RNA Perceptrón Multicapa está constituida por varias capas de nodos con interconexión completa entre ellos. El caso más sencillo en este tipo de red consiste en sólo 2 capas de neuronas, las de entrada y las de salida. De esta manera podemos obtener un modelo adecuado para problemas lineales del tipo de la regresión lineal múltiple.

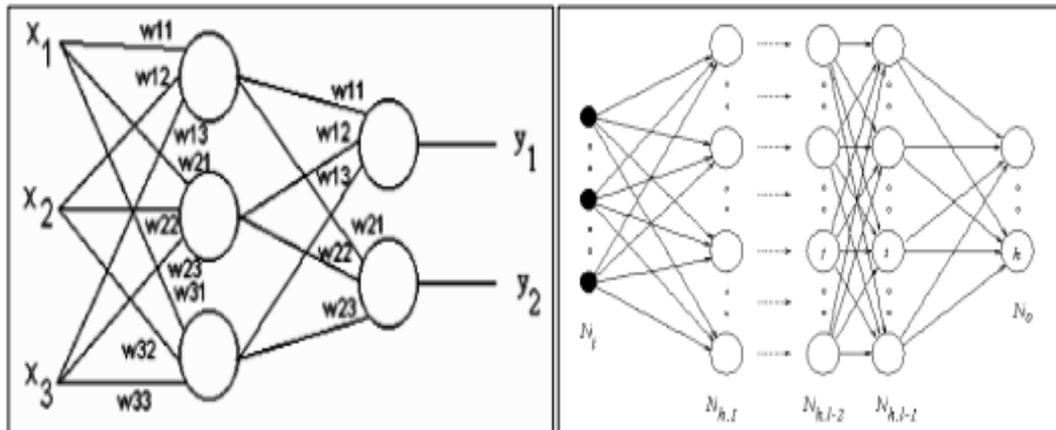


Figura 14. **Modelo de un Perceptrón Multicapa** Tomado de Diseño y construcción de una red neuronal artificial de propósito general.

En este tipo de red, una neurona recibe distintas entradas y activa una función de red (o regla de propagación) con unos pesos de entrada asociados. La computación de estos pesos se sigue de la aplicación de la función de activación que determina el nivel de activación de salida de la neurona.

Algoritmo Backpropagation

Flores (2006) señala: “Es un algoritmo de aprendizaje supervisado que se usa para entrenar RNA de arquitectura multicapa y con conexiones hacia adelante. El algoritmo consiste en minimizar un error (comúnmente cuadrático) por medio de gradiente descendiente, por lo que la parte esencial del algoritmo es el cálculo de las derivadas parciales de dicho error con respecto a los parámetros de la RNA”.

Este consta de los siguientes pasos:

Inicialización:

1. Construcción de la red.
2. Inicialización aleatoria de pesos y umbrales (-0.5, 0.5).
3. Criterio de terminación (número máximo de iteraciones,...).
4. Contador de iteraciones $n=0$.

Hacia adelante

5. Seleccionamos el primer par de entrenamiento.
6. Calcular la salida de la red para cada patrón de entrada.
7. Calcular el error total cometido (SSE).
8. Si la condición de terminación se satisface, parar. Hacia Atrás

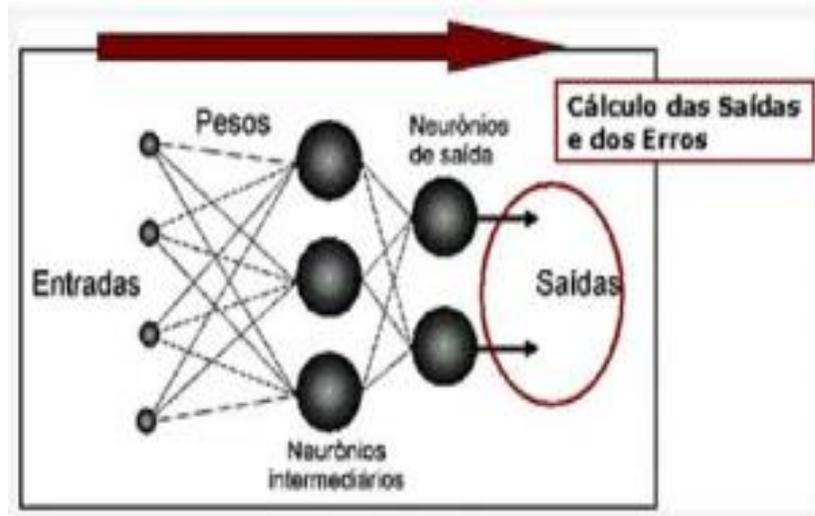


Figura 15. Backpropagation hacia Adelante tomado de Reconocimiento de patrones mediante redes neuronales para la determinación del nivel de contenido calcio en la Zanahoria

Hacia atrás

9. Incrementar el contador $n=n+1$.
10. Propaga los errores hacia la capa de entrada, ajustando los pesos de las capas ocultas de forma que se minimice el error.

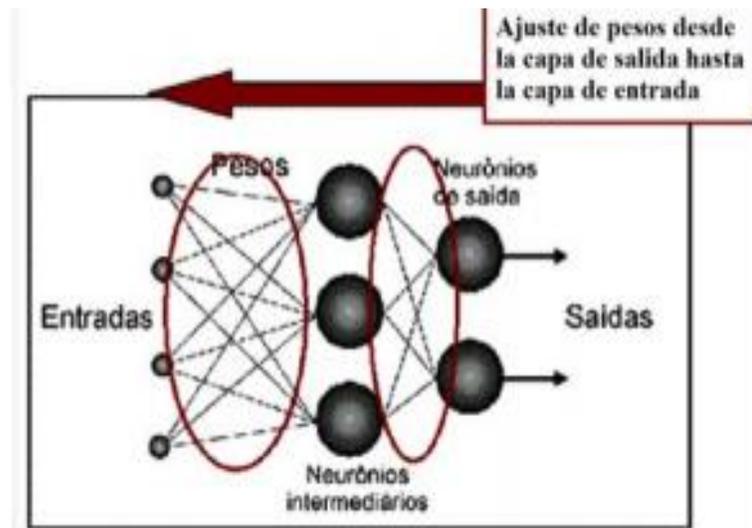


Figura 16. **Backpropagation hacia Atrás** Tomado de *Reconocimiento de patrones mediante redes neuronales para la determinación del nivel de contenido calcio en la Zanahoria.*

- Para cada neurona de salida calcular

$$\delta_k = (o_k - y_k) f'(net_k) \text{ donde } net_j = \sum_i w_{ij} x_i + b_j$$

- Para cada unidad oculta calcular

$$\delta_j = f'(net_j) \sum_k \delta_k w_{jk}$$

- Actualizar los pesos

$$\Delta w_{ij}(n+1) = \eta \delta_j o_i + \alpha \Delta w_{ij}(n)$$

11. Repetimos desde el paso 5 para cada par de entrenamiento hasta que el error para todos los conjuntos de entrenamiento sea aceptable.

Máquina de Aprendizaje (Machine Learning)

Según Brownlee (2016) la máquina de aprendizaje (MA) “es el desarrollo de técnicas que permitan a las aplicaciones aprender sin necesidad de programarlas explícitamente para resolver un problema, esta logra que una computadora pueda trabajar con más precisión a medida que recopila y aprende de los datos que se le dan” (p.40).

Enfoques para definir el aprendizaje de la máquina

Para resolver un problema se recopilan datos y se estipula el rendimiento esperado. Como primer paso el problema se enmarca de manera que la máquina pueda entenderlo y que los resultados tengan significado para un ser humano, de acuerdo a esto, se presentan seis formas de afrontar el problema:

- Exploratorio: Se analizan los datos y se busca patrones tales como una tendencia o relación entre las variables, esta exploración a menudo conduce a hipótesis, ejemplo de ello puede ser la vinculación de la dieta con la enfermedad, o la tasa de criminalidad entre las viviendas urbanas.

- **Descriptivo:** Intenta resumir las características específicas de los datos, un ejemplo es la esperanza de vida media, la temperatura media o los zurdos en una población.

- **Inferencial:** Trata de apoyar una hipótesis, es decir, demostrar o refutar una proposición, por ejemplo, un vínculo general entre la esperanza de vida y los ingresos económicos usando diferentes conjuntos de datos.

- **Predictivo:** Se trata de anticipar el comportamiento futuro, como ejemplo, la predicción de la esperanza de vida mediante el análisis de los ingresos económicos.

- **Casual:** Se intenta de averiguar qué causa algo, como podría ser ¿Los bajos ingresos causan una esperanza de vida más baja?

- **Mecanismo:** Trata de responder las preguntas que vinculan los mecanismos con los resultados.

En gran medida, los problemas de aprendizaje de la máquina necesitan varios de estos tipos de preguntas durante su desarrollo, se puede explorar los datos buscando patrones o tendencias y luego describir ciertas características clave de los datos. Esto puede permitir hacer una predicción, y encontrar una causa o un mecanismo detrás de un problema particular.

Clasificación según el estilo de Aprendizaje

Aprendizaje supervisado: Con el aprendizaje supervisado, el modelo es entrenado con datos y valores de característica de entrada a los valores de salida correctos. Los algoritmos de aprendizaje supervisado se dividen en dos tipos: corrección de error y gradiente.

Corrección de error: el cual cambia los pesos de la NA luego de la manifestación de un patrón para corregir el error de la salida.

Gradiente: Altera los pesos de la NA luego de la presentación de un patrón para minimizar el error cuadrático medio sobre todos los patrones. Para esto se modifica los pesos en dirección opuesta al gradiente de la función de error.

Aprendizaje no supervisado: En el aprendizaje sin supervisión, los datos no contienen resultados, pero se espera que el modelo determine relaciones por sí mismo.

Semi-supervisado: Esta técnica usa una pequeña cantidad de datos etiquetados que contienen la respuesta con una mayor cantidad de datos no etiquetados. La combinación puede conducir a resultados.

Aprendizaje de refuerzo: Es similar al aprendizaje supervisado pero se proporciona una recompensa para obtener buenos resultados.

Aprendizaje profundo: este enfoque modela abstracciones de alto nivel usando un gráfico que contiene múltiples niveles de procesamiento.

Dentro de las técnicas que utilizan el aprendizaje supervisado se encuentran:

Árboles de decisión: Se construye un árbol utilizando las características del problema como nodos internos y los resultados.

Soporte de máquinas de vectores: Generalmente se utiliza para la clasificación mediante la creación de un hiperplano que separa el conjunto de datos y luego hace predicciones.

Redes Bayesianas: Modelos utilizados para representar relaciones probabilísticas entre eventos dentro de un ambiente

Aprendizaje reforzado (Reinforcement Learning)

Según Brownlee (2016) el aprendizaje reforzado es un tipo de aprendizaje en la vanguardia de la investigación actual en las redes neuronales y el aprendizaje automático. A diferencia del aprendizaje no supervisado y supervisado, el aprendizaje reforzado toma decisiones basadas en los resultados de una acción. Se trata de un proceso de aprendizaje orientado a las metas, similar al que utilizan muchos padres y maestros de todo el mundo.

Saturación en la Red Neuronal Artificial

A través del proceso de aprendizaje de una RNA, los pesos neuronales se afirman en su tejido dentro de la red, estos están ajustados para cumplir con las características específicas que le dan una especialización, las neuronas colindantes confían en esto y si el alcance va demasiado lejos puede resultar en un modelo frágil demasiado especializado para los datos de entrenamiento.

Esta dependencia del contexto de una NA durante el entrenamiento se conoce como co-adaptaciones complejas.

Regularización de Abandono

Según Brownlee (2016) “la deserción es una técnica en la que las neuronas seleccionadas al azar son ignoradas durante el entrenamiento. Se abandonan aleatoriamente” (p.19).

Esto significa que su contribución a la activación de otras neuronas hacia adelante se elimina temporalmente y no se aplican las actualizaciones de peso a las neuronas en el paso hacia atrás, esta técnica se denomina regularización de abandono y puede contribuir a subsanar el problema de la saturación.

Si las NA se abandonan o excluyen aleatoriamente de la red a través del entrenamiento, otras neuronas asumirán la representación necesaria para realizar las predicciones de las neuronas desaparecidas y se estima que esto resulta en múltiples representaciones internas independientes aprendidas por la red.

El resultado es que la RNA se torna menos sensible a los pesos específicos de las neuronas, dando como resultado que la RNA esté capacitada para una mejor generalización, siendo menos probable que se sobrecargue con los datos de entrenamiento y así evitar la saturación.

Topologías de redes neuronales

De manera muy resumida se puede decir que las RNA se pueden dividir en dos grandes grupos de acuerdo a su conexión, que a saber son:

Redes de propagación hacia adelante

La característica principal de este modelo es que la suma ponderada de las señales de entrada es comparada con un umbral para determinar la salida de la neurona. Si la suma es mayor o igual al umbral, la salida es igual a 1, de lo contrario la salida es 0.

La regla de aprendizaje simple confluirá a los pesos correctos de la red, la regla de aprendizaje es simple y en forma automática la red aprende de sus errores.

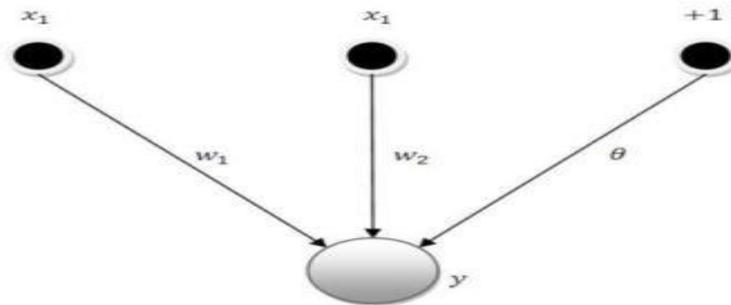


Figura 17. Red con propagación hacia adelante. Tomado de Redes Neuronales Artificiales aplicadas a la estadística y análisis de fundamentos

Redes Recurrentes

Para Colmejo (2011) "Son redes con una arquitectura que implementan memoria, para ello algunas neuronas reciben como entrada la salida de una de las capas e inyectan su salida en una de las capas de un nivel anterior a ella" (p.32), es decir, son redes que presentan un diseño con memoria propia

que actúan de acuerdo a como fueron entrenadas para dar respuesta de una problemática en general, por consiguiente cuando entra una información esta es recibida por la salida de una información en un espacio de más bajo nivel al de la entrada

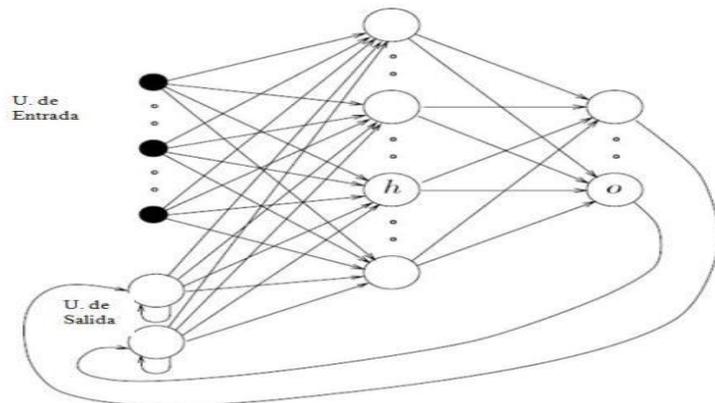


Figura 18. **Red Recurrente.** Tomado de *Redes Neuronales Artificiales aplicadas a la estadística y análisis de fundamentos*

Aprendizaje Profundo (Deep Learning) y Redes Neuronales Profundas (DNN)

Según López (2007) “se entiende por aprendizaje profundo o redes neuronales profundas, a las que poseen al menos dos o más capas oculta” (p.33), en relación a ello es importante señalar que agregar capas añade niveles de complejidad que pueden ser mucho más difíciles de controlar y requiere de muchas más neuronas para funcionar, por lo que es vital conocer la cantidad de capas que presenta una red.

Además, en el aprendizaje profundo, la red no aprende solo a predecir una salida Y dada una entrada X, también entiende las características básicas de la entrada, la red neural es capaz de hacer abstracciones de las características que comprenden los ejemplos de entrada, entender las características básicas de los ejemplos y hacer predicciones basadas en esas características. En el aprendizaje profundo, hay un nivel de abstracción que falta en otros algoritmos básicos de aprendizaje de máquina o en redes neurales poco profundas.

Las ventajas de aprendizaje profundo con respecto a otros algoritmos de aprendizaje de máquina son:

- El aprendizaje profundo puede aprender representaciones.
- El aprendizaje profundo es menos sensible al ruido.
- El aprendizaje profundo puede ser un algoritmo generativo.

Los algoritmos generativos son una forma de modelar por medio del cómputo el comportamiento orgánico de la naturaleza.

Se puede definir más generalmente el aprendizaje profundo como una clase de técnicas de aprendizaje automático en las que la información se procesa en capas jerárquicas, para comprender representaciones y características de los datos en niveles crecientes de complejidad. En estos algoritmos se pueden incluir:

Perceptrones Multi-capas: red neuronal con muchas capas ocultas, con propagación hacia adelante.

Máquinas Boltzmann: es una red estocástica simétrica con una función bien definida.

Máquinas Boltzmann Restringidas: están compuestas de conexiones simétricas entre dos capas, una visible y otra oculta, pero a diferencia de las máquinas generales de Boltzmann, las neuronas no tienen conexiones intra-capas y pueden ser juntados para formar Redes profundas de creencia.

Redes profundas de creencia: es un modelo generativo estocástico en el que las capas superiores tienen conexiones simétricas entre ellas (no dirigidas, a diferencia de las redes de avance), mientras que las capas inferiores reciben la información procesada desde las conexiones dirigidas desde las capas superiores.

Autoencoders: Una clase de algoritmos de aprendizaje no supervisados en los que la forma de salida es la misma que la entrada, que permite a la red aprender mejor las representaciones básicas.

Redes Neuronales Convolucionales (RNC): Las capas convolucionales aplican filtros a la imagen de entrada (o sonido) deslizando este filtro a través de la señal entrante para producir un mapa de activación bidimensional. Las RNC permiten la mejora de las funciones ocultas en la entrada.

Cada una de estas implementaciones de aprendizaje profundo tiene sus propias ventajas y desventajas, y puede ser más fácil o más difícil de entrenar dependiendo del número de capas y neuronas de cada capa.

GPU contra CPU

El aprendizaje profundo es muy popular debido en gran proporción a la capacidad de procesamiento de las GPU (Unidades de Procesamiento Gráfico). Visto desde la arquitectura de la computadora, la CPU (Central Processing Unit) se compone de unos cuantos núcleos que pueden manejar unos cuantos hilos a la vez, en cambio las GPU están formadas de cientos de núcleos que pueden manejar miles de hilos al mismo tiempo, a su vez la GPU es altamente paralelizable, en comparación con la CPU que es principalmente una unidad serial.

Factores que Influyen en el Rendimiento del Aprendizaje de la Red

Efecto del número de muestras de aprendizaje.

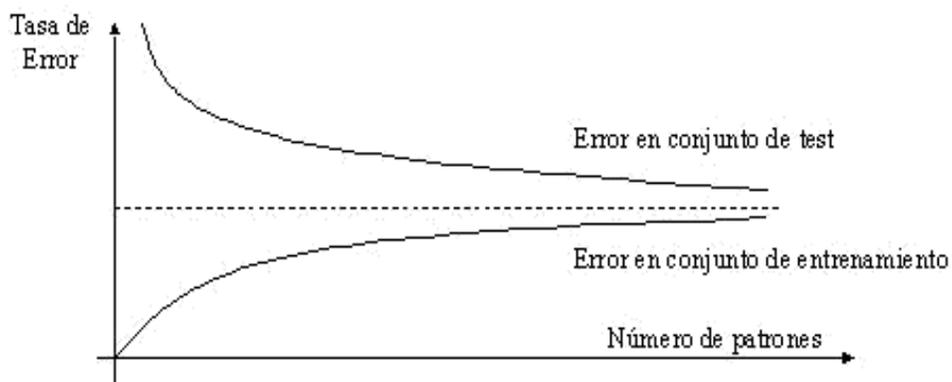


Figura 19: Efecto del Número de Muestras de Aprendizaje Tomado de Reconocimiento de patrones mediante redes neuronales para la determinación del nivel de contenido calcio en la Zanahoria.

Efecto del número de unidades ocultas.

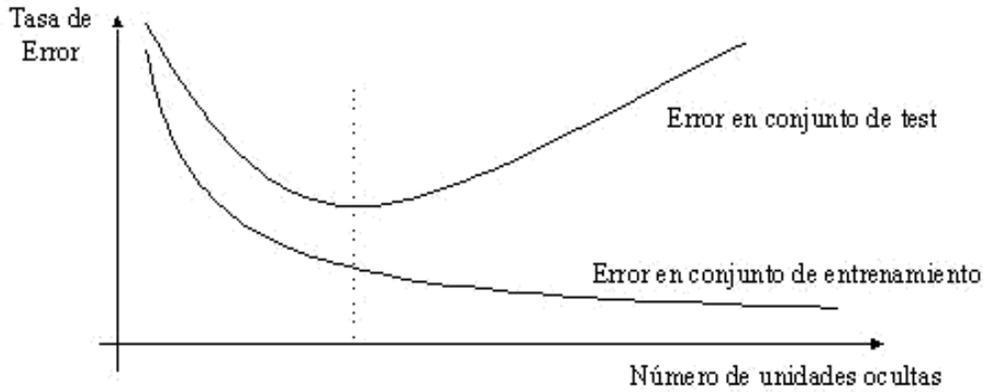


Figura 20. Efecto del Número de Unidades Ocultas Tomado de Reconocimiento de patrones mediante redes neuronales para la determinación del nivel de contenido calcio en la Zanahoria

Efecto del número de iteraciones.

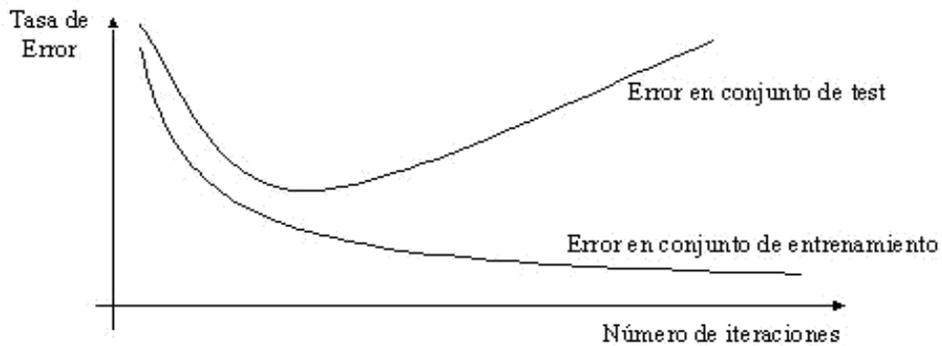


Figura 21. Efecto del Número de Iteraciones Tomado de Reconocimiento de patrones mediante redes neuronales para la determinación del nivel de contenido calcio en la Zanahoria.

Efectos sobre-entrenamiento

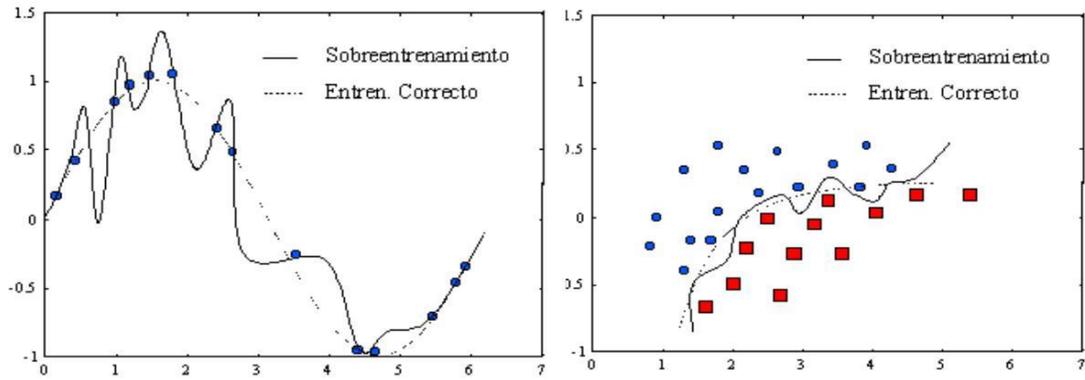


Figura 22. **Efecto de Sobre-Entrenamiento.** Tomado de Diseño y construcción de una red neuronal artificial de propósito general

Términos Básicos

Aprendizaje Profundo (Deep Learning): es un conjunto de técnicas y algoritmos de aprendizaje automático que procesan datos para comprender representaciones y características de alta complejidad como abstracciones.

Data Science: en español Ciencia de datos, es una interdisciplina que comprende métodos científicos, procesos y sistemas para obtener conocimiento de los datos en sus diferentes formas, para ello se vale de campos de análisis de datos como la estadística, minería de datos, aprendizaje automático y analítica predictiva, entre otros.

Máquina de Aprendizaje (Machine Learning): se refiere a la creación y uso de modelos que aprenden de los datos.

Neurona Artificial: es la unidad básica de la red neuronal artificial, consta de una o más entradas y una sola salida, controlada por un peso asociado a ella y una función que activa dicha salida.

Neurona Biológica: es una célula nerviosa que posee la capacidad de excitarse y de propagar el impulso nervioso a otra neurona.

Python: es un lenguaje de programación creado por Guido van Rossum a principios de los años 90 con una sintaxis muy limpia y que favorece un código legible, es interpretado o de script, con tipado dinámico, multiplataforma y orientado a objetos.

Redes Neuronales Artificiales: Son estructuras algorítmicas que intentan imitar la forma en que las neuronas funcionan en el cerebro para resolver y modelar problemas que presentan un alto grado de dificultad en el área de la computación.

CAPÍTULO III

MARCO METODOLÓGICO

La fundación metodológica, mejor conocido como el marco metodológico, es el camino científico que sigue el investigador para desarrollar la metodología que empleará en la investigación, en relación a ello Hurtado y Toro (2010) señalan “el marco metodológico está definido como la esencia de la investigación, se refiere a como se ha desarrollado propiamente el trabajo investigativo indicando que métodos son usados para lograr los objetivos propuestos” (p. 90), es decir, es la metodología que debe seguir el investigador para llevar a cabo el trabajo de manera correcta y confiable, teniendo en consideración el tipo de diseño en el cual se trabajara.

Tipo de Investigación

La presente investigación busca la descripción del impacto del uso de un sistema basado en redes neuronales en los EPP, de allí que se señala que el presente trabajo se enmarca en una investigación de tipo descriptiva que para Sabino (2012) es aquella que “describe las características fundamentales de conjuntos homogéneos de fenómenos, utilizando criterios sistemáticos que permitan poner de manifiesto su estructura o comportamiento. De esta forma se pueden obtener las notas que caracterizan a la realidad estudiada” (p.51).

Por tanto, el autor señala que por medio de este tipo de investigación se busca describir las características fundamentales del fenómeno a objeto de estudio, que permitan hacer ver su comportamiento desde cualquier punto de vista, por medio de este tipo de investigación se deja en evidencia todos

y cada uno de los elementos y factores que inciden en la problemática para ser estudiados y analizados en su propia realidad y así poder presentar soluciones eficaces que conlleven a la solución de fenómeno a objeto de estudio.

Modalidad de Estudio

La presente investigación se enmarca en los lineamientos de proyecto factible, dado a que se propone como viable el desarrollo de técnicas de redes neuronales para la evaluación de los programas de postgrado de Ingeniería, el cual es definido por la Universidad Experimental Libertador UPEL (2014) como aquel que:

Consiste en la investigación, elaboración y desarrollo de una propuesta de un modelo operativo viable para solucionar problemas, requerimientos o necesidades de la organización o grupos sociales. El Proyecto debe tener apoyo en una investigación de tipo documental, de campo o un diseño que incluya ambas modalidades (p. 16).

En relación a lo planteado por el autor, este diseño de investigación propone la elaboración de planes o propuestas de modelos que ayuden en la operatividad eficaz de los diferentes procesos de a las organizaciones empresariales, es por ello que de acuerdo con las características específicas que se plantean para la elaboración de la investigación, se determina que este trabajo se presenta para dar solución eficaz, pronta y oportuna a un problema de tipo práctico que se viene presentando en el área de Postgrado de Ingeniería de la Universidad de Carabobo.

Diseño de Investigación

En relación al diseño de la investigación, se señala que el presente trabajo será estudiado en el entorno del área de posgrado de la Facultad de Ingeniería de la Universidad de Carabobo, por lo que se puede señalar que es un trabajo con diseño de campo, en relación a ello Sabino (2012), afirma “Los diseños de campo son los que se refieren a los métodos a emplear cuando los datos de interés se recogen directamente de la realidad, mediante el trabajo directo y concreto del investigador y su equipo” (p. 64).

En relación a lo señalado por el autor se puede decir que una investigación con diseño de campo es aquella que se lleva a cabo dentro del contexto donde ocurre el fenómeno a objeto de estudio, es decir la problemática, por lo que el investigador no puede manipular las variables que se presenten durante el desarrollo del trabajo, en ese sentido es importante resaltar que dicho trabajo de investigación se estará desarrollando dentro de la Universidad de Carabobo, específicamente en el área de posgrado de la Facultad de ingeniería lugar en donde se propone la implementación del programa de redes neuronales, el cual tiene como objetivo proponer un sistema computacional de redes neuronales para la evaluación de los programas de postgrado de la Facultad de Ingeniería de la Universidad de Carabobo.

Fases del Proyecto

Diagnóstico

La fase de diagnóstico de la presente investigación será llevada a cabo

mediante el empleo de técnicas e instrumentos de recolección de información, entre los que se emplearan la técnica de la observación directa y la entrevista, los instrumentos de la observación directa será el block de notas y el diario de campos en los cuales se llevara a cabo las anotaciones de lo observado por el investigador en el área de posgrado de la universidad de Carabobo, en relación al tiempo que emplean los profesores en cuanto a la corrección de las encuestas de los estudiantes.

El instrumento empleado en la aplicación de la entrevista será un cuestionario de preguntas con respuestas dicotómicas “si o no”, ello con la finalidad de hacer una serie de interrogantes a los profesores encargados del departamento para conocer de primera fuente información relacionada con la problemática a objeto de estudio y de allí poder realizar el estudio de factibilidad para conocer cómo se aplicara el diseño que se propone.

Estudio de factibilidad

Para llevar a cabo dicha fase, se aplicó un análisis cuantitativo de los resultados arrojados por la aplicación de técnicas de recolección de datos, con los mismos se procedió al levantamiento de cuadros y gráficos que serán acompañados de sus respectivos análisis todos en función del sistema computacional de redes neuronales propuesto para la solución de la problemática.

Diseño

La propuesta de un sistema computacional de redes neuronales para la evaluación de los programas de posgrado de la Facultad de Ingeniería de la Universidad de Carabobo, se llevó a cabo con el empleo de la tecnología

computacional, para lo cual se empleó una serie de técnicas, herramientas y estrategias que se fusionaron para el diseño de la red en función del desarrollo del sistema

Población Y Muestra

Población

Según Tamayo y Tamayo, (2010), “La población se define como la totalidad del fenómeno a estudiar donde las unidades de población posee una característica común la cual se estudia y da origen a los datos de la investigación” (p.114). La población de la investigación será la Coordinación de los programas de Postgrado de Ingeniería, la cual está conformada por los docentes que laboran en dicha área (ver cuadro 1)

Cuadro 1

Personal que labora en el área de Posgrado de la Facultad de Ingeniería de la Universidad de Carabobo

Personal del área de Posgrado	Población
Docentes	15

Nota: Fuente propia

Muestra

Según Tamayo y Tamayo (2010), afirma que la muestra “es el grupo de individuos que se toma de la población, para estudiar un fenómeno estadístico” (p.38). Dado de que la población es pequeña se tomará toda la población como objeto para el estudio, en tal sentido la muestra se denominó

muestreo censal, lo que quiere decir que la muestra estuvo conformada por un total de 15 docentes

Técnicas de Recolección de Datos

De acuerdo con Arias (2012), “Se entiende por técnica, el procedimiento o forma de recoger los datos” (p.68), es decir, el procedimiento por medio del cual el investigador puede recolectar la información de una determinada problemática por consiguiente en la presente investigación, se utilizarán las siguientes técnicas e instrumentos:

a. Observación Directa: Para diagnosticar la situación del sistema de redes neuronales para la evaluación de los programas de postgrado de Ingeniería y ver los procedimientos que se llevan a cabo manualmente, se empleara en primera instancia la observación directa, definida según Méndez (2009) como “el proceso mediante el cual se perciben deliberadamente ciertos rasgos existentes en la realidad por medio de un esquema conceptual previo y con base en ciertos propósitos definidos, generalmente por una conjetura que se requiere investigar” (p.25).

b. La entrevista no estructurada según Arias (2012), “no dispone de una guía de preguntas elaboradas previamente” (p.73), es decir, es un instrumento que sirve para conocer información de los entrevistados relacionada con el problema a estudiar mediante una conversación que se enmarca en la problemática a objeto de estudio.

Instrumentos de Recolección de Datos

En relación a los instrumentos para la recolección se utilizarán:

- a. Diario de campo o block de notas, con la finalidad de emplearlo para registrar lo observado mediante diagramas de flujos de datos, algoritmos, estructuras de bases de datos, cuestionarios y pruebas.
- b. Grabador de voz que será empleado para grabar para las diferentes entrevistas a que diera lugar la investigación.

Técnicas de Análisis de Datos

Según Méndez (2011), “Las técnicas de análisis se basan en descomponer un todo en sus partes constitutivas para profundizar el conocimiento de cada una de ellas” (p. 118), en tal sentido el análisis de los datos arrojados en la fase de diagnóstico se llevará a cabo, mediante la aplicación de un análisis cualitativo, a través de un análisis de contenido.

CAPÍTULO IV

RESULTADOS

Para el reporte de los resultados de la investigación se procedió en primera instancia a la recolección de los datos mediante el empleo de la observación directa y la entrevista no estructurada, dichas técnicas arrojaron resultados importantes para el desarrollo de la investigación y la propuesta de un sistema computacional de redes neuronales para la evaluación de los programas de postgrado de la Facultad de Ingeniería de la Universidad de Carabobo.

En la aplicación de la observación directa se procedió en primera instancia a conocer como era el procedimiento que se ejecutaban en el departamento de Posgrado en relación a la evaluación de los diferentes programas existentes en dicha área, para ello el investigador se valió de instrumentos para la recolección de la información necesaria.

En relación a la entrevista, se procedió a la realización de una entrevista con el personal involucrado en el área de posgrado de la Facultad de ingeniería de la Universidad de Carabobo, para ello el investigador se valió de un cuestionario de preguntas abiertas relacionadas con la problemática en la cual se buscó tener una conversación fluida a través de la cual los entrevistados dieron sus opiniones para luego someter la información a un análisis de contenido.

Teniendo en consideración la antes señalado se abordó la entrevista con la temática de los sistemas computacionales como dispositivos de interacción en función de solucionar problemas mediante la aplicación de la

inteligencia artificial, para lo cual la mayoría de los entrevistados señalaron que mediante la aplicación de dichos sistemas muchas son las aplicaciones que se pueden llevar a cabo en función de ayudar al hombre para la solución de eventualidades que requieren de tiempo y mediante este sistema el tiempo se puede minimizar al máximo y así las personas podrán ocuparse en otras actividades.

De igual forma se indagó sobre los procesos de instrucciones existentes en un proceso los cuales ayudan a minimizar el tiempo de quienes lo van a aplicar, facilitan la realización del mismo y minimizan el tiempo en realizar las actividades para lo cual se implementarán las instrucciones, así mismo los entrevistados señalaron que los sistemas computacionales son una ayuda para la producción de información de manera más clara, rápida y sencilla, pues dichos sistemas son elaborados en función de dar una respuesta precisa.

Otra de las temáticas abordadas guardo relación con las redes neuronales como modelo computacional, el cual fue considerado por los entrevistados como un sistema idóneo para el procesamiento de información, ya que aplicando dicho modelo previamente adiestrado para un fin, el mismo puede generar información necesaria e importante para la solución de un problema para el cual fue diseñado, en tal sentido es importante señalar que una red neuronal debe ser entrenada para formular una respuesta en función de una solución, de allí que las Redes Neuronales Artificiales (RNA) se constituyen en un sistema de procesamiento para obtener una información relevante a un tema en particular.

En ese particular para los entrevistados el procesamiento de información mediante la aplicación de redes neuronales se emplea como

análisis para obtener resultados de datos estudiados, es decir, se entrena la red neuronal para que esta emita una respuesta como resultado de una consulta que previamente fue administrada al sistema.

De igual forma los entrevistados coincidieron en que se puede almacenar conocimiento por medio de las Redes Neuronales Artificiales, ya que estas imitan funciones cerebrales, lo que ayuda a mantener una organización por medio de las Redes Neuronales Artificiales (RNA), es decir, la red se entrena la primera vez, luego su conocimiento se almacena para ser utilizado posteriormente sin necesidad de repetir el mismo proceso.

Por otra parte, los entrevistados señalaron que aplicar un programa computacional por medio de inteligencia artificial ayuda a minimizar los tiempos empleados en las correcciones de las evaluaciones donde se emplea mucho tiempo para obtener resultados específicos de grupos de participantes.

Teniendo en consideración los resultados arrojados se puede señalar que con la aplicación de la propuesta se podrán minimizar los tiempos empleados en la corrección de evaluación de los programas del área de postgrado lo que ayudaría a los docentes a tener mayor tiempo en sus espacios de trabajo y por ende las diferentes actividades realizadas por estos serían más satisfactorias y aportarían mayor desempeño en cuanto a las áreas que imparten en la facultad.

Ello es debido a que se emplea la herramienta tecnológica como lo es el sistema computacional para implementar la propuesta, la cual consiste en adiestrar un programa como lo es la Red Neuronal Artificial para que cumpla con la tarea de evaluar los diferentes programas del departamento antes

señalado, teniendo en consideración que las RNA, son un modelo computacional inspirado en redes neuronales biológicas que puede ser considerada como un sistema de procesamiento de información, es una estructura distribuida, formada de neuronas artificiales, las cuales son usadas para procesar el conocimiento e intentar emular las funciones computacionales elementales de la red nerviosa del cerebro humano.

Por consiguiente se trata de una imitación de las funciones del cerebro con el empleo de un sistema computacional, que implica la inteligencia artificial donde la computadora es el canal para que se cumpla tal función, de allí que tanto la tecnología como el proceso de evaluación se conforman para trabajar a la par por medio de un sistema que tiene como función la evaluación de los diferentes programas del área de posgrado de la facultad de Ingeniería de la Universidad de Carabobo, de allí que la propuesta es la alternativa más viable para la solución de la problemática.

CAPÍTULO V

LA PROPUESTA

En este capítulo se trata lo concerniente al sistema de redes neuronales propuesto, diseño, entrenamiento, ejecución, eficacia y resultados del mismo.

Encuesta al Estudiante

La encuesta aplicada al estudiante y de la cual provienen los datos que serán suministrados a la RNA para su entrenamiento y posterior validación se encuentra en el anexo A y consta de 33 preguntas y son las siguientes:

Para cualquiera de las anteriores preguntas las posibles respuestas son:

- Muy Insatisfactorio
- Insatisfactorio
- Poco Insatisfactorio
- Indiferente
- Poco Satisfactorio
- Satisfactorio
- Muy Satisfactorio

Preparación de la data

Previo al desarrollo del modelo computacional, es necesario hacer una conversión y depuración de los datos existentes.

La data suministrada por la Universidad de Carabobo se recibió en formato de hoja de cálculo Excel de nombre “EVALUACIÓN DEL FUNCIONAMIENTO ACADÉMICO ADMINISTRATIVO.xlsx”, he aquí una vista parcial de dicho archivo.

1	A	B	C	D	E	F	G	H	I	J
1	Marca temporal	Es usted participante del f Programa		(1) El participante es info	(2) El participante es info	(3) La selección del persc	(4) El costo de la matrícul	(5) El costo de los arance	(6) Los procedimientos es	(7) La atención
2	8/1/2016 17:31:36	Estudiante	Maestría en Gerencia de l	ni satis fecho ni insatis fec	satis fecho	ni satis fecho ni insatis fec	satis fecho	insatis fecho	total mente insa	
3	8/1/2016 17:45:17	Estudiante	Maestría en Gerencia de l	total mente satis fecho	satis fecho	satis fecho				
4	8/1/2016 17:53:39	Estudiante	Maestría en Ingeniería Am	satis fecho	satis fecho	satis fecho	total mente satis fecho	total mente satis fecho	insatis fecho	satis fecho
5	8/1/2016 18:26:14	Estudiante	Maestría en Ingeniería Me	total mente satis fecho	satis fecho	total mente satis fecho	total mente satis fecho	total mente satis fecho	total mente satis fecho	total mente satis fecho
6	8/1/2016 18:27:03	Estudiante	Doctorado en Ingeniería	ni satis fecho ni insatis fec	ni satis fecho ni insatis fec	total mente satis fecho	total mente satis fecho	total mente satis fecho	insatis fecho	insatis fecho
7	8/1/2016 18:34:48	Estudiante	Maestría en Ingeniería Am	satis fecho	satis fecho	total mente insatis fecho	insatis fecho	ni satis fecho ni insatis fec	satis fecho	ni satis fecho n
8	8/1/2016 19:19:39	Estudiante	Maestría en Ingeniería Ind	insatis fecho	ni satis fecho ni insatis fec	ni satis fecho ni insatis fec	insatis fecho	ni satis fecho ni insatis fec	insatis fecho	insatis fecho
9	8/1/2016 19:31:00	Estudiante	Maestría en Ingeniería de l	ni satis fecho ni insatis fec	ni satis fecho ni insatis fec	total mente satis fecho	total mente satis fecho	satis fecho	satis fecho	satis fecho
10	8/1/2016 21:21:30	Estudiante	Maestría en Gerencia de l	ni satis fecho ni insatis fec	satis fecho	total mente satis fecho	satis fecho	satis fecho	ni satis fecho ni insatis fec	satis fecho
11	8/1/2016 22:13:26	Estudiante	Maestría en Ingeniería Eli	ni satis fecho ni insatis fec	satis fecho	total mente satis fecho	total mente satis fecho	total mente satis fecho	insatis fecho	ni satis fecho n
12	8/1/2016 22:18:02	Estudiante	Maestría en Ingeniería Am	total mente satis fecho	total mente satis fecho	satis fecho	total mente satis fecho	total mente satis fecho	total mente satis fecho	total mente satis fecho
13	8/2/2016 7:33:27	Estudiante	Maestría en Ingeniería Am	total mente satis fecho	total mente satis fecho	satis fecho	total mente satis fecho	total mente satis fecho	total mente satis fecho	total mente satis fecho
14	8/2/2016 7:55:00	Estudiante	Maestría en Ingeniería Ind	satis fecho	satis fecho	satis fecho	insatis fecho	satis fecho	insatis fecho	insatis fecho
15	8/2/2016 8:49:13	Estudiante	Maestría en Ingeniería de l	total mente satis fecho	total mente satis fecho	satis fecho	total mente satis fecho	total mente satis fecho	total mente satis fecho	total mente satis fecho
16	8/2/2016 9:53:03	Estudiante	Maestría en Gerencia de l	total mente satis fecho	total mente satis fecho	satis fecho	total mente satis fecho	total mente satis fecho	satis fecho	satis fecho
17	8/2/2016 12:32:27	Estudiante	Maestría en Ingeniería Am	satis fecho	satis fecho	satis fecho	insatis fecho	insatis fecho	satis fecho	satis fecho
18	8/2/2016 13:08:15	Estudiante	Maestría en Ingeniería Ind	total mente satis fecho	total mente satis fecho	insatis fecho	total mente satis fecho	total mente satis fecho	total mente satis fecho	satis fecho
19	8/2/2016 13:41:15	Estudiante	Maestría en Gerencia de l	total mente satis fecho	satis fecho	total mente satis fecho	total mente satis fecho	total mente satis fecho	satis fecho	satis fecho
20	8/2/2016 13:48:05	Estudiante	Doctorado en Ingeniería	total mente satis fecho	ni satis fecho ni insatis fec	satis fecho	ni satis fecho ni insatis fec	ni satis fecho ni insatis fec	total mente satis fecho	insatis fecho
21	8/2/2016 17:14:50	Estudiante	Maestría en Gerencia de l	total mente satis fecho	total mente satis fecho	satis fecho	satis fecho	satis fecho	satis fecho	satis fecho
22	8/2/2016 18:32:04	Estudiante	Maestría en Ingeniería Eli	satis fecho	satis fecho	total mente satis fecho	total mente satis fecho	total mente satis fecho	satis fecho	ni satis fecho n
23	8/2/2016 20:08:37	Estudiante	Maestría en Ingeniería Am	total mente satis fecho	satis fecho	satis fecho	total mente satis fecho	total mente satis fecho	total mente insatis fecho	total mente insa
24	8/2/2016 21:21:06	Estudiante	Maestría en Ingeniería Ind	total mente satis fecho	total mente satis fecho	satis fecho	satis fecho	satis fecho	insatis fecho	satis fecho
25	8/3/2016 11:35:31	Estudiante	Maestría en Ingeniería Eli	satis fecho	satis fecho	satis fecho	insatis fecho	insatis fecho	satis fecho	satis fecho
26	8/3/2016 18:23:00	Estudiante	Maestría en Gerencia de l	satis fecho	total mente satis fecho	satis fecho	satis fecho	satis fecho	satis fecho	satis fecho
27	8/4/2016 8:22:21	Estudiante	Maestría en Ingeniería Am	satis fecho	satis fecho	satis fecho	insatis fecho	ni satis fecho ni insatis fec	ni satis fecho ni insatis fec	ni satis fecho n

Figura 23. Vista parcial del archivo base de la data “cruda” en formato Excel. Fuente propia.

En una primera fase se procedió a convertir el formato Excel a “csv”, un formato de archivo de texto delimitado por caracteres especiales como pueden ser las comas “,” en este caso.

Luego se procedió a la eliminación de datos nulos o con errores. Posteriormente se hizo una conversión de datos categóricos en etiquetas de texto a valores numéricos para su posterior utilización en el modelo neuronal en forma cuantitativa y así finalmente se dispone de una base de

datos adecuada para entrenar y probar el modelo como se muestra a continuación.

```

1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25
, 26, 27, 28, 29, 30, 31, 32, 33, R
7, 6, 3, 6, 1, 5, 2, 7, 2, 7, 3, 5, 6, 3, 4, 3, 6, 2, 6, 6, 5, 4, 4, 7, 6, 3, 2, 3, 7, 5, 4, 7, 3
, 6
3, 4, 2, 7, 5, 7, 5, 2, 2, 2, 6, 7, 1, 3, 5, 3, 5, 3, 2, 7, 7, 3, 2, 7, 3, 1, 2, 1, 3, 1, 6, 3, 6
, 7
7, 4, 3, 4, 3, 7, 1, 3, 2, 2, 4, 2, 5, 5, 6, 4, 7, 2, 2, 6, 7, 1, 4, 2, 4, 3, 7, 4, 1, 5, 1, 6, 5
, 5
4, 3, 2, 7, 6, 2, 4, 3, 7, 7, 6, 4, 6, 1, 3, 3, 6, 5, 3, 6, 6, 3, 1, 2, 6, 5, 7, 4, 7, 6, 7, 3, 2
, 4
4, 6, 5, 4, 1, 6, 5, 7, 4, 4, 6, 2, 2, 5, 6, 1, 1, 2, 2, 6, 6, 6, 2, 2, 3, 7, 3, 2, 1, 6, 4, 2, 4
, 5
1, 4, 4, 6, 5, 7, 5, 4, 7, 4, 6, 5, 3, 7, 3, 3, 7, 2, 3, 6, 5, 4, 4, 7, 6, 6, 7, 3, 7, 4, 7, 6, 7
, 5
1, 6, 4, 5, 3, 1, 7, 7, 1, 5, 6, 1, 7, 5, 5, 3, 5, 6, 1, 6, 6, 7, 2, 2, 5, 6, 5, 4, 3, 3, 1, 7, 4
, 5
2, 7, 3, 1, 3, 5, 5, 3, 1, 7, 4, 7, 5, 7, 6, 3, 6, 7, 3, 3, 7, 7, 1, 1, 1, 6, 2, 6, 6, 5, 3, 4, 6
, 6
7, 5, 6, 7, 6, 2, 4, 7, 7, 4, 3, 6, 6, 6, 6, 3, 1, 5, 7, 3, 1, 1, 1, 1, 1, 5, 2, 6, 3, 2, 3, 2, 7
, 1
5, 4, 3, 2, 3, 3, 4, 3, 2, 2, 6, 5, 7, 7, 6, 4, 4, 2, 3, 2, 2, 7, 1, 4, 3, 1, 7, 2, 7, 5, 1, 7, 5
, 7
2, 3, 6, 4, 7, 4, 1, 6, 4, 7, 6, 7, 2, 7, 3, 3, 6, 3, 7, 6, 6, 3, 4, 7, 4, 1, 4, 6, 3, 1, 4, 6, 4
, 5
5, 3, 2, 4, 7, 6, 5, 5, 5, 7, 4, 4, 6, 2, 5, 1, 5, 7, 4, 1, 4, 7, 3, 5, 3, 1, 5, 5, 2, 2, 6, 5, 1
, 1
5, 7, 2, 4, 1, 5, 5, 6, 2, 3, 1, 2, 6, 7, 1, 6, 7, 7, 1, 4, 5, 7, 3, 7, 7, 3, 1, 3, 1, 5, 6, 5, 2
, 5
5, 6, 1, 2, 7, 7, 2, 2, 7, 5, 5, 6, 7, 6, 2, 1, 2, 6, 1, 2, 6, 4, 3, 3, 1, 5, 2, 6, 1, 4, 5, 4, 1
, 8
1, 4, 2, 4, 4, 6, 5, 6, 3, 6, 3, 5, 7, 6, 5, 3, 2, 7, 7, 1, 7, 7, 3, 7, 1, 6, 4, 4, 4, 6, 4, 6, 3
, 2
1, 1, 2, 7, 7, 2, 4, 4, 7, 7, 6, 7, 2, 7, 5, 1, 6, 6, 4, 3, 5, 3, 3, 1, 3, 3, 7, 3, 7, 1, 7, 1, 1
, 7
7, 6, 5, 5, 1, 3, 5, 4, 7, 4, 4, 7, 4, 7, 4, 3, 3, 2, 3, 7, 2, 1, 3, 1, 3, 7, 7, 3, 4, 3, 4, 4, 2
, 5
7, 7, 3, 6, 3, 2, 6, 7, 6, 3, 6, 5, 1, 7, 6, 1, 7, 1, 1, 3, 7, 2, 4, 3, 1, 1, 7, 5, 1, 3, 1, 4, 2
, 2
4, 3, 6, 5, 7, 6, 4, 7, 2, 4, 4, 5, 7, 1, 1, 3, 6, 7, 4, 6, 7, 1, 3, 7, 7, 6, 7, 6, 5, 3, 5, 7, 4

```

Figura 24. **Vista de la data de entrada del modelo neuronal (entrenamiento y prueba).** Fuente Propia.

```
1, 2, 3, 4, 5, 6, 7
0, 0, 0, 0, 1, 0, 0
0, 0, 1, 0, 0, 0, 0
0, 0, 0, 1, 0, 0, 0
0, 0, 0, 0, 1, 0, 0
0, 0, 0, 1, 0, 0, 0
0, 0, 0, 0, 1, 0, 0
0, 0, 0, 0, 1, 0, 0
0, 0, 0, 0, 1, 0, 0
0, 0, 0, 1, 0, 0, 0
0, 0, 0, 1, 0, 0, 0
0, 0, 0, 1, 0, 0, 0
0, 0, 0, 1, 0, 0, 0
0, 0, 0, 0, 1, 0, 0
0, 0, 0, 1, 0, 0, 0
0, 0, 0, 1, 0, 0, 0
0, 0, 0, 1, 0, 0, 0
0, 0, 1, 0, 0, 0, 0
0, 0, 0, 1, 0, 0, 0
0, 0, 0, 1, 0, 0, 0
0, 0, 0, 1, 0, 0, 0
0, 0, 0, 1, 0, 0, 0
0, 0, 0, 1, 0, 0, 0
0, 0, 1, 0, 0, 0, 0
0, 0, 0, 1, 0, 0, 0
0, 0, 0, 1, 0, 0, 0
0, 0, 0, 0, 1, 0, 0
0, 0, 0, 1, 0, 0, 0
0, 0, 0, 1, 0, 0, 0
0, 0, 1, 0, 0, 0, 0
0, 0, 0, 1, 0, 0, 0
0, 0, 0, 1, 0, 0, 0
0, 0, 0, 1, 0, 0, 0
0, 0, 0, 1, 0, 0, 0
0, 0, 0, 1, 0, 0, 0
0, 0, 0, 1, 0, 0, 0
```

Figura 25. Vista de la data de salida del modelo neuronal (entrenamiento y prueba). Fuente Propia.

Muestra para el entrenamiento de la RNA

Descripción de la muestra

Para el entrenamiento de RNA se debe tener una muestra que contenga las respuestas a todas las preguntas de la encuesta y al mismo tiempo representen toda la variedad posible de su población.

Para resolver problemas mediante el entrenamiento de RNA es importante cuan diferentes son entre sí los elementos que conforman dicha población. Como el objetivo de la presente investigación es realizar una predicción de la evaluación, entonces se debe tomar el total de las preguntas a evaluar para conseguir la información más completa que logre

entrenar correctamente la RNA. La muestra consta de una base de datos existente en la unidad de postgrado de antiguas encuestas, esta alcanza a 256 registros con respuestas a 33 preguntas por cada uno de ellos y una evaluación general.

Cada pregunta hecha tiene solo 7 posibles respuestas y vienen expresadas en forma cualitativa, luego la asociamos a un número para darle el grado de satisfacción de manera cuantitativa (ver cuadro 2):

Cuadro 2
Respuesta ponderada numéricamente

Respuesta	Ponderación
Muy Insatisfactorio	1
Insatisfactorio	2
Poco Insatisfactorio	3
Indiferente	4
Poco Satisfactorio	5
Satisfactorio	6
Muy Satisfactorio	7

Nota: Fuente propia

La siguiente especificación matemática muestra la estructura de cada registro:

$$R = \{P1, P2, P3, P4, \dots, P33, R E\}; \quad \text{Encuesta a un estudiante.}$$

Donde R es un registro cualquiera, P1, P2, ..., P33 son las respuestas a cada pregunta y E es la evaluación general a esa encuesta individual.

Generación de datos de entrada a partir de la muestra seleccionada

Representación Funcional entre las RNA y sus Datos de Entrada y Salida

En teoría, la dependencia funcional entre una RNA y sus datos de entrada y salida es la siguiente:

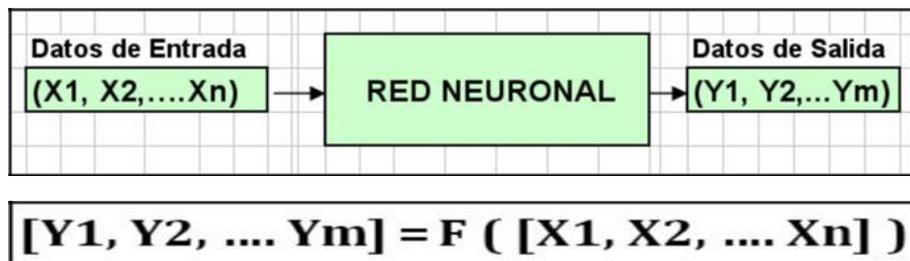


Figura 26: Representación funcional entre la RNA y los datos de entrada y salida. Fuente propia

Para el caso que nos ocupa la dependencia funcional entre los datos de entrada y salida y de la RNA es la siguiente:

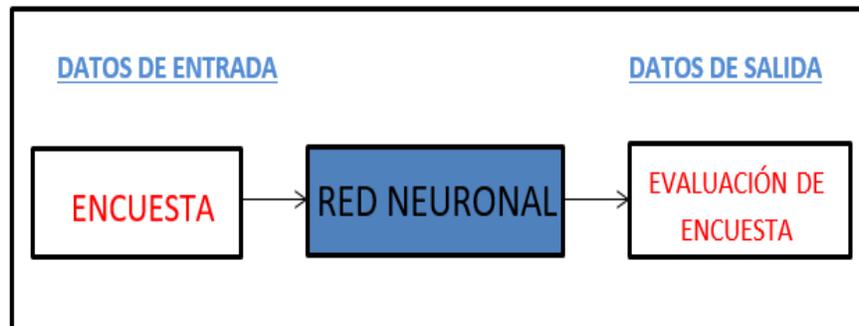


Figura 27: Dependencia Funcional entre la Encuesta y la Evaluación de la encuesta. Fuente propia

Como el objetivo es demostrar que una RNA puede ser entrenada para predecir las evaluaciones de las encuestas, se explicará el proceso en detalle, la dependencia funcional de la RNA y sus datos de entrada y salida será la siguiente:

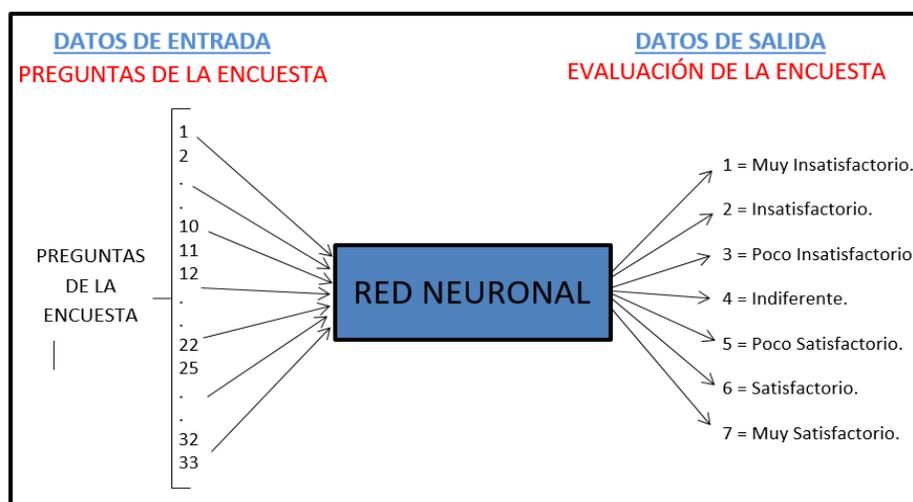


Figura 28: Relación entre los Elementos de la Muestra y la Evaluación. Fuente propia

El problema será generar por cada encuesta de la muestra un vector lineal de siete posiciones con información binaria y solo habrá un número uno en la posición que corresponda con la predicción hecha.

Almacenaje de los Vectores Lineales en una Tabla

A cada encuesta le corresponde como salida de la RNA un vector lineal con 7 valores binarios, que corresponden a los valores obtenidos por el modelo para predecir la evaluación.

La unión de todos los vectores lineales con información binaria generados por cada encuesta, conformará una matriz de $n \times 33$ con la predicción de todos los datos suministrados. En la siguiente figura se aprecia la conformación de la matriz de entrada de la RNA:

Cuadro 3
Matriz de datos de entrada de la RNA.

	Pregunta 1	Pregunta 2	Pregunta 3	Pregunta 4	Pregunta 33
Registro 1	1	2	3	4	5	6	7
Registro 2	2	1	3	4	5	6	7
Registro 3	3	4	2	1	5	7	6
Registro 4	4	5	6	7	2	1	3
...	5	6	7	1	2	3	4
...	6	5	1	2	3	4	7
...	7	6	5	4	3	2	1
...	3	2	1	5	4	6	7
Registro n	4	5	6	7	1	2	3

Nota: Fuente propia

De igual manera, la salida de la RNA producirá una matriz de valores binarios, cada fila estará compuesta por 7 celdas y solo una de ellas tendrá un 1 como valor, el resto serán 0, indicando así la posición donde se realizó la predicción.

Cuadro 4:
Matriz de datos de salida de la RNA.

	1	2	3	4	5	6	7
Predicción 1	0	0	0	1	0	0	0
Predicción 2	0	1	0	0	0	0	0
Predicción 3	0	0	0	1	0	0	0
Predicción 4	1	0	0	0	0	0	0
...	0	0	0	0	0	0	1
...	0	0	1	0	0	0	0
Predicción n	0	0	0	0	1	0	0

Nota: Fuente propia

Arquitectura de la RNA

Tipo de Red Neuronal

El tipo de red neuronal que utilizaremos es Deep Neural Network (DNN) o Red Neuronal Profunda, que se distinguen de las demás redes neuronales porque poseen más de una capa oculta, de allí el término profundo. En las redes de aprendizaje profundo, cada capa de nodos se entrena en un conjunto distinto de características basadas en la salida de la capa anterior. Cuanto más avance en la red neuronal, más complejas serán las características que sus nodos pueden reconocer, ya que agregan y recombinan características de la capa anterior. En nuestro caso tendremos dos capas ocultas.

Número de Neuronas en la Capa de Entrada

El vector de entrada X almacena los valores de respuesta de la encuesta, en un formato de datos que consta de valores entre el 1 al 7, cada elemento del vector lineal de entrada debe ser ingresado a la RNA a través de una neurona, por lo tanto, la red neuronal debe tener 33 neuronas de entrada, debido a que es una por cada valor de las respuestas a las 33 preguntas. Es decir, nuestra entrada será un vector lineal con 33 posiciones y sus valores oscilaran entre 1 y 7.

Número de Neuronas en la Capa de Salida

Cada neurona de salida de una red neuronal sirve para reconocer una evaluación diferente, como el objetivo de la investigación es pronosticar la evaluación de la totalidad de las preguntas hechas a cada encuestado,

entonces debemos tener siete neuronas en la capa de salida, es decir, una por cada valor posible que pueda tomar la predicción de la evaluación.

De las 7 neuronas de salidas solo una estará activada y se hace con el conjunto binario, es decir, el valor de 1 para la activación de la neurona y 0 para el estado de desactivado, así la respuesta al pronóstico esperado se obtiene mediante una de las neuronas de la capa de salida.

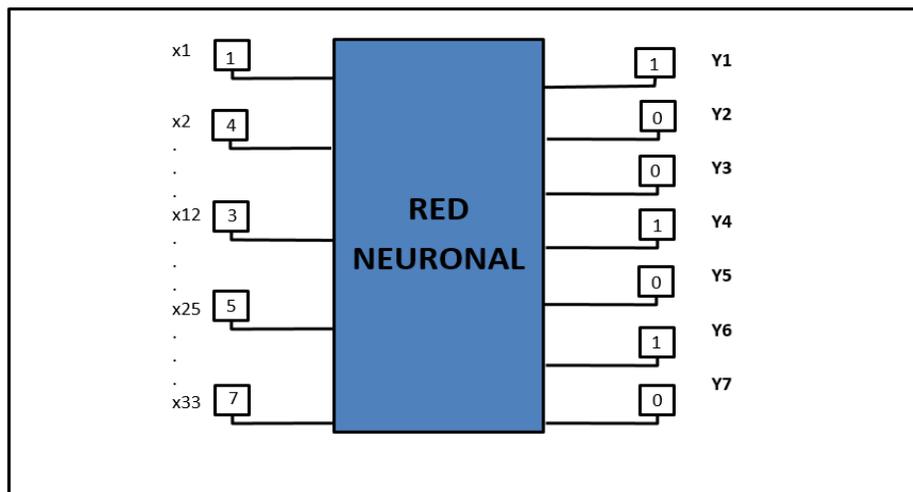


Figura 29: Arquitectura de la RNA para la Predicción de las encuestas.
Fuente Propia

Desde el punto de vista funcional el vínculo entre nuestra RNA y los datos de entrada y salida será como sigue:

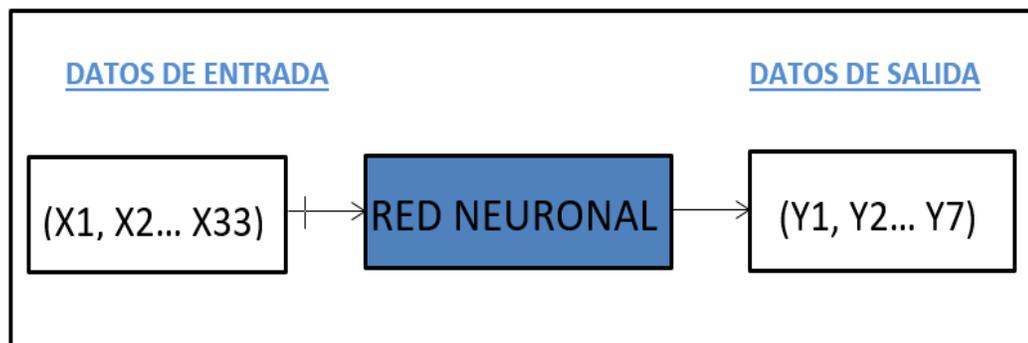


Figura 30. Representación Funcional entre la RNA y los Datos de Entrada y Salida para la predicción de Evaluación de Encuestas. Fuente propia

Número de Neuronas en las capas ocultas

El número de neuronas adecuadas de la capa intermedia es muy importante, sin embargo, aunque el funcionamiento de la red depende en forma importante del número de nodos en las capas ocultas, no existe aún un método confiable que permita determinar con precisión el número óptimo de estos, por lo tanto, la realizaremos de una manera experimental.

Para esta investigación se utilizan dos capas ocultas, cada una de ellas tendrá 128 neuronas, de este modo las dos capas ocultas tendrán $2 \times 128 = 256$ neuronas. Si sumamos las siete neuronas de entradas y las siete de salida obtendremos un total de 270 neuronas.

Funciones de activación para las neuronas

La función de activación que usan las neuronas ocultas es "ReLU" (Rectified Linear Unit) y su fórmula es la siguiente:

$$f(x) = x^+ = \max(0, x)$$

donde x es la entrada de la neurona.

Esto también se conoce como función de rampa y es análogo a la rectificación de media onda en ingeniería eléctrica. Es la función de activación más popular para redes neuronales profundas. Para la activación de las neuronas de salida se usará la función “SoftMax” (Función Exponencial Normalizada).

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}}$$

Donde z es un vector de las entradas a las capas de salida, j indexa las unidades de salida siendo $j=1,2,\dots,K$. Es una generalización de la función sigmoidea y es con frecuencia la función de activación de la capa final. Trabaja apilando capas de funciones lineales y no lineales, detectando una amplia gama de patrones y prediciendo con precisión una etiqueta para una imagen determinada. Actúa básicamente como un normalizador y produce un vector discreto de distribución de probabilidad y la salida de la RNA es una probabilidad de que los valores ingresados se corresponda con una clase en particular.

Momentum

El momentum permite reducir las oscilaciones y consigue que la convergencia de la función de activación del gradiente sea más rápida, pero

tiene una dificultad y es que no detecta cuándo se ha alcanzado un mínimo. Debido a que a la inercia se le da un valor elevado y constante, sucede con frecuencia que cuando se alcanza el mínimo de la función este no se detecta y el algoritmo se dirige a un óptimo local.

Para solucionar este problema, Nesterov propuso un algoritmo que lleva su nombre y se conoce como método del gradiente acelerado que es el siguiente: Una vez computado el gradiente en la manera habitual, se calculan cuáles son los nuevos parámetros tras ser actualizados con el factor momento seleccionado.

Luego, se calcula el gradiente del error respecto de dichos parámetros futuros. El gradiente a futuro admite comprobar si los nuevos parámetros permiten acercarse al mínimo o, si por el contrario lo alejan, esto nos da la opción de corregirlos si se da el segundo caso mediante la identificación del momentum en la siguiente iteración.

$$\tilde{\theta} = \theta_t + \alpha v_t$$

$$g_{NAG} = \frac{1}{n} \sum_{i=1}^n \nabla_{\theta} \mathcal{L}(x^{(i)}, y^{(i)}, \tilde{\theta}).$$

Donde v_t es la velocidad para los parámetros θ que calculan los parámetros provisionales $\tilde{\theta}$. Luego se calcula el gradiente usando los parámetros interinos.

Luego se actualiza los parámetros utilizando una regla de actualización similar con al método Momentum.

$$v_{t+1} = \alpha v_t - \eta g_{NAG}$$

$$\theta_{t+1} = \theta_t + v_{t+1}.$$

Función de Pérdida

La función de pérdida o función objetivo, o función de puntuación de optimización, es uno de los parámetros requeridos para construir nuestro modelo y usaremos 'categorical - crossentropy' (entropía categórica cruzada), que trabajará recortando los valores de predicción de la red neuronal por el bien de la estabilidad numérica antes de que se active la función 'SoftMax' y esta es su ecuación:

$$Loss_{NN} = -\frac{1}{D} \sum_{i=1}^{batch} (Loss_i \mathcal{F}_{correction,i})$$

Generación de la base de entrenamiento

Representación en una Tabla el Conocimiento de la Evaluación de las Encuestas

La Base de Entrenamiento para la RNA, es la representación del conocimiento que tiene el experto humano, en forma de datos procesables por una computadora.

Para la presente investigación, se seleccionaron 256 encuestas, pertenecientes a los programas de Postgrado de Doctorado en Ingeniería,

Maestría en Gerencia de la Construcción, Maestría en Ingeniería Ambiental, Maestría en Ingeniería Mecánica, Maestría en Ingeniería de Procesos, Maestría en Ingeniería Eléctrica, Maestría en Matemática y Computación, Maestría en Ingeniería Industrial y Especialización Técnica en Gerencia de Servicios Automotrices.

Las muestras seleccionadas representan para nuestra investigación el conocimiento del experto humano y se puede representar mediante la siguiente tabla:

**Cuadro 5:
Representación del Conocimiento**

Predicción	Evaluación
1	Muy Malo.
2	Malo.
3	Deficiente.
4	Regular.
5	Bueno.
6	Muy Bueno.
7	Excelente.

Nota: Fuente propia

Generación de la Base de Entrenamiento

Para la generación de la base de entrenamiento de la RNA se debe transformar la tabla de representación del conocimiento del experto, en datos procesables por el computador.

Como se ha descrito antes, la representación mediante el vector lineal 33 valores de X (uno por cada encuesta) y otro vector lineal de 7 valores binarios (que representa el valor de la predicción de la RNA) conformara la base de conocimiento para el aprendizaje de nuestro modelo.

Predicción	Evaluación
1	Muy Malo.
2	Malo.
3	Deficiente.
4	Regular.
5	Bueno.
6	Muy Bueno.
7	Excelente.

BASE DE CONOCIMIENTO PARA EL ENTRENAMIENTO DE LA RED													
Datos de Entrada							Datos de Salida						
Pregunta	Pregunta	Pregunta	Pregunta	Pregunta	Y1	Y2	Y3	Y4	Y5	Y6	Y7
1	2	3	4	33							
1	2	3	4	5	6	7	1	0	0	0	0	0	0
2	1	3	4	5	6	7	0	1	0	0	0	0	0
3	4	2	1	5	7	6	0	0	1	0	0	0	0
4	5	6	7	2	1	3	0	0	0	1	0	0	0
5	6	7	1	2	3	4	0	0	0	0	1	0	0
6	5	1	2	3	4	7	0	0	0	0	0	1	0
7	6	5	4	3	2	1	0	0	0	0	0	0	1

Figura 31: Base de Entrenamiento de la RNA. Fuente propia

De esta manera se generan los datos de entrada y salida para cada una de las encuestas de la muestra, por lo tanto, para esta investigación la Base de Entrenamiento de la RNA tendrá un total de 256 registros.

Los datos de esta base de conocimiento se guardan en archivos tipo texto con extensión 'csv' con el nombre de 'trainX.csv' y 'trainY.csv'. Estos archivos se utilizarán en la RNA para su entrenamiento.

Generación de los datos de salida de la RNA a partir de la muestra seleccionada

De acuerdo a arquitectura definida los datos de salida se generan de la siguiente manera:

a. Si la predicción corresponde a “Muy Insatisfactorio” entonces el vector de salida $(y_1, y_2, y_3, y_4, y_5, y_6, y_7)$ será igual a $(1, 0, 0, 0, 0, 0, 0)$.

b. Si la predicción corresponde a “Insatisfactorio” entonces el vector de salida $(y_1, y_2, y_3, y_4, y_5, y_6, y_7)$ será igual a $(0, 1, 0, 0, 0, 0, 0)$.

c. Si la predicción corresponde a “Poco Insatisfactorio” entonces el vector de salida $(y_1, y_2, y_3, y_4, y_5, y_6, y_7)$ será igual a $(0, 0, 1, 0, 0, 0, 0)$.

d. Si la predicción corresponde a “Indiferente” entonces el vector de salida $(y_1, y_2, y_3, y_4, y_5, y_6, y_7)$ será igual a $(0, 0, 0, 1, 0, 0, 0)$.

e. Si la predicción corresponde a “Poco Satisfactorio” entonces el vector de salida $(y_1, y_2, y_3, y_4, y_5, y_6, y_7)$ será igual a $(0, 0, 0, 0, 1, 0, 0)$.

f. Si la predicción corresponde a “Satisfactorio” entonces el vector de salida $(y_1, y_2, y_3, y_4, y_5, y_6, y_7)$ será igual a $(0, 0, 0, 0, 0, 1, 0)$.

g. Si la **predicción** corresponde a “Muy Satisfactorio” entonces el vector de salida $(y_1, y_2, y_3, y_4, y_5, y_6, y_7)$ será igual a $(0, 0, 0, 0, 0, 0, 1)$.

Entrenamiento de la RNA en la predicción de las encuestas

Para construir y entrenar la RNA usaremos varios softwares, estos son: el sistema operativo Linux, lenguaje de programación Python, las librerías numpy, tensorflow, tflearn y pandas.

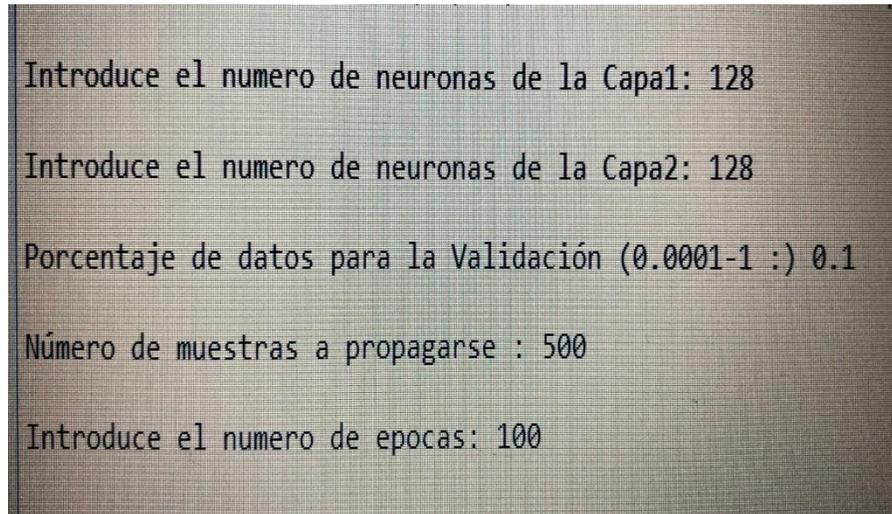
Mostraremos el procedimiento de entrenamiento de la RNA con el sistema de redes neuronales constuido suministrándole la base de entrenamiento almacenada en 'trainX.csv' y 'trainY.csv'.

El primer paso consiste en subir los datos al programa y transformarlos en matrices para que pueda ser utilizado por nuestro sistema de redes neuronales. Utilizando la librería Pandas lo almacenamos en la variable 'dataX' y luego lo transformamos a matriz bajo el nombre de 'matrizX', estos son los vectores lineales de los valores de las encuestas. A continuación se hace lo mismo con los vectores que contienen los valores de predicción, los almacenamos en la variable dataY y posteriormente los transformamos en una matriz de nombre 'matrizY'.

Luego de tener los datos preparados, la tarea es suministrar a la RNA los datos de entrada para que sea capaz de aprender a reconocer los datos de entrada y vincularlos con los datos de salida. Previamente hemos definido el número de neuronas de entrada y de salida, así como también las capas ocultas, por lo tanto, la arquitectura de la RNA está completamente definida y conformada.

Ingreso de Parámetros

En el entorno del Python ejecutamos el programa Tesis.py e ingresamos los parámetros requeridos.



```
Introduce el numero de neuronas de la Capa1: 128
Introduce el numero de neuronas de la Capa2: 128
Porcentaje de datos para la Validación (0.0001-1 :) 0.1
Número de muestras a propagarse : 500
Introduce el numero de epocas: 100
```

Figura 32: *Ingreso de Parámetros en RNA*. Fuente Propia

Justificación de los Hiper-Parámetros Ingresados

Neuronas Capa1: [0] Asume por defecto 128 / [n]

Son el número de neuronas que tendrá la capa oculta número 1.

Neuronas Capa2: [0] Asume por defecto 128 / [n]

Son el número de neuronas que tendrá la capa oculta número 2.

Porcentaje de Datos para la Validación: [0] Asume por defecto 0.1 / [n]

Este parámetro le dice al modelo que porcentaje de datos va a tomar para realizar las predicciones, el restante se utilizará para entrenamiento. Por defecto se asigna 0.1 (asume una muestra del 10% para la validación del modelo)

Número de muestras a propagarse: [0] Asume por defecto 500 / [n]

Son el número de muestras que se propagaran a través de la red

Épocas: [0] Asume por defecto 100 / [n]

Son el número de épocas en la cual será entrenado el modelo.

Resultado del entrenamiento

Se realizaron un total de 21 pruebas de entrenamiento, buscando obtener el más óptimo cambiando los hiper-parámetros, funciones de activación internas de las neuronas, función de pérdida utilizada por el optimizador de capa y función de momentum. Finalmente se llegó a la siguiente configuración con la cual logramos 95,41% de acierto o certeza, lo cual es muy bueno.

Resumen de ajuste de Hiperparámetros utilizados

Porcentaje de datos de validación (**learning_rate**) = **0.1**

Tipo de Red Neuronal: Red Neuronal Profunda (**DNN**)

Número de neuronas de entrada: **34**

Número de neuronas de la primera capa oculta: **128**

Número de neuronas de la segunda capa oculta: **128**

Número de neuronas de la capa de salida: **7**

Épocas de entrenamiento: **200**

Cantidad de paquetes a propagar en la red (**batch_size**): **500**

Función de activación de las capas ocultas: Unidad Lineal Rectificada (**ReLU**, por sus siglas en inglés).

Función de activación de las neuronas de salida: **“SoftMax”**

Función de pérdida del optimizador de capa:
“categorical_crossentropy”.

Función de **Momentum: Nesterov**.

Número de núcleos para el cálculo sobre la tarjeta gráfica (**GPU**): **8**

Fracción de memoria para el cálculo sobre la tarjeta gráfica (**GPU**): **0.5**

Análisis de la ejecución del Entrenamiento

Con los parámetros antes mencionados se obtuvieron los siguientes indicadores del proceso de entrenamiento y funcionamiento de la RNA.

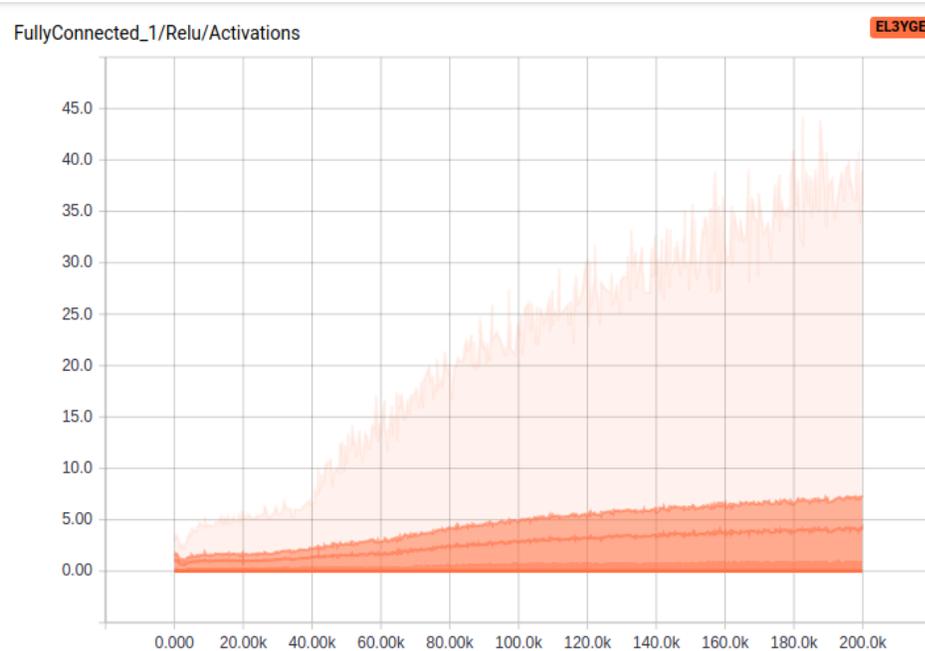


Gráfico 1: Función de activación ReLu en las capas ocultas.

En el gráfico 1 se puede apreciar como se comporta el rectificador a traves de los entrenamientos, va incrementado a medida que se realizan los entrenamientos.

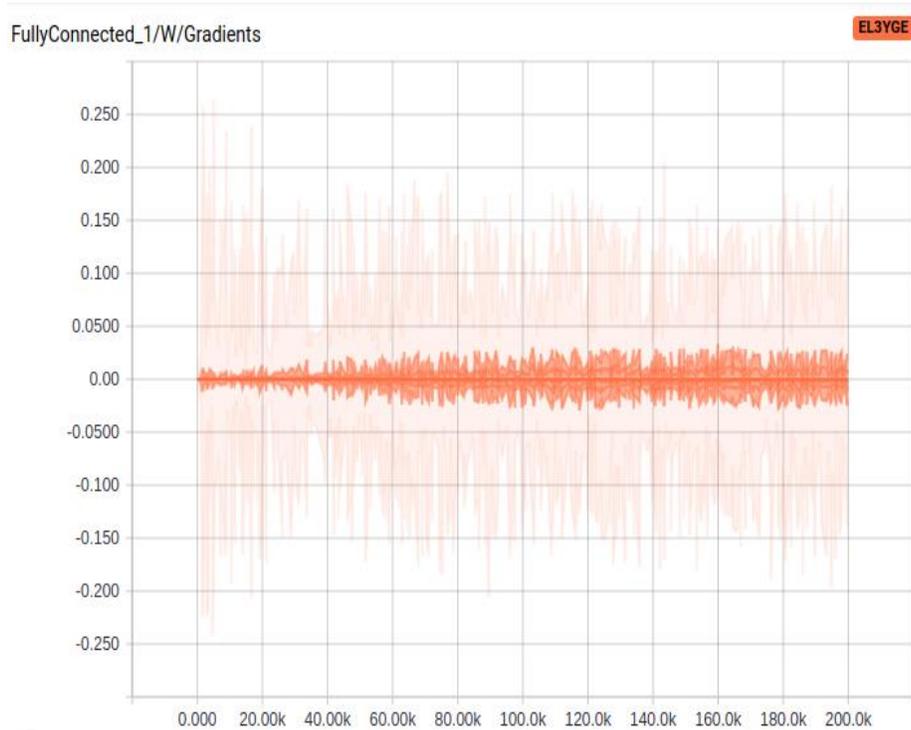


Gráfico 2: Función de Gradiente en la Primera capa oculta de la RNA.

En el gráfico 2 se observa el ajuste de la función de gradiente descendiente a medida que avanza el entrenamiento.

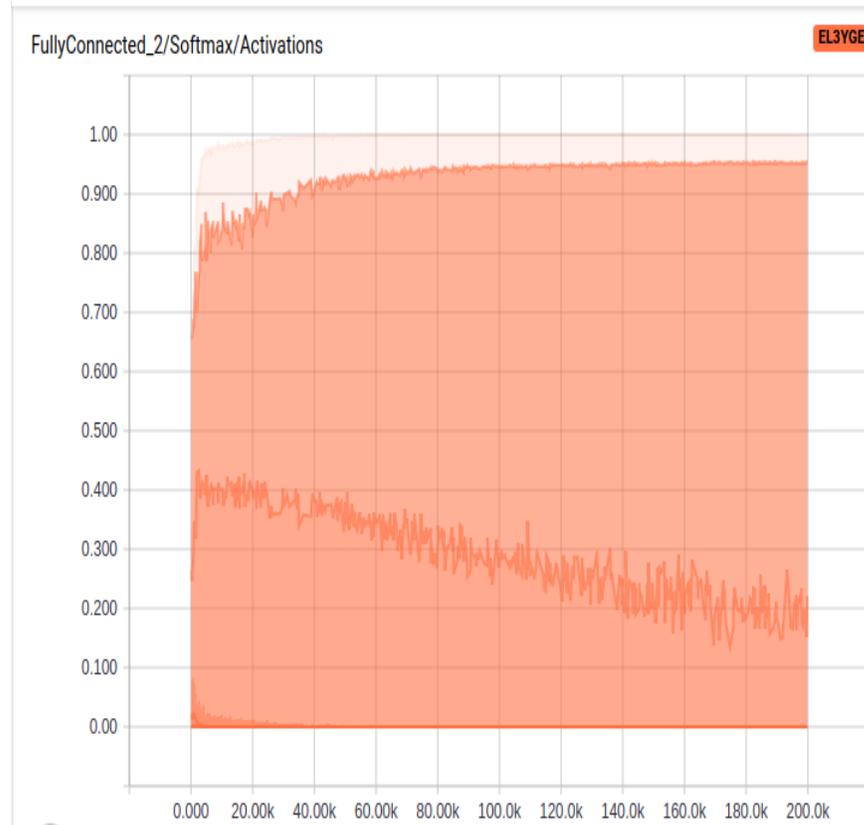


Gráfico 3: Función Softmax en las neuronas de la capa de salida

En el gráfico 3 observamos como la función Softmax aumenta las probabilidades del valor máximo (1) de la capa anterior en comparación con el otro valor (0).

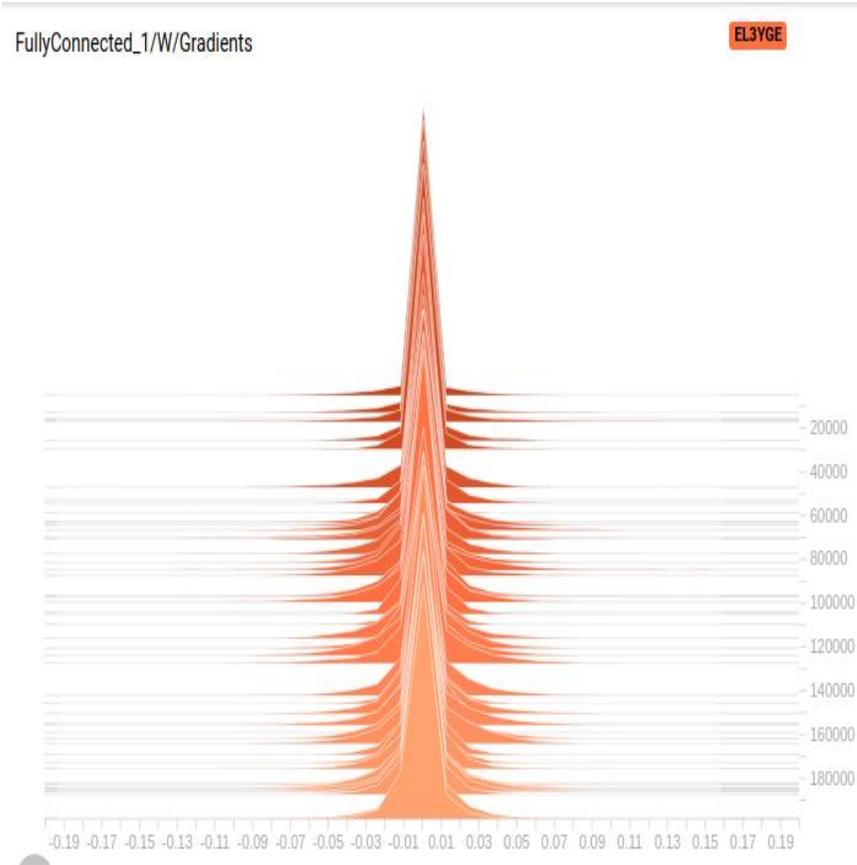


Gráfico 4: **Función de Gradiente de acuerdo al learning-rate 1.**

En el gráfico 4 se observa el ajuste de la función de gradiente descendiente a medida que avanza el entrenamiento de acuerdo al parámetro de learning-rate 1.

FullyConnected_1/b/Gradients

EL3YGE

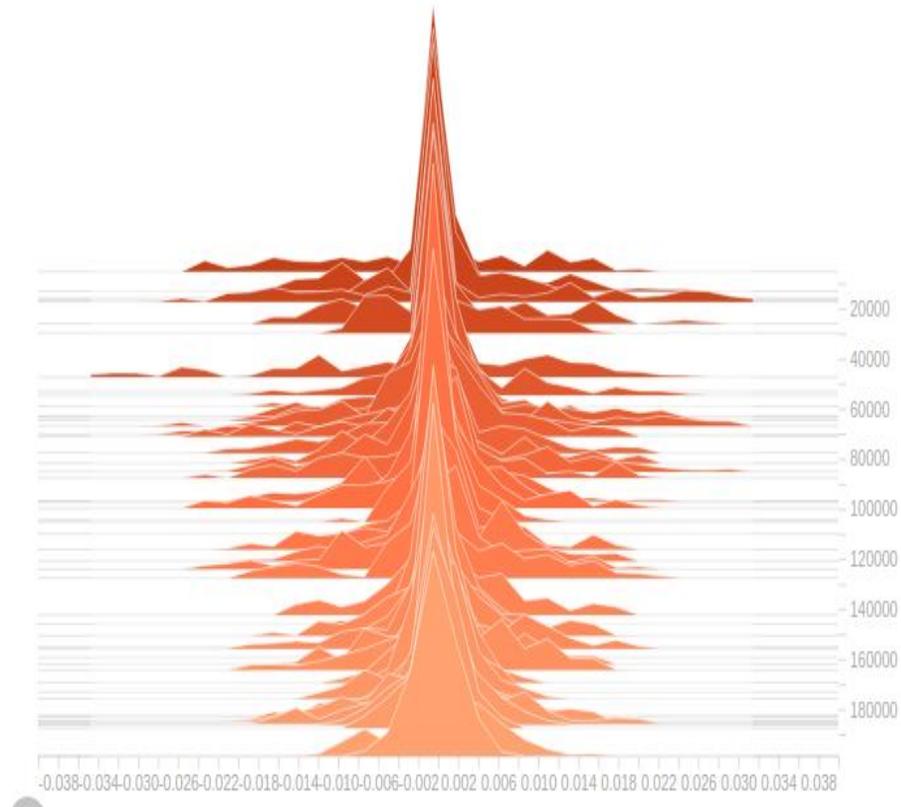


Gráfico 5: **Función de Gradiente de acuerdo al learning-rate 2.**

En el gráfico 5 se observa el ajuste de la función de gradiente descendente a medida que avanza el entrenamiento de acuerdo al parámetro de learning-rate 2.

FullyConnected_2/b/Gradients

EL3YGE

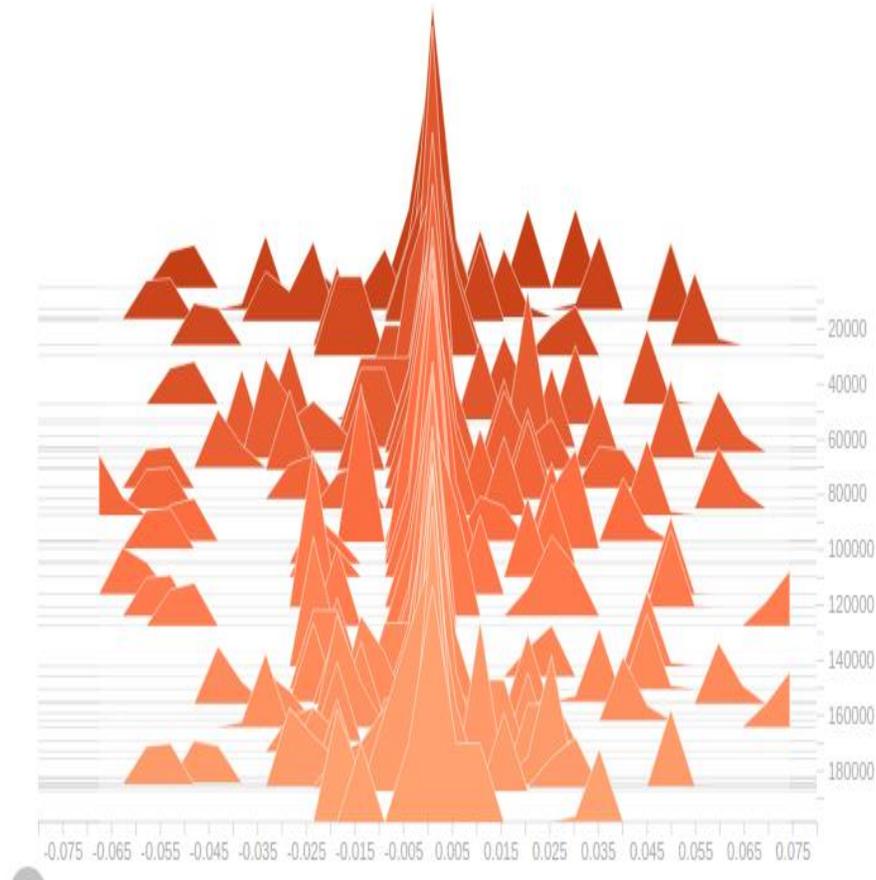


Gráfico 6: **Función de Gradiente en la capa oculta 2 de la RNA.**

En el gráfico 6 se observa el ajuste de la función de gradiente descendiente en la capa oculta número 2 a medida que avanza el entrenamiento.

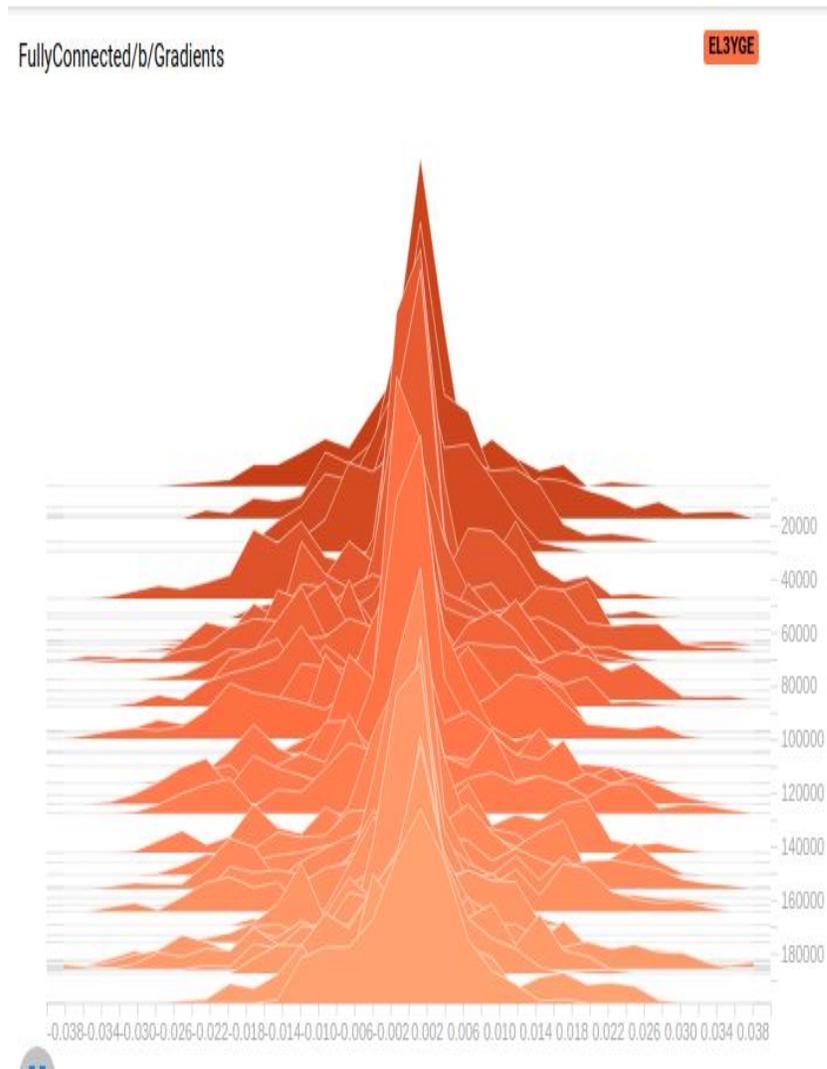


Gráfico 7: Función de Gradiente en la capa oculta 1 de la RNA.

En el gráfico 7 se observa el ajuste de la función de gradiente descendiente en la capa oculta número 1 a medida que avanza el entrenamiento.

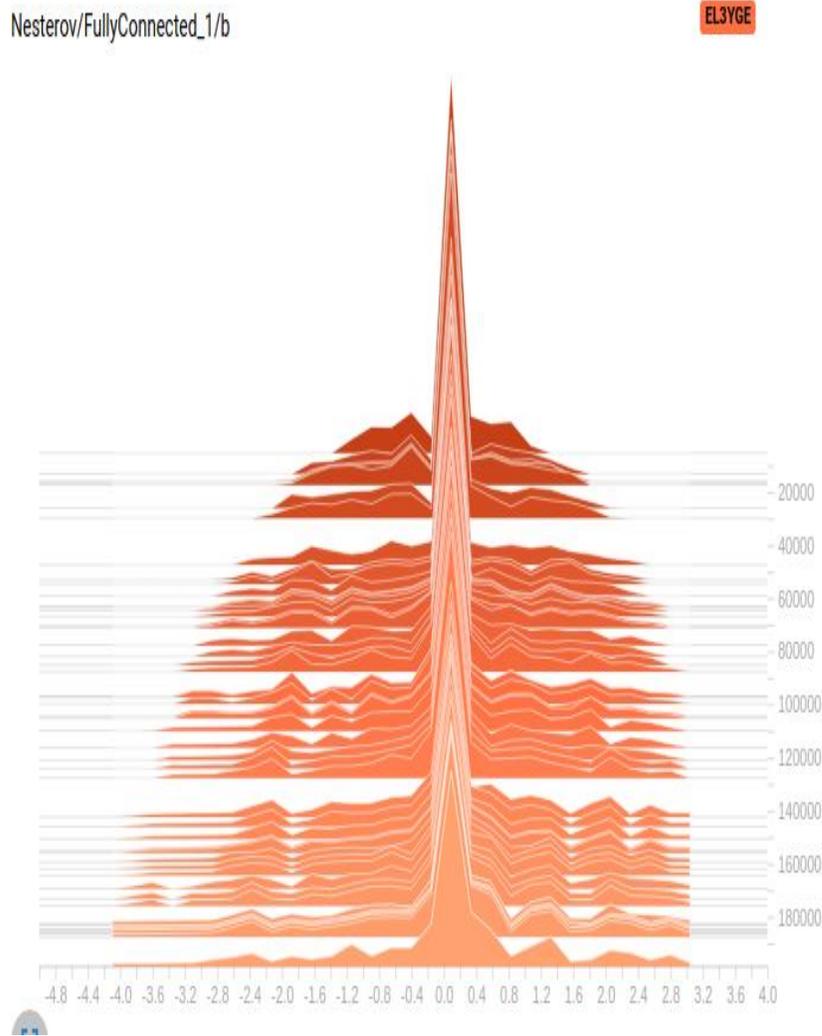


Gráfico 8: Función de Momentum Nesterov Capa Oculta 1 de la RNA

En el gráfico 8 se puede apreciar el nivel de corrección de la función de gradiente acelerado (Nesterov) luego de calcular el gradiente del error respecto a los parámetros futuros, comprueba si los nuevos parámetros permiten acercarse al mínimo o, si por el contrario lo alejan, tomando una decisión al respecto en la capa oculta número 1.

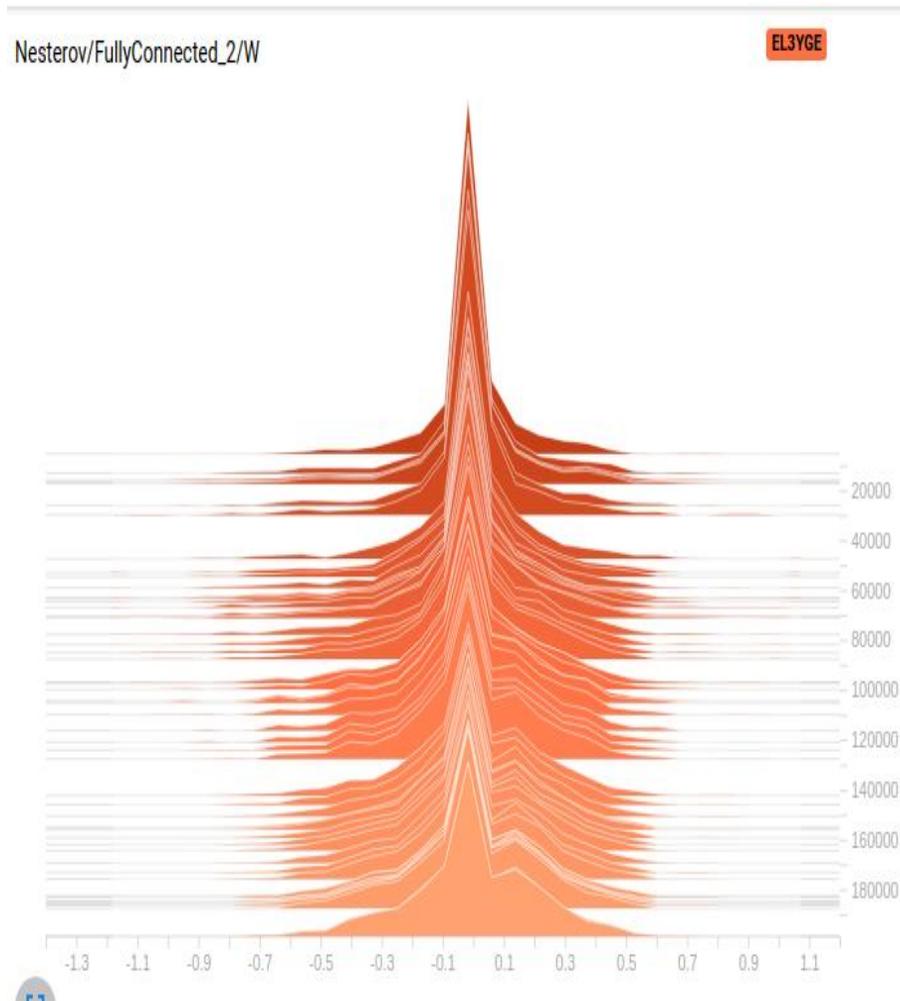


Gráfico 9: **Función de Momentum Nesterov Capa Oculta 2 de la RNA**

En el gráfico 9 se puede apreciar el nivel de corrección de la función de gradiente acelerado (Nesterov) luego de calcular el gradiente del error respecto a los parámetros futuros, comprueba si los nuevos parámetros permiten acercarse al mínimo o, si por el contrario lo alejan, tomando una decisión al respecto en la capa oculta número 2.

Nesterov/FullyConnected_2/b

EL3YGE

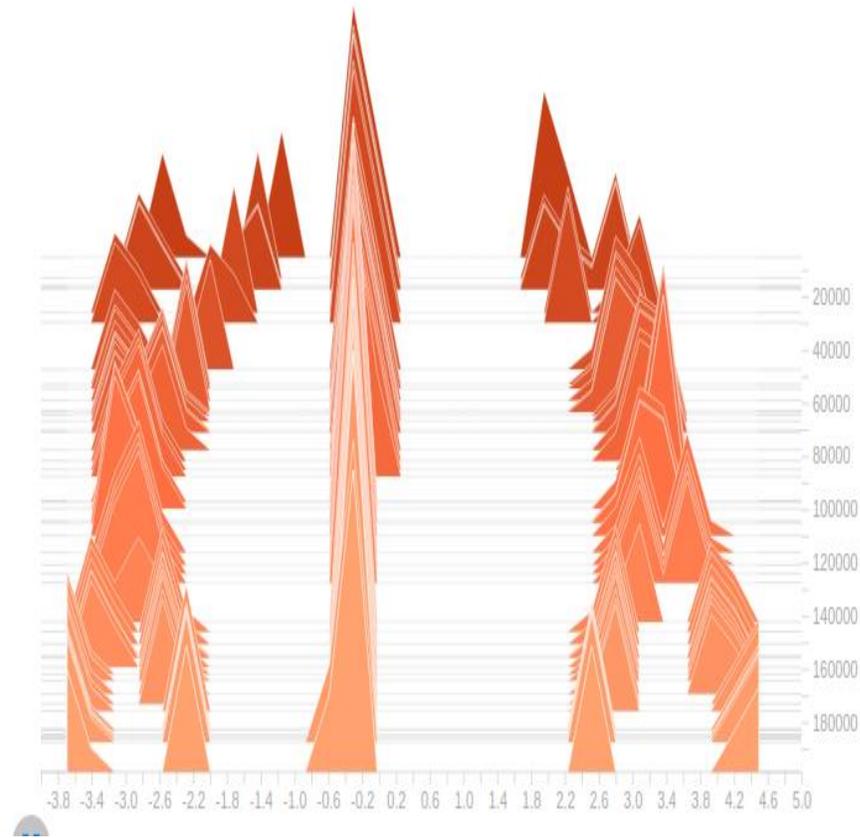


Gráfico 10: **Función de Momentum Nesterov Capa Oculta 1 y 2.**

En el gráfico 10 se puede apreciar el nivel de corrección de la función de gradiente acelerado (Nesterov) luego de calcular el gradiente del error respecto a los parámetros futuros, comprueba si los nuevos parámetros permiten acercarse al mínimo o, si por el contrario lo alejan, tomando una decisión al respecto en la capa oculta número 1 y 2.

Finalmente, luego del entrenamiento y la validación, se concluye guardando en archivos todos los pesos sinápticos de la RNA, esto lo hacemos en “tesis.modelo.index” y “tesis.modelo.meta” y es muy importante ya que una vez entrenada y validada la RNA, se pueden recuperar todos los datos de su aprendizaje sin necesidad de volver a realizar dichos procesos y de esta manera podemos utilizarla en el momento que se requiera.

Eficiencia de la RNA entrenada

Los siguientes gráficos muestran de manera muy precisa como el proceso de aprendizaje va aumentando la certeza con la que la RNA realiza las predicciones. En el primero vemos solo la exactitud de las predicciones conforme avanzan las épocas de entrenamiento. En la segunda figura se puede comparar la exactitud y la validación del modelo que de igual forma van en forma ascendente, es decir, aprendiendo más con cada iteración.

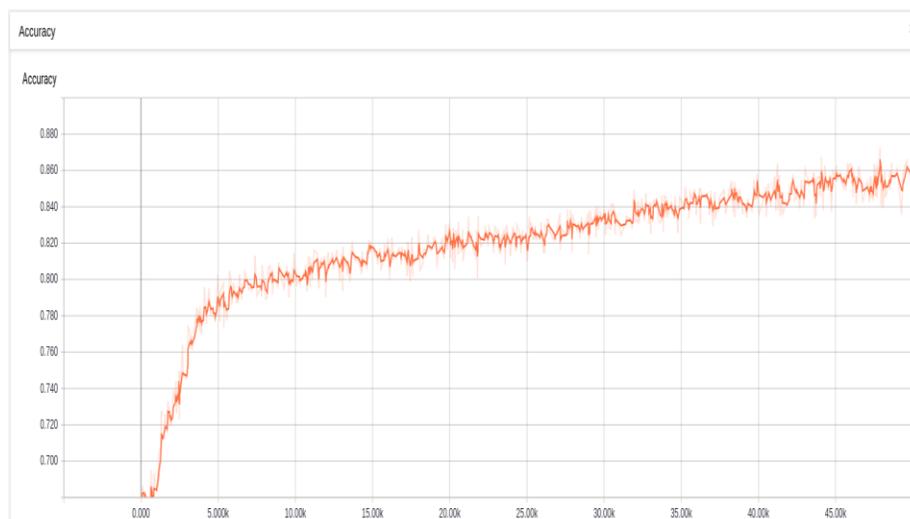


Gráfico 11: Progreso de la Exactitud de la predicción de la RNA.

En el gráfico 11 se puede ver como la RNA va aumentando la exactitud de las predicciones acorde aumenta el número de entrenamientos, también se observa como tiende a estabilizarse.

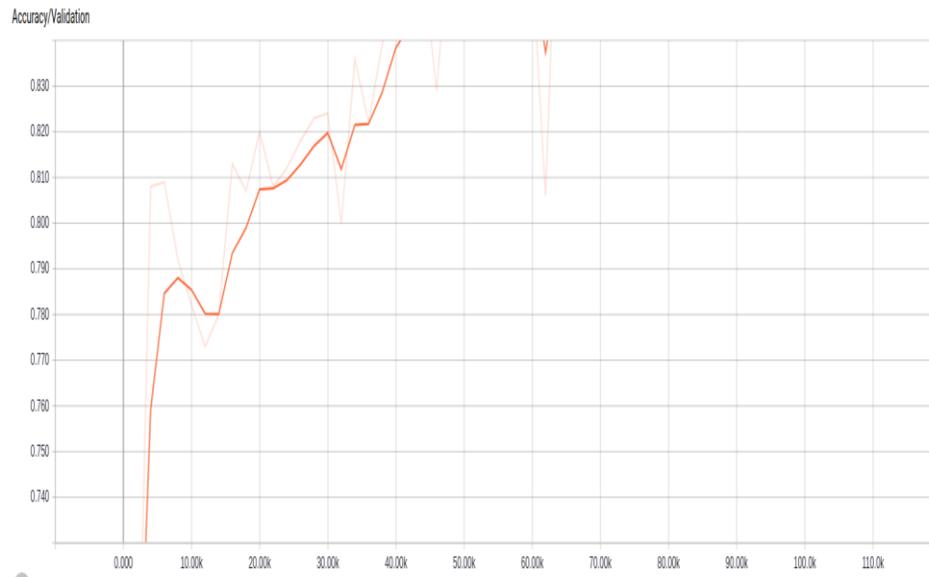


Gráfico 12: Progreso de la Exactitud vs Validación de la RNA.

En el gráfico 12 se puede ver como la RNA va aumentando la exactitud de las predicciones vs los datos de prueba (validación) acorde aumenta el número de entrenamientos.

Gráficos del funcionamiento de los Nodos de Tensor Flow

A continuación, se muestran una serie de gráficos computacionales que describen las operaciones que la librería de TensorFlow realiza en forma de nodos y que organiza el trabajo de la red neuronal. TensorFlow utiliza un gráfico de flujo de datos para representar su cálculo en términos de las dependencias entre operaciones individuales. El flujo de datos es un modelo de programación común para la computación paralela.

En un gráfico de flujo de datos, los nodos representan unidades de cálculo y los bordes representan los datos consumidos o producidos por un cálculo. La estructura gráfica de los nodos y los bordes del gráfico indican cómo se componen las operaciones individuales, pero no prescriben cómo deben usarse. La estructura del gráfico es como un código de ensamblaje: inspeccionarlo contiene cierta información útil, pero no todo lo que el código fuente transmite. De esta manera se pueden observar tres elementos en estos gráficos:

Nodos: Realizan el cálculo y tienen cero o más entradas y salidas. Los datos que se mueven entre nodos se conocen como tensores y son matrices multidimensionales de valores reales.

Bordes: Señalan el flujo de los datos, bifurcaciones, bucles y actualizaciones de los estados. También se usan para sincronizar el comportamiento dentro del gráfico.

Operación: Es un cálculo abstracto con nombre que puede tomar atributos de entrada y producir atributos de salida.

En la información contenida de los siguientes gráficos, se puede observar relaciones entre funciones utilizadas por la RNA, el entrenamiento, la exactitud de las predicciones y su entropía.

Graph Principal

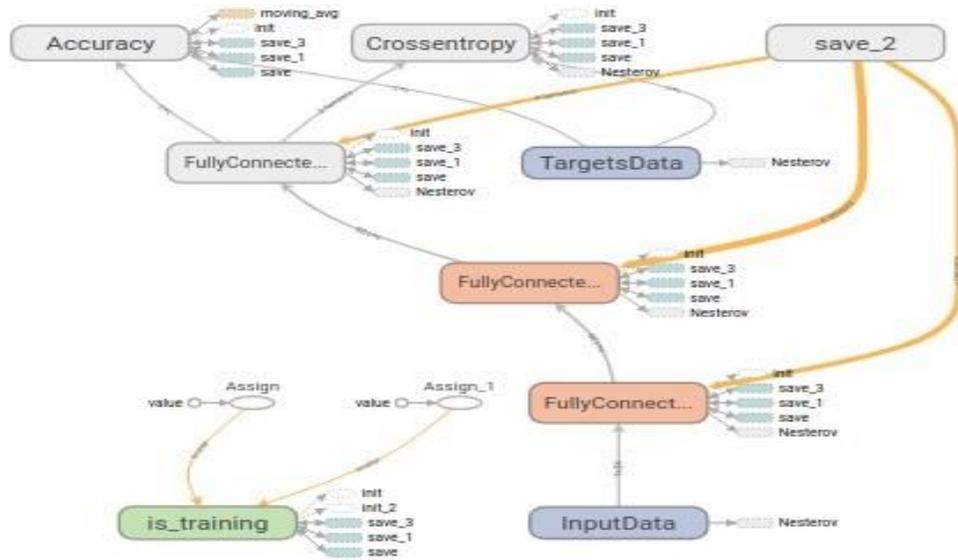


Figura 33. Estructura principal de los nodos creados por TensorFlow. Fuente propia

Graph Principal 2

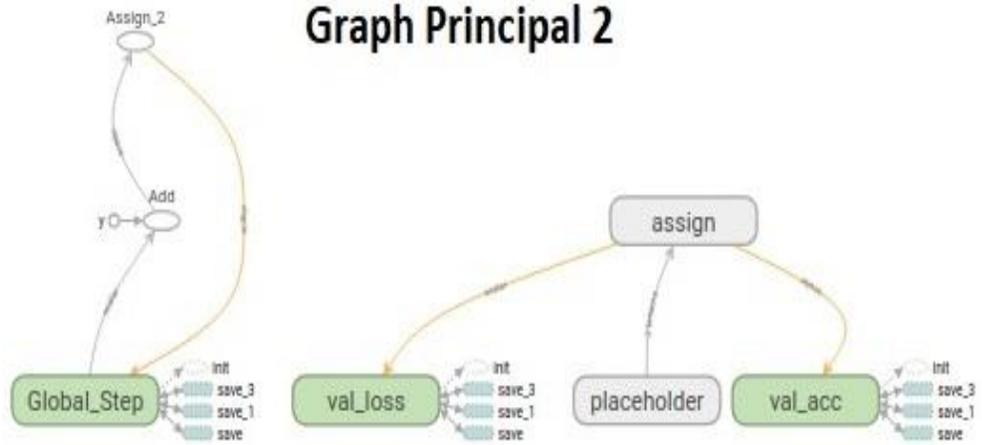


Figura 34. Estructura principal 2 de los nodos creados por TensorFlow. Fuente propia

Nodos Auxiliares

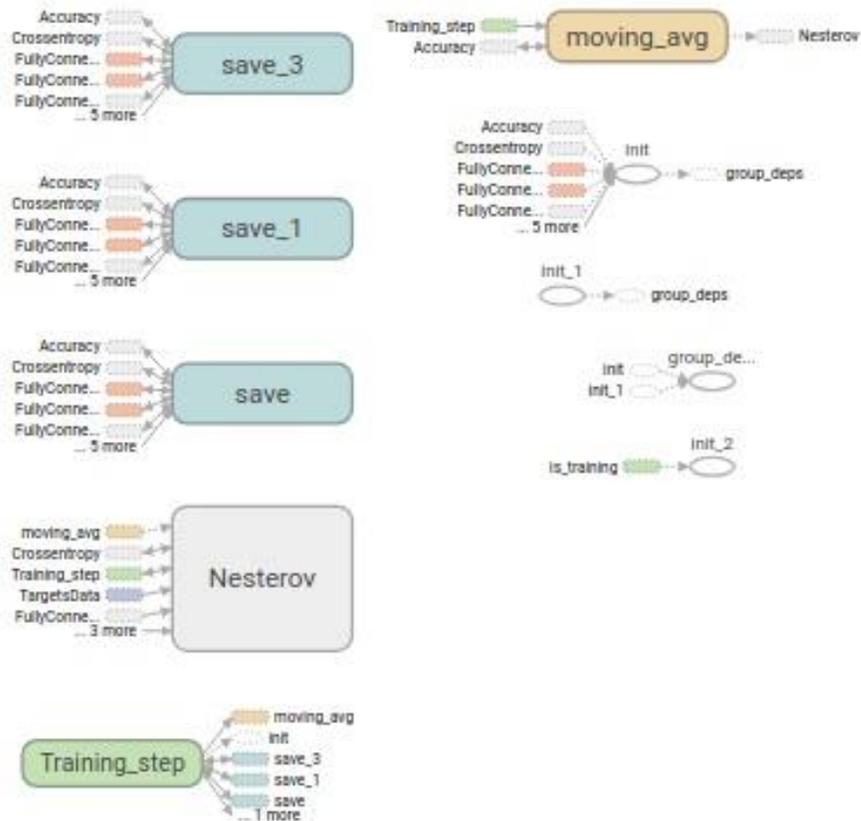


Figura 35. **Nodos auxiliares creados por TensorFlow.** Fuente propia

Interpretación de los Datos de Salida

Para los valores de salida del proceso de reconocimiento primero se colocan los valores en una matriz, luego con esos valores se reemplazan los valores respectivos de la tabla de Datos de Reconocimiento presentados en la tabla.

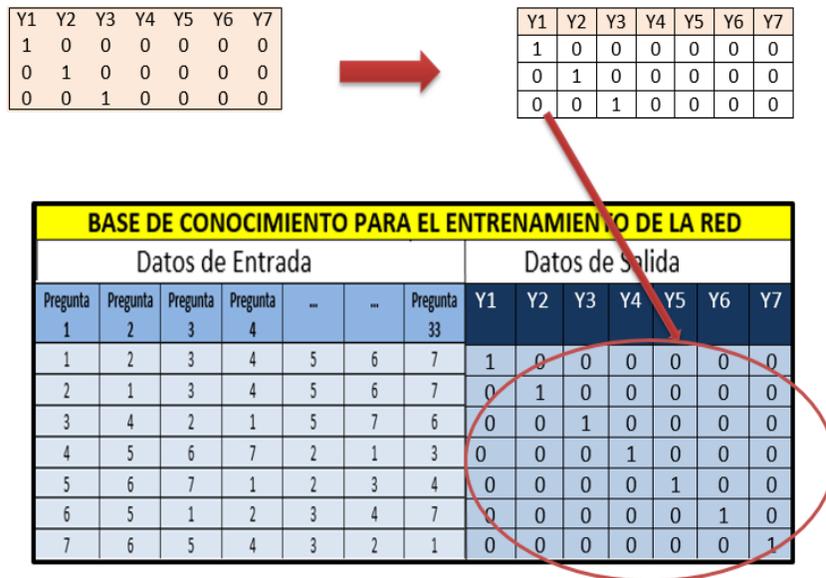


Figura 36: **Datos de Entrada y Salida del Proceso de Reconocimiento.** Fuente propia

Vectores resultantes por cada predicción:

$(1, 0, 0, 0, 0, 0, 0) =$ Muy Insatisfactorio = Se transforma a 1

$(0, 1, 0, 0, 0, 0, 0) =$ Insatisfactorio = Se transforma a 2

$(0, 0, 1, 0, 0, 0, 0) =$ Poco Insatisfactorio = Se transforma a 3

$(0, 0, 0, 1, 0, 0, 0) =$ Indiferente = Se transforma a 4

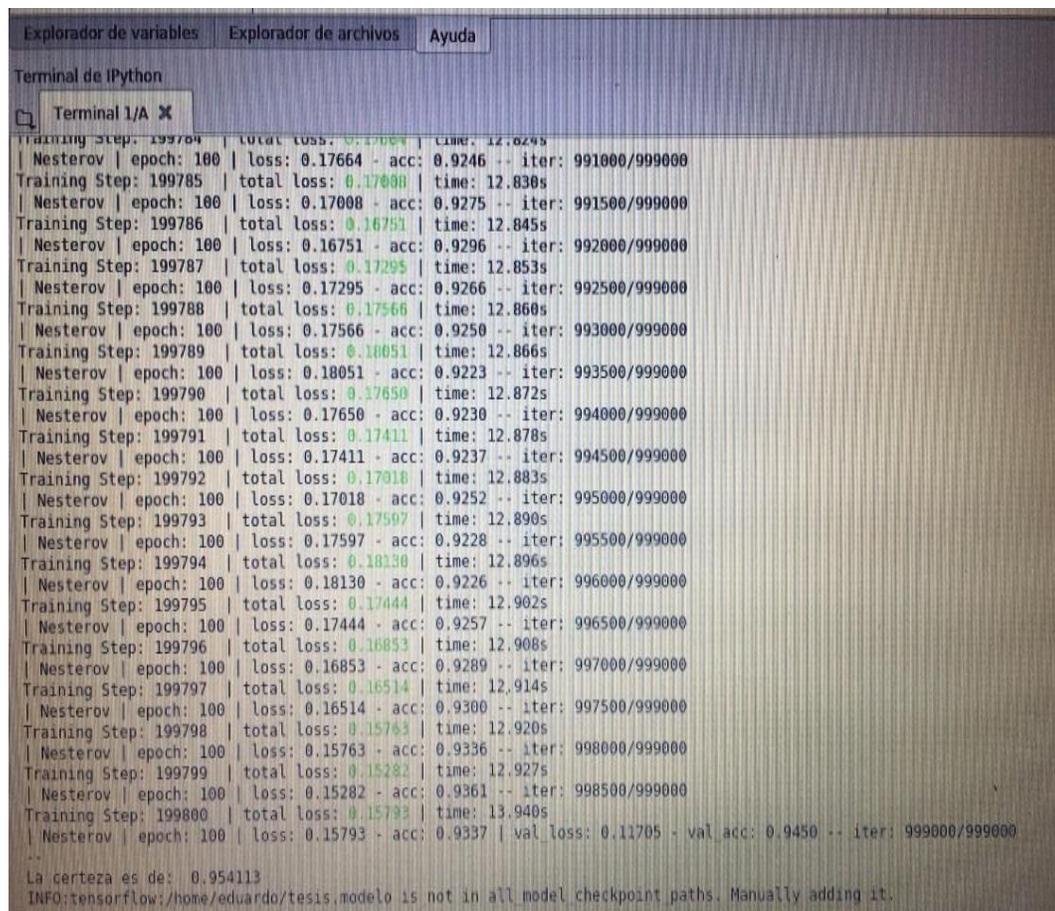
$(0, 0, 0, 0, 1, 0, 0) =$ Poco Satisfactorio = Se transforma a 5

$(0, 0, 0, 0, 0, 1, 0) =$ Satisfactorio = Se transforma a 6

$(0, 0, 0, 0, 0, 0, 1) =$ Muy Satisfactorio = Se transforma a 7

Resultado de la Validación (Predicción)

Como se puede apreciar, a medida que transcurren las 200 iteraciones o épocas el aprendizaje de la RNA va mejorando hasta llegar a un muy buen margen de certeza en la predicción de 95,41%, lo que quiere de decir que nuestro error es de apenas 5.59%, como ya dijimos esto es muy bueno. Hemos logrado el objetivo y se puede afirmar que tenemos una RNA bien entrenada para la predicción de las encuestas estudiadas en este trabajo.



```

Explorador de variables  Explorador de archivos  Ayuda
Terminal de IPython
Terminal 1/A X
Training Step: 199784 | total loss: 0.17064 | time: 12.829s
| Nesterov | epoch: 100 | loss: 0.17664 - acc: 0.9246 -- iter: 991000/999000
Training Step: 199785 | total loss: 0.17008 | time: 12.830s
| Nesterov | epoch: 100 | loss: 0.17008 - acc: 0.9275 -- iter: 991500/999000
Training Step: 199786 | total loss: 0.16751 | time: 12.845s
| Nesterov | epoch: 100 | loss: 0.16751 - acc: 0.9296 -- iter: 992000/999000
Training Step: 199787 | total loss: 0.17295 | time: 12.853s
| Nesterov | epoch: 100 | loss: 0.17295 - acc: 0.9266 -- iter: 992500/999000
Training Step: 199788 | total loss: 0.17566 | time: 12.860s
| Nesterov | epoch: 100 | loss: 0.17566 - acc: 0.9250 -- iter: 993000/999000
Training Step: 199789 | total loss: 0.18051 | time: 12.866s
| Nesterov | epoch: 100 | loss: 0.18051 - acc: 0.9223 -- iter: 993500/999000
Training Step: 199790 | total loss: 0.17650 | time: 12.872s
| Nesterov | epoch: 100 | loss: 0.17650 - acc: 0.9230 -- iter: 994000/999000
Training Step: 199791 | total loss: 0.17411 | time: 12.878s
| Nesterov | epoch: 100 | loss: 0.17411 - acc: 0.9237 -- iter: 994500/999000
Training Step: 199792 | total loss: 0.17018 | time: 12.883s
| Nesterov | epoch: 100 | loss: 0.17018 - acc: 0.9252 -- iter: 995000/999000
Training Step: 199793 | total loss: 0.17597 | time: 12.890s
| Nesterov | epoch: 100 | loss: 0.17597 - acc: 0.9228 -- iter: 995500/999000
Training Step: 199794 | total loss: 0.18130 | time: 12.896s
| Nesterov | epoch: 100 | loss: 0.18130 - acc: 0.9226 -- iter: 996000/999000
Training Step: 199795 | total loss: 0.17444 | time: 12.902s
| Nesterov | epoch: 100 | loss: 0.17444 - acc: 0.9257 -- iter: 996500/999000
Training Step: 199796 | total loss: 0.16853 | time: 12.908s
| Nesterov | epoch: 100 | loss: 0.16853 - acc: 0.9289 -- iter: 997000/999000
Training Step: 199797 | total loss: 0.16514 | time: 12.914s
| Nesterov | epoch: 100 | loss: 0.16514 - acc: 0.9300 -- iter: 997500/999000
Training Step: 199798 | total loss: 0.15763 | time: 12.920s
| Nesterov | epoch: 100 | loss: 0.15763 - acc: 0.9336 -- iter: 998000/999000
Training Step: 199799 | total loss: 0.15282 | time: 12.927s
| Nesterov | epoch: 100 | loss: 0.15282 - acc: 0.9361 -- iter: 998500/999000
Training Step: 199800 | total loss: 0.15793 | time: 13.940s
| Nesterov | epoch: 100 | loss: 0.15793 - acc: 0.9337 | val_loss: 0.11705 - val_acc: 0.9450 -- iter: 999000/999000
..
La certeza es de: 0.954113
INFO:sensorflow:/home/eduardo/tesis.modelo is not in all model checkpoint paths. Manually adding it.

```

Figura 37: Resultados del entrenamiento de la RNA. Fuente propia

CAPÍTULO VI

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

Se logró hacer un diagnóstico de la situación de los procesos de evaluación de programas de postgrado de Ingeniería de la Universidad de Carabobo y se pudo verificar la necesidad de un sistema que automatice el proceso.

Luego de estudiar la factibilidad de incorporar un sistema de redes neuronales se observó la viabilidad del mismo y se procedió a diseñar el mismo para cumplir con la evaluación de los programas de postgrado de Ingeniería de la Universidad de Carabobo.

Al finalizar el presente trabajo, Se puede concluir que la RNA ha realizado las predicciones de las evaluaciones de manera correcta con una certeza del 95,41%, siendo un margen de acierto muy bueno para un modelo aprendizaje profundo.

El modelo propuesto permite automatizar los procesos de evaluación de los programas de postgrado de la Facultad de Ingeniería de la Universidad de Carabobo, así como también la creación de un registro histórico de las encuestas para futuras generaciones de reportes estadísticos que guarden relación con dichas evaluaciones. El diseño de la RNA ha sido capaz de pronosticar de manera eficiente las evaluaciones que son realizadas por personas.

Se pudo constatar que muchos de los algoritmos de Machine y Deep Learning están automatizados en librerías disponibles en lenguajes como Python y R. Luego de la creación del programa surgen dos aspectos muy importantes para lograr el éxito del proyecto: la adecuación de los datos para el modelo y el ajuste de los hiperparámetros del sistema, esto es crítico y puede ser crucial en la obtención de los resultados deseados.

Es probable que el modelo de red neuronal profunda utilizado, posea más potencia de la necesaria para resolver el problema planteado en esta investigación. Quizás con algoritmos de Machine Learning como Regresión Lineal, Árboles de Decisión o Modelos Mixtos Gaussianos, por mencionar algunos, también se podría lograr de manera satisfactoria los objetivos, es decir, es factible que al haber utilizado las RNA se incurrió en el uso exagerado de la fuerza de este algoritmo para un problema que no requiere tanta potencia, pero se demostró su efectividad y adaptabilidad para este proceso, además del valor investigativo que se obtuvo al utilizar técnicas de vanguardia como lo son las Redes Neuronales Profundas.

Recomendaciones

Se sugiere proponer proyectos afines al presente trabajo que utilicen técnicas de Machine Learning, de esta forma pueden surgir comparaciones entre las bondades y conveniencias de utilizar diferentes algoritmos en diversos tipos de problemas.

Se recomienda el uso del presente modelo en otras áreas de la Universidad que requieran la evaluación de encuestas, con algunas modificaciones se haría factible su uso para resolver problemas planteados que sean afines a los de esta investigación.

Se sugiere el uso e investigación de las técnicas de RNA para futuros desarrollos ya que son de mucha utilidad y gracias al vertiginoso desarrollo del hardware de las computadoras se hacen posible desarrollar en la actualidad. Estas son técnicas que constituyen el corazón de la Inteligencia Artificial, utilizadas en reconocimiento de imágenes y patrones, voz, automatización de procesos, educación, ingeniería, conducción autónoma de vehículos, visión artificial y muchos otros. Se trata de una tecnología que llegó para quedarse y que se encuentra muy en boga en estos momentos, son más que suficientes las razones para embarcarse en este tren de conocimientos que seguramente es y seguirá siendo protagonista de la cuarta revolución industrial.

También se sugiere seguir con los proyectos de automatización de procesos en la Universidad de Carabobo, ya que pueden agilizar y modernizar el funcionamiento de dicha institución de cara al mundo 2.0, esto requiere de talento y dedicación, precisamente lo que abunda en las Universidades, aprovechar y desarrollar estas capacidades sería de gran beneficio para ambas partes, los estudiantes se capacitarían en las tecnologías que dominan el mercado para asegurarse un futuro laboral brillante y la institución se modernizaría para lograr prestar un mejor servicio educativo.

BIBLIOGRAFIA

Arias, F. (2012). *“El proyecto de Investigación”*. 4^{ta} Edición Caracas: EPISTEME, C.A.

Brownlee, J. (2016) *“Machine Learning Mastery with python Understand your data, create accurate models and Work Projects End-To-End”* Aprendizaje automático con Python, Comprenda sus datos, cree modelos precisos y proyectos de trabajo de extremo a extremo.

Burgos y Escalona (2015) *“Redes Neuronales Artificiales para predecir variables Antropométricas de trabajadores que laboran en postura decente”* Universidad de Carabobo. Valencia Venezuela.

Cazorla y Sander (2015) *“Reconocimiento de patrones mediante redes neuronales para la determinación del nivel de contenido calcio en la Zanahoria”* Universidad Gastón Dachary. Argentina

Colmejo, R. (2011) *“Redes Neuronales Artificiales aplicadas a la estadística y análisis de fundamentos”* Universidad De Chile

Esparza, Fuentes y del Real (2017) *“Un modelo basado en el Clasificador Naïve Bayes para la evaluación del desempeño docente”* Universidad Politécnica de Aguascalientes Ciudad de México. México.

Kaplan A., y Haenlein M. (2009) *“who's the Fairest in the Land? On the Interpretations, Illustrations and Implications of Artificial Intelligence”* Berlin Germany: Business Horizons

Hurtado y Toro (2010) *“Metodología de la Investigación Holística”* 6^{ta} Edición. Caracas: Fundación Sypal.

López, D. (2007) *“Diseño y construcción de una red neuronal artificial de propósito general”* Universidad Politécnica Salesiana. Quito Ecuador.

Méndez, C. (2009). *“Metodología: guía para la Elaboración de Diseños de Investigación en Ciencias Económicas, Contables y Administrativas”* Colombia: Editorial McGraw Hill.

Parella, S y Martins, F. (2012). *“Metodología de la investigación cuantitativa”* Caracas. Venezuela: FEDEUPEL

Rojas y otros (2011) *“Implementación de una Red Neuronal Multicapa en un Microcontrolador PIC para la Identificación de Color”* Instituto Politécnico Nacional. Puerto Vallarta, Jalisco. México.

Sabino, C. (2012). *“El Proceso de Investigación”* 3ra Edición. Caracas Venezuela: Panapo.

Santamaría y Rivero (2015) *“Detección de la dirección de arribo usando técnicas de Redes Neuronales”* Universidad de Carabobo. Valencia Venezuela.

Santoyo y González (2013) *“Comparación de predicción basada en redes neuronales contra métodos estadísticos en el pronóstico de ventas”* Universidad de Carabobo.

Tamayo, M. y Tamayo (2010) *“El proceso de la Investigación Científica”* 4ta edición. México: Limusa

Universidad Pedagógica Experimental Libertador. Upel. (2010). Manual de Trabajos de Grado de Especialización y Maestría y Tesis Doctorales. 4^{ta} Edición. Caracas: FEDUPEL.

ANEXOS

ANEXO A-1 ENCUESTA A LOS ESTUDIANTES DE SATISFACCIÓN

1) ¿El participante es informado por la dirección de postgrado acerca del proceso de selección de cada programa?

2) ¿El participante es informado por la dirección de postgrado acerca del proceso de inscripción de cada programa?

3) ¿La selección del personal docente responsable de la facilitación de los cursos en el programa de parte de las instancias directivas es idónea?

4) ¿El costo de la matrícula está ajustado a las posibilidades presupuestarias de un profesional en su área?

5) ¿El costo de los aranceles es adecuado?

6) ¿Los procedimientos establecidos para realizar los trámites son expeditos?

7) ¿La atención dispensada por funcionarios adscritos a las unidades administrativas está acorde a un público profesional?

8) ¿El tiempo que se realizan los trámites administrativos en una unidad está ajustado a las necesidades del participante?

9) ¿El trato de los funcionarios adscritos a cada unidad académico-administrativa es cortés y amable?

10) ¿La orientación ofrecida por los funcionarios adscritos a cada unidad es asertiva, consistente y oportuna?

11) ¿El horario de atención al público en las unidades administrativas esta adecuado a las actividades de docencia e investigación que se realizan en el programa?

12) ¿El horario de atención de la biblioteca está ajustado a las necesidades de documentación e información de los participantes?

13) ¿El horario de atención de control de estudios está ajustado a los requerimientos de documentación de los participantes?

14) ¿El proceso de inscripción automatizado en cada período lectivo es expedito y seguro?

15) ¿El tiempo de una automatizado en cada período lectivo?

16) ¿El tiempo de respuesta en la inscripción del trabajo de investigación es adecuado?

17) ¿Los procedimientos académico-administrativos correspondientes al trabajo de investigación se ejecutan en acuerdo a lo establecido en el Reglamento de estudios de postgrado?

18) ¿Los procedimientos inherentes a los procesos académico-administrativos se divulgan suficientemente?

19) ¿Es fácil el contacto con la Comisión Coordinadora del Programa?

20) ¿La respuesta de la comisión coordinadora ante una solicitud formulada es asertiva, oportuna y diligente?

21) ¿La Comisión Coordinadora informa sobre los pormenores académico-administrativos del Programa en una jornada de inducción?

22) ¿Los procedimientos establecidos para realizar trámites son conocidos por los participantes?

23) ¿La Comisión coordinadora del programa mantiene contacto permanente con los participantes?

24) ¿El organigrama organizacional es del conocimiento de los participantes para facilitar la ejecución de los trámites académico-administrativos?

25) ¿El horario establecido para el desarrollo de las actividades docentes y de investigación está ajustado a las necesidades del participante?

26) ¿El participante recibe información sobre sus deberes y derechos como estudiante de postgrado?

27) ¿El participante recibe de las unidades académico-administrativas los manuales de procedimiento para orientarle en la ejecución de los trámites correspondientes?

28) ¿La información suministrada por funcionarios adscritos a las unidades académico-administrativas es uniforme y consistente?

29) ¿El personal adscrito a las unidades académico-administrativas evidencia tener una formación especializada?

30) ¿Los criterios de selección previstos en el reglamento y aplicados para ingresar al programa están adecuados a la población que demanda los estudios?

31) ¿Las restricciones en el horario de atención en las unidades académico-administrativas favorecen la gestión de los procesos correspondientes?

33) ¿El participante de postgrado recibe de la institución la identificación que le acredita como tal durante sus estudios?