



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

ESCUELA NACIONAL DE ESTUDIOS SUPERIORES
UNIDAD LEÓN

PRONÓSTICO DEL ÍNDICE S&P500 DE LA BOLSA DE
VALORES DE NEW YORK, MEDIANTE EL ANÁLISIS DE
NOTICIAS DE REUTERS HACIENDO USO DE UNA RED
NEURONAL ARTIFICIAL (RNA)

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

LICENCIADO EN ECONOMÍA INDUSTRIAL

PRESENTA:

FRANCISCO JAVIER PEDROZA CASTRO



TUTOR:

DR. JOSÉ FRANCISCO REYES DURÁN

ASESORA:

DRA. ARELI VÁZQUEZ JUÁREZ

León, Guanajuato, 2020



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

A mi familia.

Agradecimientos

Agradezco a mi madre, quien siempre me ha apoyado tanto económicamente como emocionalmente, quien jamás me ha juzgado por ser una persona que se interesa por diversos temas, un gran pilar en mi vida; a mi padre y hermanas, quienes también me han apoyado; a mis sobrinos Leonel y Kida, que cuando llegaba de la universidad dejaban de jugar en la PC para permitirme hacer mi Tesis. A la Dra. Areli Vázquez Juárez, por su paciencia y dedicación en su enseñanza docente; pues ella me impartió la materia de Cálculo Vectorial, y gracias a esas enseñanzas pude entender con facilidad las Redes Neuronales Artificiales (RNA) y el algoritmo de BackPropagation, además, le agradezco su paciencia de leer y releer varias veces mi tesis y hacerme sus observaciones y correcciones, fue mucho el esfuerzo que hizo; pues su apoyo no fue únicamente en el área matemática, sino también en la redacción, ya que no soy un docto en ambas áreas. Agradezco a Israel Vázquez Juárez, quien a pesar de no haber sido mi profesor, y no ser profesor, de la ENES Unidad León de la UNAM, dedicó parte de su tiempo para revisar mi tesis y hacer sus observaciones. Al Dr. José Francisco Reyes Durán, por su tiempo. A la Dra. Adriana Martínez Martínez y el Mtro. Cristian F. Ruiz Aguado, por sus recomendaciones para mejorar la tesis. Al Dr. Juan Miguel Ruiz Zepeda, a quien en alguna ocasión me acerqué para preguntarle un problema muy específico de despeje del Perceptron, algo fundamental para poder comprender las RNA's, quizás él no lo recuerde, pero fue significativo para mí y por sus observaciones a mi tesis, que con ojos de halcón encontró errores incluso en la bibliografía. Agradezco también al Dr. Juan Martín Carpió Valadez, del Instituto Tecnológico de León, quien sin su ayuda, muy probablemente seguiría buscando como solucionar el problema de la falta de datos para mi tesis, pues él me orientó sobre los polinomios de Jacobi, y me permitió usar un programa que él mismo hizo para obtener polinomios de Jacobi.

Índice general

Agradecimientos	II
Introducción	1
Antecedentes	1
Planteamiento del problema y delimitación del tema	2
Justificación e hipótesis de trabajo	4
Objetivo general	5
Objetivos particulares	5
1 Marco Teórico	6
§1.1 Pronósticos económicos y financieros con información de noticias y redes sociales	6
§1.2 Pronósticos y análisis económico - financieros con algoritmos de inteligencia artificial	15
§1.3 Crítica a los algoritmos computacionales aplicados en economía y finanzas	26
§1.4 Teoría de la Inteligencia Artificial (IA)	29
§1.4.1 Breve historia de la inteligencia artificial	30
§1.4.2 El método de la inteligencia artificial	32
§1.4.3 Flujo de trabajo de Machine Learning, etapas del proceso de Big Data y Knowledge Discovery in Database (KDD)	32
§1.4.4 Predicción y adaptación	38
§1.4.5 Algoritmos de clasificación	39
§1.4.6 Tipos de redes neuronales artificiales	40
§1.4.7 Validación	49

<i>ÍNDICE GENERAL</i>	IV
§1.4.8 Overfitting y Underfitting	50
§1.5 Polinomio de Jacobi	51
2 Metodología	53
§2.1 Tipo de investigación	53
§2.2 Objeto de estudio	53
§2.3 Modelo de clasificación de variaciones positivas y negativas	54
§2.4 Pasos	56
§2.4.1 Crear base de datos	56
§2.4.2 Pre - procesamiento de datos	58
§2.4.3 Entrenamiento de Red Neuronal Artificial y prueba de diferentes arquitecturas	58
§2.5 Alcances y consideraciones	59
3 Análisis y Resultados	62
4 Discusiones	79
§4.1 Interpretación	79
§4.2 Conclusiones	79
§4.3 Consideraciones	81
§4.4 Recomendaciones	82
Bibliografía	83
Anexos	95
A Anexo I: Python	95
§A.1 Python	95
§A.1.1 Variables en Python	95
§A.1.2 Listas en Python	96
§A.1.3 Tuplas en Python	96
§A.1.4 Diccionarios	96
§A.1.5 If statements en Python	97

§A.1.6 While Loop en Python	97
§A.1.7 For Loop en Python	98
§A.1.8 Funciones en Python	98
§A.1.9 Clases en Python	98
B Anexo II: Glosario	100
C Anexo III: Obtención de los datos	104
D Anexo IV: Modelo entrenado con todos los datos	106
E Anexo V: Link para descargar modelo	108

Índice de gráficas y diagramas

2.1 Diagrama de flujo de trabajo.	61
3.1 Error de Modelo 22.	64
3.2 Validación de Modelo.	65
3.3 Error de Modelo 1.	66
3.4 Validación de Modelo 1.	66
3.5 Error de Modelo 2.	67
3.6 Validación de Modelo 2.	67
3.7 Error de Modelo 3.	68
3.8 Validación de Modelo 3.	68
3.9 Error de Modelo 4.	69
3.10 Validación de Modelo 4.	69
3.11 Error de Modelo 5.	70
3.12 Validación de Modelo 5.	70
3.13 Error de Modelo 6.	71

3.14 Validación de Modelo 6.	71
3.15 Error de Modelo 11.	72
3.16 Validación de Modelo 11.	72
3.17 Error de Modelo 13.	73
3.18 Validación de Modelo 13.	73
3.19 Error de Modelo 14.	74
3.20 Validación de Modelo 14.	74
3.21 Error de Modelo 17.	75
3.22 Validación de Modelo 17.	75
3.23 Error de Modelo 19.	76
3.24 Validación de Modelo 19.	76
3.25 Error de Modelo 20.	77
3.26 Validación de Modelo 20.	77
3.27 Error de Modelo 21.	78
3.28 Validación de Modelo 21.	78
4.1 Propuesta de Flujo de Trabajo.	84
D.1 Error cuadrático medio (modelo 23).	107
D.2 Validación (modelo 23).	107

Introducción

En la presente introducción se expone: los antecedentes, delimitación del tema, planteamiento del problema, justificación e hipótesis del trabajo y, por último, objetivos.

Antecedentes

En la actualidad la inteligencia artificial (IA) ha tenido un gran auge al solucionar problemas complejos, por ejemplo: regular la temperatura de un reactor nuclear usando redes neuronales artificiales (Viñuela & Galván, 2004); reconocimiento de imágenes para la discriminación entre pacientes con cáncer y los que no lo tienen; generación de texto (tan realista como el caso de la empresa OpenAI quien decide no liberar un modelo entrenado de generación de texto con IA, para evitar el uso indebido de la IA (OpenAI, 2019)); entre otros. Todo esto gracias a la capacidad de cómputo y el Internet. Este último es destacable, ya que genera una gran cantidad de datos que en varias ocasiones son utilizados como bases de datos para entrenar inteligencias artificiales. Por ejemplo, el *Mannquin Challenge* (reto que consistía en grabar una escena donde todos los participantes fingen ser un maniquí) usado para encontrar la profundidad en imágenes (Freeman, *et al.*, 2019), algo que para un humano es muy sencillo, pero para una máquina, no; del mismo modo la inteligencia artificial de OpenAI que se mencionó anteriormente, fue entrenada con datos generados en Internet (OpenAI, 2019). Ahora, en esta tesis se busca realizar un pronóstico en economía usando inteligencia artificial específicamente el subcampo de redes neuronales artificiales, ya que actualmente los modelos usados por los economistas se basan en supuestos que no siempre empatan con la realidad; mientras que las redes neuronales artificiales supervisadas no se basan en supuestos *a priori*, sino que ellas buscan en el espacio del error la función

que mejor se ajuste a los datos.

El presente trabajo se inspira en dos obras: El libro de Viñuela, & Galván (2004) llamado *Redes Neuronales Artificiales, Un Enfoque Práctico*, Madrid, Pearson Educación, S.A., en este libro se expone la teoría matemática de las redes neuronales artificiales (RNA) supervisadas como no supervisadas, donde menciona que las RNA tienen la capacidad de adaptarse a las situaciones de ruido, y las cuales se han usado para aproximar funciones y pronosticar, por ejemplo el nivel del mar en una playa, entre otros problemas. La segunda obra es el artículo de Gómez (2013), llamado “*Señales de inversión basada en un índice de aversión al riesgo*”, Investigaciones Europeas de Dirección y Economía de la Empresa, Vol. 19, pp. 147 - 157, Madrid, España. En este trabajo se hace un Índice de aversión al riesgo usando búsquedas en Google; con lo cual disminuye tiempos y costos, elementos que tienen peso en cualquier industria. Estas dos obras y el contexto mencionado en el párrafo anterior, estimulan la presente tesis para realizar una investigación donde se encuentra la capacidad de cómputo, los datos generados en Internet, la inteligencia artificial y el análisis económico - financiero.

Planteamiento del problema y delimitación del tema

Actualmente existen diversos métodos de análisis matemático para analizar los mercados financieros, no obstante la mayoría de ellos, como: el análisis de regresión, Black-Scholes, AR, MA, ARIMA, ARMACH; son modelos con supuestos que no empatan con la realidad económica, supuestos como: *random walk*, información perfecta, mercados eficientes, entre otros; tales modelos son aplicados en un contexto donde no se sigue un paseo aleatorio, no se encuentra estacionalidad, existe ruido, no - linealidad, e información imperfecta (Parisi *et al.*, 2007: 791; Bao & Yang, 2008: 1; León *et al.*, 2018: 604-605). Además, tradicionalmente, los análisis económicos se basan en análisis de datos históricos y estructurados, mientras que, por otro lado, hoy en día existen datos, no estructurados, como puede ser un texto de algún blog, o los comentarios realizados a productos de alguna página web, que en varias ocasiones son generados por las tecnologías de la información; por lo que, en contraste con las observaciones anteriores, existen diferentes modelos de

inteligencia artificial (IA), sus sub - áreas como el *Machine Learning*, técnicas de *Big Data* y pre - procesamiento de datos; con las cuales se pueden analizar datos no estructurados, adaptarse a los datos, y presentar características de no - linealidad, por lo que pueden ser una alternativa para el análisis financiero (e incluso económico), ya que dichas técnicas tienen la capacidad de ajustarse a situaciones de ruido, no estacionalidad (como se destaca renglones atrás) y generar modelos no lineales (incluso lineales) (Viñuela & Galván, 2004). Por otro lado, haciendo énfasis en los análisis clásicos con datos históricos, y en contraste, tenemos, hoy en día, la posibilidad de hacer análisis y pronósticos financieros, con datos como *tuits* de Twitter (ellos se consideran datos no estructurados) (Bollen *at el.*, 2010), incluso poder pronosticar la situación económica de EU con búsquedas de Google (Varian, 2009); toda esta información puede ser analizada con los algoritmos de inteligencia artificial, gracias a diferentes métodos de representación de información y capacidad de cómputo existente. Varios de estos algoritmos tienen la capacidad de encontrar patrones en los datos que difícilmente un humano puede detectar, o que métodos estadísticos clásicos, como regresión simple, difícilmente pueden encontrar.

Por lo tanto, la presente investigación pretende determinar si es posible realizar una red neuronal artificial (RNA) que pueda pronosticar el índice S&P500 con noticias del portal Reuters (<https://www.reuters.com/>).

Ahora bien, la elección del objeto de estudio, el Índice S&P500, se debe a su disponibilidad y fácil acceso a la información. Ya que al realizar la presente tesis se intentó hacer con el INPC mexicano, no obstante el acceso a esta información es más difícil y de elevados costos, ya que si bien la proporciona el Banco de México, la ofrece de pocos años, y en esta tesis se ocupa un periodo largo para tener varios ejemplos y entrenar la RNA. En cuanto a la elección del portal, se seleccionó por la sencillez de generar un programa que automatizara las descargas de más de 50 mil noticias en comparación de otros, que en ocasiones requiere suscripción. También se decidió hacerlo en inglés debido a la gran cantidad de paquetes que ofrece Python para el procesamiento de lenguaje natural en inglés; mientras que en español existen pocos y con errores de procesamiento del lenguaje.

Por otro lado, el periodo de tiempo seleccionado fue de 2013 a 2019. El periodo se seleccionó porque el portal de Reuters no proporcionaba más información de periodos

anteriores a 2013. También solo se seleccionó el apartado de noticias económicas de Estados Unidos, ya que tomar todas las secciones que ofrece el portal implicaría aplicar modelos de Inteligencia Artificial y/o pre - procesamiento de datos más avanzados para que la RNA puede realizar el pronóstico.

Las herramientas utilizadas en la presente investigación fueron, beautifulsoup4, la cual sirvió en el proceso de creación de la base de datos, dicha herramienta extrae el texto del HTML de la página de Reuters. Para el procesamiento de los datos, las herramientas usadas fueron: Pandas, para el manejo de tablas; NLTK, para el Procesamiento de Lenguaje Natural; entre otras herramientas. La Red Neuronal Artificial fue diseñada con Keras; Keras es un *API* para realizar pruebas y entrenamiento rápido de diferentes arquitecturas de redes neuronales. Cabe mencionar que el lenguaje que se usó fue Python (las herramientas mencionadas con anterioridad funcionan con Python, a excepción de HTML), el cual es un lenguaje de programación de alto nivel, y por tanto, sencillo de usar en comparación de otros lenguajes de bajo nivel.

Justificación e hipótesis de trabajo

Tener modelos que se ajusten a la realidad son menester para tener pronósticos con un mayor grado de precisión, con esto deviene mejor toma de decisiones y por tanto disminuye la incertidumbre y/o riesgo de tales decisiones. Hoy en día los métodos estadísticos y econométricos tradicionales, no se ajustan con precisión al análisis de mercados, ya que se basan en supuestos *a priori*, por ejemplo, la regresiones lineal, la linealidad, heterocedasticidad, entre otros, lo que puede dar como resultado pronósticos erróneos. Por otro lado en un contexto donde la información que se genera en Internet y la capacidad de cómputo que permite el uso de Inteligencia Artificial, así como sus subconjuntos (*Machine Learning* [ML], por ejemplo), proveen las condiciones que abren las puertas para generar e implementar modelos con mayor grado de precisión, ya que, entre las herramientas de ML, existen las redes neuronales artificiales, que trabajan bajo la premisa del desconocimiento de la función, y a partir de los datos buscan, en el espacio del error, la función que mejor se ajustada a los datos; lo que puede generar análisis más precisos.

Cabe destacar, además de las características mencionadas en el párrafo anterior, que las RNAs, tienen la capacidad de procesar elementos de la realidad, como el lenguaje natural, entre otras abstracciones de la realidad, lo que en contraste, tales elementos no pueden ser procesados, por ejemplo, con facilidad, con una regresión lineal. Por lo que, como **hipótesis** se propone que: se puede pronosticar la variación positiva y negativa del Índice S&P500, con noticias del portal Reuters (<https://www.reuters.com/>), con un retraso de una hora en el tiempo, haciendo uso de una RNA, ya que éstas pueden procesar información como el lenguaje natural y, buscar la función que mejor se adapte a los datos del fenómeno económico o financiero.

Por otro lado, cabe preguntarse ¿por qué utilizar la información generada en Internet?, pues la información que se genera en Internet en su mayoría puede ser libre, gratuita, y en varias ocasiones generada automáticamente, por lo que acceder a ella es relativamente económica en comparación, por ejemplo levantamiento de encuestas u observación. Además, todos estos datos generados en la Internet, han llegado a sumar grandes volúmenes, y dicha información abre las posibilidades del uso de modelos matemáticos de Inteligencia Artificial para el tratamiento y análisis de los mismos, como es el *Machine Learning* (Varian, 2014).

Objetivo general

Realizar un modelo de pronóstico de Índice S&P500, haciendo uso de una Red Neuronal Artificial, que interpreta el texto de las noticias de Reuters para realizar dicho pronóstico.

Objetivos particulares

- Descargar las noticias de Reuters.
- Pre - procesar los datos para que la red neuronal artificial pueda procesar el texto.
- Generar los datos por horas de S&P500 a partir de los datos de cierre por día.
- Entrenar la red neuronal artificial para realizar el pronóstico.

Capítulo 1

Marco Teórico

En este capítulo se hará una revisión de diferentes investigaciones que han realizado estudios de análisis económico y financiero usando inteligencia artificial, así como análisis con datos no muy usados para el análisis económico y/o financiero, como búsquedas en Google, información de Twitter, entre otras. En las últimas secciones se revisará el modelo del perceptron multicapa, el cual es un modelo de redes neuronales artificiales, y por último una breve revisión del modelo de polinomio de Jacobi.

1.1. Pronósticos económicos y financieros con información de noticias y redes sociales

Feuerriegel & Gordon (2019), realizaron un pronóstico de diferentes indicadores macroeconómicos con noticias *ad hoc* de Alemania (Feuerriegel & Gordon, 2019), en su búsqueda de trabajos de pronósticos con noticias de indicadores macroeconómicos previos, al respecto fueron nulos, más no en el campo financiero (Feuerriegel & Gordon, 2019), con lo cual ellos proponen su propio método usando modelos de *Machine Learning*, y representación de lenguaje natural. En su trabajo, hicieron pronósticos para diferentes indicadores macroeconómicos: PIB de la Euro zona, PIB de Alemania, desempleo, inflación, producción industrial, confianza empresarial, confianza del consumidor y, bonos gubernamentales. Los resultados obtenidos, comparándolos con modelos clásicos como series de tiempo, dieron

resultados de precisión superiores a los modelos de pronósticos clásicamente utilizados por los economistas. De acuerdo Feuerriegel & Gordon, (2019), ello abre las puertas a nuevos métodos de análisis. Realizar estudios económicos a nivel macroeconómico implica costos elevados. Sin embargo los costos del trabajo de Feuerriegel, & Gordon (2019) fueron bajos, debido al aprovechamiento de la “*era de los grandes datos*” (Feuerriegel, & Gordon, 2019: 163), *hardware* y algoritmos (capaces de analizar grandes volúmenes de datos, tanto los datos estructurados como no estructurados). Al aprovechar dichas tecnologías deviene estudios menos costosos, en cuanto tiempo y dinero. En el estudio de Feuerriegel & Gordon (2019), cabe destacar que los pronósticos de mayor precisión fueron los de largo plazo, es decir, periodos de más de 10 meses, en contraste con los de corto plazo, los cuales no fueron satisfactorios, como lo es en la tasa de desempleo e inflación.

Cabe resaltar que Feuerriegel & Gordon (2019) realizan la investigación bajo la premisa de que los mercados fluctúan por las expectativas, de tal modo que las noticias influyen en las expectativas de los agentes económicos, y por tanto influyen sobre en los diferentes mercados ya mencionados, PIB de la Euro zona de Alemania, Confianza del Empresario, entre otros. Otro punto a destacar en el trabajo de Feuerriegel & Gordon (2019), es considerar el problema de pronóstico como un problema de clasificación, la visión, con la que se emprendió esta tesis.

De acuerdo Chen, *et al.* (2019) los inversionistas toman decisiones con información de datos, visualización de los mismos, y noticias, ya que dicha información les ayuda a predecir tendencias en función de su experiencia pasada. Chen, *et al.* (2019) emprenden su investigación complementando con trabajos de dos autores Guo (2017) y Lio (2018), el primero indica que los sentimientos de los inversores influyen en su toma de decisiones, el segundo realiza minería de texto para la clasificación de sentimientos de los textos analizados. Por lo que Chen, *et al.* (2019) realizan su pronóstico financiero de acciones con análisis de sentimientos mediante noticias; las noticias las obtuvo de diversos sitios web usando Google. Chen, *et al.* (2019) realizan su investigación con diferente tipo de acciones financieras. En su trabajo primero realizó un proceso de clasificación de los sentimientos y luego con ello el pronóstico financiero, haciendo uso de una Red Neuronal Artificial Recurrente, conocida como *Long Short-Term Memory* (LSTM). Para tal trabajo Chen, *et al.*

(2019) realizaron un trabajo de pre - procesamiento de los datos, ya que las noticias son, al igual que en trabajos de Feuerriegel, & Gordon, (2019), datos no estructurados. Una de las herramientas a destacar en el trabajo de Chen, *et al.* (2019) en el pre - procesamiento de texto se usó de *Word2vec* (*Wordembeddings*), que es un tipo de Red Neuronal Artificial que busca agrupar en un espacio vectorial palabras que son similares y/o estén relacionadas, guardando el sentido y significado de las palabras; esto facilita una parte importante del pre - procesamiento de los datos. Esta estructura se usó en el presente trabajo. El pronóstico realizado obtuvo un precisión de entre 70 % y 80 % para diferentes acciones aproximadamente.

En otro trabajo realizado por Chen & Chen (2019), poco antes de la investigación de Chen, *et al.* (2019), realizaron una investigación similar para la bolsa de Taiwan (tomaron las 50 empresas más grandes que cotizan en ella) como variable dependiente (*output*), y variable independiente (*input*) noticias de diversos portales de noticias populares como *ChinaTime.com*, *cnYES.com*, y Google, para el análisis usaron un flujo de trabajo muy parecido al *Knowledge Discovery in Data Base* (KDD); los resultados de pronóstico para diferentes acciones varían entre 30 % y 59 %. Lo que cabe destacar en la investigación de Chen & Chen (2019) fue que al probar un conjunto de datos de mayor léxico, es decir: una mayor cantidad de palabras, aumentó la precisión del análisis.

Fang, *et al.* (2018), analizaron los mercados financieros de la bolsa de China. Tomaron como variable independiente los *tuits* de la red social de inversores chinos, llamada Xueqiu, y como variable dependiente diversas acciones de la bolsa de China. En su estudio identificaron sentimientos usando *Naive Bayes*, y para el problema de clasificación de los movimientos de las acciones usaron *Support Vector Machines* (SVM) y Redes Neuronales Artificiales. Los resultados de su trabajo fueron satisfactorios, realizando dos modelos, uno sin incluir los sentimientos y otro que los incluye, logrando pronosticar los movimientos de las acciones, teniendo mayor precisión añadiendo los sentimientos.

Guo, *et al.* (2016), en su revisión de antecedentes, encuentran que existen varias investigaciones sobre el análisis de texto y análisis financiero, de las cuales varias encuentran una relación predictiva de textos - prensa sobre los mercados financieros, por ejemplo noticias positivas o negativas, pueden pronosticar una alza o baja de la bolsa, respectivamente,

o noticias diarias y semanales, pueden tener capacidad predictiva de corto y largo plazo, respectivamente. De la extensa literatura revisada por Guo, *et al.* (2016), identifican tres modelos ampliamente usados para el análisis de texto en finanzas y contabilidad, *Naive Bayes*, *Support Vector Machines* (SVM) y *Redes Neuronales Artificiales* (RNA). Con tales métodos realizan experimentos de clasificación, llegando a la conclusión de que las RNA, o Redes Neuronales Artificiales superan a las otras técnicas de *Machine Learning* en clasificación de texto.

“*What would you search for if you thought you might lose your job? Typical searches might be queries like [unemployment office], [apply for unemployment], [jobs], and so on.*” (Choice & Varian, 2009: 1), así inicia su artículo Choice & Varian, titulado “*Predicting Initial Claims for Unemployment Benefits*”; ya que, ellos, versan su artículo bajo el supuesto de que el clasificador de búsquedas *Google Trend* puede arrojar información para predecir los reclamos de beneficios de desempleo en Estados Unidos; “*el ‘Initial Jobless Claims’ es un reporte semanal que emite el Department of Labor*” (Choice & Varian, 2009: 1) de Estados Unidos; el cual es, considerado, un informe representativo del mercado laboral (Choice & Varian, 2009: 1). Choice & Varian (2009) aplicaron un modelo ARIMA con la información de búsquedas en Internet, con lo cual lograron pronosticar los reclamos de beneficios de desempleo de EU. Además de eso, Choice & Varian (2012), han realizado otras investigaciones donde han encontrado que las búsquedas en Internet (*Google Trend*), pueden ser usadas para pronosticar otros indicadores económicos como venta de automóviles, viajes y confianza del consumidor (Choice & Varian, 2012).

Por otro lado, Bollen *et al.* (2010) hacen uso de las representaciones de alto nivel, analizando el estado de ánimo (clasificándolos en seis estados diferentes) de los mercado a través de los *tuits*, con el objeto de pronosticar el *Down Jones Industrial Average* (DJIA). Bollen *et al.* (2010), realizaron dos modelos: *Granger causality* (Método Lineal) y *A Self-organizing Fuzzy Neural Network* (SOFNN) (Método no-lineal). En el último modelo se probaron tres sub - modelos, el primero implementando los datos del estado de ánimo, y de los otro dos: el primero con un retraso en el tiempo, más el estado de ánimo, utilizando 6 diferentes estados, (*Calm, Alert, Sure, Vital, Kind, Happy*). En el otro modelo se realizó con dimensiones de estado de ánimo binarias (Positivo/negativo). De la misma manera, los

sub - modelos fueron realizados con *granger causality*. Bollen *et al.* (2010) no encontraron predicciones acertadas para las dimensiones binarias de estado de ánimo para el pronóstico del DJIA; por otro lado, en los modelos de 6 dimensiones, se encontró un pronóstico más acertado para el DJIA, con la variable *Calm*, y, ligeramente, con la variable *Happy*. La variable *Calm*, tuvo una correlación positiva y significativa con tres retrasos en el tiempo. Lo cual, quiere decir: que puede pronosticar con tres días de anticipación el cambio en la bolsa de valores, con el estado emocional *Calm*, el cual es el estado emocional obtenido de los *tuits*. Cabe agregar que Bollen *et al.* (2010) basan sus pronósticos de DJIA usando los *tuits*, bajo el precepto de que un número agregado de *tuits* proporciona el estado de ánimo general de los mercados. Bollen *at el.*, (2010), crearon sus propia herramienta de análisis de emociones de *tuits*, GPOMS; además usaron una herramienta gratuita para el mismo fin, OpinionFinder.

Otro trabajo similar para pronósticos haciendo uso de los *tuits* de *Twitter*, fue el de Bing Li *et al.* (2017), en el cual al igual que Bollen *et al.* (2010), primero realizan una clasificación de sentimientos de los *tuits*, para ello realizar un pre - procesamiento de 200 millones de datos, donde agrupa palabras similares en un espacio vectorial, después obtuvieron las palabras que son más significativas en el texto, entre otros pasos realizados, para luego clasificar los *tuits* en sentimientos, con el objeto de usar las clases para realizar el pronóstico de diferentes sectores económicos desde tecnologías de la información, hasta medicina. En total fueron 30 empresas de las cuales lograron pronosticar el precio de las acciones de la mayoría de las empresas de los diferentes sectores industriales, teniendo mayor precisión en la industria de tecnologías de la información. En su trabajo hicieron uso de SMeDA-SA, un modelo propuesto por los autores, de acuerdo con ellos, novedoso; además Li *et al.* (2017) hicieron uso de *Support Vector Machines* (SVM) y *Naive Bayesian Classifier*; el uso de estas herramientas tienen como objetivo, en la investigación de Bing Li *et al.* (2017), compararlas con su modelo. En contraste sus resultados fueron superiores. Cabe agregar, que una de las herramientas usadas por Li *et al.* (2017) fue MongoDB, una sistema NoSQL, es decir un sistema de administración de datos no relacionales. Al tener conocimiento de éste tipo de trabajo, se considera, únicamente, realizar esta investigación con una menor cantidad de datos, y sólo un activo financiero, el cual es el Índice de S&P500

de la bolsa de Nueva York y guardar la información en una base de datos SQLite.

Gómez (2013) realiza un trabajo similar al de Bollen *et al.* (2010). Él creó un Índice de Aversión al Riesgo, haciendo uso de la información de las estadísticas de búsquedas en la Internet. Si bien, medir el estado de ánimo (confianza) de los consumidores e inversionistas puede realizarse mediante encuestas, ello, es un método lento y costoso (Gómez, 2013) . Por lo que Gómez (2013) propone realizar un Índice de Aversión al Riesgo mediante las búsquedas en Google, utilizando *Google Insights* (cabe destacar, que a día de hoy, *Google Insights* se encuentra fuera de servicio, y es sustituida por *Google Trends*, el cual ofrece la misma información estadística), considerando que ciertas palabras buscadas en Google reflejan el estado de ánimo, de acuerdo al modelo realizado por Gómez & Prado (2010), en su artículo “*Sentimiento del inversor, Selección Española de Fútbol y su influencia sobre el Ibox 35.*” XXI Congreso Internacional AEDEM, Bucarest. Gómez (2013) utilizó una lista de palabras que “*fueron máximas o mínimas en momentos coincidentes con algún evento económico o financiero relevante*” (Gómez, 2013:149), como las búsquedas realizadas en el año 2008, año en el cual estalló la crisis hipotecaria de Estados Unidos de Norteamérica; por lo cual, la lista de palabras que Gómez (2013) seleccionó califica relevantes; para elegir dichas palabras, Gómez (2013) considero 4 idiomas: inglés, español, francés y portugués; de acuerdo al trabajo realizado por George Weber en 1997 (Gómez, 2013: 149), quien hace un estudio de los idiomas con más hablantes. En la tabla 2.1 se muestra una lista de palabras seleccionadas en los idiomas mencionados.

De acuerdo con Gómez (2013) el estado de ánimo incide en los índices bursátiles, el cual puede ser medido con encuestas, sin embargo, el desarrollo de las tecnologías de información, y de Internet, da la posibilidad de usar las búsquedas en Google, por su costo bajo y rápido acceso. Ello le permitió a Gómez (2013) realizar un Índice de Aversión al Riesgo. Para realizar el índice usa las búsquedas realizadas en Google; con ello Gómez (2013: 156), “*analizó la rentabilidad esperada*” (2013: 156) proveyendo de información sobre las “*rentabilidades positivas para las señales de compra y negativas para las señales de venta*” (2013: 156), con lo que provee información de compra-venta. Gómez (2013) con las variables (conjunto de palabras) mencionadas en la tabla 2.1, formó un modelo de regresión lineal múltiple, con lo que obtuvo su Índice de Aversión al Riesgo; las búsquedas de Google

Instituciones y mercados financieros	Productos financieros	Análisis y coyuntura económica
Stock market	Shares	Analisis tecnico
Stock exchange	Acciones	Technical analysis
Bolsa de valores	Securities account	Analyse tecnica
Bourse	Cuenta de valores	Analyse technique
Wall Street	Securities	Analisis fundamental
Money Market	Bonds	Dundamental analysis
Mercado monetario	Bonos	Analyse fundamental
Marche monetaire	Bbligaciones	Analyse fondamentale
	Bund	Bull market
	Mutual funds	Mercado alcista
	Fondos de inversion	Marche haussier
	Fundos mutuos	Bear market
	Fonds communs de placement	Mercado bajista
	Etf	Marche baissier
	Hedge funds	Economic recession
	Derivatives	Recesion economica
	Derivados	Recession economique
	Derives	Recessao economica
	Dhort selling	Economic growth
	Ventas en corto	Desarrollo economico
	Vente a decouvert	Croissance economique
	Cenda a descoberto	Crescimento economico
	Securities lending	Crisis economica
	Prestamo de valores	Economic crisis
	Pret de titres	Crise economique
	Emprestimo de titulos	Crise economica
	Contract for differences	Crisis financiera
	Contratos por diferencias	Financial crisis
	Cfd	Crise financiere
	Commodities	Crise financeira
	Euro value	Stock market crash
	Cotizacion euro	Crash bursatil
	Dollar value	Crash da bolsa
	Cotizacion dolar	Krach boursier
		Liquidez
		Liquidity
		Liquidite
		Volatility
		Volatilidad V
		Olatilite
		Volatilidade

Tabla 1.1: Listado de palabras usada por Gómez (2013:150).

como variables independientes y el índice Eurostoxx 50, como variable dependiente; los resultados de varios experimentos, usando por ejemplo retrasos y, eliminando variables, obtuvo un resultado, en su mayoría, con las diferentes pruebas, un índice de correlación alrededor del 90 % (para sus diferentes modelos).

Existen otras investigaciones donde se hace uso de la información de Internet para el pronóstico de los sentimientos, por ejemplo el uso de *microblogging*. Al día de hoy el uso de *microblogging* es sumamente popular, lo cual proporciona la posibilidad de usar, la información generada, en esta actividad, para Minería de Datos, o análisis de datos para diferentes actividades, como lo realizó Pak & Paroubek (2010), quienes hicieron un trabajo de clasificación de sentimientos, mediante dos tipos de clases, clase positiva y negativa. La información fue obtenida de *Twitter*, haciendo uso de su API (*Application programming interface*) oficial; el proceso de clasificación fue usando el método *Multinomial Naive Bayes*, entrenado con una Máquina de Soporte Vectorial (*Machine Learning* para clasificación) (Pak, A. & Paroubek, P. 2010: 1323-1324). Pak, A. & Paroubek, P. (2010) concluyen que los autores de los *tuits* utilizan estructuras semánticas que describen sentimientos. Con la información de *tuits*, los autores mencionados anteriormente logran clasificar sentimientos positivos, negativos y, neutros. Estas investigaciones dan cabida a la posibilidad de crear diferentes índices que midan los sentimientos del consumidor, su aversión o no aversión al riesgo, similar al trabajo de Gómez (2013), con los cuales tomar de decisiones de menor riesgo.

Edmans *et al.* (2007) bajo el conocimiento de investigaciones que aseveran que los resultados de deportes influyen en el estado de ánimo encuentran en su investigación que los mercados reaccionan de manera negativa ante la pérdida de los equipos nacionales.

En otras investigaciones, Schmidt & Vosen (2011) encuentran que el agregar información de búsquedas en Internet (*Google Insights*) a los pronósticos económicos, en su mayoría, puede mejorar el desempeño de los modelos de pronósticos, e incluso mejora, aún más, si a esto se añaden datos cuantitativos.

Cardona & Rojas (2017) realizaron un trabajo de pronóstico de la tasa de desempleo en Colombia. Su pronóstico lo realizan haciendo uso de las búsquedas en *Google Trend*, para ello implementaron tres modelos: Regresión lineal, ARIMA, y ARIMAX; su trabajo obtuvo

resultados satisfactorios con una serie de palabras y cadenas de texto, especialmente con la palabra “trabajo”.

De acuerdo con Piotr & Lambe (2011) las noticias tienen un efecto de influencia sobre los rendimientos de los activos financieros, ya que en su estudio, realizado con datos de la crisis subprime, o también conocida como crisis hipotecaria, encontró con un análisis de Autorregresión Vectorial, que los periodistas causaron los rendimientos de la banca, es decir, los noticieros no sólo informaron del fenómeno económico, sino que también contribuyeron a acentuar la crisis; ya que sus noticias afectan los sentimientos de los inversionistas; ideas que empatan con las de Keynes, (2003), quien asegura que las tasas de interés también son afectadas por las noticias.

Hernández (2007) en su tesis de maestría busca si es factible o no,

“incorporar el sentimiento de mercado utilizando ‘colonias de hormigas’ para la administración de portafolios y que la incorporación de éste pueda beneficiar la eficiencia de la administración de portafolios” (Hernández, 2015: 7)

de acuerdo a su investigación, dicha hipótesis fue correcta. El modelo de colonias de hormigas de Hernández (2007), fue complementado y comparado con el modelo de Markowitz; el modelo de colonias de hormigas de Hernández (2015) hace una clasificación de las tendencias del mercado apoyado en los sentimientos del mercado. Hernández (2007) incorpora los sentimientos de mercado a su modelo de colonias de hormiga usando la variable VIX (*Chicago Board Options Exchange Market Volatility Index*) como variable de medición de los sentimientos de mercado; el VIX es conocido como Índice de Volatilidad del CBOE, “*el cual mide las expectativas del mercado en el plazo cercano reflejado por los precios de las opciones sobre el S&P 500, considerado como un barómetro de sentimiento de mercado*” (Hernández, 2007: 25). Actualmente el VIX es un modelo para estimar la volatilidad esperada, basado en el Index S&P500, como refiere Hernández (2007: 25), en el 2003, el índice se realiza agregando los pesos de los precios ponderados del S&P500 de las *call options* y *pull options* a un rango amplio de *stock price* y para el 2014 se le añade el S&P500 *Weeklys* (CBOE, 2019:3). VIX es un indicador, también, conocido como “*fear gauge*” (CBOE, 2019: 3). En la última etapa de la investigación de Hernández (2007) hizo una simulación donde obtuvo un rendimiento mayor, en comparación de los modelos

de Markowitz. Cabe aclarar que tal resultado fue sin añadir los sentimientos de mercado; después, añade los sentimientos, donde tiene un rendimiento de portafolio de, citando: “2.3 % al incorporar el sentimiento del mercado” (Hernández, 2007: 60). Este resultado fue haciendo un re-balanceo del portafolio, considerando los costos de transacción, y a pesar de tener una mejora en los beneficios respecto al modelo de Markowitz, el rendimiento fue menor a lo esperado en la investigación de Hernández (2007).

Siguiendo las ideas de Piotr & Lambe (2011) y Feuerriegel & Gordon (2019), quien dice que las noticias tienen efectos sobre los mercados, las ideas de Choice & Varian (2009), quienes en sus estudios pronostican las ventas de carros, reclamos de subsidios gubernamentales para desempleados y Cardona & Rojas (2017), que en su trabajo logra medir el comportamiento del mercado de trabajo con las búsquedas en la Internet; de esas investigaciones podemos concluir que la información generada en noticias, *twits*, y blogs en Internet, es usada para generar pronósticos, así como también, aquella información afecta en distintos mercados.

La información que nos proporciona esta revisión de las investigaciones mostradas anteriormente, nos da indicios para poder tener un mayor acercamiento a la veracidad de nuestra hipótesis.

1.2. Pronósticos y análisis económico - financieros con algoritmos de inteligencia artificial

En esta sección se hace una revisión de diferentes investigaciones donde se hace uso modelos de inteligencia artificial, aplicados en diferentes campos de la economía desde la economía como tal, hasta las mismas finanzas, objeto de estudio principal de esta investigación. Estos estudios implementan diferentes modelos de inteligencia artificial (o conjuntos de dicho campo) como: redes neuronales artificiales, algoritmos genéticos, métodos aplicados al análisis del genoma humano, modelado de agentes inteligentes como: simulación de enjambres de abejas para asignación de porcentajes de activos en portafolios de inversión, y simulación de agentes en la bolsa de valores, como inversionistas, especuladores, entre

otros; máquinas de soporte vectorial, modelos de ARIMA, AR, ARCH, combinados con redes neuronales artificiales, o algoritmos genéticos, entre otros.

Iniciando, Varian (2014) sugiere a sus alumnos de economía que se introduzcan a las Ciencias de la Computación y adquirir conocimiento de *Machine Learning*, ya que, desde su punto de vista, el hecho de que se junten estas dos ciencias proveerá resultados fructíferos. Porque, a diferencia de las herramientas usadas comúnmente por economistas que se enfocan a problemas de regresión, buscando la predicción de puntos específicos, con modelos *a priori*, mientras que, por otro lado, las herramientas de *Machine Learning*, como las redes neuronales artificiales, realizan la búsqueda de la función que mejor realice la predicción de un valor dada la información que se le provea a los algoritmos (Google/Tensorflow, 2019).

“Una gran cantidad de circunstancias inciden en el mercado de valores como pueden ser las tasas de interés, sucesos políticos, políticas monetarias, factores tecnológicos, situaciones económicas, y expectativas de los inversionistas” (León et al., 2018: 604),

donde las series de tiempo presentan características de no - linealidad, no estacionarias, y ruidosas, donde si se toma la hipótesis del mercado eficiente como cierta, es casi imposible poder predecir los mercados financieros. Los métodos estadísticos y econométricos tradicionales, no se ajustan con precisión a los mercados con las características ya mencionadas, lo que puede dar como resultado un pronóstico erróneo. Una alternativa a este problema son las redes neuronales artificiales, las cuales tienen la capacidad de ajustarse a fenómenos no lineales y situaciones de ruido (León *et al.*, 2018); estos modelos además de las características anteriores, tienen la capacidad de procesar información como el lenguaje natural; elementos abstractos que los métodos estadísticos y econométricos tradicionales no podrían procesar con facilidad. Por lo que León *et al.* (2018) implementa un modelo de Red Neuronal Artificial (RNA); a la RNA se le aplicó algoritmos genéticos con el fin de encontrar la arquitectura o topología con la cual convergieron en una menor cantidad de tiempo, logrando con ello una arquitectura simple y una optimización o convergencia en menor tiempo. En la investigación de León *et al.* (2018) obtuvieron resultados significativamente positivos en la predicción del Índice del Precio al Consumidor (IPC), logrando predicciones mejores a modelos como ARIMA y ARCH, incluso pudo mejorar

las predicciones de situaciones atípicas como crisis; esto último fue logrado haciendo una descomposición de IPC, con el método Descomposición Empírica en Modos (DEM). No obstante León *et al.* (2018) consideran que tomar como variable endógena para el análisis, como datos históricos, puede limitar el modelo y por tanto su pronóstico, agregando:

“los acontecimientos históricos han demostrado, que en un momento de crisis los inversionistas son propensos a entrar en pánico y reaccionar de manera visceral vendiendo todos sus activos. Por ello, el investigador al igual que el inversionista debe considerar abordar el estudio del mercado bursátil desde otra perspectiva” (León *et al.*, 2018: 628).

Por otro lado, Amornwattana *et al.* (2007) presentan tres modelos financieros para el pronóstico de opciones financieras: Black-Scholes para pronosticar los precios, usando la Volatilidad Histórica; Black-Scholes para pronosticar los precios, usando la volatilidad proveída por una Red Neuronal Artificial; Black-Scholes y Red Neuronal Artificial para pronosticar los precios, usando como supervisor o error, la diferencia entre la salida de estos dos últimos, usando la volatilidad proveída por una Red Neuronal. La Red Neuronal Artificiales utilizada por Amornwattana, *et al.* (2007) es una *General Regression Neural Network* (GRNN), la cual se entrena en un solo proceso o paso (*forward pass*), es decir, converge en una menor cantidad de tiempo; ella proporciona las mismas características de los algoritmos de aprendizaje entrenada con *“Backpropagation”* (Amornwattana, *et al.*, 2007:12). Los modelos que recibieron la volatilidad generada por la Red Neuronal Artificial tuvieron mayor precisión al estimar los precios de las opciones; en comparación con la volatilidad Histórica. Así como el modelo entrenado con la diferencia entre el Black-Scholes y la Red Neuronal Artificial, presentaron una mejora en la estimación de precios (Amornwattana, *et al.*, 2007). Lo cual, de acuerdo a los autores Amornwattana *et al.* (2007) los modelos de Redes Neuronales Artificiales proporcionan mayor estimación en comparación con el modelo Black-Scholes, usando la volatilidad.

Yang & Bao (2008) realizaron un sistema artificial de *trading*, en el cual usaron la representación de alto nivel, es decir, realizaron una abstracción de dos variables que utilizan, desde su punto de vista, los inversionistas: Comprar cuando existen precios bajos y vender cuando hay precios altos; con la expectativa de un cambio de dirección de los precios. Para hacer esto, midieron los puntos de inflexión, donde consideraban cuándo la tendencia

deviene cambio de tendencia. Decisiones basadas en indicadores técnicos (*Moving Average, RSU Oscillator, Stochastic slow K-slow D crossoveroscillator, Trendline system*). El primer punto mencionado anteriormente, se le asignó un umbral, para identificar el punto de inflexión, y pronosticar un posible cambio de dirección de la tendencia; el segundo, se le asignó una señal positiva o negativa al valor más cercano, dependiendo del indicador y sus rezagos en el tiempo. En el modelo se aplicó Cadenas de Markov, optimizadas con el método Monte Carlo Markov Chain. El resultado obtenido después del proceso de aprendizaje y prueba del modelo, se logró obtener un modelo generalizable a diferentes situaciones de inversión de corto plazo y largo plazo, así como a diferentes mercados; dicho modelo se adaptó a las situaciones de ruido.

En otras investigaciones Medina & Manco (2007) diseñaron un sistema experto de lógica con el objeto de asignar créditos y reducir su riesgo. Medina & Manco (2007) ponen de manifiesto que quienes otorgan crédito, además de hacer uso de razones financieras, ellos hacen uso de su experiencia, y análisis de expertos; por lo tanto, Medina & Manco (2007), integraron al sistema experto los elementos de: análisis técnico, análisis fundamental, y otros elementos internos de la empresa, como la capacidad de respuesta al cliente. Un procedimiento relevante en el diseño del sistema experto de Medina & Manco (2007) fue la eliminación de variables en el cual señala:

“Dado que existen varios indicadores que miden el ejercicio de las compañías en relación con diferentes aspectos (liquidez, rentabilidad, endeudamiento cobertura), fue necesario realizar un proceso de selección entre los indicadores propuestos, con el ánimo de no ingresar información redundante al modelo” (Medina & Manco, 2007: 106).

En este paso, para la reducción de variables, hicieron correlaciones y pruebas de hipótesis de Pearson, en el cual se eliminaron las variables con correlaciones positivas y negativas, y, además, significativas. De este modo, se eliminó la redundancia. Medina & Manco (2007) obtuvieron resultados positivos con su sistema experto difuso, sin embargo, los autores no pretenden que se sustituya el análisis de las personas, sino que su sistema sea una herramienta de apoyo para la toma de decisiones de otorgamiento crediticio; y que dicha herramienta, agilice el proceso de otorgamiento de éstos.

Parisi *et al.* (2004) aplicaron un modelo multivariado dinámico generado por algoritmos

genéticos, para predecir las variaciones positivas o negativas de índices bursátiles, el trabajo fue comparado con un AR y ARIMA; Estos fueron puestos a prueba para comparar los rendimientos obtenidos por cada modelo. El resultado fue un modelo predictivo superior a ARIMA, además de significativo; no obstante, su precisión no superó el 68 %; los índices a pronosticar fueron el IPC, TSE, Nasdaq, y DJI, los cuales son considerados de los más importantes en América del Norte (Parisi *et al.*, 2004). Los índices donde se obtuvo mayor rentabilidad fueron el: IPC, TSE y el DJI. Al final se puede concluir que los algoritmos genéticos mejoran los pronósticos de series de tiempo. Una característica importante del trabajo de Parisi *et al.* (2007) fue el utilizar *block bootstrap* para generar datos faltantes de la serie de datos históricos.

Parisi, *et al.* (2018), en otros trabajos, realizan un modelo multivariado usando modelos computacionales de fuerza bruta; éste integra dos variables exógenas, “*el NASDAQ y el factor conciencia*” (Parisi, *et al.*, 2018: 52); aunque la conciencia es la capacidad de pensar en si mismo, es decir: ser consciente de sí y de su entorno, desde la psicología, sin embargo Parisi *et al.* (2018) determinan como conciencia el hecho de retroalimentar el sistema sobre el conocimiento de los resultados de rendimiento obtenidos anteriormente, con el objeto de mejorar las predicciones futuras. Estas dos variables (NASDAQ y factor conciencia) se usaron para tener una aproximación predictiva porcentual de las acciones TSLA, empresa automotriz Tesla, Inc., comercialmente conocida como Tesla Motors. El objeto de la investigación es “*proyectar la variación del signo [es decir, si va haber una variación positiva o negativa en el mercado] que tiene el valor estudiado*” (Parisi *et al.*, 2018: 61). El trabajo arrojó una precisión de un rango entre 60 % y 70 % de aciertos, esto con ayuda del factor conciencia, teniendo una precisión, haciendo pronósticos de hasta 4 rezagos en el tiempo. La precisión fue evaluada con el test de Acierto Direccional de Pesaran y Timmermann:

“El cual mide la capacidad de predicción de los modelos de Redes Neuronales Artificiales, permitiendo medir la significancia estadística de los modelos y su capacidad de predicción”/.../“establece[siendo] como Hipótesis Nula (Ho) que las variaciones proyectadas por el modelo están independientemente distribuidas de las variaciones observadas reales” (Parisi, et al., 2018: 57);

si se obtiene un valor observado fuera del rango 90-95 % se rechaza la Hipótesis Nula, luego

entonces se dice que el modelo “*tiene capacidad de predicción*”(Parisi, *et al.*, 2018: 57). En su trabajo rechaza la Hipótesis Nula, por tanto su modelo tiene capacidad predictiva de acuerdo al los autores.

En contraste con las investigaciones de Parisi *et al.* (2018), al usar algoritmos genéticos entrenados por fuerza bruta, de acuerdo a Isasi (2014) puede llegar a situaciones donde sea casi imposible la convergencia, de ahí la diferencia con otros algoritmos genéticos que buscan optimizar una funciones de utilidad, o bien, Redes Neuronales Artificiales (RNA), que usa el algoritmo *Stochastic Gradient Descent* que converge minimizando el error de manera iterativa (Isasi, 2004); este algoritmo se usará en esta investigación.

Parisi *et al.* (2006) realizaron dos modelos multivariados de inteligencia artificial para pronosticar los cambios positivos o negativos de las acciones de: *Nikkei225*, *Hang Seng*, *Shanghai Composite*, *Seoul Composit* y *Taiwan Weighted*. Dichos modelos fueron realizados con Algoritmos Genéticos y Redes Neuronales Artificiales; estos se compararon con un modelo de AR(1) y puestos a prueba considerando el rendimiento como comparación de desempeño. A lo largo de la historia del análisis de pronósticos financieros se ha intentado pronosticar valores precisos, disminuyendo la varianza, no obstante Parisi *et al.* (2006) en lugar de disminuir la varianza y tener un valor preciso o nivel, ellos buscan pronosticar las variaciones positivas o negativas de los valores. El modelo para predecir las variaciones fue hecho mediante modelos de clasificación (Parisi *et al.*, 2006); ya que, al comparar el modelo con los de estimación de nivel, los primeros se desempeñan mejor (Parisi *et al.*, 2006; Leung, *et al.*, 2000); de ahí que Parisi *et al.* (2006) optaron por modelos de clasificación, y con ello clasificar las variaciones. Parisi *et al.* (2006) obtuvieron que los modelos de Algoritmos Genéticos se desempeñaron mejor que los modelos de Redes Neuronales, en cuanto a rendimiento obtenido; sin embargo, en cuanto a predicción, los modelos de Redes Neuronales se desempeñaron mejor. Algo que puede ser contradictorio; aunque de acuerdo con los autores, las Redes Neuronales solo tuvieron mejor desempeño en la predicción de dos acciones, mientras que los genéticos fueron en las cinco acciones elegidas; lo cual puede tener explicación, ya que al poner a prueba sus rendimientos en el mercado, las Redes Neuronales solo usaron dos activos, mientras que los Algoritmos Genéticos usan cinco. Por otro lado, añadiendo, los algoritmos de Inteligencia Artificial tuvieron mejores

rendimientos en contraste con el modelo AR(1).

Rosenberg *et al.* (2017) realizaron una comparación de la predicción de acciones hechas por enjambres de abejas artificiales en contraste con las predicciones de individuo y grupos de inversión. Los resultados fueron favorables para los enjambres de abejas, tanto en nivel de predicción y significancia, en relación con los individuos y grupos de inversión, así como en rendimiento obtenido en pruebas.

Chidmbaran *et al.* (1998) realizaron un modelo de Algoritmos Genéticos para el pronóstico de opciones; este obtuvo un mejor resultado en comparación con el modelo Black-Scholes. Chidmbaran *et al.* (1998) se lo atribuye a que los Algoritmos Genéticos son flexibles, y tienen la capacidad de adaptarse a los cambios, además de ser un algoritmo de autoaprendizaje y baja demanda de datos (en contraste con las Redes Neuronales Artificiales, que demandan una mayor cantidad de datos (Viñuela & Galván, 2004)).

Spiegeleer *et al.* (2018), realizaron un modelo de *Machine Learning* llamado *Gaussian Process Regression* (GPR) para el pronóstico de diferentes activos financieros, con el objetivo de disminuir el periodo de entrenamiento; que en este caso se hace de una sola pasada (un único ciclo entrenamiento), logrando el objetivo, empero, disminuye la precisión, no obstante es compensable por el menor tiempo de entrenamiento. Su modelo logró pronosticar de manera confiable los activos como opciones vanilla y opciones exóticas.

Canelas *et al.* (2013), exponen que el *High Frequency Trading* (método usado en gran medida en el sector financiero, Canelas *et al.* (2013)) es el uso de algoritmos computacionales para realizar transacciones en los mercados financieros, con posibilidad de adaptarse a los *new patterns*, por lo cual propone una metodología: SAX-GA, el cual puede ser superior al método *Chart Patterns*, este se basa en el análisis gráfico para tener una intuición aproximada del futuro comportamiento de la bolsa, el cual es conocido, también, como análisis técnico. SAX es un método de reducción de dimensional, así como representación simbólica, este método es, generalmente, usado en proyectos de bioinformática *v.gr.*: *Human Genome Project*; SAX es considerado como mejor método de reducción dimensional para grandes volúmenes de datos, ya que otros métodos como *Discrete Fourier Transform* (DFT), o *Singular Value Decompositions* (SVD) genera una gran pérdida de información, a diferencia de SAX (Canelas *et al.*, 2013). En el trabajo SAX es combinado con *Genetic*

Algorithm (GA), para optimizar sus resultados, teniendo dos objetivos, analizar las tendencias de mercados alcistas y mercados bajistas, para posteriormente hacer inversiones de corto y largo plazo. La idea principal del trabajo es que con SAX-GA, el algoritmo se adapte a los datos y encuentre nuevos patrones conforme se presentan en las series de tiempo. Canelas *et al.* (2013) realizaron una comparación entre SAX-GA y *Buy & Hold*, con el objeto de tener un punto de referencia que pueda guiar en el saber de si es o no un mejor modelo. Así pues, el resultado concluye que SAX-GA tiene rendimiento mayor que *Buy & Hold*, siempre y cuando el GA no se sobre ajuste.

Rodríguez *et al.* (2014) realiza una comparación de “éxito” de diagnóstico de fracaso financiero de empresas, comparándolo con modelos paramétricos y de inteligencia artificial; se aplicaron 3 modelos paramétricos: análisis de regresión lineal múltiple (MDA), análisis discriminante lineal (MRL), análisis de regresión logística (LOGIT); y un modelo de Redes Neuronales Artificiales (RNA) tipo perceptrón multicapa, entrenado con el algoritmo *Backpropagation*. Los modelos fueron ajustados con Razones Financieras; posteriormente se contrastaron los modelos paramétricos con la Red Neuronal Artificial; esta última obtuvo un rendimiento superior; a pesar de utilizar una menor cantidad de parámetros; esto se le atribuye debido a su capacidad de disminuir el ruido (Rodríguez *et al.*, 2004) (sin embargo, Viñuela (2004) atribuye el rendimiento de las RNA a su capacidad de trabajar en situaciones de ruido). No obstante, tanto los modelos paramétricos como la RNA pronostican de manera satisfactoria el riesgo de fracaso financiero. Posteriormente, se contrastó con las auditorías externas; teniendo, tanto los modelos, como la auditoría, resultados satisfactorios; y no el sesgo esperado de las auditorías externas, como refieren Rodríguez *et al.* (2014). Cabe destacar que Rodríguez *et al.* (2014), en su investigación, no realizaron normalización de los datos; sino que utilizaron las razones financieras sin algún tipo de modificación, para su modelo, ya que consideran que se perdería información; otra parte relevante de su trabajo es el realizar un análisis factorial con “*objetivo de reducir el conjunto inicial de variables a un pequeño grupo de regresores sintéticos, no correlacionados*” (Rodríguez *et al.*, 2014:193); de tal modo que reducen la redundancia de sus variables.

Gordillo & Christopher (2003) diseñaron una simulación de mercados financieros, haciendo uso de algoritmos genéticos, para obtener las mejores respuestas posibles de los

agentes a participar en la simulación de los mercados financieros. Las conclusiones obtenidas en su investigación es que los inversionistas de largo plazo obtienen mejores rendimientos, en contraposición de los de corto plazo; la falta de información o información negativa sobre los mercados disminuye la inversión por aversión a la pérdida (ojo, y no al riesgo, sino a la pérdida, como lo dice la teoría Prospectiva). Una de las conclusiones a destacar es que a mayor información hay mayor volatilidad, así como también disminuye la volatilidad cuando disminuyen las restricciones. Los resultados fueron obtenidos con base en la simulación de los algoritmos genéticos. Por otro lado, los cuatro modelos probados en el artículo solo considera a los inversionistas y los *market makers*, en el mercado y si, por ejemplo, existen modelos con información integrada de alta representación (Depei & Zehong, 2008). Puede considerarse, integrar, también, las relaciones de poder en las simulaciones, por ejemplo; aunque eso sería en otra investigación, pues no es el objeto de estudio de esta tesis. Cabe resaltar, que las investigaciones de Gordillo & Christopher (2013) llegan a resultados similares a los de Eugene Fama, quién demostró, en 1960, que predecir los precios de las acciones en el corto plazo es imposible; contrastando, además, su investigación, con el otro trabajo de Robert Shiller, que a su vez menciona que se pueden predecir los precios de las acciones en el largo plazo; luego entonces coincide con los resultados de los autores, ya que Gordillo & Christopher (2013) encuentran que los agentes en niveles de incertidumbre alta, en el corto plazo no tienen rendimientos, mientras que por otro lado, los agentes que invierten en el largo plazo tienen mayores rentabilidad; coincidiendo con la idea económica que asegura: “*High future returns are then viewed as compensation for holding risky assets during unusually risky times*” (_____, 2013).

Haciendo hincapié en los resultados de Eugene Fama, como se dijo antes, aclara que la predicción de los activos financieros en el corto plazo, es extremadamente complicado, y además - haciendo énfasis - la nueva información que constantemente aparece en el mercado “*is very quickly incorporated into prices*” (_____, 2013), lo que hace complicado la predicción de activos financieros.

Binner *et al.* (2010) realizan un análisis para generar un modelo de pronóstico de la inflación de EU mediante la oferta monetaria; para ello realizaron dos modelos, uno de Redes Neuronales Recursivas, y otro de *Kernel Recursive Least Squares Regression*; los

resultados arrojaron que la oferta monetaria tiene un efecto marginal sobre la inflación de EU. Refieren, los autores, que desde 1980 la inflación se ha vuelto más difícil de pronosticar en referencia a los trabajos de Stock & Watson (Binner *et al.* 2010); y que es un fenómeno más general y complejo. Además, Binner *et al.* (2010) mencionan que la oferta monetaria no proporciona gran información cuando la inflación es estable, es decir: no presenta cambios drásticos (volátil) ni, tampoco, es alta.

Benhmad (2012) realiza una investigación para determinar la causalidad entre el Tipo de Cambio (TC) estadounidense y el precio del petróleo, Benhmd (2012) utilizó Wavelet (Benhmad, 2012), el cual es un modelo que sirve para estudiar fenómenos no estacionarios, variantes en el tiempo, no-lineales (Benhmad, 2012; Gonzáles, 2010), o dicho en otras palabras, es robusto para el análisis o tratamiento de datos; no obstante en su investigación prefirió utilizar transformación de diferencia logarítmica, con el objetivo de realizar la prueba de causalidad lineal y no - linealidad; Benhmad (2012) hace uso de dos modelos de causalidad no lineal, uno basado en la serie de Taylor y el otro basado en Redes Neuronales Artificiales. Las conclusiones a las que llegó Benhmad (2012) fue que el TC y el precio del petróleo tienen una fuerte relación bidireccional, con lo cual logró realizar pronósticos.

Choudhary (2010) hizo uso de diferentes modelos de Redes Neuronales Artificiales (RNA) para pronosticar la inflación de 28 países pertenecientes a la OCDE, dichos modelos fueron comparados con un AR(1); donde las RNA tuvieron un rendimiento superior, en la mayoría de los casos, al modelo AR; por lo que Chaudhary (2010) sugiere probar las RNA con modelos econométricos para la elección adecuada de la herramienta, es decir, probar diferentes modelos de inteligencia artificial, para así encontrar el modelo adecuado; lo que coincide con el proceso de diseño de *Machine Learning System*, que consiste en probar diferentes modelos para encontrar aquel que se ajuste mejor a los datos (Raschka *at al.*, 2016).

Por otro lado tenemos la investigación de Marwalka (2013) que diseñan un modelo de Determinación Automática de Relevancia, para pronosticar el índice de precios al consumidor (*consumer price index*, CPI) de EU, haciendo uso de Redes Neuronales Multicapa, aplicando el Teorema de Bayes (Marwalka, 2013); de este modo, logró obtener la relevancia de cada uno de los sectores que utilizó, determinando que los sectores con mayor relevancia

para el CPI es:

la minera, transporte, almacenamiento y comunicación, intermediarios financieros, seguros, bienes raíces, servicios para empresas, servicios sociales y personales, valor agregado bruto a precios básicos, impuestos menos subsidios a los productos, asequibilidad, crecimiento económico, tasa de interés, producto interno bruto, consumo familiar e inversión." (Marwalka, 2013: 61).

Cabe destacar que Marwalka (2013) no menciona para qué país se realizó su investigación.

Bogda *et al.* (2013) proponen un arquitectura de inteligencia artificial para el pronóstico de crisis financieras. Creando dos redes neuronales artificiales conectadas a un circuito lógico, las cuales clasifican los factores que causan la crisis; sin embargo no obtuvieron resultados satisfactorios; ello lo atribuyen a la arquitectura de las Redes Neuronales, las cuales fueron realizadas con dos capas ocultas.

Qi (2001) realizó un trabajo para pronosticar las recesiones de EU utilizando una Red Neuronal Artificial, con la que logró pronosticar cuatro recesiones, en cuanto al periodo utilizado para su análisis, las variables que determinaron la crisis fueron: el índice del Departamento de Comercio, el índice Stock & Watson, el índice S&P500, entre otros, cabe destacar que de acuerdo con su análisis en diferentes periodos las variables tienen diferente relevancia.

Estos trabajos nos dan indicios o información de que el uso de algoritmos de inteligencia artificial (sus diferentes áreas) tienen ventajas significativas con respecto a los modelos usados tradicionalmente en las finanzas y economía, principalmente, debido a que se ajustan a los datos mediante la búsqueda de soluciones en el espacio del error y eliminado el ruido (aunque otros autores mencionan que trabajan adecuadamente en situaciones de ruido, lo que es diferente de eliminar el ruido). No obstante, esto no quiere decir que sea una ley el tener ventaja significativa sobre otros modelos, ya que como se observa en las diferentes investigaciones, no en todos los casos se obtuvo un resultado satisfactorio y, además los modelos no eliminan del todo la incertidumbre, ya que aún queda un margen de error que implica riesgo en la toma de decisiones, aunque ciertamente menor en comparaciones de los resultados obtenidos con otros modelos.

Por otra parte, como se observa en la revisión de las investigaciones anteriores, al momento de trabajar con inteligencia artificial, es, en la mayoría de los casos, necesario hacer un pre - procesamiento de los datos. Por lo que en la presente tesis, se realizará un pre - procesamiento de datos, ya que se tratará con datos no numéricos como lo son las palabras de las noticias emitidas por Reuters a través de su portal <https://www.reuters.com/>.

1.3. Crítica a los algoritmos computacionales aplicados en economía y finanzas

“Un intelectual aparece a partir del momento en que el ejercicio de este oficio hace surgir una contradicción, entre las leyes de ese trabajo y las leyes de la estructura capitalista /.../ un científico nuclear no es un intelectual. Es un científico a la medida que hace sus investigaciones. Si el mismo científico, llevando a cabo sus investigaciones nucleares se da cuenta de que con su trabajo va a posibilitar la guerra atómica, y si denuncia esto, es porque lo siente como una contradicción /.../ Si al mismo tiempo, como ha sucedido a menudo cierto número de científicos nucleares se reúnen y declaran que no quieren que sus trabajos se utilicen para esos fines, ellos viven su propia contradicción. Si denuncia esa contradicción exterior, yo digo que son intelectuales. En esa condición, como vemos, el intelectual tiene un doble espacio /.../ Pero si el intelectual no descubre constantemente su contradicción en sí, si no ejerce constantemente su oficio de intelectual, de científico o de técnico de saber práctico, es un marginal /.../ Es decir que debe a la vez ejercer su oficio y comprometerse a la manifestación de las contradicciones de la sociedad.”

(Sartre, J., 1967).

“Los robots ocupan el lugar de las personas en los empleos”

(Bènèfice, I, *et al.*, 2013).

En el mundo de las finanzas, la imagen de hombres gritando y/o aquellas personas frente a una computadora realizando operaciones de compra/venta de activos financieros ha cambiado. Hoy en día existen algoritmos computacionales que gestionan los mercados financieros. En 2013, dos tercios de las transacciones realizadas en la bolsa de Estados Unidos son llevados a cabo por algoritmos computacionales (Bènèfice, *et al.*, 2013: min. 5.39 - 6.10 min) y en 2019, el 75 % de las transacciones son realizados por Bots (Gershenson, 2019: Bots, seg. 56 - min. 2.10); y, además, en París Francia, un tercio de las transacciones,

en la bolsa de París, se hacen por algoritmos computacionales (Bènèfice, *et al.*, 2013). Estos algoritmos pueden generar cambios abruptos en los mercados financieros y alta volatilidad (Gershenson, 2019: Bots, seg. 56 - min. 2.10) en la carrera del *High Frequency Trading* (Gershenson, C., 2019: Bots, seg. 56 - min. 2.10); de la misma manera Steliaros (2019), en entrevista en QuantMinds TV, asevera que los algoritmos de *Machine Learning* generarán mayor volatilidad (Steliaros, 2019).

Los algoritmos computacionales tienen la capacidad de hacer transacciones en milisegundos, pasando “órdenes al carnet de órdenes” (Bènèfice, *et al.*, 2013: Rainelli, min. 3.57 - min. 4.28), todo a una alta velocidad, transacciones realizadas por algoritmos computacionales, ésto se conoce como *High Frequency Trading* (Bènèfice, *et al.*, 2013: Rainelli, min. 3.57 - min. 4.28). Estos consisten en usar las tecnologías para estar por delante de los demás, “*se trata, pues, de un carrera informática y tecnológica*” (Bènèfice, *et al.*, 2013: Philipponat, min. 4.35 - 4.57) para tener mayores beneficios teniendo una mayor velocidad y por tanto mejores precios; estos algoritmos son “*muy inteligentes, las personas, no les pueden hacer sombra, a su lado no son nada*” (Bènèfice, *et al.*, 2013: Jorion, min. 4.50 - min 5.14).

Como se menciona en párrafos anteriores, las transacciones de *High Frequency Trading* generan mayor volatilidad y de acuerdo a Thierry Francq (Bènèfice, *et al.*, 2013) el secretario general de la Autoridad de Mercados Financieros (AMF) en 2013 de Francia, entrevistado por La noche Temática en 2013 en el documental *Trading* de Alta Frecuencia dice:

“Se puede generar una caída súbita en cualquier momento, en París o en Londres, es perfectamente posible. Todo mundo tiene en la cabeza la caída súbita de Estados Unidos (el Flash Crash de 2010); Pero recordemos que el año pasado, se produjeron en Francia, pequeñas caídas súbitas en determinados valores /.../ lo cual es un riesgo que tenemos muy presente”,

las caídas en bolsa pueden durar muy poco, pero generar catástrofes y ésto puede suceder en pocos segundos, de acuerdo con Francq (Bènèfice, *et al.*, 2013) en entrevista.

En el documental de *Trading* de Alta Frecuencia del canal La Noche Temática se realizó una entrevista a un Anónimo del sector financiero: él, asegura que en el sector financiero,

en materia de *High Frequency Trading*, nadie le tiene miedo a la AMF, y a quién le llevan por mucho la delantera.

Un medio de especulación con algoritmos de *High Frequency Trading*, es hacer solicitudes de compra, pujando en los precios, una vez que estos suban, se cancela la solicitud antes de ejecutarse, después se realizan solicitudes de venta y se ejecutan, de este modo se obtienen beneficios de la diferencia del precio creada por pujar en los precios; esta estrategia se puede realizar ya sea una posición de compra o venta, es decir, en términos más financieros, posición corta o posición larga, respectivamente (Bènèfice, *et al.*, 2013); tal actividad se conoce como *spoofing*, y se considera una práctica fraudulenta. No obstante, son operaciones complicadas de identificar, ya que los algoritmos lo hacen a una alta velocidad.

Las caídas en bolsa en segundos o minutos es un fenómeno económico que antes del siglo XIX jamás se hubieran pensado y es conocido como Flash Crash. El primer Flash Crash del que se tiene información es el que tuvo lugar el 6 de mayo de 2010, los algoritmos computacionales generaron un Flash Crash. Este sucedió a las 14 horas, del 6 de mayo de 2010, Wall Street sufrió un hundimiento, el Dow Jones se desplomó, en solo 5 minutos (Bènèfice, *et al.*, 2013: min. 7.40 - min. 11.32); los ordenadores del fondo de pensiones estadounidense Waddell & Reed, vendieron cerca de 3 mil contratos en un periodo de tiempo muy corto (Bènèfice, *et al.*, 2013: min. 9.40 - min. 11.32). Aunque no se le atribuye toda la responsabilidad a Waddell & Reed (Bènèfice, *et al.*, 2013: min. 9.40 - min. 11.32), cabe destacar que se le atribuye el inicio de generar la caída, ya que el emitir la venta de cerca de 3 mil contratos, influyó en los demás operadores computacionales de *High Frequency Trading* para generar ventas tales que provocaron la abrupta caída en la bolsa de valores en pocos minutos (Bènèfice, *et al.*, 2013). Esta caída de bolsa de valores “no duró más de 14 minutos” (Bènèfice, *et al.*, 2013: min.). La bolsa de Chicago, para solucionar el problema, detuvo las operaciones durante 5 segundos; pero en 14 minutos se esfumaron más de 150 millones de euros a valor de 2013. En un informe de *Commodity Futures Trading Commission* y la *Securities and Exchange Commission* de Estados Unidos enunciaron, con respecto a dicho fenómeno declaran:

“One key lesson is that under stressed market conditions, the automated execu-

tion of a large sell order can trigger extreme price movements, especially if the automated execution algorithm does not take prices into account. Moreover, the interaction between automated execution programs and algorithmic trading strategies can quickly erode liquidity and result in disorderly markets. As the events of May 6 demonstrate, especially in times of significant volatility, high trading volume is not necessarily a reliable indicator of market liquidity.” (CFTC & SEC, 2010: 6).

Por otro lado Mirowski (2002) refiere que los grupos de economistas que buscan renovar o hacer nuevos aportes a la economía haciendo uso de análisis computacional, aplicando nuevos modelos para el análisis económico, recaen sobre las mismas bases y supuestos de los economistas neoclásicos, reduciendo la economía a números, sin ir a las bases del análisis económico, es decir, siguen haciendo análisis matemático, sin analizar el fenómeno que subyace, es decir, analizan el síntoma, y no la causa del fenómeno. Empero algunos algoritmos de inteligencia artificial como los de *Machine Learning* difieren considerablemente de los modelos neoclásicos, ya que estos modelos son realizados con supuestos *a priori*, mientras que los modelos de *Machine Learning* son modelos que se ajustan a los datos, y no al revés, ajustar los datos al modelo, *v. gr.*, la minería de datos, que comúnmente se utiliza k-NN (*k* Nearest Neighbors, este algoritmo se utiliza para la clasificación y predicción, el cual calcula *k* clases, donde dependiendo a la distancia a esta clase, ciertas observaciones pertenecen o no a dicha clase); aunque, cabe destacar, que se pueden presentar problemas como el overfitting (estos temas serán retomados más adelante).

1.4. Teoría de la Inteligencia Artificial (IA)

En esta sección se hará una breve revisión de la historia de la inteligencia artificial; revisión de algunos modelos de este campo, como los modelos de clasificación, y se profundiza en las redes neuronales artificiales, rama de la inteligencia artificial que nos compete en esta tesis, ya que será la herramienta a utilizar para realizar el pronóstico financiero, también se revisará algunos de los problemas que presenta y como solucionarlos.

1.4.1. Breve historia de la inteligencia artificial

Los inicios de la inteligencia artificial se remontan a la Edad Media y el Renacimiento, épocas donde se desarrollaron los primeros autómatas, los cuales tenían sus respectivas limitaciones; después en la “*Edad Moderna gracias a avances en relojería e ingeniería se construyeron varios autómatas y andróides*” (Gershenson, 2019). Ya en el siglo XX, con las primeras computadoras, inició la discusión de crear máquinas pensantes. En 1943, Rosenbluth, Wiener y Bigelow concluyen que se puede crear máquinas pensantes con las computadoras; en ese mismo tiempo, surgen las bases de las redes neuronales, creado por McCulloch y Pitts (Gershenson, 2019). En 1956 surge la inteligencia artificial como disciplina, en una reunión realizada en la Universidad Dartmouth. En los primeros años de la inteligencia artificial como disciplina se generaron una corriente llamada conexionismo, que dio origen a lo que hoy se conoce como redes neuronales. El éxito destacable de la época fue la creación del Perceptrón, de Frank Rosenblatt, que permite el aprendizaje por prueba y error; aunque tenía algunas limitaciones, y a pesar de que se propuso en su tiempo hacer perceptrones de varias capas, no existía un algoritmo que fuera capaz de entrenar a estos sistemas (Gershenson, 2019). Fue hasta los años 80’s donde se diseñaron algoritmos para poder entrenar a perceptrones multicapa. En los 50’s surgieron las simulaciones evolutivas que dieron origen a los algoritmos genéticos, que se desarrollaron en los 60’s y 70’s (Gershenson, 2019). En 1962, cuando el enfoque cambió a aprendizaje automático, en la representación del conocimiento, esto dio origen a lo que hoy se conoce como sistemas expertos (Gershenson, 2019); esta situación incentivó a los investigadores a generar los lenguajes de programación para poder desarrollar estos sistemas, naciendo, como consecuencia, Prolog y Lisp.

En los años 70’s, los fondos destinados a la investigación comenzaron a disminuir, debido a las altas expectativas no cumplidas que se tenían sobre la inteligencia artificial; este periodo de reducción de investigación se le conoce como el invierno de la inteligencia artificial (Gershenson, 2019).

En 1982, Japón revive el sueño roto de la inteligencia artificial, con el proyecto a la Quinta Generación (Gershenson, 2019), con un programa de inversión de 10 años en

tecnologías de la computación. Dicho proyecto no cumplió las expectativas, empero revivió el interés por esta área, reviviendo la corriente coleccionista como la de David Rumelhart, McClelland y Hopfield, quienes criticaban los sistemas basados en símbolos, y por otro lado se encontraban los computacionalistas, que criticaban a las redes neuronales artificiales, a las cuales, ello “*las catalogan como cajas negras, donde no se sabía dónde estaba su lógica*” (Gershenson, 2019). Otra corriente, posterior, inspirada en la etología y conceptos de cibernética, desarrollo sistemas de inteligencia artificial basados en el comportamiento; ella “*se complementó con el campo de la vida artificial*” (Gershenson, 2019) y cambiando “*el enfoque de razonamiento a la interacción con el mundo a través de ciclos de percepción - acción*” (Gershenson, 2019). En los noventas, surgieron los agentes, el cual “*es una entidad que puede actuar en su entorno*” (Gershenson, 2019), guiados por reglas; en ese mismo tiempo, 90’s, también se “*popularizó la investigación sobre emociones artificiales y cómputo afectivo*” (Gershenson, 2019); Roselind Picard, por ejemplo, publicó “*Cómputo afectivo*”, donde considera que las emociones son necesarios para regular el comportamiento de los agentes.

En las última década se han dado grandes avances en el reconocimiento de patrones, como los avances en carros autónomos, y el juego de Go, aunque estos avances no son generalizables, ya que el algoritmo que juega Go, no puede jugar otro tipo de juego, o el algoritmo del carro autónomo no servirá para una moto (Gershenson, 2019). Aunque existen, por ejemplo técnicas de *transfer learning* en redes neuronales artificiales, las cuales pueden transferir el conocimiento a otra red neuronal.

Hoy en día empresas como DeepMind de Google han realizado grandes avances en el *Deep Learn y Reinforcement Machine Learning*, desde sus publicaciones en 2017 de Alpha Go, una inteligencia que utiliza *Deep Learn y Reinforcement Machine Learning*, esta Inteligencia Artificial está diseñada para jugar ajedrez, la cual a superado las expectativas, ganando a los mejores programas de juego de ajedrez, sin embargo el realizar generalizaciones a otras áreas de optimización como los superconductores, aún es complicado, ya que, en el mundo real existen más variables que pueden influir en los resultados, no así en el ajedrez (Gershenson, 2019). Por otro lado tenemos la inteligencia artificial de reconocimiento del lenguaje natural y respuesta de Amazon, Alexa (Gershenson, C., 2019), que se

puede definir como un asistente personal virtual que utiliza algoritmos de reconocimiento de voz; sin embargo aún hay retos en la inteligencia artificial, y sus aplicaciones pueden ser sorprendentes, como lo son aplicaciones a la economía, de las cuales se habló en los artículos mencionados en capítulos anteriores.

1.4.2. El método de la inteligencia artificial

La inteligencia artificial usa el método sintético (Gershenson, 2019) el cual consiste en cuatro etapas principales:

1. Observación de un fenómeno natural, “*que a través de la abstracción o generalización, nos llevan a una*”(Gershenson, 2019):
2. Teoría; luego, esta teoría se usa para crear, con la ingeniería (Gershenson, 2019) un
3. Sistema artificial, el sistema artificial genera cierto comportamiento y
4. Se mide el desempeño (Gershenson, 2019), para verificar o falsificar la teoría.

1.4.3. Flujo de trabajo de Machine Learning, etapas del proceso de Big Data y Knowledge Discovery in Database (KDD)

En esta sección se muestran dos modelos para trabajar con grandes volúmenes de datos, modelo de *Machine Learning System*, que para propósito de este trabajo solo llamaremos *Machine Learning* (ML), y el modelo de *Knowledge Discovery in Database* que llamaremos KDD.

El *Machine Learning*, es un campo de la inteligencia artificial que involucra algoritmos de aprendizaje no supervisado, y aprendizaje supervisado; técnicas usadas por ejemplo: para obtener conocimiento a partir de datos. El ML permiten formar conocimiento a partir de grandes datos, algo que el humano, con dificultad haría, por lo que regresando a la idea de generación de conocimiento, esto abre las puertas a la toma de decisiones basándose en datos. El ML no solo está tomando gran relevancia en Ciencias de la Computación e Informáticas, sino también en la vida cotidiana (Raschka, & Mirjalili, 2017), por ejemplo,

gracias a él tenemos una bandeja de entrada de correos y una para spams, y no se diga de la inmensa información analizada por Amazon quien nos ofrece el producto que deseamos cuando lo necesitamos.

Raschka & Mirjalili (2017) señalan que el proceso de flujo de trabajo de construcción de *Machine Learning systems*, se debe de realizar un pre - procesamiento de los datos, el cual mejora el rendimiento de los modelos; dicho proceso incluye la reducción de dimensiones; esto contrasta con algunos de los modelos revisados en la sección 2.1 y 2.2 de este capítulo. Cabe resaltar que los métodos realizados no son los únicos, sino que existen diferentes métodos que van de acuerdo a cada problema a resolver y sistema a construir, como se observa en las secciones mencionadas.

A continuación se muestra el Flujo de Trabajo para la construcción de “*Machine Learning System in predictive modeling*” (Raschka, & Mirjalili: 12-13):

1. Pre - procesamiento de los datos
2. Selección del modelo que mejor se ajusta a los datos de entrenamiento
3. Se evalúa el modelo, es decir: evaluar la precisión.
4. Se realizan las predicciones con el modelo.

Por otro lado el proceso de *Big Data* o analítica de *Big Data*, es “*el proceso de examinar grandes volúmenes de datos de una gran variedad de tipos (Big Data) para descubrir patrones ocultos, correlaciones desconocidas y otras informaciones útiles*” (Joyanes, 2013); por otro lado existe una gran cantidad de significados que se le da al *Big Data*, aunque cabe resaltar que es de origen inglés y se traduce simplemente como Datos Masivos, pero de las diferentes concepciones que se tiene de él se puede resumir que, este, se refiere a “*datos cuyo tamaño supera considerablemente la capacidad de captura, almacenado, gestión y análisis del software convencional de bases de datos*” (García, et al., 2018: 14).

Los datos del *Big Data* tiene que cumplir las “3V”, de acuerdo a la empresa Gartner según García, et al. (2013), que significa, Volumen, Velocidad, y Veracidad de los datos:

- Volumen: Se refiere a datos masivos; generados ya sea por empresas, u organizaciones de cualquier otra índole, por ejemplo el millón de transacciones generadas en Walmart cada hora (García, *et al.*, 2018: 14).
- Veracidad: Se refiere a la velocidad en que los datos son generados y a su vez procesados para satisfacer a la demanda (García, *et al.*, 2018: 14).
- Variedad: “*Big data ha de tener la capacidad de combinar una gran variedad de información digital en los diferentes formatos en los que se puedan presentar*” (García, *et al.*, 2018: 14), refiriéndose a bases de datos relacionadas y no relacionadas, bases de datos estructuradas y no estructuradas, documentos, correos, imágenes, audios, entre otros. (García, *et al.*, 2018: 14).

Por lo que el *Big Data* es

“el conjunto de arquitecturas y herramientas informáticas destinadas a la manipulación, gestión y análisis de grandes volúmenes de datos desde todo tipo de fuentes, diseñadas para extraer valor y beneficio de los mismos, con una amplia variedad en su naturaleza, mediante procesos que permiten capturar, descubrir y analizar información a alta velocidad y con un costo reducido” (García, *et al.*, 2018: 14).

“BD involucra un proceso iterativo e interactivo de búsqueda de modelos, patrones o parámetros” (García, *et al.*, 2018: 19), que es el proceso de descubrimiento de conocimiento. De manera esquemática se presenta, a continuación, el proceso de descubrimiento de conocimiento de acuerdo a García, *et al.* (2018: 20):

1. *“Formular el problema.*
2. *Determinar la representación (atributos y clases). Esta determinación se puede realizar directamente a la vista de los datos, mediante la intervención de expertos o utilizando técnicas automáticas como son los filtros.*
3. *Identificar y recolectar datos de entrenamiento (bases de datos, ficheros, etc.).*
4. *Preparar datos para análisis.*

5. *Selección de modelo, construcción y entrenamiento.*
6. *Evaluar lo aprendido. La evaluación puede realizarse automáticamente, como por ejemplo la validación cruzada o mediante la intervención de un experto que valore los resultados obtenidos.*
7. *Integrar la base de conocimiento a la espera de nuevos datos tras acciones.” (García, et al., 2018: 14).*

Cabe resaltar que García, *et al.* (2018) en su libro *Ciencia de Datos*, expone los pasos anteriores para luego hablar de manera más formal y específica del proceso de *Big Data*; el cual es el siguiente:

1. *“Comprensión del dominio de la aplicación, del conocimiento relevante y de los objetivos del usuario final.*
2. *Creación del conjunto de datos: Consiste en la selección del conjunto de datos, o del subconjunto de variables o muestras de datos, sobre los cuales se va a realizar el descubrimiento.*
3. *Limpieza y pre - procesamiento de los datos. Se compone de operaciones, tales como la recolección de la información necesaria sobre la cual se va a realizar el proceso, decidir las estrategias sobre la forma en que se van a manejar los cambios de los datos no disponibles, estimación del tiempo de la información y su posibles cambios, etc.*
4. *Reducción de los datos y proyección. Encontrar las características más significativas para representar los datos, dependiendo del objetivo del proceso. En este paso se puede utilizar métodos de transformación para reducir el número de variables a ser consideradas o para encontrar otras representaciones de los datos.*
5. *Elegir la tarea de minería de datos. Decidir si el objetivo del proceso es: regresión, clasificación, agrupamiento, etc.*

6. *Elección del algoritmo(s) de minería de datos. Selección del método(s) a ser utilizado para buscar los patrones en los datos. Incluye además la decisión sobre qué modelos y parámetros pueden ser los más apropiados.*
7. *Minería de datos. Consiste en la búsqueda de los patrones de interés en una determinada forma de representación o sobre un conjunto de representaciones, utilizando para ello métodos de clasificación, reglas o árboles, regresión, agrupación, etc.*
8. *Interpretación de los patrones encontrados. Dependiendo de los resultados, a veces se hace necesario regresar a uno de los pasos anteriores.*
9. *Consolidación del conocimiento descubierto. Consiste en la incorporación de este conocimiento al funcionamiento del sistema, o simplemente documentación e información a las partes interesadas.” (García, et al., 2018: 14).*

El *Knowledge Discovery in Database* (KDD) hoy se considera estrechamente relacionado con el concepto de Minería de Datos (la minería de datos se considera como el proceso mediante el cual se encuentran patrones en los datos). García, et al. (2018) menciona que es lo mismo que Machine Learning; aunque en años anteriores, la Minería de Datos se le consideraba solo una pequeña parte del proceso de KDD, sin embargo, para evitar complicaciones conceptuales y teóricas, ya que este no es el objeto de estudio, se mencionara, únicamente, 4 modelos de KDD mostrados por Squire (2016), en su libro “Mastering Data Mining with *Python* - Find Patterns hidden in your data”.

Proceso de KDD por Fayyad et al.

1. Selección de datos: De un conjunto de datos extraer un subconjunto de datos para procesar.
2. Pre - procesamiento de datos: Se eliminan datos raros o atípicos, así como la generación de datos faltantes. El realizar ésto se dice que se obtienen datos limpios.
3. Transformación de datos: Se organizan los datos de manera apropiada para trabajar con ellos, así como la reducción de variables o características de los datos.

4. Minería de Datos: Se realiza el proceso de extracción de patrones mediante diferentes algoritmos.
5. Interpretación/evaluación de los datos: Se evalúan los patrones para resolver el problema en cuestión.

Proceso de KDD por Han *et al.*

1. Limpieza de los datos.
2. Integración de datos: Los datos limpios se integra, en caso de que vengan de diferentes fuentes.
3. Selección de datos: Para el problema en cuestión.
4. Transformación de los datos: se le da la forma a los datos en formularios para la minería de datos.
5. Minería de datos: Se obtienen patrones a través de los algoritmos inteligentes.
6. Evaluación de patrones: Se evalúa(n) el(los) patrón(es) de acuerdo al interés del estudio y el problema en cuestión.
7. Representación del conocimiento: El conocimiento se comunica, añadiendo la visualización de los datos.

CRoss - Industry Standard Process for Data Mining (CRISP-DM process):

1. Entendimiento del negocio: El analista entiende el problema de minería de datos desde una visión comercial.
2. Comprensión de los datos: El analista se familiariza con los datos, sus posibles promesas respecto al negocio, y comienza a generar hipótesis. En caso de ser necesario re - evaluar el proyecto comercial.
3. Preparación de los datos:

- Selección de datos,
- integración de datos y,
- Pre - procesamiento.

Nota: No se espera seguir un proceso seriado, a diferencia de los modelos mencionados con anterioridad.

4. Modelado: Se aplican los algoritmos para encontrar los patrones; de ser necesario, se re - evalúa la preparación de los datos.
5. Evaluación: El modelo y los patrones se evalúan para el problema comercial en cuestión.
6. Aplicación: El modelo se ponen en producción, se aplica.

Como se puede observar, los métodos mencionados tienen grandes similitudes y convergencias en su flujo de trabajo. Para objeto de este estudio se toman elementos de los flujos de trabajo mencionados para generar el modelo de predicción financiero. El flujo de trabajo propuesto se muestra en la metodología.

1.4.4. Predicción y adaptación

Existen dos subconjuntos de modelos en el conjunto de la inteligencia artificial: el de predicción y el de adaptación. La predicción busca generar un cambio en el comportamiento antes de que se presente una perturbación en el sistema para, de este modo, lograr sus metas (Gershenson, 2019). Las predicciones tienen límites inherentes por la complejidad (Gershenson, 2019), ya que “*las interacciones relevantes de un sistema complejo pueden generar información que no está presente ni en condiciones iniciales, ni de frontera*” (Gershenson, 2019), “*por lo que no podemos saber a priori el estado futuro del sistema*” (Gershenson, 2019).

Por otro lado, la adaptación pretende resolver las limitaciones de las predicciones (Gershenson, 2019), por lo que busca generar “*la capacidad de cambiar el comportamiento en*

presencia de una perturbación” (Gershenson, 2019) y así lograr las metas deseadas (Gershenson, 2019).

Además de esos dos subconjuntos de modelos, la inteligencia artificial se enfrenta a dos subconjuntos de problemas: los estáticos y los dinámicos (Gershenson, 2019). Los primeros actúan en espacios estacionarios; por lo que utilizan la predicción, “*Generalmente se intenta predecir a través de la optimización*” (Gershenson, 2019), empleando métodos de búsqueda, técnicas heurísticas y meta - heurísticas, el cómputo evolutivo y el razonamiento automatizado. Con respecto a los problemas dinámicos actúan en espacios no estacionarios, por lo que emplean la adaptación, aunque depende de la escala temporal, ya que se puede usar métodos conexionistas (como redes neuronales) en escalas tiempo cortas.

1.4.5. Algoritmos de clasificación

En el mundo real, las respuestas a problemas, no siempre son de variables cuantitativas, sino de variables cualitativas (Caselle *et al.*, 2013), estas pueden ser: Color de ojos, género, gusto, entre otros. El proceso mediante el cual se busca predecir una variable cualitativa (categoría), se le conoce como clasificación, dicho proceso busca clasificar las observaciones observadas (Caselle *et al.*, 2013). Los problemas de clasificación son más comunes de lo que parecen, de hecho pueden ser más frecuentes que los problemas de regresión (Caselle *et al.*, 2013); por ejemplo: Un joven llega con problemas respiratorios, teniendo una serie de síntomas, similares a tuberculosis, pero las imágenes de rayos x muestran características similares a algún tipo de cáncer, ¿cuál es el diagnóstico correcto? ¿Cáncer o tuberculosis? Otro ejemplo: saber si una persona hizo una transacción fraudulenta en la banca en función de su uso y consumo de tarjeta de crédito.

Existe una gran gama de algoritmos para la solución de problemas de clasificación entre ellos: *Logistic Regression*, *Linear Discriminant Analysis*, *K-Nearest Neighbors*, *Support Vector Machines*, Redes Neuronales Artificiales. Estas últimas tienen la característica de funcionar como regresión (utilizando, por ejemplo una función lineal como función de activación de salida) (Marwalka, 2013), y como clasificador, dependiendo de la forma en que se introduzcan los datos y se usen las funciones de activación (las funciones de activación se verán más adelante); por tales motivos, y sobre todo, en las investigaciones mencionadas,

se hará uso de este tipo de algoritmo para desempeñar la tarea de pronóstico en esta tesis.

Astorga, (2005), pone en contraste la linealidad y no linealidad, en los problemas de clasificación:

“Para finales de 1970, muchas técnicas de aprendizaje fueron desarrolladas; sin embargo, eran exclusivamente métodos lineales, pues ajustar de forma no lineal la información era casi imposible con el poder computacional de aquellos días. En la actualidad, el poder computacional es superior y ajustar modelos no lineales a los datos se ha vuelto más accesible” (Astorga, 2015: 1).

Incluso, hoy en día puedes adquirir por menos de 4000 MXN (pesos de 2019) un TPU Coral Beta de Google en Internet, con el cual puedes acelerar el proceso de entrenamiento de una IA, o una Jetson de NVIDIA, tarjetas externas para el entrenamiento de IA; estas tarjetas dan una ventaja a bajo costo para los investigadores en IA, ya que son procesadores diseñados especialmente para entrenamiento de modelos de *Machine Learning*.

En la siguiente sección se hablará de las redes neuronales artificiales, de los diversos problemas a los que se puede enfrentar el investigador cuando realice el entrenamiento de alguna red neuronal artificial, sesgos en la información, *Underfitting*, entre otros.

1.4.6. Tipos de redes neuronales artificiales

Existen diferentes tipos de Redes Neuronales artificiales:

- De Aprendizaje Supervisadas: Donde los parámetros de la RN se ajustan en función del error obtenido de la salida esperada y la obtenida:
- De Aprendizaje No Supervisada: El cual dispone únicamente de valores de entrada, sin salida, donde los parámetros se ajustan por los elementos internos de la red, tratando de encontrar características similares de los datos de entrada;
- De Aprendizaje por Refuerzo;
- Y Mixta, las cuales pueden ser una mezcla de las anteriores.

El tipo de Red Neuronal más conocida es la red perceptrón multicapa, quizá se deba a su capacidad de generalización. El perceptrón multicapa puede considerarse como un modelo que busca minimizar el error.

Surge a partir del perceptrón, el cual simula una neurona, ese modelo es conocido por su capacidad de clasificación.

Perceptrón

“Éste [perceptrón] modelo se concibió como un sistema de capaz de realizar tareas de clasificación. /.../La información sobre la que se basaba el sistema estaba constituida por ejemplos existentes de las diferentes clases” (Viñuela & Galván, 2004:26).

A los ejemplos se les llama patrones o ejemplos de entrenamiento (Viñuela & Galván, 2004:26) , y gracias a esos patrones, *“se construye la superficie discriminante”* (Viñuela & Galván, 2004:26). A continuación se menciona la estructura del perceptrón de acuerdo a Viñuela & Galván, (2004):

- Conjunto de células de entrada, tantas como sea necesario.
- y una o varias células de salida
- cada célula de entrada está conectada a las células de salida
- Las conexiones determinan la superficie de discriminación.

Esquema del modelo

“Se trata de una estructura mono capa, en la que hay un conjunto de células de entrada, tantas como sea necesario, según los términos del problema; y una o varias células de salida. Cada una de las células de entrada tiene conexiones con todas las células de salida, y son estas conexiones las que determinan las superficies de discriminación del sistema” (Viñuela & Galván, 2004:26).

Para ejemplificar, se usara un modelo de un perceptron con dos entradas y una capa oculta. Se calcula la activación de la salida, mediante la suma ponderada de los pesos de entrada, como se muestra en la siguiente ecuación:

$$y' = \sum_{i=1}^n w_i x_i$$

(Cabe resaltar que los pesos w y el valor θ se inician de manera aleatoria en el rango de entre 0 y 1) La salida definitiva se pasa por una función de activación de acuerdo a Viñuela, & Galván (2004:27), en este caso, por razones prácticas se usa una función escalar. La función depende del umbral. Sin embargo es en apariencia, ya que el umbral θ sirve como factor de comparación.

$$1. y = F(y', \theta)$$

$$2. F(y', \theta) = \begin{cases} 1 & \text{si } y' > \theta \\ -1 & \text{si } y' \leq \theta \end{cases}$$

Reescribiendo:

$$1. s(y', \theta) = y' + \theta = \sum_{i=1}^n w_i x_i + \theta$$

$$2. F(s) = \begin{cases} 1 & \text{si } s > 0 \\ -1 & \text{si } s \leq 0 \end{cases}$$

y como se observa, cuando $s > 0$ el resultado de la función es 1 lo que indica que la información de la entrada pertenece a la clase A y cuando $s \leq 0$ el resultado de la función es -1, y por tanto la información de la entrada pertenece a la clase B .

Y, dado que tenemos dos entradas, la ecuación:

$$y = F\left(\sum_{i=1}^n w_i x_i + \theta\right)$$

se convierte en:

$$w_1 x_1 + w_2 x_2 + \theta = 0$$

donde la pendiente es:

$$-\frac{w_2}{w_1}$$

y su intersección es:

$$-\frac{\theta}{w_1}$$

(Viñuela & Galván, 2004:28)

Cabe resaltar, que de acuerdo a Viñuela & Galván (2004), las redes neuronales artificiales no cuentan con medidas como varianza o significancia, y sólo se miden por la división entre el número de aciertos obtenidos y el número de datos con que se hizo el entrenamiento.

Descenso del gradiente (o regla delta)

Antes de iniciar con el perceptrón multicapa se mostrara el algoritmo del descenso del gradiente, el cual es fundamental para el proceso de aprendizaje de un red neuronal perceptrón multicapa.

La idea del Descenso del Gradiente es realizar un cambio en cada peso (parámetro) proporcional a la derivada del error, medida en el patrón actual, respecto de los pesos:

Error:

$$E = \sum_{p=1}^m E^p = \frac{1}{2} \sum_{p=1}^m (d^p - y^p)^2$$

Donde d es la salida esperada y y la salida obtenida.

Ecuación del Descenso del Gradiente:

$$\Delta_p w_j = -\gamma \frac{\partial E^p}{\partial w_j}$$

Dado que no hay función de activación en la neurona de salida, tenemos:

$$\frac{\partial y^p}{\partial w_j} = x_j$$

$$\frac{\partial E^p}{\partial w_j} = -(d^p - y^p)$$

Sustituyendo:

$$\Delta_p w_j = \gamma (d^p - y^p) x_j$$

Uno de los problemas que tiene este algoritmo, es que recorre el espacio del error hasta encontrar un punto mínimo. Es decir, puede caer en algún mínimo local, perjudicando el

proceso de aprendizaje; ya que en el entrenamiento puede converger en el mínimo local. Para solucionar este tipo de problemas, por lo general, se añade ruido al modelo.

Perceptrón Multicapa

Ya que vimos las bases de una red neuronal tipo perceptrón multicapa, podemos resumir que esta es una extensión del perceptrón simple, siendo una red multicapa con conexiones hacia adelante (Viñuela & Galván, 2004). En la cual se usará la regla delta generalizada, para modificar los parámetros de la red. Este algoritmo se basa en el descenso del gradiente o regla delta. Y en esta tesis, al revisar el algoritmo Backpropagation, se encuentra que es lo mismo, sin embargo en la literatura menos reciente se le llama regla delta generalizada.

A continuación se mostrara la estructura de una red neuronal artificial tipo perceptrón multicapa o *full layer* (como se le llama en literatura más reciente); este tipo de red se compone de tres capas:

1. Capa de entrada,
2. Capa oculta (puede haber más de una capa oculta) y,
3. Capa de salida.

El proceso de activación de las neuronas de la red neuronal es el siguiente.

- Las neuronas de la primera capa (entrada), no tienen ninguna función de activación. Y ellas transmiten la información a la capa oculta (Viñuela, & Galván, 2004:26). Por tanto:

$$a_i^1 = x_i, \text{ para } i = 1, 2, \dots, n_1$$

donde:

$$X = \{x_1, x_2, \dots, x_{n_1}\}$$

- Las neuronas de la capa oculta, reciben y procesan la información, ésta cuenta con una función de activación:

$$a_i^c = f \left(\sum_{j=i}^{n_{c-1}} w_{ji} a_j^{c-1} + u_i^c \right) \text{ para } i = 1, 2, \dots, n_c \text{ y } c = 2, 3, \dots, C - 1$$

donde a_j^{c-1} es de activación de la capa c-1 (Viñuela, & Galván, 2004:26).

- Las neuronas de la capa de salida tienen la siguiente activación:

$$y_i = a_i^C = f \left(\sum_{j=i}^{n_{C-1}} w_{ji}^{C-1} a_j^{C-1} + u_i^C \right) \text{ para } i = 1, 2, \dots, n_C$$

para

$$Y = (y_1, y_2, \dots, y_{n_C})$$

(Viñuela, & Galván, 2004:26)

Funciones de Activación mas usadas

- Sigmoidal:

$$f(x) = \frac{1}{1 + e^{-x}}$$

- Tangente Hiperbólica:

$$f(x) = \frac{1 - e^{-x}}{1 + e^{-x}}$$

- ReLU

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

- PreLU

$$f(x) = \begin{cases} \alpha x & \text{si } x < 0 \\ x & \text{si } x \geq 0 \end{cases}$$

- Softmax

$$f(x)_k = \frac{e^{x_k}}{\sum_{j=1}^k e^{x_j}}$$

- Donde k es el número de clases.
- Donde x es el vector que tiene los puntajes de cada clase.
- $f(x)$ es la probabilidad estimada de x para cada clase k .

Backpropagation

Aquí se verá el método mediante el cual se modifican los parámetros de la red neuronal (pesos y umbrales). En el caso del perceptrón multicapa se busca minimizar el error, tratando de que la salida obtenida sea lo más parecida a la salida esperada. El problema se formula de la siguiente manera:

$$\text{Min}_W E$$

W = Parámetros (pesos y umbrales)

E = Función de error o función de coste

Generalmente la función de coste es (Aunque no es la única):

$$E = \frac{1}{N} \sum_{n=1}^N e(n)$$

$$e(n) = \frac{1}{2} \sum_{i=1}^{n_c} (s_i(n) - y_i(n))^2$$

s = salida deseada

y = salida obtenida

$$S_i(n) = (s_1(n), \dots, s_{n_c}(n)), \text{ (Salida esperada)}$$

$$Y_i(n) = (y_1(n), \dots, y_{n_c}(n)), \text{ (Salida obtenida)}$$

Aplicando el descenso del gradiente estocástico:

$$s(n) = w(n-1) - \alpha \frac{\partial e(n)}{\partial w}$$

Cálculo de Backpropagation

La función de coste o de error, como se dijo en la sección anterior, no es la única, aquí se calcula el Backpropagation con la función de coste mostrada líneas atrás, y la función de activación que se usará es la sigmoide, ya que es más sencillo realizar las operaciones para obtener la ecuación del Backpropagation.

Para el caso de la capa oculta C-1 a la capa de salida y umbrales de la capa de salida.

$$w_{ji}^{C-1}(n) = w_{ji}^{C-1}(n-1) + \alpha \frac{\partial e(n)}{\partial w_{ji}^{C-1}}$$

$$\frac{\partial e(n)}{\partial w_{ji}^{C-1}} = -(s_i(n) - y_i(n)) \frac{\partial y_i(n)}{\partial w_{ji}^{C-1}}$$

$$\frac{\partial y_i(n)}{\partial w_{ji}^{C-1}} = f' \left(\sum_{j=1}^{n_{C-1}} w_{ji}^{C-1} a_j^{C-1} + u_i^C \right) a_j^{C-1}(n)$$

$$\delta_i^C(n) = -(s_i(n) - y_i(n)) f' \left(\sum_{j=1}^{n_{C-1}} w_{ji}^{C-1} a_j^{C-1} + u_i^C \right)$$

$$\frac{\partial e(n)}{\partial w_{ji}^{C-1}} = \delta_i^C(n) a_j^{C-1}(n)$$

y al final obtenemos la ley para modificar los pesos:

$$w_{ji}^{C-1} = w_{ji}^{C-1}(n-1) + \alpha \delta_i^C(n) a_j^{C-1}(n)$$

$$\text{para } j = 1, 2, \dots, n_{C-1} \quad i = 1, 2, \dots, n_C$$

Esto se generaliza para los umbrales

$$u_i^C(n) = u_i^C(n-1) + \alpha \delta_i^C(n) \quad \text{para } i = 1, 2, \dots, n_C$$

Para el caso de la capa c a la capa c + 1 y umbrales de las neuronas de la capa c + 1 para c = 1, 2, ..., C-2. Se inicia desde esa capa para entender el proceso y luego generalizarlo para las demás capas (Viñuela, & Galván, 2004:26):

$$w_{kj}^{C-2} = w_{kj}^{C-2}(n-1) + \delta \frac{\partial e(n)}{\partial w_{kj}^{C-2}}$$

$$\frac{\partial e(n)}{\partial w_{kj}^{C-2}} = - \sum_{i=1}^{n_C} (s_i(n) - y_i(n)) \frac{\partial y_i(n)}{\partial w_{kj}^{C-2}}$$

$$\frac{\partial y_i(n)}{\partial w_{kj}^{C-1}} = f' \left(\sum_{j=1}^{n_{C-1}} w_{ji}^{C-1} a_j^{C-1} + u_i^C \right) w_{ji}^{C-1} \frac{\partial a_j^{C-1}}{\partial w_{kj}^{C-2}}$$

Sustituimos

$$\frac{\partial e(n)}{\partial w_{kj}^{C-2}} = \sum_{i=1}^{n_C} \delta_i^C(n) w_{ji}^{C-1} \frac{\partial a_j^{C-1}}{\partial w_{kj}^{C-2}}$$

$$\frac{\partial a_j^{C-1}}{\partial w_{kj}^{C-2}} = f' \left(\sum_{k=1}^{n_{C-2}} w_{kj}^{C-2} a_k^{C-2} + u_j^{C-1} \right) a_k^{C-2}(n)$$

δ para las neuronas de la capa $C-1$, $\delta_j^{C-1}(n)$, teniendo, luego entonces:

$$\delta_j^{C-1} = f' \left(\sum_{k=1}^{n_{C-2}} w_{kj}^{C-2} a_k^{C-2} + u_j^{C-1} \right) \sum_{i=1}^{n_C} \delta_i^C(n) w_{ji}^{C-1}$$

Sustituyendo en la derivada del error respecto a los pesos:

$$\frac{\partial e(n)}{\partial w_{kj}^{C-2}} = \delta_j^{C-1}(n) a_k^{C-2}(n)$$

Y dado que:

$$w_{kj}^{C-2} = w_{kj}^{C-2} + \alpha \delta_j^{C-1}(n) a_k^{C-2}(n)$$

para $k = 1, 2, \dots, n_{C-2}$ y $j = 1, 2, \dots, n_{C-1}$

Luego entonces, generalizamos para n capas ocultas:

$$w_{kj}^C = w_{kj}^C(n-1) + \alpha \delta_j^{c+1}(n) a_k^c(n)$$

para $j = 1, 2, \dots, n_{c+1}$ y $c = 1, 2, \dots, C-2$

Dado:

$$\delta_j^{c+1} = f' \left(\sum_{k=1}^{n_c} w_{kj}^c a_k^c + u_j^c \right) \sum_{i=1}^{n_{c+1}} \delta_j^{c+1}(n)$$

Es posible generalizarlo para los umbrales:

$$u_j^{c+1}(n) = u_j^{c+1}(n-1) + \alpha \delta_j^{c+1}(n)$$

para $j = 1, 2, \dots, n_{c+1}$ y $c = 1, 2, \dots, C-2$

Word Embedding

Word Embedding, es un método para convertir palabras a vectores, estos vectores capturan la relación semántica entre palabras (Kulkarni, & Shivananda, 2019:82). El *Word Embedding* es una red neuronal artificial, la cual consiste en una capa de entrada, una capa oculta y una capa de salida, recibiendo como entrada un conjunto de palabras pre-procesadas con *tokenizer*, teniendo como entrada, una secuencia de palabras y como salida, la palabra que se encuentra en medio de ellas o bien, el otro método consiste en introducir una sola palabra y luego las palabras que se encuentran a los lados de ella. Una vez terminado el proceso, la capa oculta es la que se toma, ya que al introducir una palabra, los pesos de la capa oculta retornan el vector. Los resultados de ambas vías (de los métodos) ofrecen resultados similares.

Word2vec es un *framework* de *deep learning* de *Google*, para entrenar una red neuronal tipo *Word Embeddings* (Kulkarni, & Shivananda, 2019).

En el diseño de la red neuronal del presente trabajo, la primera capa oculta será una capa *Word Embedding*, con lo cual el proceso de creación de vectores, se realizará al momento de introducir la información y entrenamiento de la red neuronal.

1.4.7. Validación

De acuerdo al *Curso intensivo de aprendizaje automático con TensorFlow* de *Google*, en un modelo de *Machine Learning* se deben de separar los datos en 3 conjuntos, conjunto de entrenamiento, validación, y prueba; ya que si solo se tiene conjunto de prueba y conjunto

de entrenamiento, no podremos observar un comportamiento adecuado de nuestro modelo, o podemos caer en errores de sobre ajuste, ya que no se hizo una validación antes de realizar la prueba (test). Por lo general para realizar el test, se considera tomar el 20 % de los datos (Gerón, 2019).

La validación es un paso importante para seleccionar el modelo que “mejor” se ajuste a los datos, una vez se encuentra un modelo que “mejor” se adapta a los datos, el modelo debe ser entrenado con todos los datos (Gerón, 2019).

1.4.8. Overfitting y Underfitting

El *Overfitting* y *Underfitting*, son problemas que afectan a los modelos de *Machine Learning*, en nuestro caso puede ser un pronóstico erróneo. A continuación se realiza una breve explicación del *Overfitting* y *Underfitting*.

El *Overfitting*, también conocido como sobreajuste, es un problema típico en modelos de *Machine Learning*, cuando hay un proceso de aprendizaje; dicho problema surge cuando el algoritmo de inteligencia se sobre adapta a los datos, lo cual impide la generalización de los mismos, y no generar soluciones adecuadas cuando se le presentan nuevos datos; como analogía podemos decir que un agente quiere pasar un examen, para pasarlo, puede elegir dos estrategias: estudiar los conceptos para resolver los problemas, o aprender cada problema paso a paso para solucionar el examen, es decir: memorizar. En la primera estrategia el agente produce una generalización para la solución de problemas nuevos, mientras que por otro lado, la segunda estrategia, cuando se presenten nuevos datos, no podrá solucionar el problema, ya que solo memorizó un conjunto de problemas, por tanto, si encuentra un problema nuevo no sabrá como resolverlo, teniendo, por tanto, pésimos resultados; a esto en inteligencia artificial, se le conoce como *Overfitting*.

Por otro lado el *Underfitting* produce resultados similares al *Overfitting*, ya que no puede hacer una generalización del modelo, la diferencia radica en que el *Underfitting* tuvo una menor cantidad de entrenamiento para aprender de los datos; refiriéndonos a la analogía del párrafo anterior, el estudiante, no aplicó ninguna estrategia, es decir: no estudió.

Para evitar el *Overfitting* se requiere de experiencia, sin embargo se puede hacer uso de

diferentes técnicas como: disminuir el número de épocas de entrenamiento; agregar ruido a los modelos, por ejemplo, en Redes Neuronales, se hace mediante el apagado neuronas de manera aleatoria durante el entrenamiento (“*Dropout*”). Otro modo de evitar el *Overfitting*, es hacer menos complejo el algoritmo, o bien, agregando mayor cantidad de datos; también se suele pre - procesar los datos de otro modo, entre otras técnicas.

Por lo que considerar el *Overfitting*, *Underfitting*, y las formas de evitarlo, es menester para lograr un pronóstico preciso.

1.5. Polinomio de Jacobi

En los diferentes trabajos mostrados en la sección 1.2, 1.2, y 1.4.3 se muestra que en la mayoría de los casos es necesario completar espacios vacíos en las bases de datos, o se necesitan más datos de los disponibles para entrenar los algoritmos de inteligencia artificial supervisados, en tales casos se hace uso de diferentes métodos para generar esos datos, como: *Block bootstrap*, y redes neuronales artificiales, entre otros. Otro tipo de modelos matemáticos son los polinomios ortogonales de Jacobi. A continuación se define:

Los polinomios de Jacobi son ortogonales en el intervalo $[-1, 1]$ con una función de peso $\rho(x) = (1 - x)^\alpha(1 + x)^\beta$ (Szegő, 1958), donde $\alpha > -1$ y $\beta > -1$ y el polinomio tiene la forma de :

$$P_n^{\alpha,\beta}(x) = \frac{\Gamma(n + \alpha + 1)}{\Gamma(\alpha + 1)n!} {}_2F_1 \left(-n, 1 + \alpha + \beta + n; \alpha + 1; \frac{1 - x}{2} \right)$$

Donde:

$$\int_{-1}^1 P_n^{\alpha,\beta}(x) P_m^{\alpha,\beta}(x) \rho(x) dx = \delta_{mn} \frac{2^{\alpha+\beta+1} \Gamma(n + \alpha + a) \Gamma(n + \beta + a)}{(2n + \alpha + \beta + a) \Gamma(n + \alpha + \beta + 1) n!}$$

Donde $\Gamma(x)$ es la función Gamma de Euler:

$$\Gamma(x) = \int_0^\infty e^{-x} x^{z-1} dt$$

${}_2F_1$ es la función hipergeométrica de Gauss:

$${}_2F_1 = \sum_{k=0}^{\infty} [\text{frac}(a)_k (b)_k z^k (c)_k k!]$$

Donde a , b , c son parámetros reales o complejos y z un número complejo.

La δ_{mn} es la función delta de Kronecker:

$$\delta_{mn} = \begin{cases} 0 & \text{si } m \neq n \\ 1 & \text{si } m = n \end{cases}$$

Cabe destacar que el programa del Dr. Juan Martín Carpió Valadez que se usa para generar el polinomio de Jacobi utiliza mínimos cuadrados para el ajuste el modelo.

Capítulo 2

Metodología

En este capítulo se muestra el tipo de investigación a realizar, el objeto de estudio, periodo de análisis y los pasos a seguir en la investigación.

2.1. Tipo de investigación

El presente trabajo es de tipo cuantitativo, dado que se realizará un pronóstico de las variaciones positivas o negativas del Índice S&P500. Para realizar el análisis se usa una red neuronal artificial, la cual buscará en las noticias (emitidas una hora antes de que suceda la variación en el índice) los patrones que pronostican las variaciones mencionadas líneas atrás.

2.2. Objeto de estudio

La presente tesis opta por usar el S&P500 por ser representativo como indicador económico de la industria de Estados Unidos y, la relativa facilidad de acceso a ellos; ya que al intentar obtener el INPC (Índice Nacional de Precios y Cotizaciones) de México, implicaba un costo de adquisición no asequible para esta investigación, por otro lado el Banco de México ofrece la base de datos, sin embargo el periodo histórico era reducido, y en el caso de otras plataformas no eran confiables para realizar la investigación, como Investing, Yahoo, y Google, las cuales ofrecían la base de datos de manera gratuita. Mientras que

el S&P500 se ofrecía en la plataforma oficial de la Reserva Federa (Fed, por sus singlas en inglés) de Estados Unidos(<https://fred.stlouisfed.org>) de manera gratuita y con un amplio periodo histórico.

La elección de usar el noticiero Reuters (<https://www.reuters.com/>) se debe a la relativa facilidad de realizar un programa que automatizara la descarga de las noticias; ya que hacer el mismo programa automatizado para otras páginas como por ejemplo: el Financiero o El Economista, implica un mayor conocimiento de WebScraping (método para descargar contenido de la web), y otros portales de noticias como New York Time requieren suscripción.

2.3. Modelo de clasificación de variaciones positivas y negativas

En la presente sección se muestra el modelo propuesto para realizar el pronóstico:

Tenemos la ecuación:

$$f(N_t) = X_{t+1}$$

Donde X_{t+1} es la variación positiva o negativa del Índice S&P 500, en el tiempo $t + 1$, donde la unidad de t es en horas, y donde N_t es la noticia en el tiempo t , donde la unidad de t es en horas.

Y, dado que trabajamos el modelo de la red neuronal perceptrón multicapa como un modelo de clasificación, en el cual, él, buscará el hiperplano que separa las clases A , y B , donde, respectivamente, son las variaciones positivas y negativas, tenemos que:

$$\mathbb{R}^n; A = \{\vec{a}_1, \dots, \vec{a}_{na}\} \text{ y } B = \{\vec{b}_1, \dots, \vec{b}_{nb}\},$$

entonces el hiperplano es:

$$w_1x_1 + \dots + w_nx_n + \theta = 0$$

de tal forma que:

$$\forall \vec{a} \in A : w_1 a_1 + \dots + w_n a_n + \theta > 0$$

$$\forall \vec{b} \in B : w_1 b_1 + \dots + w_n b_n + \theta \leq 0$$

luego, con el entrenamiento la red neuronal encontrará y , la cual es la salida de la red neuronal, que puede ser 0 ó 1.

$$\forall \vec{a} \in A : y(\vec{a}) = 1$$

$$\forall \vec{b} \in B : y(\vec{b}) = 0$$

De tal modo que la red neuronal recibirá la información de las noticias, y hará una discriminación para pronosticar la variación positiva o negativa del Índice S&P500.

Para realizar el análisis, se realizó un pre - procesamiento de los datos, eliminación de puntuaciones, caracteres especiales y, realización de *Tokenizing*.

Después se diseñó la arquitectura de la red neuronal, tipo perceptrón multicapa, del siguiente modo:

- Una capa de entrada de 400 neuronas
- Capa *Embedding* de 500 neuronas; esta capa es la red neuronal pre - entrenada que ofrece Google.
- Capa oculta de 33 neuronas y una función de activación ReLU
- Capa oculta de 15 neuronas y una función de activación tangente hiperbólica
- Capa de salida de 2 neuronas y una función de activación *softmax*.
- Nota: Los pesos de las dos capas ocultas se iniciaron en una distribución normal.

2.4. Pasos

La metodología sigue un proceso similar al de creación de modelos de pronóstico de *Machine Learning* y el KDD; en este trabajo se sigue del siguiente proceso:

1. Crear base de datos: Obtener las noticias de Reuters, e Índice S&P500;
2. Pre - procesar datos: En esta etapa se pre - procesan los datos, limpian y completan;
3. Realizar red neuronal artificial;
4. Probar diferentes arquitecturas de redes neuronales artificiales para encontrar una que mejor se ajuste a los datos.

2.4.1. Crear base de datos

Obtener las noticias de Internet para el período de 2013 a 2019 (del portal de noticias <https://www.reuters.com>), se realizó un método de obtención de información de documentos contenidos del HTML de diferentes enlaces de la página de noticias Reuters enfocadas únicamente en la versión en inglés estadounidense. El HTML es el formato que se usa para presentar una página web; el proceso de obtención de contenido de una página web se conoce como Webscraping. Para este proceso se hizo uso de beautifulsoup4, de Python; para comprender cómo se usó esta herramienta imaginemos que tenemos el siguiente documento HTML:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Página Web de www.reuters.com</title>
  </head>
  <body>
    <header>
      <h1>Título de la noticia</h1>
    </header>
```

```
<p> Texto de la noticia </p>
</body>
</html>
```

Y lo que nosotros deseamos hacer es obtener el texto de la noticia que se encuentra en la página web, la cual, en código, se ve en el formato anterior. Supongamos que el contenido de nuestro interés se encuentra en el párrafo `< p >< /p >`. Ahora, bajo el supuesto que este documento se encuentra guardado en tu computadora como *noticiatext.html* y que ya se han instalado las librerías necesarias, y que además se trabaja mediante un *Jupyter Notebook*, pasaremos, enseguida, al proceso de *scraping*:

```
# Primero, importamos las librerías necesarias.
# Si no se ha instalado, y suponiendo que se está haciendo uso de
# un notebook, se puede hacer uso del siguiente comando para
# instalar la librería
# !pip install beautifulsoup4
# Ya instalado importarlo
from bs4 import BeautifulSoup

soup = BeautifulSoup(noticiatext.html, 'html.parser')

text = soup.p

text = text.get_text()

print(text)

>>> Texto de la noticia
```

Si se desea ver el ejemplo con mayor claridad puede entrar al siguiente link:

<https://colab.research.google.com/drive/1ukzV6EdxESSinrCcPM07cSsK0A3AVnw7>.

Si se desea saber más sobre BeautifulSoup puede leer la documentación oficial BeautifulSoup4.

Haciendo énfasis y aclaraciones, se obtuvieron las noticias de la url <https://www.reuters.com/> de la edición de Estados Unidos de periodo de 2013 a 2019. Decidiendo obtener las noticias de dicha página debido a su menor complejidad en su estructura de código HTML, ya que de elegir otro tipo de portales *online* de noticias, exigía un conocimiento más avanzado de técnicas de WebScraping. El Índice S&P500 fue obtenido de la página <https://fred.stlouisfed.org>, la pagina de base de datos económicos de la *Federal Reserve*. Se obtuvo una base de datos del periodo 2012-01-09 al 2019-12-09, con 73 datos NaN y 1993 datos del cierre del día. Los datos NaN, se sustituyeron por el dato del cierre del día anterior.

2.4.2. Pre - procesamiento de datos

En este proceso se hizo una limpieza del texto, quitando caracteres especiales, puntos, comas, haciendo uso de una herramienta open source llamada NLTK escrita en Python.

Para convertir los datos del cierre del día a datos por hora del Índice S&P500, se hizo un polinomio de Jacobi de grado 160, para el proceso de ajuste del polinomio de Jacobi, se usaron datos suavizados de media móvil de grado 9; se usó el programa realizado por el Dr. Juan Martín Carpió Valadez, del Instituto Tecnológico de León. Dado el polinomio se estimarán los valores por hora. Se puede consultar el modelo obtenido en el siguiente enlace: <https://bit.ly/34IoQv9>.

2.4.3. Entrenamiento de Red Neuronal Artificial y prueba de diferentes arquitecturas

Posteriormente se iteró entre diferentes arquitecturas de redes neuronales artificiales para encontrar la que mejor se ajusta a los datos.

Para el entrenamiento se dividieron los datos en tres conjuntos:

1. Conjunto de entrenamiento

2. Conjunto de validación
3. Conjunto de prueba

Esto de acuerdo con las ideas del Curso Intensivo de Aprendizaje Automático de *Google*, también con las ideas del curso *Introduction to Machine Learning by Tensorflow* y el proceso de *Workflow Machine Learning*.

2.5. Alcances y consideraciones

En este trabajo, se hará el análisis con un único noticiero, ya que si bien, como se ve en el marco teórico, existen diferentes trabajos donde usan más portales de noticias y más variables, eso implicaría mayor cantidad de trabajo y aprender más conocimiento, lo cual, evidentemente, lleva más tiempo. Por otro lado, de acuerdo a los flujos de trabajo que se mencionan en la sección 1.4.4 llamada *Machine Learning*, etapas del proceso de *Big Data* y *Knowledge Discovery in Database* (KDD), se expone que para hacer un pronóstico con *Machine Learning* se deben probar diferentes tipos de algoritmos para encontrar el modelo que mejor se ajusta a los datos; realizar eso lleva más tiempo, tanto de aprendizaje como de entrenamiento de los algoritmos; por lo cual, solo se probarán diferentes arquitecturas de redes neuronales para encontrar la que mejor se ajusta a los datos, cabe aclarar, en el supuesto de que se prueben diferentes modelos de IA, en las RNA siempre se tendrá que probar diferentes arquitecturas de la red, ya que no siempre en la primer propuesta del modelo se encuentra la más optima. Cabe aclarar que para aplicar redes neuronales, uno de los pasos es ese, encontrar la arquitectura que mejor se ajusta a los datos; tampoco se crearan los programas desde cero para entrenar las redes neuronales, por lo que se usara el FrameWork de Keras para reutilizar el código, modificarlo, y tener las redes neuronales en una menor cantidad de tiempo. Esto también se aplicará para el pre - procesamiento de los datos, usando librerías y FrameWorks como Matplotlib, NLTK, Pandas, Numpy, entre otras herramientas; las cuales ayudan a realiza el trabajo en una menor cantidad de tiempo.

Otro detalle a considerar es que las redes aprenden de los datos que se les suministre,

pudiendo causar sesgos. En este caso se usa un solo noticiero, lo cual puede causar sesgos. Para solucionar este inconveniente se debe entrenar la Red Neuronal Artificial con más información, no obstante, entrenar la RNA con más datos de diferentes portales de noticias, implicaría adquirir mayor conocimiento sobre Webscraping, conocimiento que llevaría más tiempo adquirirlo, lo cual sobre pasa los tiempos de esta investigación.

Cabe también, aclarar que al revisar las fechas de las noticias descargadas y contrastarlas de la URL de donde se tomó dicha información, diferían entre una y dos horas. Al ser más de 40 mil noticias, se hace imposible revisar todas y hacer que coincidan. No se sabe porque se genera esta diferencia; puede ser a los *metadatos* del momento exacto en que se subieron o algún otro tipo de motivo. Es por ello que el pronóstico realizado se hace con las fechas obtenidas de las noticias descargadas.

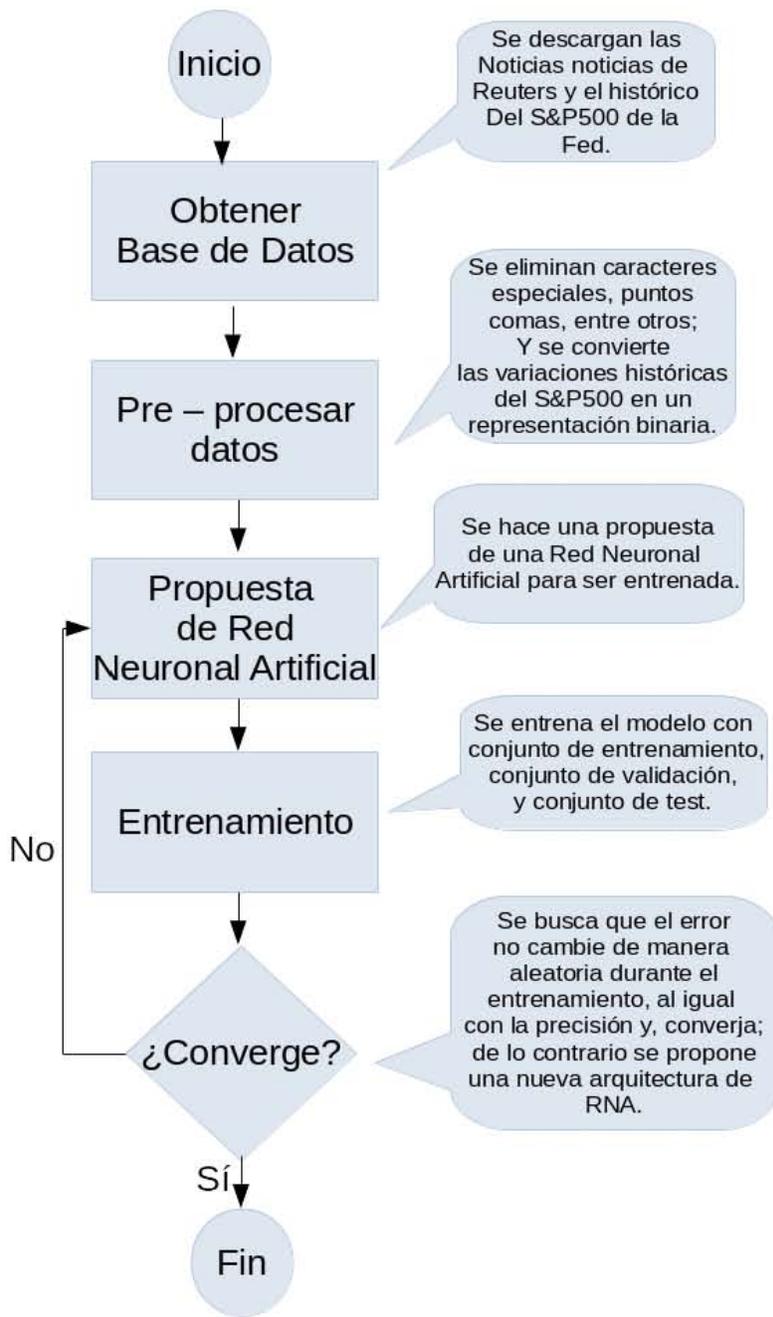


Diagrama 2.1: Diagrama de flujo de trabajo.

Capítulo 3

Análisis y Resultados

En el presente capítulo, se muestran los resultados obtenidos de entrenar diferentes arquitecturas de redes neuronales artificiales, el análisis de lo obtenido, así como la interpretación de los datos.

Para poder entrenar la RNA, se hizo uso de la base de datos del S&P500 generada por la interpolación polinomial de Jacobi. Cabe destacar que el Error Cuadrático Medio de la interpolación fue de 29.368332967152 y el grado del modelo es de 160. Como el objeto de estudio no es el análisis de este modelo, y evitar llenar la tesis de varios datos se decide agrega un link con el modelo ajustado: <https://bit.ly/3euY9yA>.

La red neuronal se entrenó con 169 ciclos, un factor de aprendizaje de 0.00001, se hizo uso del algoritmo *backpropagation* aplicando el descenso del gradiente estocástico, promediando 40 datos con la función de error cuadrático medio mostrada en el apartado Descenso del gradiente (o regla delta) de la sección 1.4.6. *Tipo de redes neuronales artificiales* del marco teórico.

Se probaron 20 arquitecturas de Redes Neuronales Artificiales para encontrar la arquitectura que ‘mejor’ se ajusta a los datos. Es importante resaltar que el modelo número 20, 21, 22 y 23 es la misma arquitectura entrenada con sus pesos iniciados de manera diferente, como se ve en las gráficas 3.1, 3.2, 3.25, 3.26, 3.27, 3.28 no obstante, se guardaron y presentaron los resultados del modelo 22, ya que surgieron complicaciones al guardar el modelo 20, y los datos históricos del modelo 21 se reescribieron a mano para posteriormente hacer la gráfica, no obstante a falta de decimales, la gráfica pierde suavidad, y no se aprecia

con precisión el entrenamiento, por lo que se entrenó una vez más el modelo, teniendo el modelo 22. El modelo 23 es un modelo de anexo entrenado con todos los datos, el cual se puede consultar en los anexos, ya que, como se ve en la sección 1.4.7 del marco teórico titulada Validación, al encontrar el “mejor” modelo, este debe ser entrenado con todos los datos para ponerlo en producción, no obstante la precisión se consulta con el modelo hecho con los 3 conjuntos (de entrenamiento, validación y prueba). Por tales razones se presenta el modelo 22.

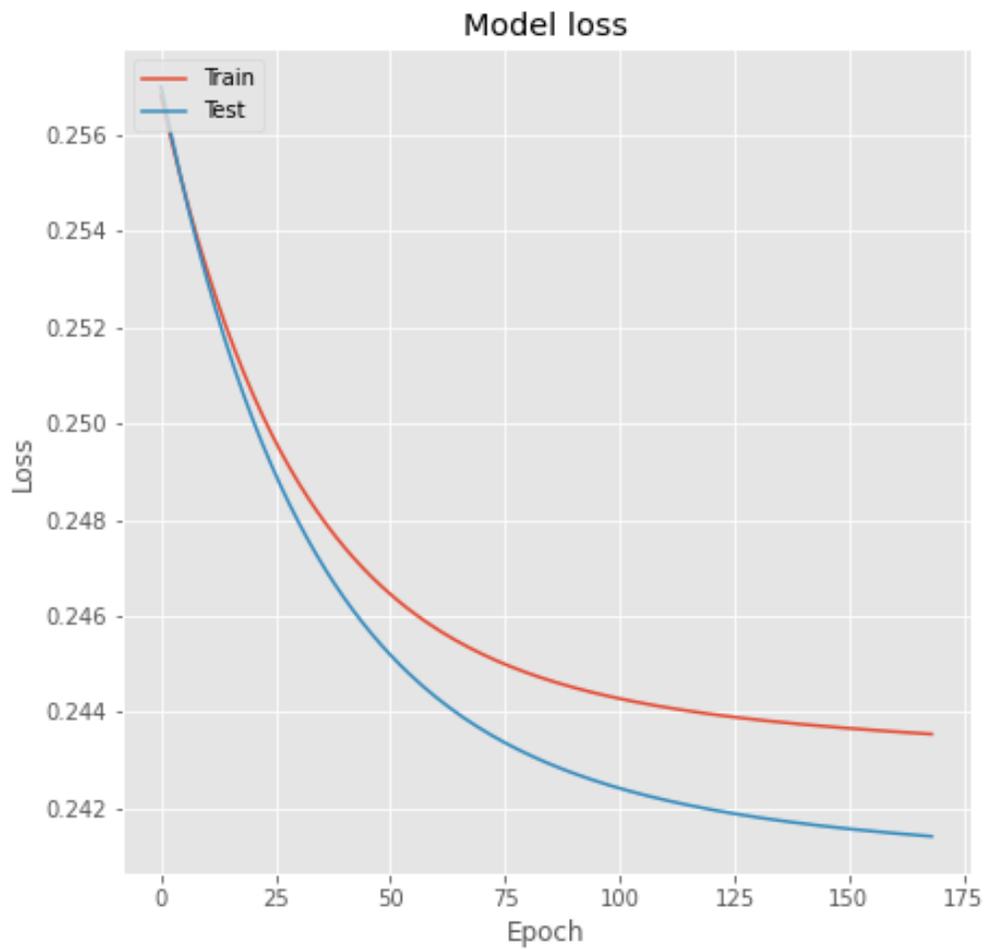
El error cuadrático medio (E) obtenido en el entrenamiento del modelo 22 fue de: 0.2435; y el error (E) de validación fue de: 0.2414, teniendo una precisión de entrenamiento de 0.5839, y de validación de 0.5973; después se comprobó la precisión con datos que no se usaron para entrenar el modelo, teniendo una precisión de 0.5861, por otro lado para las variaciones positivas y negativas, los resultados fueron: 0.59145129 y 0.42366412, respectivamente. Es menester resaltar que el concepto de validación se usa para decir que la precisión del modelo con datos de validación fue de n porcentaje y el concepto precisión para decir que el modelo tuvo cierta precisión con datos de prueba (test), es decir, datos que jamás se usaron para el entrenamiento.

Cabe agregar que el entrenamiento se hizo con 26590, la validación con 6648, el test con 8310, de este último 3399 para variaciones negativas, y 4911 para variaciones positivas.

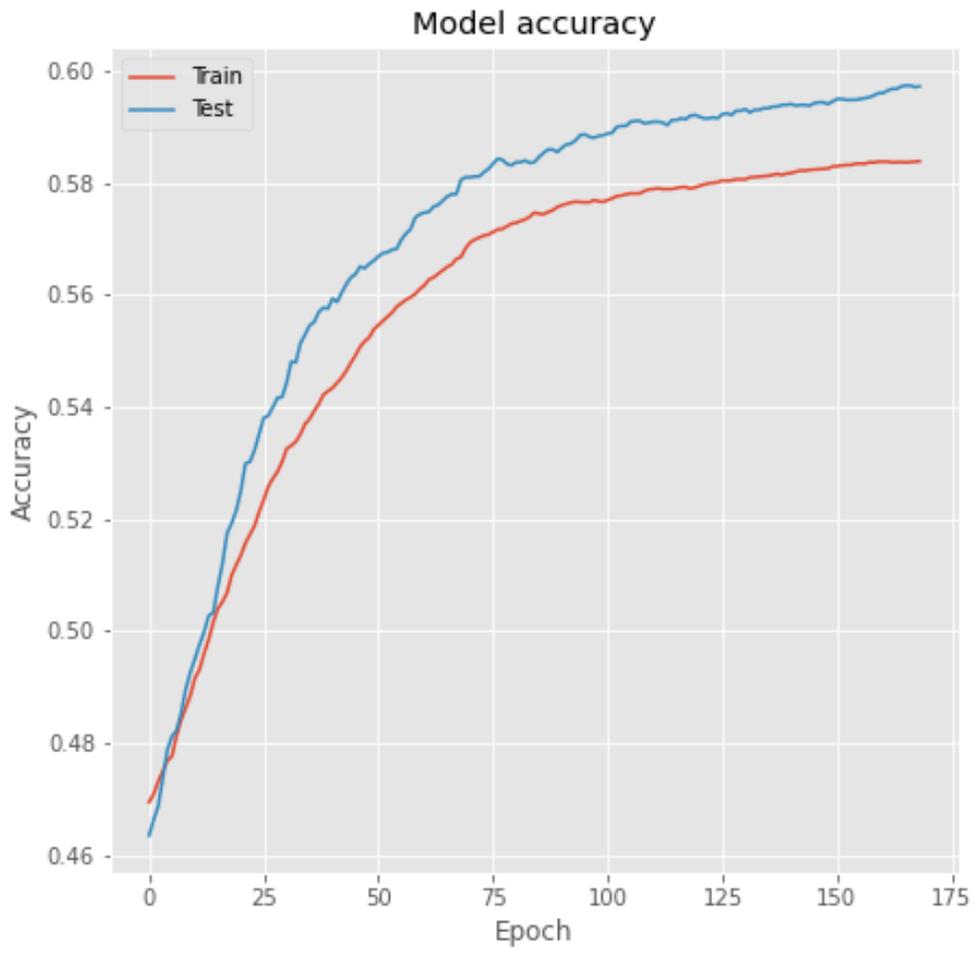
En la gráfica 3.1 se muestra el error cuadrático medio que se obtuvo con datos de entrenamiento y datos de validación, en la gráfica 3.2 se muestra la precisión del modelo con datos de entrenamiento y datos de validación. Como se puede apreciar en cada epoch (se le dice epoch a cada ciclo de entrenamiento) el error va disminuyendo y la validación aumenta, pero la velocidad con la que sube esta última va decayendo conforme la red va aprendiendo en cada época, de igual manera la velocidad con la que baja el error va disminuyendo; cuando comienza a observarse un cambio mínimo en las gráficas es por que red esta convergiendo, no obstante comienza a generarse un *Overfitting*, por lo que se recomienda detener el entrenamiento en éste punto; de igual manera, se debe detener el proceso de entrenamiento cuando la precisión obtenida con datos de entrenamiento comienza a aumentar y la precisión con datos de validación comienza a bajar, ya que es señal de *Overfitting*. Por esa razón se decidió parar el entrenamiento en el ciclo número

169, y así evitar el sobre ajuste, y de ese modo se obtuvo los datos mencionados en párrafos anteriores.

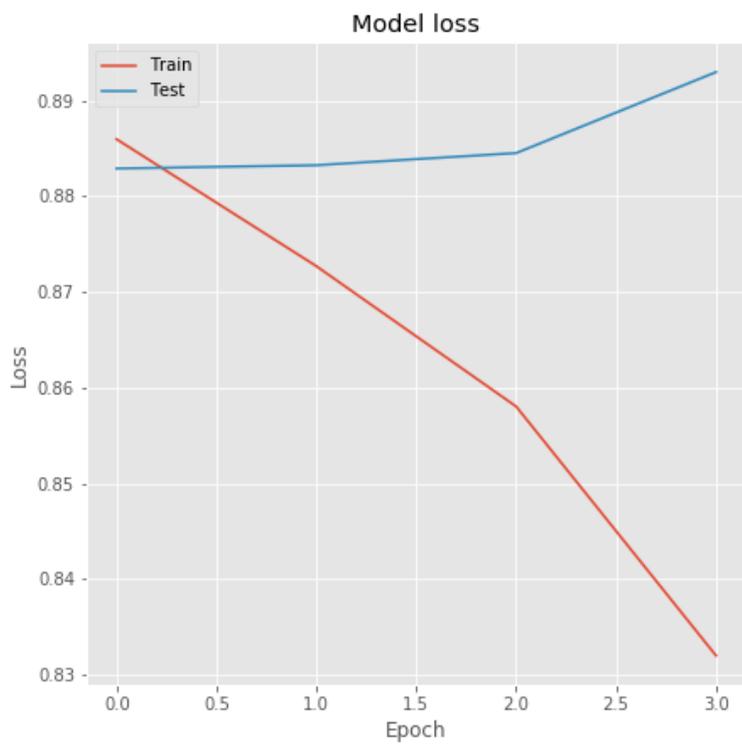
A continuación, se muestra las gráficas del entrenamiento de las diferentes redes neuronales artificiales.



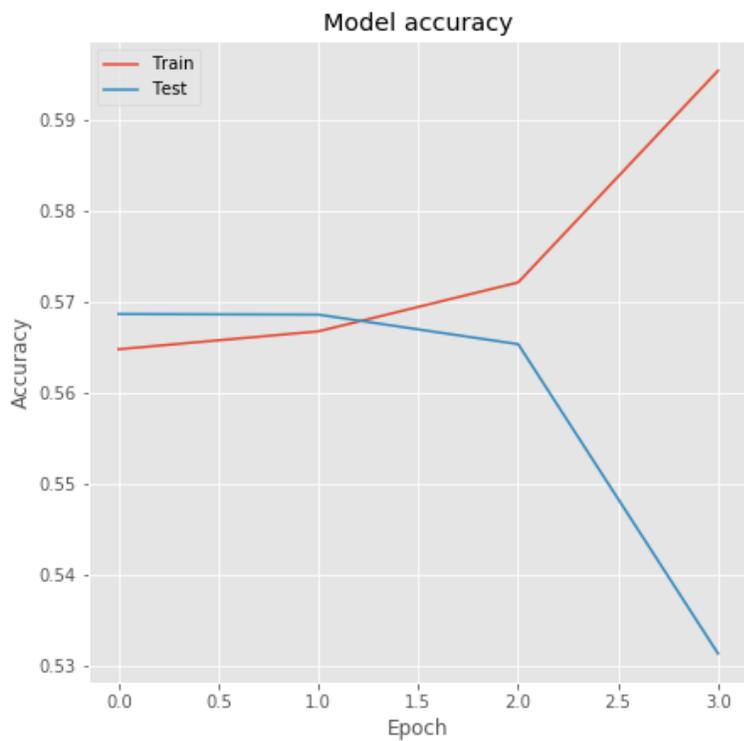
Gráfica 3.1: Error de Modelo 22.



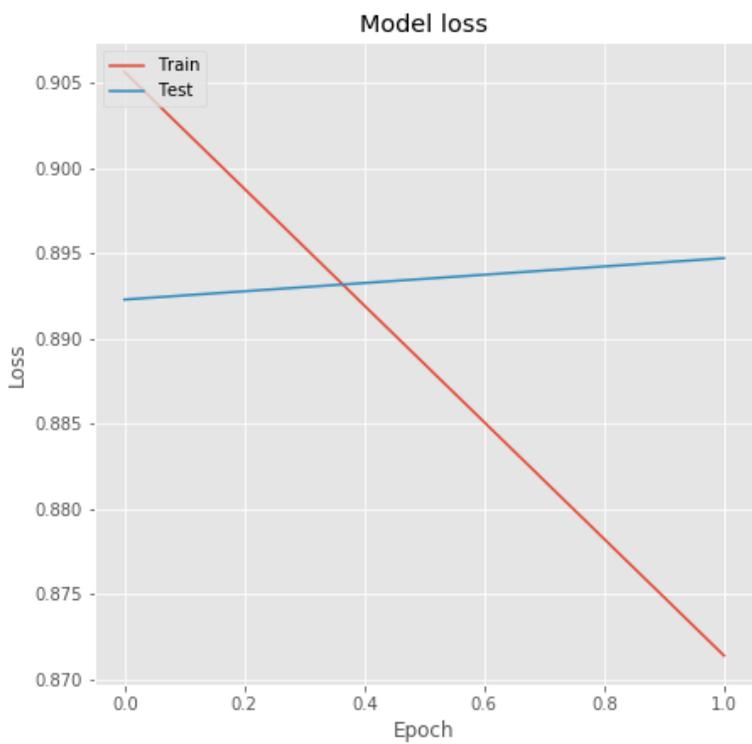
Gráfica 3.2: Validación de Modelo.



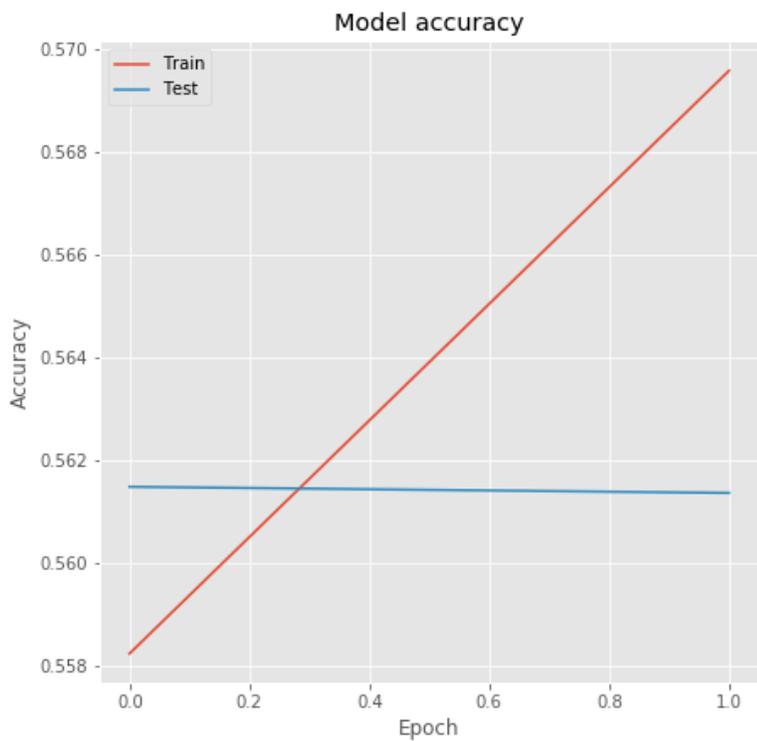
Gráfica 3.3: Error de Modelo 1.



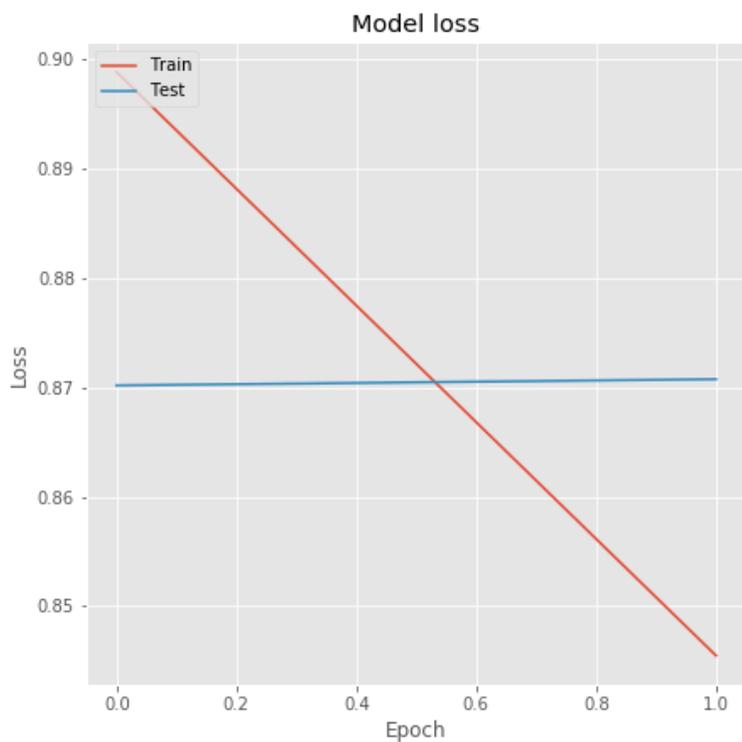
Gráfica 3.4: Validación de Modelo 1.



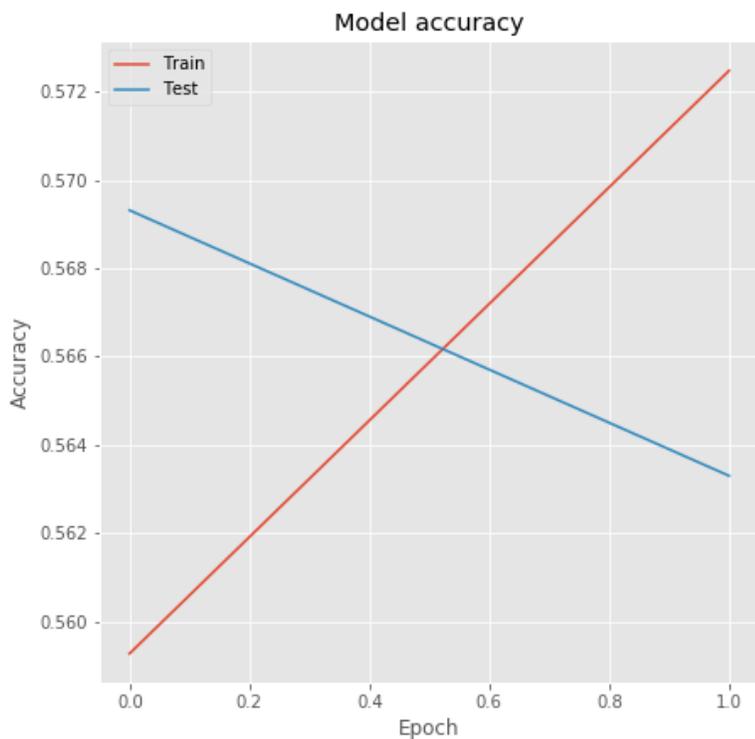
Gráfica 3.5: Error de Modelo 2.



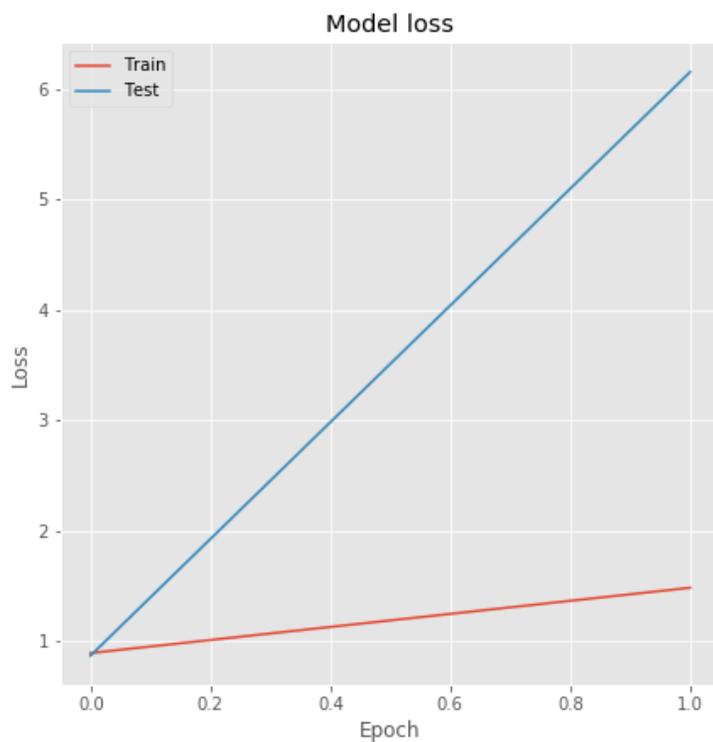
Gráfica 3.6: Validación de Modelo 2.



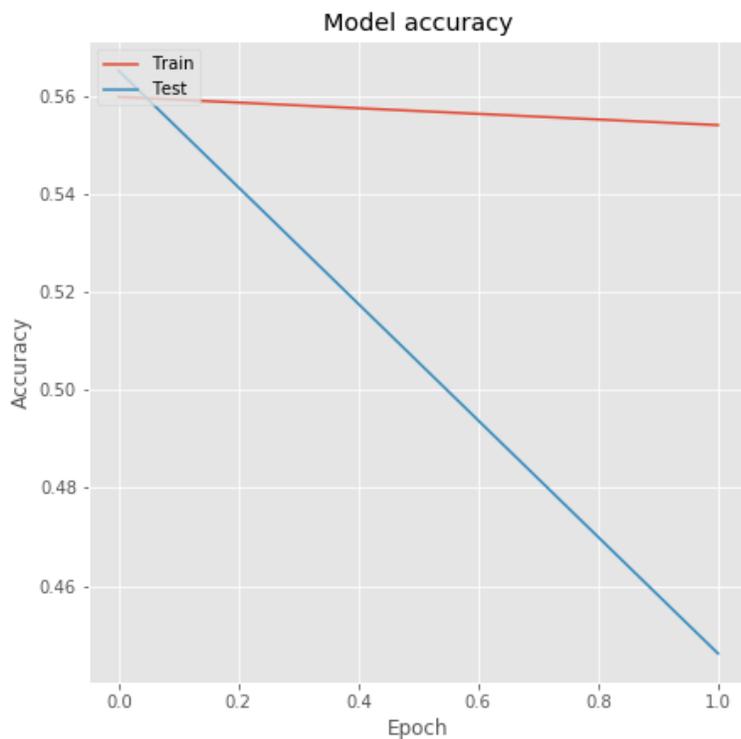
Gráfica 3.7: Error de Modelo 3.



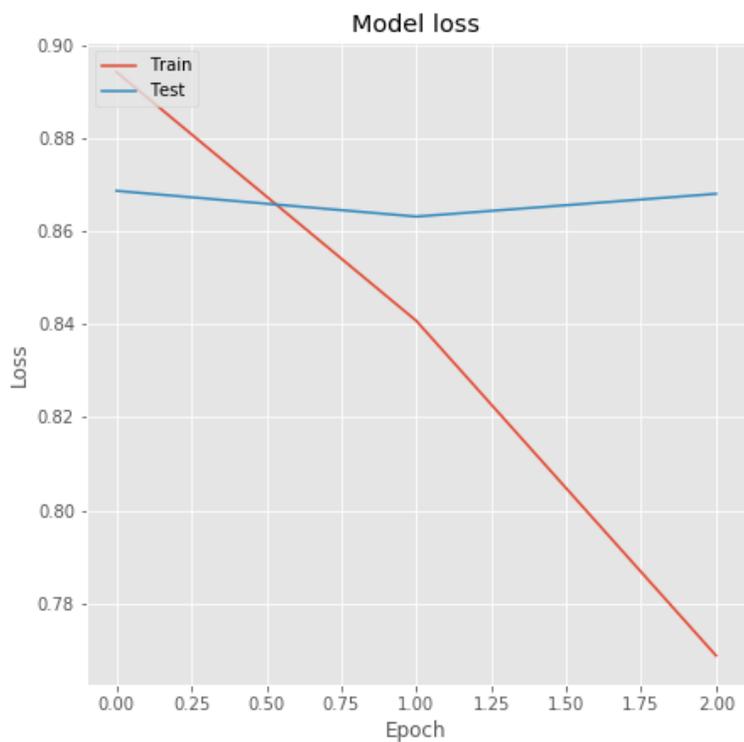
Gráfica 3.8: Validación de Modelo 3.



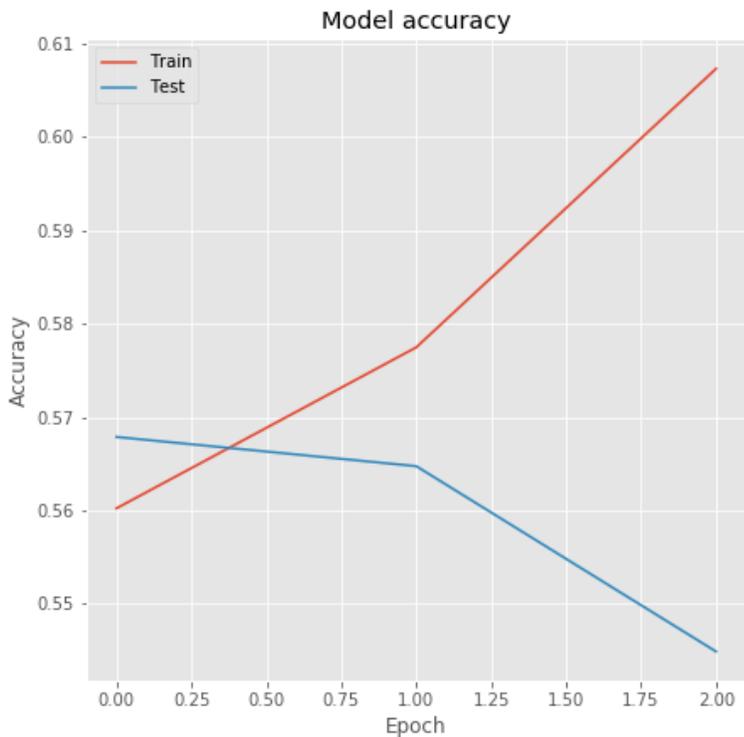
Gráfica 3.9: Error de Modelo 4.



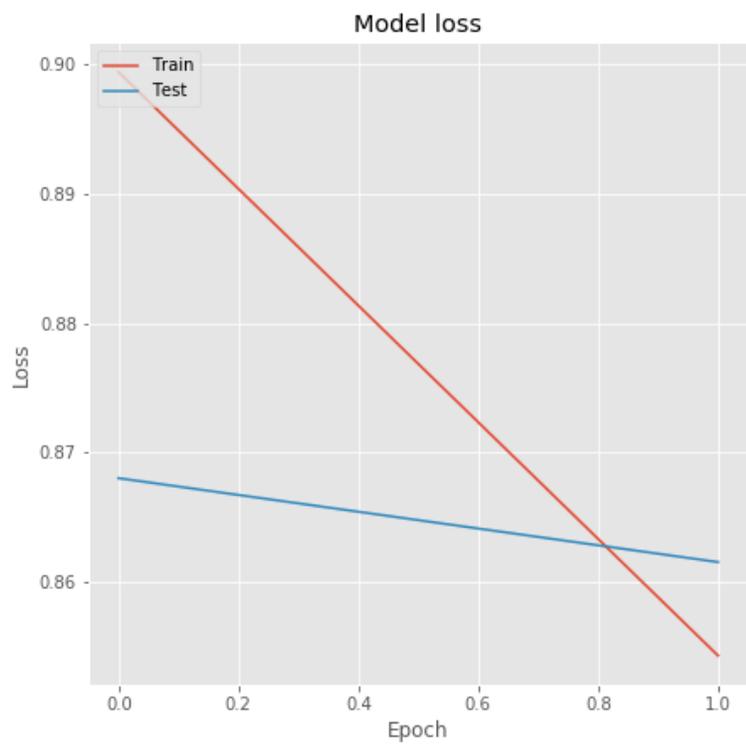
Gráfica 3.10: Validación de Modelo 4.



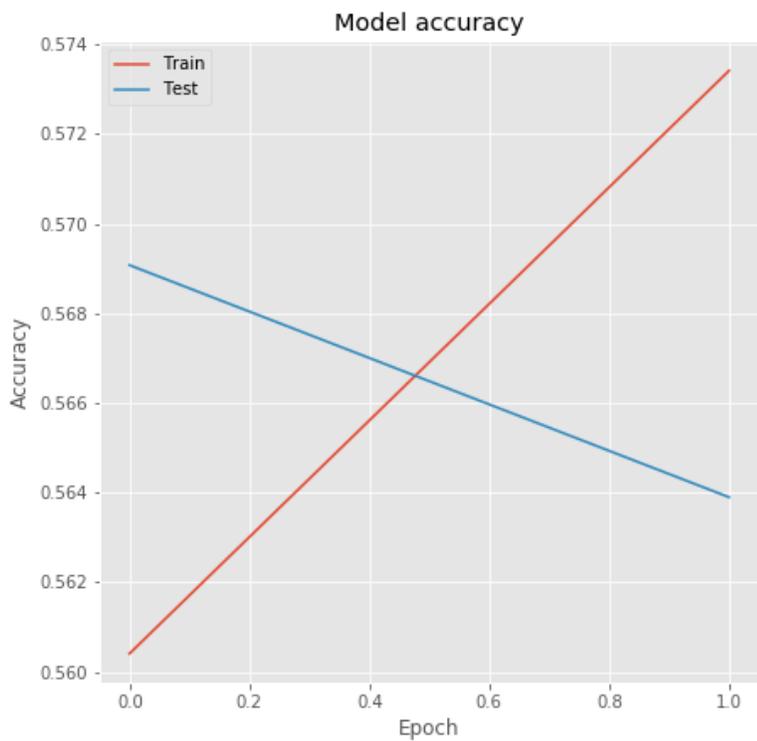
Gráfica 3.11: Error de Modelo 5.



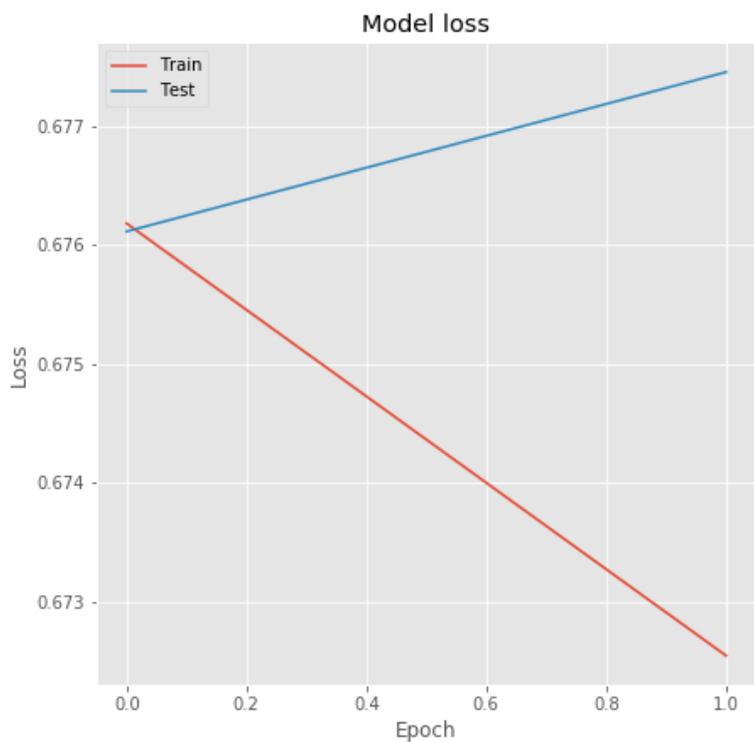
Gráfica 3.12: Validación de Modelo 5.



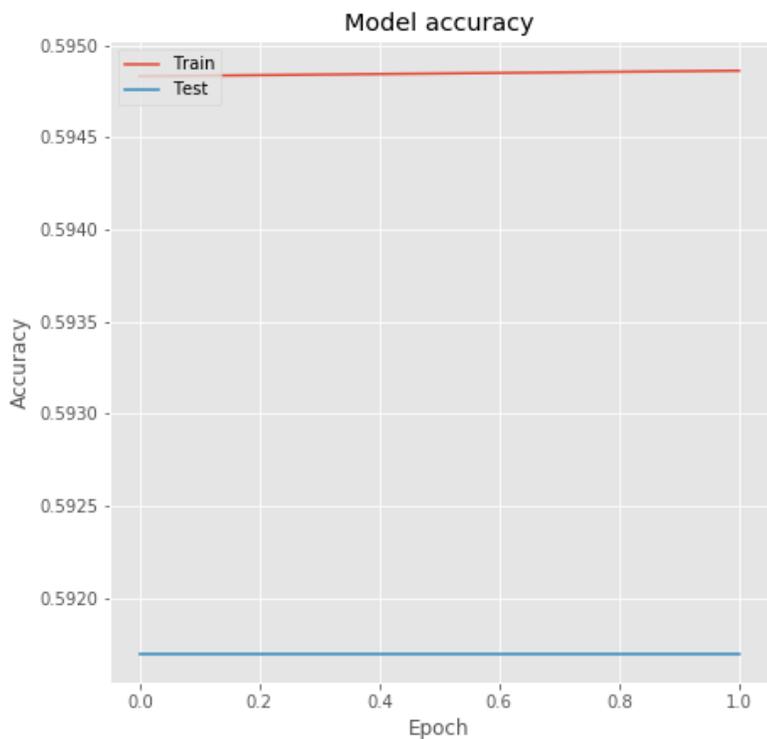
Gráfica 3.13: Error de Modelo 6.



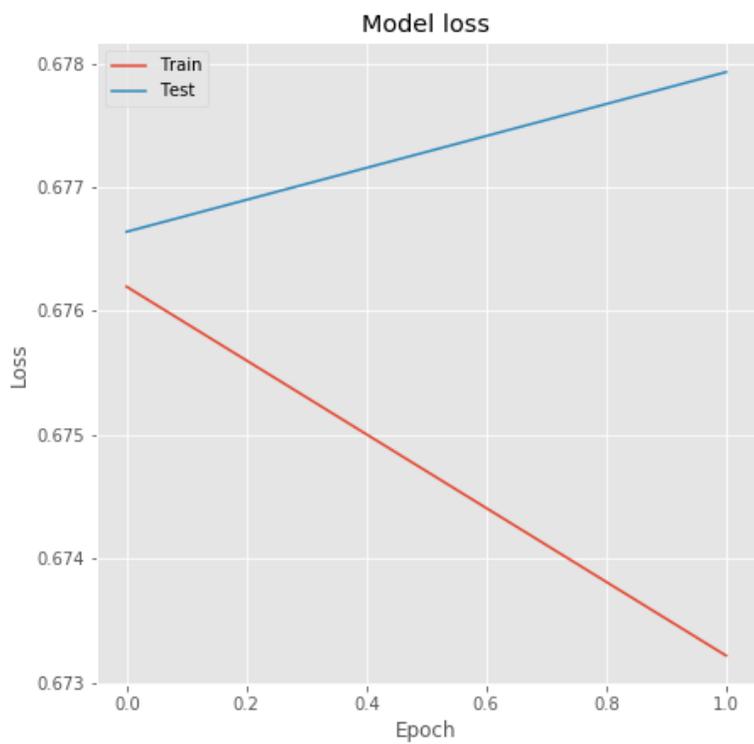
Gráfica 3.14: Validación de Modelo 6.



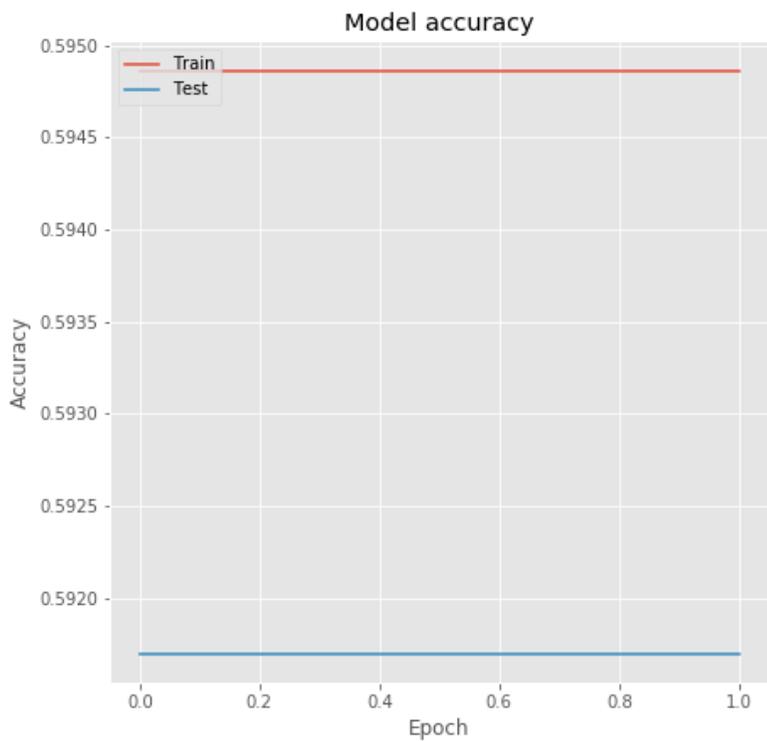
Gráfica 3.15: Error de Modelo 11.



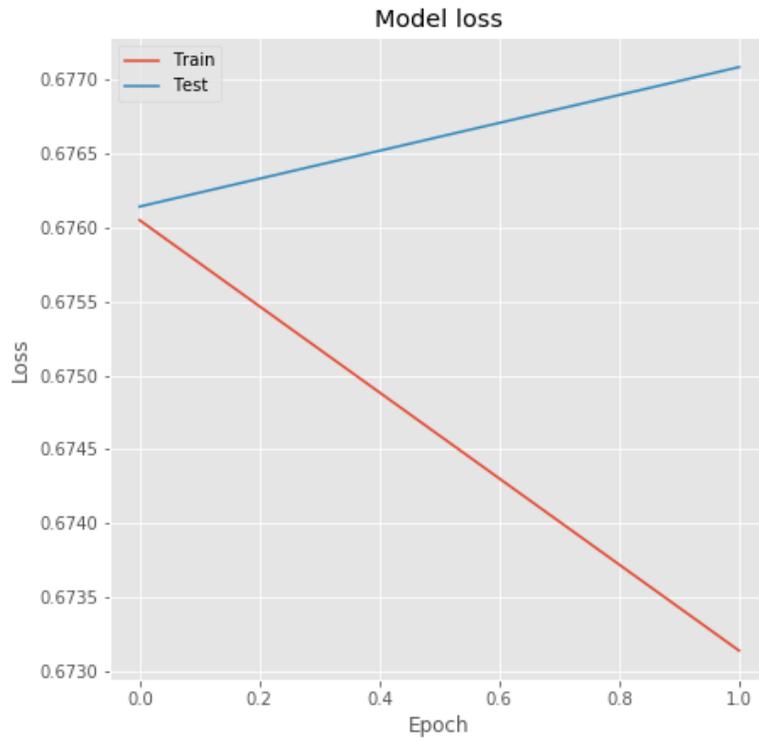
Gráfica 3.16: Validación de Modelo 11.



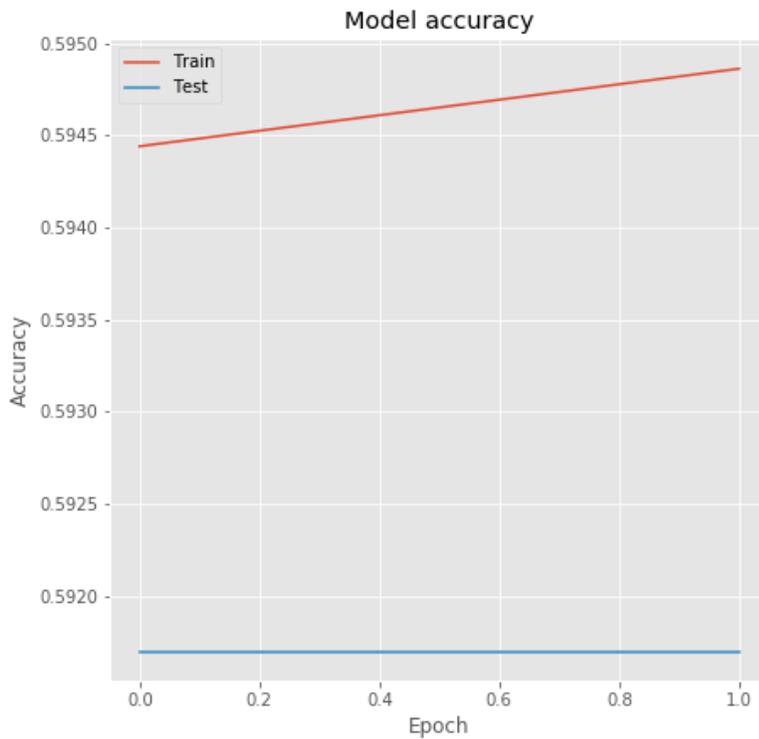
Gráfica 3.17: Error de Modelo 13.



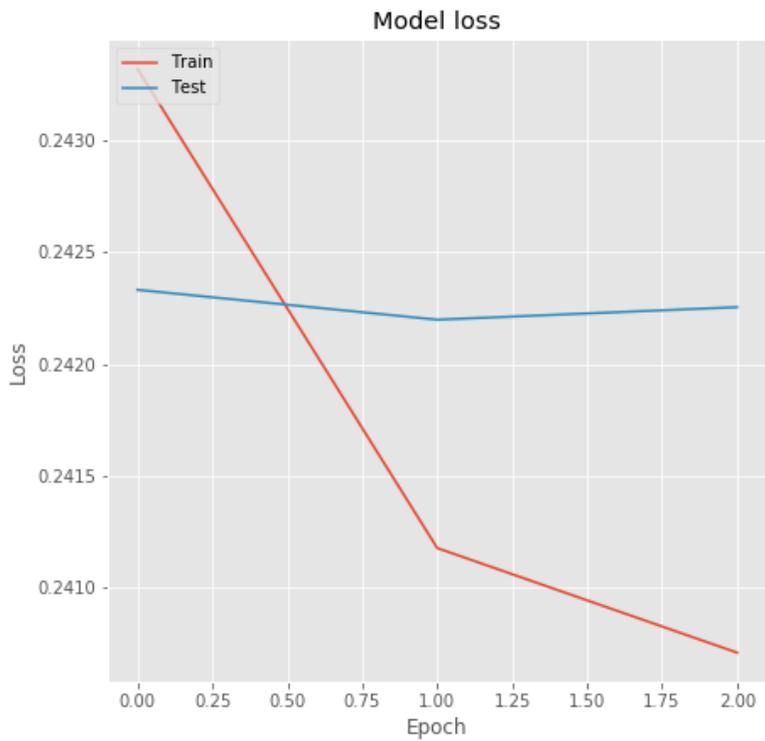
Gráfica 3.18: Validación de Modelo 13.



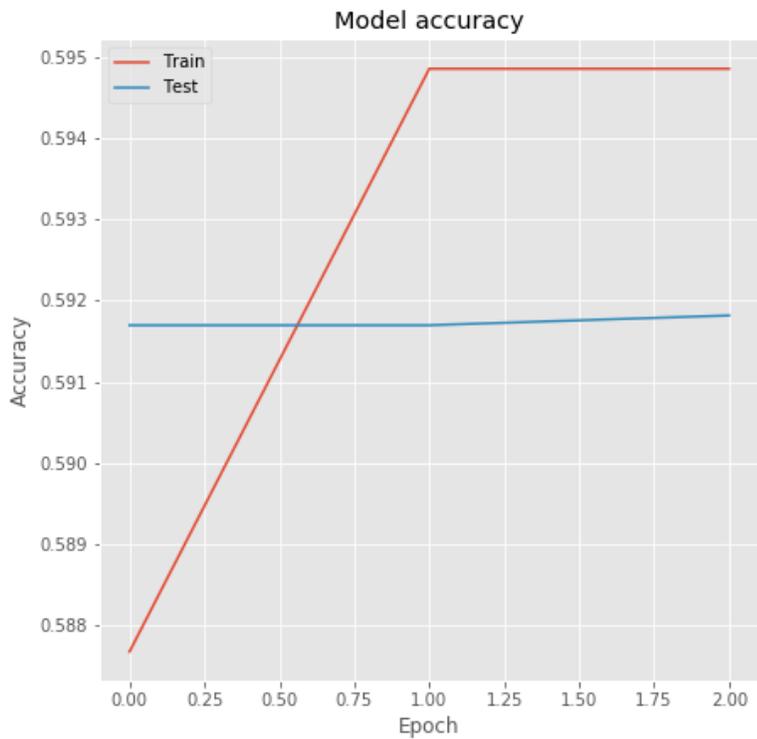
Gráfica 3.19: Error de Modelo 14.



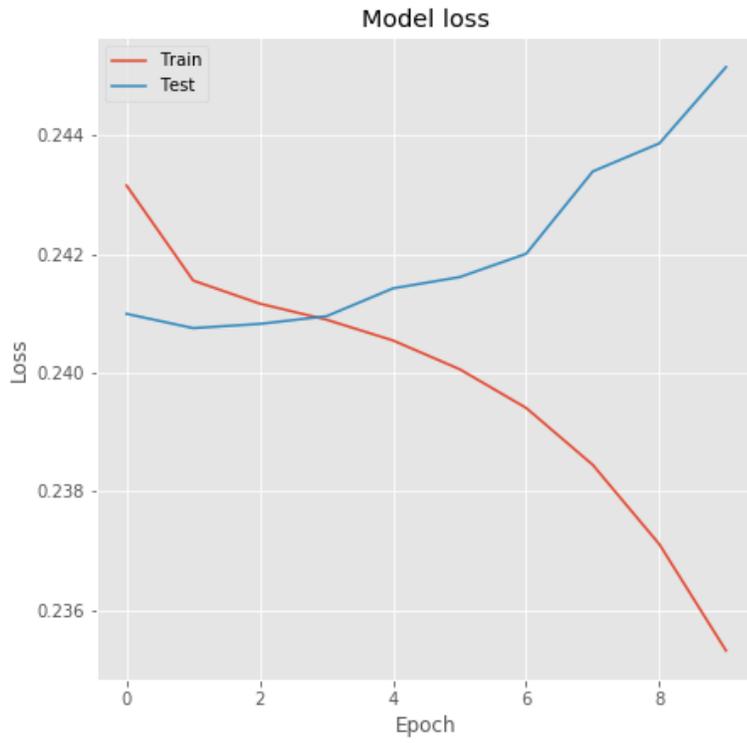
Gráfica 3.20: Validación de Modelo 14.



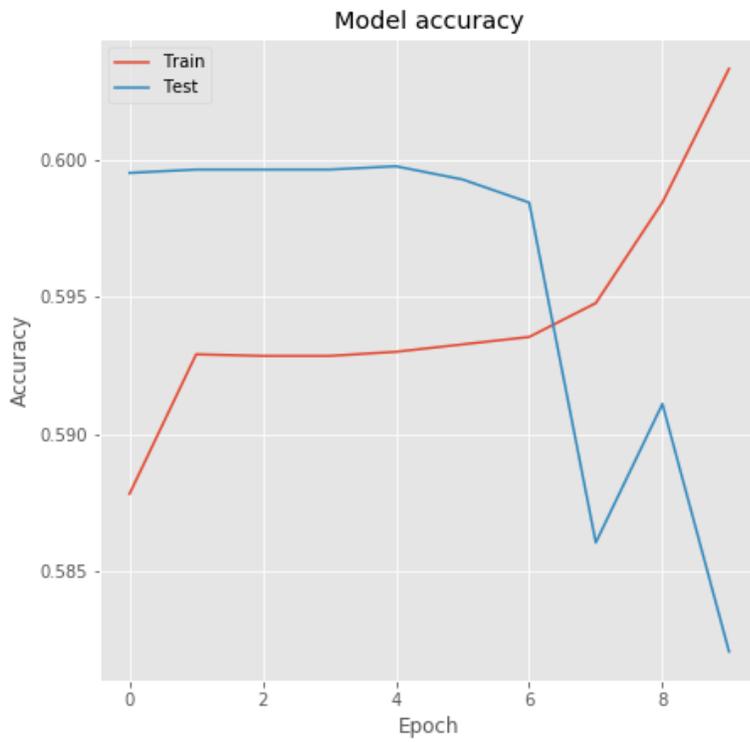
Gráfica 3.21: Error de Modelo 17.



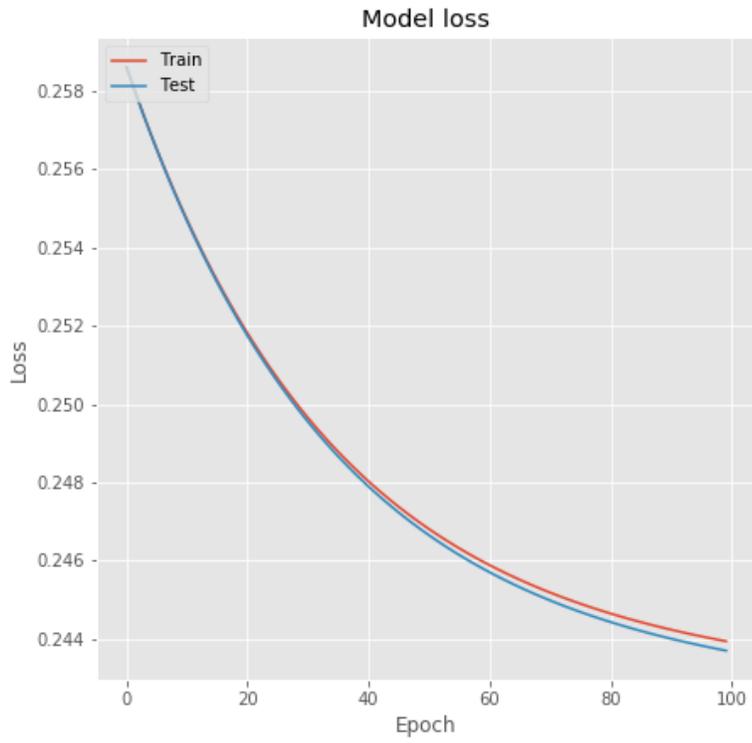
Gráfica 3.22: Validación de Modelo 17.



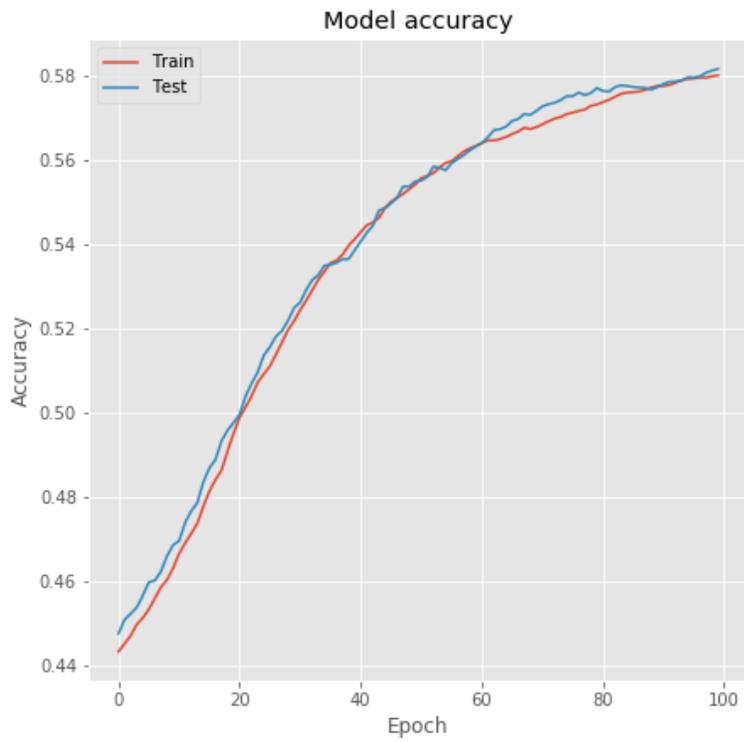
Gráfica 3.23: Error de Modelo 19.



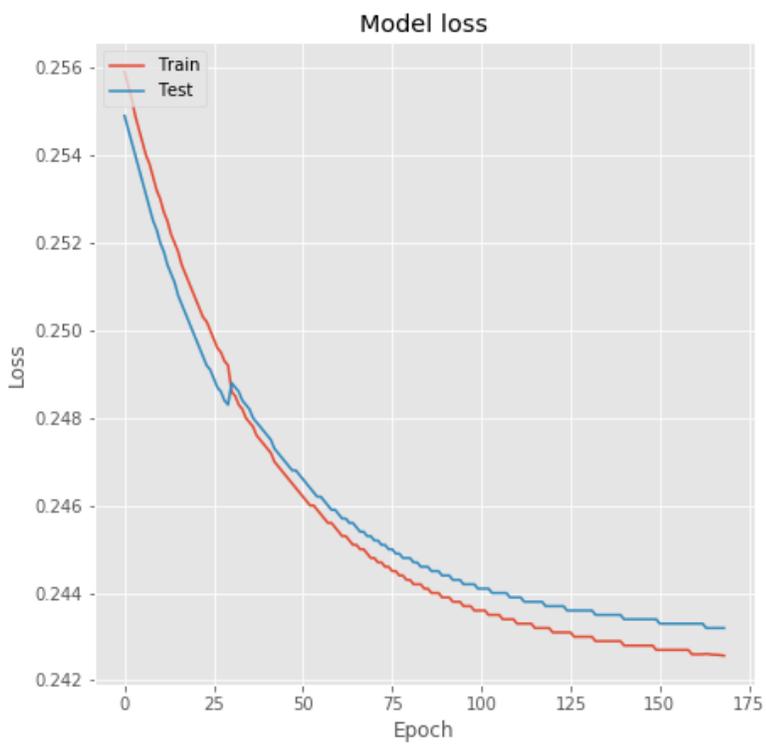
Gráfica 3.24: Validación de Modelo 19.



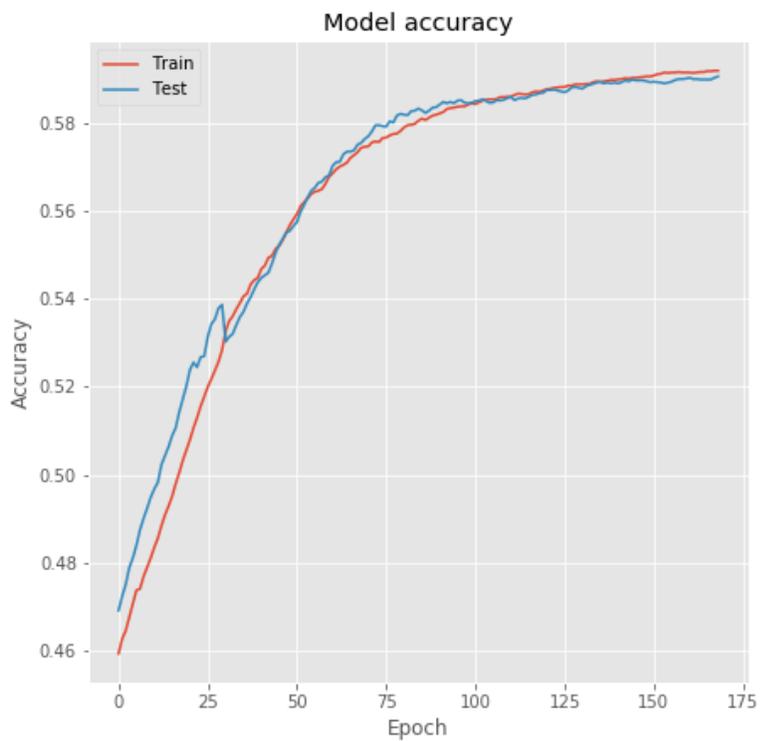
Gráfica 3.25: Error de Modelo 20.



Gráfica 3.26: Validación de Modelo 20.



Gráfica 3.27: Error de Modelo 21.



Gráfica 3.28: Validación de Modelo 21.

Capítulo 4

Discusiones

4.1. Interpretación

De acuerdo a Viñuela & Galván (2004) la manera de medir la precisión de una Red Neuronal Artificial es la división de los casos acertados en el test entre el total de datos del test; en este caso la RNA al realizar el proceso de pronóstico obtiene una precisión de 0.5861, y para variaciones positivas y negativas, 0.5914 y 0.4236, respectivamente; es decir, acierta en el 58.61 % de veces; para los datos positivos se acierta el 59.14 % de veces, y por último para los datos negativos se acierta el 42.36 % de veces.

4.2. Conclusiones

Existen, actualmente, un gran avance en el análisis económico donde se realizan métodos muy sofisticados para realizar pronósticos económicos y financieros; todo gracias a las tecnologías de la información, Ciencias de la Computación y datos que se generan en Internet. Cuando se juntan estos puntos se obtienen análisis y pronósticos novedosos. Los cuales, dos décadas atrás eran imposibles por la tecnología disponible.

Los trabajos desempeñados por los diversos autores, mencionados en el marco teórico, siguen un flujo de trabajo muy similar: crear base de datos, pre - procesar datos, probar modelos, y generar pronósticos. Por lo cual, se usa un esquema de trabajo muy similar, explicado en la metodología.

Las diversas investigaciones a destacar en esta tesis son las que usaron información, para realizar su pronóstico, como: noticias, *tuits*, búsquedas en Google, entre otras fuentes similares. Estos trabajos hacen investigaciones que convergen con la hipótesis del presente trabajo, ya que, en esta tesis, se buscó realizar un pronóstico usando noticias del portal Reuters. Aquellas investigaciones que hicieron pronósticos con información de texto tuvieron resultados que se encuentran entre: 40 %, 50 %, 60 %, 70 % y, 80 %, de precisión; en la mayoría de esas investigaciones, para poder hacer el análisis de texto, tuvieron que aplicar algoritmos de inteligencia artificial, entre otros algoritmos, por ejemplo: Máquinas de Soporte Vectorial, Algoritmos Genéticos, Modelos de Optimización, Redes Neuronales Artificiales, entre otros. Reiterando, el aplicar esos modelos fue menester, ya que otro tipo de herramientas difícilmente se ajustan a las noticias, las cuales se consideran información no estructurada.

Otro punto a destacar, es que, las RNA, tienen la capacidad de ajustarse a los datos, incluso en situaciones de ruido. Una desventaja, es que ellas necesitan una gran cantidad de datos para ser entrenadas, lo cual puede ser una desventaja cuando el investigador no tiene los datos suficientes para entrenar un modelo de RNA, sin embargo se pueden hacer técnicas de remuestreo como *bootstrapping*, generar interpolación con otra RNA's, o polinomios, entre otros. Entre otras desventajas es encuentra el tiempo de entrenamiento, en comparación de otros modelos, por ejemplo, algoritmos genéticos y K-NN, los cuales llevan menor tiempo de entrenamiento; sin embargo, la capacidad de ajuste de la RNA es mayor.

En las investigaciones citadas en el marco teórico algunos autores han optado por realizar pronósticos no de valores específicos, sino, más bien, de clasificación, es decir, crear un modelo de inteligencia artificial que clasifique alzas o bajas de ciertos indicadores económicos o de algún activo financiero en el mercado. Lo cual en comparación con otros modelos tienen mayor capacidad de predicción.

Los puntos mencionados en los párrafos anteriores así como todo lo expuesto en el marco teórico de la presente sección nos proveen de varias razones por la que es justificable y viable realizar un pronóstico con una RNA, y, en esta tesis, realizar un pronóstico con una RNA para clasificar las variaciones positivas o negativas del S&P500 de la Bolsa de

New York. No obstante la precisión obtenida del modelo obtenido no superó los modelos entrenados por los diferentes autores mostrados en el marco teórico, sin embargo, no se obtuvo un pronóstico inferior a ellos. Por lo cual y retomando la pregunta de hipótesis: “se puede pronosticar la variación positiva y negativa del Índice S&P500, con noticias del Reuters (<https://www.reuters.com/>), con un retraso de una hora en el tiempo, haciendo uso de una RNA, ya que éstas pueden procesar información como el lenguaje natural y, buscar la función que mejor se adapte a los datos del fenómeno económico.”, entonces con nuestros resultados, se puede considerar que el modelo entrenado acierta en el 58.61 % de veces y, 59.14 % y 42.36 % de veces para variaciones positivas y negativas, respectivamente.

4.3. Consideraciones

Realizar un pronóstico económico o financiero usando Redes Neuronales Artificiales debe considerar, en gran medida, los datos que se usan para entrenar el modelo, ya que si los datos no son representativos, o sesgados, puede generar pronósticos erróneos. También es importante el pre - procesar datos, ya que los datos “crudos” en la mayoría de las ocasiones son inconsistentes, como: datos que faltan, estar desordenados, entre otros. El pre - procesamiento busca corregir los problemas mencionados anteriormente. En la presente tesis la falta de datos puede ser una limitante importante en un trabajo de este tipo, como fue el caso de encontrar los datos del S&P500 solo por día, obtenerlos por hora puede implicar un costo monetario elevado; por lo que optar por modelos de interpolación es viable, no obstante implica un mayor tiempo de dedicación para aprender dichos modelos.

Por otro lado, el generar la base de datos de sitios web puede ser un reto para personas que carecen del conocimiento técnico, por ejemplo, para hacer la presente tesis se requirió emprender dicho conocimiento técnico; del mismo modo se comenzó el proceso de aprendizaje del funcionamiento y programación de Redes Neuronales Artificiales, y las técnicas de pre - procesamiento de datos.

Cabe destacar, además, que los algoritmos, y algoritmos de inteligencia artificial aplicados a los mercados financieros puede ser perjudicial por la velocidad de sus transacciones como pasó en el Flash Crash de 2010, el cual generó una abrupta caída del Dow Jones en

un tiempo muy reducido; no obstante los algoritmos aplicados por ejemplo al pronóstico de crisis o éxito financiero de una empresa, entre otras, pueden ser viables sin implicar efectos negativos en la economía.

4.4. Recomendaciones

Es menester resaltar que la investigación se hizo con un único noticiero (Reuters), no obstante con una precisión del 58.61 %, de ahí que realizar futuras investigaciones considerando agregar al modelo más noticias y variables cualitativas como variables independientes, es decir, introducir, por ejemplo, retrasos del índice, indicadores macroeconómicos, entre otros, ello, puede mejorar el comportamiento de los modelos de IA para pronósticos y análisis económico.

También puede considerarse aplicar este tipo de modelos a otros ámbitos, como los realizados por los diversos autores mostrados en el marco teórico, por ejemplo: pronóstico del PIB, pronóstico de pago de créditos, pronóstico de empleo, crisis, ventas, entre otros. Lo cual puede ser más viable y evitar repercusiones como las mostradas en la sección 1.3 del marco teórico.

Cabe resaltar que de los diversos autores encontrados Parisi *et al.* (2018) fueron los únicos en realizar una prueba estadística para medir la capacidad de los modelos para clasificar variaciones positivas o negativas, el cual es llamado: test de Acierto Direccional de Pesaran y Timmermann; empero, el aplicar ese tipo de prueba al modelo de la presente tesis no es viable, ya que el modelo funciona bajo distribución normal, lo cual no es el caso del histórico del índice del presente trabajo. Por lo que se sugiere buscar algún otro modelo para encontrar una significancia estadística o aplicar otro tipo de test de reconocimiento de patrones o aprendizaje estadístico, para verificar el rendimiento del modelo.

Otra manera de realizar el trabajo es mediante un flujo de trabajo donde se itere con diferentes algoritmos y más propuestas de RNA, como se ve en la sección 1.4.3 del marco teórico, ya que cabe la posibilidad de que otro algoritmo se ajuste mejor, o que la RNA entrenada en el presente trabajo haya convergido en un mínimo local. La propuesta puede seguir el flujo mostrado en el diagrama 4.1.

Por otro lado, este trabajo también puede ser un introducción para aquellos economistas industriales o economistas que deseen introducirse a las herramientas de inteligencia artificial para análisis económico, ya que este trabajo cuenta con una bibliografía citada relativamente extensa, de la cual el lector se puede nutrir.

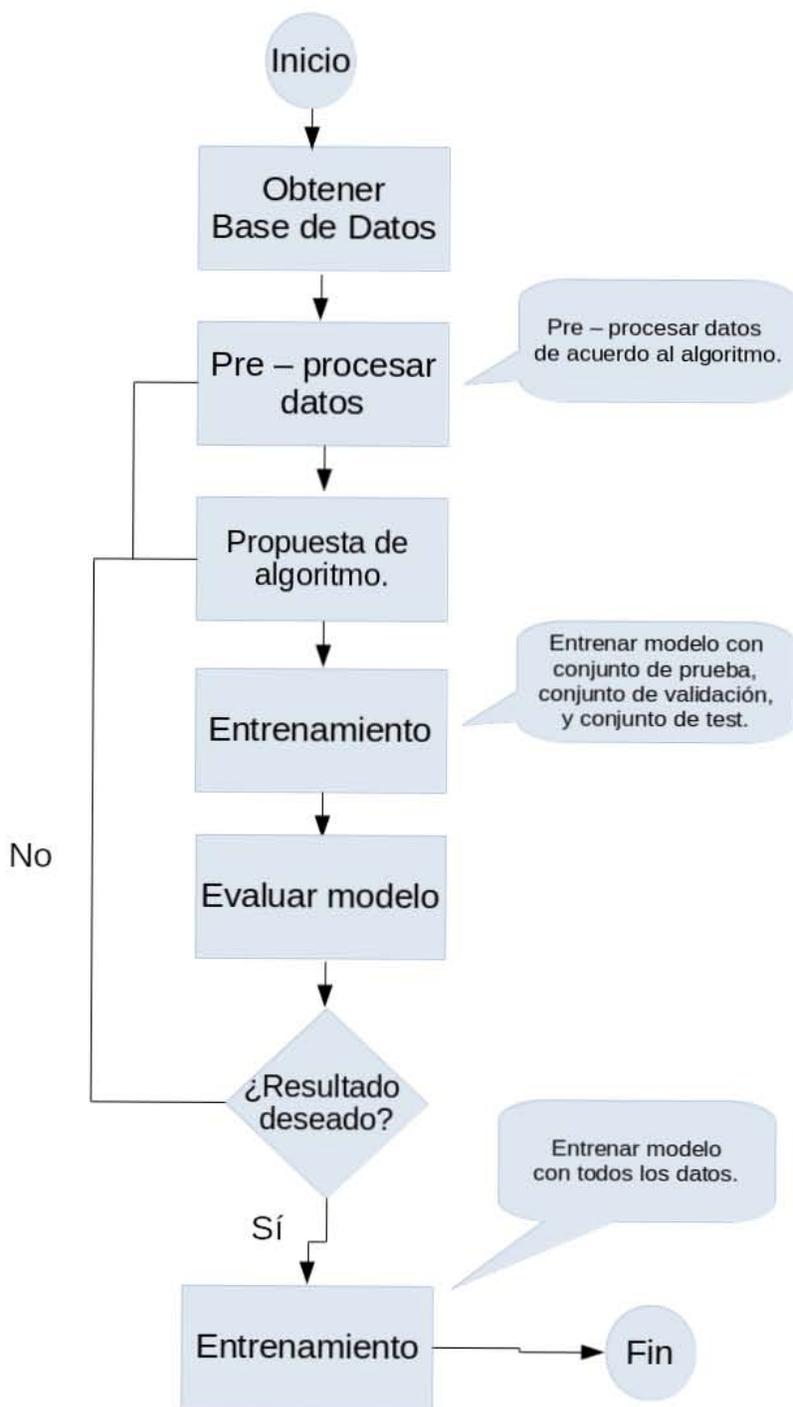


Diagrama 4.1: Propuesta de Flujo de Trabajo.

Bibliografía

1. Álvarez, R., (2001), "*Polinomios ortogonales: historia y aplicaciones*", Universidad de Granada, Bol. Soc. Esp. Mat. Apl. No. 18, pp. 19 - 45.
2. Amournwattana, S. *et al.*, (2007), A hybrid option pricing model using a neural network for estimating volatility International Journal of General Systems , Vol. 36, No. 5, pp.558-573, USA.
<https://www.tandfonline.com/doi/pdf/10.1080/03081070701210303?needAccess=true>.
3. Astorga, A., (2015), Redes Neuronales Bayesianas: Una Mirada al Aprendizaje Estadístico en el Problema de Clasificación, México, Universidad Nacional Autónoma de México.
4. Bao, D. & Yang, Z., (2008), "*Intelligent stock trading system by turning point confirming and probabilistic reasoning*" , Expert System with Applications, No. 34, pp. 620 - 627, China
5. Bènèfice, I, *et al.*, (2013), Trading de Alta Frecuencia (Documental) - La Noche Temática, Televisión Española, S.A., España,
<https://www.youtube.com/watch?v=a3pVqMT9DXA>.
6. Benhmad, F., (2012), "*Modeling nonlinear Granger causality between the oil price and U.S. dollar: A wavelet based approach*", Economic Modelling, Vol. 29 pp. 1505 - 1514, Francia.
7. Bing Li *et al.*, (2017), "*Discovering public sentiment in social media for predicting stockn movement of publicly listed companies*", Information Systems, No. 96C, pp.

- 81-92.
8. Binner, J. *et al.*, (2010), “*Does Money Matter in Inflation Forecasting?*”, FMI, USA, <https://core.ac.uk/download/pdf/6608082.pdf>.
 9. Bogda, Ion B. *et al.*, (2013), “*The Financial Crisis Prediction Using an Artificial Intelligence Method*”, European Social Fund, Rumanía.
 10. Bollen, J. *et al.*, (2010), “*Twitter mood predicts the stock market*”, Cornell University, USA, <https://arxiv.org/abs/1010.3003>.
 11. Canelas, A. *et al.*, (2013), *Investment Strategies Optimization Based on a SAX-GA Methodology*, London, UK , Springer Heidelberg New York Dordrecht London.
 12. Cardona R. & Rojas A., (2017), “*Pronósticos para la tasa de desempleo en Colombia a partir de Google Trends*”, Archivos de Economía, 2017, Departamento Nacional de Planeación, <https://ideas.repec.org/p/col/000118/016050.html>.
 13. Caselle, G. *et al.*, (2013), *An Introduction to Statistical Learning with Applications in R*, London, UK, Springer London Heidelberg New York Dordrecht.
 14. CBOE, (2019), “*White Paper Cboe Volatility Index*”, <http://www.cboe.com/micro/VIX/vixwhite.pdf>.
 15. CFTC & SEC, (2010), “*Findings regarding the market events of may 6, 2010*”, <https://www.sec.gov/news/studies/2010/marketevents-report.pdf>.
 16. Chen, Mu-Yen & Chen, Thing-Hsuan, (2019), “*Modeling public mood and emotion: Blog and news sentiment and socio-economic phenomena*”, *Future Generation Computer Systems*, Vol. 96, pp.692-699, Taiwan.
 17. Chen, Mu-Yen *et al.*, (2019), “*Modeling public mood and emotion: Stock market trend prediction with anticipatory computing approach*”, *Computers in Human Behavior*, pp. 402-408, Taiwan.
 18. Chidmbaran, N. *et al.*, (1998), “*A An Adaptive Evolutionary Approach to Option Pricing via Genetic Programming*”, NYU Working Paper, No. FIN-98-086, NY, USA.

19. Choice, H. & Varian, H., (2009), “*Predicting Initial Claims for Unemployment Benefits*”, Publish Online.
20. Choice, H. & Varian, H., (2012), “*Predicting the Present with Google Trends*”, The Economic Record, Vol. 88, Special Issue, June, 2012, pp.2 - 9, California, USA.
21. Choice, H. & Varian, H., (2009), “*Predicting the Present with Google Trends*”, Google, Inc., California, USA,
22. Choudhary, M., (2010), “*Neural Network Models for Inflation Forecasting: An Appraisal*”, Applied Economics, UK,
<https://hal.archives-ouvertes.fr/hal-00704670/document>.
23. Coase, R., (1937), “*The Nature of the Firm*”.
24. Coelho, L. & Richert, W., (2015), Building Machine Learning systems with *Python*, Second Edition Birmingham, UK, Packt Publishing Ltd.
25. Edmans, A. *et al.*, (2007), “*Sports Sentiment and Stock Returns*”, The Journal of Finance, Vol. LXII, No. 4, pp. 1967 - 1998, USA.
26. Fang, B. *et al.*, (2018), “*Exploiting investors social network for stock prediction in China’s market*”, Journal of Computational Science, No. 28, pp. 249-303, China.
27. Feuerriegel, S. & Gordon, J., (2019), “*News-based forecasts of macroeconomic indicators: A semantic path model for interpretable predictions*”, European Journal of Operational Research 272, pp. 162 - 157, Alemania.
28. Freeman, B., (2019), “Learning the Depths of Moving People by Watching Frozen People”, CVPR 2019 (submission), <https://research.google/pubs/pub48104/>.
29. García, A., (2018), INTELIGENCIA ARTIFICIAL: Fundamentos, práctica y aplicaciones, México, Alfaomega, Rc Libros.
30. García, J., *et al.*, (2018), Ciencia de Datos, Técnicas Analíticas y Aprendizaje Estadístico: Un Enfoque Práctico, Alfaomega, México.

31. Géron, A., (2017), Hands-On Machine Learning with Scikit-Learn & Tensorflow, 1^a ed., Gravenstein Highway North, Sebastopol, O'Reilly Media, Inc.
32. Géron, A., (2019), Hands-On Machine Learning with Scikit-Learn & Tensorflow, 2^a ed., Gravenstein Highway North, Sebastopol, O'Reilly Media, Inc.
33. Gershenson, C., (2019), "Sesenta años de inteligencia artificial - Curso", Programa especializado Introducción a la Inteligencia Artificial, UNAM.
34. Gómez, R., (2013), "*Señales de inversión basada en un índice de aversión al riesgo*", Investigaciones Europeas de Dirección y Economía de la Empresa, Vol. 19, pp. 147 - 157, Madrid, España.
35. González G., (2010), Algoritmo basado en Wavelets aplicado a la detección de incendios forestales, México.
36. *Google/Tensorflow*, (2019), "Intro to TensorFlow by TensorFlow", Udacity.
37. Gordillo, J. & Christopher, S., (2003), "*Análisis de Mercados Financieros utilizando Técnicas de Inteligencia Artificial*", Computación y Sistemas, Vol. 6 No. 4, pp. 253 - 272, México.
38. Guo, L., *et al.*, (2016), "*Textual analysis and machine leaning: Crack unstructured data in finance and accounting*", The Journal of Finance and Data Science, Vol. 2, pp. 153-170, China.
39. Hernández de Oteya, E., (2015), Portafolio de inversión e Inteligencia Artificial: Clasificación óptima de tendencias de mercados financieros mediante Colonias de Hormigas", México, Universidad Nacional Autónoma de México.
40. Heydt, M., (2015), Mastering pandas for Finance, Birmingham, UK Packt Publishing Ltd.
41. Hunter, J. *et al.*, (2020), Matplotlib: Visualization with Python (Página oficial), <https://matplotlib.org/>

42. Joyanes, L., (2013), *Big Data: Análisis de Grandes Volúmenes de Datos en Organizaciones*, Alfaomega Grupo Editor, S.A. de C.V., México.
43. Keynes, J., (2003), *Teoría General de La Ocupación, del Interés y el Dinero*, 4^a ed. México, Fondo de Cultura Económica.
44. Keras, (2020), Keras: The Python Deep Learning library (Página web oficial), <https://keras.io/>
45. Kulkarni, A. & Shivananda, A., (2019), *Natural Language Processing Recipes: Unlocking Text Data with Machine Learning and Deep Learning using Python*, Apress, India.
46. León, A. & et al., (2018), “*Predicción del IPC mexicano combinando modelos econométricos e inteligencia artificial*”, *Revista Mexicana de Economía y Finanzas Nueva Época*, Vol. 13 No. 4, pp. 603 - 629, México.
47. Luke, Y., (1969), *The Special Functions and Their Approximations*, Academic Press, INC., Vol. 1, USA.
48. Ma, X. & Lv, S., (2019), “*Financial credit risk prediction in finance driven by Machine Learning*”, *Neural Computing and Applications*, Publish Online.
49. Marwarka, T. (2013), *Economic Modeling Using Artificial Intelligence Methods* London, UK, Springer London Heidelberg New York Dordrecht.
50. Matthes, E., (2019), *Python crash course: A Hands-on, Project-Based, Introduction to Programming*, 2nd Edition, No Starch Press, Inc., San Francisco.
51. Medina H., S. & Manco, O., (2007), “*Diseño de un sistema experto difuso: Evaluación de riesgo crediticio en firmas comisionistas de bolsa para el otorgamiento de recursos financieros*”, *Estudios Gerenciales*, Vol. 23 No. 104, pp. 101 - 129, Colombia
52. Mirowski, P., (2002), *Machine Dreams, Economic Becomes a Cyborg Science*, US, Cambridge university Press.

53. Nisar, T. & Yeung, M., (2018), “*Twitter as a tool for forecasting stock market movements: A short-window event study*”, The Journal of Finance and Data Science, Vol. 4, pp. 101-119, UK.
54. NLTK Project, (2020), Natural Language Toolkit (Página web oficial), <https://www.nltk.org/>.
55. Olver, F. *et al.*, (2010), NIST Handbook of Mathematical Functions, Cambridge University Press.
56. OpenAI, (2019), “Better Language Models and Their Implications”, <https://openai.com/blog/better-language-models/>.
57. Pak, A. & Paroubek, P., (2010), “*Twitter as a Corpus for Sentiment Analysis and Opinion Mining*” Conference: Proceeding of International Conference of Language Resource and Evaluation, pp. 1320 - 1326, Francia.
58. Pandas, (2020), Pandas (Página web oficial), <https://pandas.pydata.org/>.
59. Parisi F., A., (2018), “*Inteligencia artificial en las finanzas: El factor conciencia en la predicción de la acción TSLA*”, Revista Chilena de Economía y Sociedad, Junio 2018, pp. 51 - 73.
60. Parisi, A. *et al.*, (2004), “*Algoritmos genéticos y modelos multivariados recursivos en la predicción de Índices Bursátiles de América del Norte: IPC, TSE, NASDAQ, y DJI*”, El Trimestre Económico, Vol. LXXI, No. 283, pp. 789 - 809, México, <https://www.redalyc.org/pdf/313/31328402.pdf>.
61. Parisi, A. *et al.*, (2006), “*Modelo de Algoritmos Genéticos y Redes Neuronales en la Predicción de Índices Bursátiles Asiáticos*”, Cuadernos de Economía, Vol. 43, pp. 251 - 284, Chile.
62. Patterson, Z., (2018), “*Machine Learning for Quantitative Finance: Fast Derivative Pricing, Hedging and Fitting*”. <https://engineering.nyu.edu/events/2018/11/08/machine-learning-quantitative-finance-fast-derivative-pricing-hedging-and-fitting>.

63. Pedregosa, F., *et al.*, (2011), Scikit-learn: Machine Learning in Python Journal of Machine Learning Research No. pp. 2825 - 2830,
<http://jmlr.csail.mit.edu/papers/v12/pedregosa11a.html>
64. Piotr W., T. & Lambe, B., (2011), “*The role of media in the credit crunch: The case of the banking sector*”, Journal of Economic Behavior & Organization 85, pp. 163-157.
65. Ponce C. P., (2010), Inteligencia Artificial con aplicaciones a la ingeniería, Alpha Grupo Editor, S.A. de C.V., México.
66. Project Jupyter, (2020), Jupyter (Página web oficial), <https://jupyter.org/about>.
67. Qi, M., (2001), “*Predicting US recessions with leading indicators via neural network models*”, International Journal of Forecasting, No. 17. pp. 383 - 401., USA.
68. QuantMinds TV, (2019), Michael Steliaros, Goldman Sachs, on algorithmic trading and innovations, QuantMinds TV,
<https://www.youtube.com/watch?v=FBF5UcQrKlE>.
69. Raschka, S. & Mirjalili, V., (2017), Python Machine Learning, Second edition, Birmingham, UK, Packt Publishing Ltd.
70. Raschka, S., *at al.*, (2016), Python: Deeper Insights into Machine Learning, Birmingham, UK, Packt Publishing Ltd.
71. Richardson, L., (2020), Beautiful Soup Documentation,
<https://www.crummy.com/software/BeautifulSoup/bs4/doc/>.
72. Rodríguez, M., *et al.*, (2014), “*Determinación del riesgo de fracaso financiero mediante la utilización de modelos paramétricos, de inteligencia artificial y de información de auditoría*”, Estudios de Economía, Vol. 41 - No. 2, pp.187 - 217, La Coruña, España.
73. Rosenber, L. *et al.*, (2017), “*Artificial Swarm Intelligence amplifies accuracy when predicting financial markets*”, IEEE 8th Annual Ubiquitous Computing, Electronics

- and Mobile Communication Conference (UEMCON), New York City, NY, USA,
<https://ieeexplore.ieee.org/document/8248984>.
74. Sánchez, A., (2011), “Ajuste de datos por mínimos cuadrados, usando funciones de base radial y Gram-Schmidt dual, para la obtención de forma”, Instituto Tecnológico de León,
75. Sartre, J., (1967), “Entrevista a Jean-Paul Sartre”,
<https://www.youtube.com/watch?v=fZDq1YETEBg&t=565s>.
76. Schmidt, E. & Rosenberg, J., (2015), *Cómo trabaja Google*, Aguilar, Ciudad de México.
77. Schmidt, T. & Vosen, S. (2012), “*Using Internet Data to Account for Special Events in Economic Forecasting*”, Ruhr Economic Papers, Alemania.
78. Shapiro, C. & Varian, H., (2000), *El Dominio de la Información: Una Guía Estratégica para la Economía de la Red* Barcelona, España, Antonio Bosch Editor.
79. Shivananda, A. & Kulkarni, A., (2019), *Natural Language Processing Recipes*, Apress, India.
80. Squire, M., (2016), *Mastering Data Mining with Python - Find patterns hidden in your data*, Birmingham, UK, Packt Publishing Ltd.
81. Szegő, G., (1958), *Orthogonal Polynomials*, American Mathematical Society, Colloquium Publications Vol. XXIII.
82. Tensorflow (2020), Tensorflow (Página oficial), <https://www.tensorflow.org/>.
83. Varian, H., (2014), “*Big Data: New Tricks for Econometrics*”, *Journal of Economic Perspectives*, Vol. 28, No. 2, Spring 2014. pp. 3-28,
<https://www.aeaweb.org/articles?id=10.1257/jep.28.2.3>
84. Viñuela P. & Galván L., (2004), *Redes Neuronales Artificiales, Un Enfoque Práctico*, Madrid, Pearson Educación, S.A.

85. _____ (2013), “*The Price in Economic Sciences 2013*”, Nobel Prize,
<https://www.nobelprize.org/uploads/2018/06/press-24.pdf>.

Anexos

Anexos A

Anexo I: Python

En este anexo se muestra un breve resumen del lenguaje de Python.

A.1. Python

En este anexo se dará una breve revisión del lenguaje de Python, la cual no pretende ser específica en cada uno de los subtemas, ya que la programación no es el objeto de esta investigación, sino más bien, ella es solo una herramienta para desempeñar el presente trabajo.

Python es un lenguaje de programación orientado a objetos esto significa que modela objetos como una casa, un carro, una persona, entre otros; también es un lenguaje interpretado, no necesitando un compilador y es de tipado dinámico, es decir, no se necesita declarar que tipo de variable se usará, ya que Python se lo asigna en automático (García, A., 2016).

A.1.1. Variables en Python

Las variables son unidades de información o valores que se almacenan en la memoria (Matthes, 2019) y son de diferentes tipos, los más comunes son:

- String: Una cadena de texto, la cual va entre comillas simples o dobles, por ejemplo, “Este es un String”.

- Integer: Número entero, por ejemplo, 1, 3, 4, 64.
- Float: número con punto decimal, que es llamado por Python como float (Matthes, E., 2019). Por ejemplo, 3.8, 0.2, 3.141589.
-

A.1.2. Listas en Python

Las listas son un orden de artículos o elementos con un cierto orden. Pueden contener strings, integer, o float. Y se escriben de la siguiente manera:

```
this_list = [1, 3, 4, 4.5, 1.1, "un string"]
```

A.1.3. Tuplas en Python

Las tuplas son similares a las listas, pero las tuplas son inmutables, es decir, no se pueden modificar, y se escriben entre paréntesis.

```
this_tuple = (1, 3, 4, 4.5, 1.1, "un string")
```

A.1.4. Diccionarios

Los diccionarios son un una manera de conectar variables relacionadas, y son capaces de almacenar gran cantidad de información (Matthes, 2019). Estos tienen key's y value's; a continuación un ejemplo de un diccionario de una persona:

```
this_dictionary = {  
    'Name': 'Jorge',  
    'Escolaridad': 'Licenciatura',  
    'Estado civil': 'Casado',  
    'Edad': 35,  
    'Hijos': 3  
}
```

Los key's, son el valor a la izquierda de los dos puntos, y los value's es el valor a la derecha de los dos puntos.

A.1.5. If statements en Python

“*if*” es un tipo de controlador para flujos de trabajo utilizado en Python (así como en otros lenguajes), el cual trabaja de tal modo que si se cumple una sentencia lógica, entonces se ejecuta determinada acción. Por ejemplo, si variable es igual a carro, entonces sumar dos más dos, de lo contrario sumar dos más tres. Así se vería en código de Python:

```
variable = "car"

if variable == "car":
    return 2 + 2
else:
    return 2 + 3
```

A.1.6. While Loop en Python

El controlador “*while*” funciona así: Mientras una sentencia lógica se cumpla se ejecutará una acción hasta que la sentencia no se cumpla; por ejemplo, mientras numero sea igual a menor o igual a 5, imprimir el número y luego sumar un uno al número, en Python se verá de la siguiente manera:

```
current_number = 1

while current_number <= 5:

    print(current_number)
    current_number += 1
```

```
print("current_number es mayor o igual a 5.")
```

(Ejemplo tomado de Python Crash Course, pp.118, Matthes, 2019)

A.1.7. For Loop en Python

“*for*” es un tipo de ciclo que se repite una determinada cantidad de veces asignada para la ejecución de una determinada acción. Por ejemplo, pasar en el rango del 0 a 10 e imprimir un valor en cada iteración; el código en Python se vería de la siguiente manera:

```
for num in range(0, 10):
```

```
    print(num)
```

A.1.8. Funciones en Python

Las funciones son una manera de encapsular código, y simplificar el trabajo. Por ejemplo:

```
def print_numbers(start, end):
```

```
    for num in range(start, end + 1):
        print(num)
```

de esta manera cada vez que ejecute `print_numbers(start, end)`, y asigne un número a `start` y `end`, me imprimira los números desde donde yo le indique hasta donde yo le diga que termine.

A.1.9. Clases en Python

Las clases es una manera en la que se modelan objetos. A continuación se muestra la creación de un objeto carro.

```
class Car:
    """ A simple attempt to represent a car. """

    def __init__(self, make, model, year):
        """Initialize attributes to describe a car."""
        self.make= make
        self.model = model
        self.year = year

    def get_descriptive_name(self):
        """ Return a neatly formatted descriptive name. """
        long_name = f"{self.year} {self.make} {self.model}"
        return long_name.title()

my_new_car = Car('a4', 'Audi', '2019')
```

(Ejemplo tomado de *Python Crash Course* pp. 162, Matthes, 2019)

Anexos B

Anexo II: Glosario

Base de datos Relacional

Conjunto de datos ordenado en tablas, que posee columnas y filas.

Big Data (BD)

Serie de herramientas, métodos, y técnicas, estadísticas y de *machine learning*, para el procesamiento y análisis de una gran cantidad de datos que difícilmente podrían ser analizados por un humano. También puede considerarse como un volumen de datos grande.

Framework

Marco de trabajo que ofrece una metodología, herramientas de código, para la solución de problemas.

Código HTML

El código HTML es el código de etiquetado para mostrar la información de manera estructurada en una página web.

Inteligencia artificial

Alan Turing dice que la inteligencia artificial es cuando una maquina puede comportarse como un humano (García, 2018).

Keras

Keras es una API de muy alto nivel escrita en python que se conecta a *frameworks* como TensorFlow, CNTK o Theano. Esta API se enfoca en el rápido desarrollo y pruebas de redes neuronales artificiales de una manera sencilla; para pasar de una manera rápida de la idea a los resultados y hacer una buena investigación (Keras, 2020).

Modulo

Se dice modulo a un contenedor de código. En python, generalmente es código pre-escrito y reutilizable.

Librería

Similar al modulo de python, pero escrito por terceros, y se instala con pip para usarlo como modulo de python.

Machine Learning

“*[Machine Learning is the] field of study that gives computer the ability to learn without being explicitly programmed.*” - Arthur Samuel, 1959”, (Gerón, 2019: 4).

NLTK(Natural Language Toolkit)

Es una plataforma escrita en python para trabajar con lenguaje humano, la cual cuenta con herramientas como pre - procesamiento de texto, clasificación, tokenización, entre

otros, (NLTK Project, 2020) .

Pandas

Herramienta de análisis y manipulación de datos de código abierto escrita en python.

Scikit-learn

Es un modulo de python que integra varios modelos de *machine learning* tanto supervisados como no supervisados, para la solución de problemas de mediana escala. Enfocado a personas a personas no especializadas que usan un lenguaje de programación de alto nivel(Pedregosa, *et al.*, 2011).

SQLi3

Gestor de base de datos para: administrar, agregar, editar, y eliminar datos en la base de datos.

BeutifoulSoup4

Es una herramienta escrita en Python para extraer información de código HTML y XML; está técnica es conocida como WebScraping (Richardson, 2020).

Jupyter Notebook

Project Jupyter es un proyecto de código abierto sin fines de lucro (Project Jupyter, 2020). Esta librería de Python permite ejecutar códigos de Python en formato HTML, es decir, como si fuera una página web. Esta librería es ampliamente utilizada para el análisis de datos, por la facilidad de ejecutar celdas e ir observando el comportamiento de datos o visualizar los mismos.

Tensorflow

Es un Framework para el desarrollo de inteligencia artificial creado por Google.

Matplotlib

Es una biblioteca para crear visualizaciones estáticas, animadas, o iterativas en python (Hunter *et al.*, 2020)

Tokenizer

Es el proceso de dividir texto por palabras y convertirlo en números.

Anexos C

Anexo III: Obtención de los datos

El primer paso del KDD es la obtención de los datos; estos datos pueden ser proveídos de diversas fuentes, aunque, cabe destacar, que como en este obra, es sobre el análisis de los textos y su capacidad de predicción, se pasará a tomar en cuenta las fuentes posibles de obtención de datos de textos:

- API's: Que proporcionan colecciones de datos, las cuales pueden ser obtenidas con relativa facilidad, como lo es en el caso de *Twitter* API's (Shivananda, & Kulkarni, 201).
- Datos de archivos World: Los datos de archivos World, de igual manera que el caso de PDF's, los Word's, también, se puede extraer la información de manera relativamente sencilla usando la librería "docx", librería también de *Python* (Shivananda & Kulkarni, 2019).
- Datos de JSON: JSON(JavaScript Object Notation), es un tipo de formato para el manejo de documentos basado en JavaScript; el cual, en Python, es fácil de manejar, ya que JSON tiene una estructura muy similar a los diccionarios de Python. Para poder trabajar con archivos JSON en Python se requiere el uso de la librería json (Shivananda & Kulkarni, 2019).
- Datos de HTML: También se puede obtener datos de la Web, los cuales, por lo general tienen un formato HTML, el cual, implica un reto relativamente complejo, ya que

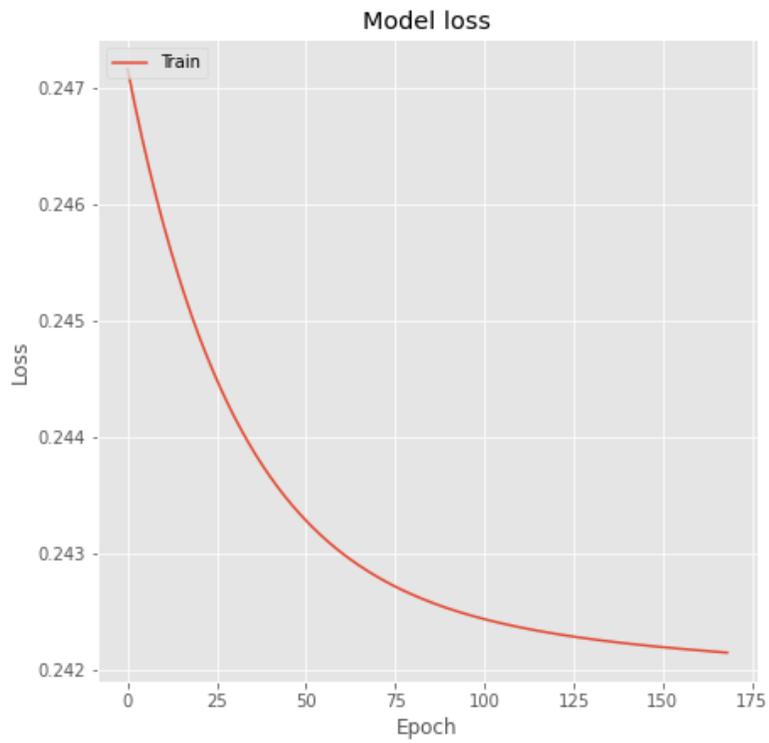
para ello, se tiene que tener nociones básicas de HTML; teniendo conocimiento básico de ello, se puede hacer uso de librerías, de Python, como BeautifulSoup4, requests o urllib.request, para realizar peticiones a páginas web y extraer la información de la página, a esto se le conoce como WebScraping (Shivananda & Kulkarni, 2019); este proceso es el que se aplica a la presente tesis para obtener la información de Reuters.

- También se puede realizar extracción de información textual de ebooks, entre otros.

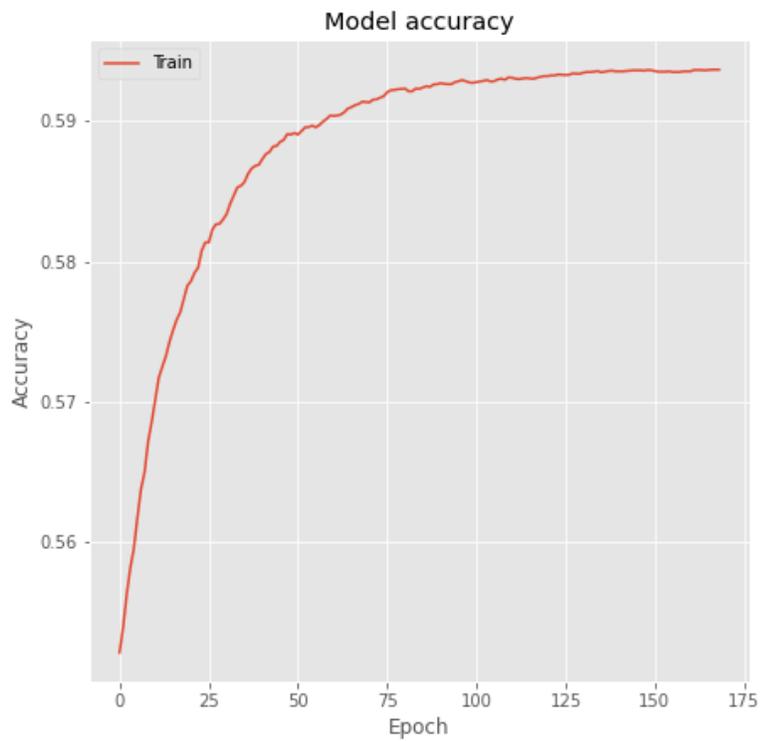
Anexos D

Anexo IV: Modelo entrenado con todos los datos

En el entrenamiento con todos los datos el error (E) fue de: 0.2422, y la precisión de 0.5937. El histórico del entrenamiento se muestra en la siguiente página:



Gráfica D.1: Error cuadrático medio (modelo 23).



Gráfica D.2: Validación (modelo 23).

Anexos E

Anexo V: Link para descargar modelo

En los siguientes link estarán el modelo y modelos entrenados, base de datos y código para cargar en un Jupyter Notebook:

- Modelo y modelos entrenados: <https://bit.ly/34IoQv9>
- Base de datos: <https://bit.ly/2VxXrbn>
- Código: <https://bit.ly/3cnYIZ7>