



Universidad de Valladolid

**ESCUELA DE INGENIERÍA INFORMÁTICA
DE VALLADOLID**

**Grado en Ingeniería Informática
Mención en Computación**

**Predicción y Clasificación
de series temporales bursátiles mediante
Redes Neuronales Recurrentes.**

Alumno: César Hernández Rodríguez

Tutor: Teodoro Calonge Cano

Cotutor: Julio Herrera Revuelta

Índice general

Lista de figuras	VII
Lista de tablas	IX
Resumen	XV
Abstract	XVII
I Memoria del Proyecto	1
1. Descripción del proyecto	3
1.1. Introducción	3
1.2. Motivación	4
1.3. Objetivos del trabajo	5
1.4. Estructura	6
2. Gestión del Proyecto	9
2.1. Metodología de trabajo	9
2.1.1. Identificación de actividades	10
2.2. Entregables del Proyecto	11
2.3. Herramientas utilizadas	11
2.3.1. Windows vs Linux	12
2.3.2. R vs Python	13
2.3.3. Pytorch vs Keras	13
3. Planificación	15
3.1. Planificación inicial del proyecto	15
3.2. Variaciones respecto a la planificación inicial	16

4. Marco Teórico	19
4.1. Redes Neuronales Artificiales	19
4.1.1. Breve Historia de las Redes Neuronales	20
4.1.2. Funcionamiento de una Red Neuronal Artificial	22
4.1.3. Perceptrón Multicapa	23
4.1.4. Entrenamiento de Redes Neuronales	23
4.2. Redes Neuronales Recurrentes	24
4.2.1. Celda LSTM	26
4.2.2. Celda GRU	28
4.3. Bolsa y Estadística Bursátil	29
4.3.1. Análisis fundamental	31
4.3.2. Análisis técnico	32
4.3.3. NASDAQ	40
4.3.4. Valores de las acciones	41
4.3.5. Inversión de capital	41
5. Metodología Experimental	43
5.1. División de los Datos	43
5.2. Elección de Modelos	44
6. Descripción y Preprocesamiento de los datos	47
6.1. Datos	47
6.2. Descripción de los datos	48
6.3. Preprocesamiento de los Datos y extracción de características	49
6.4. Decisiones	50
7. Implementaciones de las Redes Neuronales Recurrentes	51
8. Resultados	55
8.1. Elección de número de épocas	55
8.2. Elección de Redes Neuronales	56
8.2.1. SimpleRNN con secuencia de 21 días	56
8.2.2. GRU con secuencia de 21 días	58
8.2.3. LSTM con secuencia de 21 días	58
8.2.4. LSTM con secuencia de 252 días	60
8.3. ATVI - Activision Blizzard	61
8.4. AAPL - Apple Inc	64
8.5. GOOGL, Google Inc	67
8.6. BBOX, Black Box Network Services	69
8.7. MINI - Movable Mini, Inc	71
8.8. SNRC - Synchronoss Technologies, Inc	74

9. Conclusiones	77
II Apéndices	81
A. Anexos	83
A.1. Gráficas de Predicciones	85
A.2. Tablas de decisiones	91
III Bibliografía	93

Índice de figuras

1.1. Relación entre los términos de Inteligencia Artificial, Aprendizaje Automático y Aprendizaje Profundo	4
2.1. Diagrama de la metodología cascada	10
2.2. Diagrama de la metodología cascada	10
2.3. Estructura del repositorio en GitHub	12
3.1. Diagrama de GANTT	16
4.1. Partes de una neurona biológica	20
4.2. Modelo de Frank Rosenblatt	21
4.3. Modelo MultiCapa	21
4.4. Neurona recurrente y desenrolle a través del tiempo	24
4.5. Capa de neuronas recurrentes y desenrolle a través del tiempo	25
4.6. Combinaciones de entrada-salida	26
4.7. Celda LSTM	27
4.8. Celda GRU	29
4.9. Ejemplo de gráfico de líneas	34
4.10. Ejemplo de gráfico de barras	34
4.11. Ejemplo de gráfico de velas	35
4.12. Ejemplo de tendencia alcista	36
4.13. Representación de Resistencias y Soportes en tendencias alcistas y bajistas	36
4.14. Ejemplo de Media Móvil	37
4.15. Ejemplo de Media Móvil Aritmética	38
4.16. Ejemplo de Banda de Volatilidad	39
4.17. Ejemplo de Oscilador Sencillo	40
6.1. Datos Históricos	48
6.2. Datos históricos de SNCR para conjunto de entrenamiento y conjunto de datos históricos para conjunto test	50
8.1. Tasas de acierto de entrenamiento y de validación por épocas.	56
8.2. Datos históricos de ATVI para conjunto de entrenamiento	62

8.3. Datos históricos de ATVI para valores de predicción	62
8.4. Datos históricos de AAPL para conjunto de entrenamiento	65
8.5. Datos históricos de AAPL para conjunto de entrenamiento	65
8.6. Datos históricos de GOOGL para conjunto de entrenamiento	67
8.7. Datos históricos de GOOGL para conjunto de predicción	67
8.8. Datos históricos de BBOX para conjunto de entrenamiento	69
8.9. Datos históricos de BBOX para valores de predicción	70
8.10. Datos históricos de MINI para conjunto de entrenamiento	72
8.11. Datos históricos de MINI para valores de predicción	72
8.12. Datos históricos de SNRC para conjunto de entrenamiento	74
8.13. Datos históricos de SNRC para valores de predicción	74
A.1. Predicción de los modelos para el activo ATVI	85
A.2. Predicción de los modelos para el activo AAPL	86
A.3. Predicción de los modelos para el activo GOOGL	87
A.4. Predicción de los modelos para el activo BBOX	88
A.5. Predicción de los modelos para el activo MINI	89
A.6. Predicción de los modelos para el activo SNRC	90

Índice de cuadros

6.1. Tabla de primeros valores recogidos de la empresa SNCR	49
8.1. Errores de la SimpleRNN con diferentes Redes.	57
8.2. Test de Rankings de las redes SimpleRNN.	57
8.3. Errores de la GRU con diferentes Redes.	58
8.4. Test de Rankings de las redes GRU.	59
8.5. Errores de la LSTM-21 con diferentes Redes.	59
8.6. Test de Rankings de las redes LSTM-21.	60
8.7. Errores de la LSTM-252 con diferentes Redes.	60
8.8. Test de Rankings de las redes LSTM252.	61
8.9. Tasas de aciertos de los modelos elegidos para ATVI	62
8.10. Valores RMSE de las predicciones de los modelos elegidos para el activo ATVI	63
8.11. Ganancias y perdida por modelo y margen elegidos. Número de veces que se repite una decisión según cada modelo. Para el activo ATVI	64
8.12. Tasas de aciertos de los modelos elegidos para AAPL	66
8.13. RMSE de las predicciones de los modelos elegidos para AAPL	66
8.14. Ganancias y pérdidas por modelo y margen elegidos. Número de veces que se repite una decisión según cada modelo. Para el activo AAPL	66
8.15. Tasas de aciertos de los modelos elegidos para GOOGL	68
8.16. RMSE de los modelos elegidos para GOOGL	68
8.17. Ganancias y perdida por modelo y margen elegidos. Número de veces que se repite una decisión según cada modelo. Para el activo GOOGL	69
8.18. Tasas de aciertos de los modelos elegidos para BBOX	70
8.19. RMSE de los modelos elegidos para BBOX	71
8.20. Ganancias y perdida por modelo y margen elegidos. Número de veces que se repite una decisión según cada modelo. Para el activo BBOX	71
8.21. Tasas de aciertos de los modelos elegidos para MINI	73
8.23. Ganancias y perdida por modelo y margen elegidos. Número de veces que se repite una decisión según cada modelo. Para el activo MINI	73
8.22. RMSE de los modelos elegidos para MINI	73
8.24. Tasas de aciertos de los modelos elegidos para SNRC	75

8.25. RMSE de los modelos elegidos para SNRC	75
8.26. Ganancias y perdida por modelo y margen elegidos. Número de veces que se repite una decisión según cada modelo. Para el activo SNRC	75

*Dedicado a las personas que
han colaborado en de mi proceso de aprendizaje
en todos estos años*

Agradecimientos

"La religión emergente más interesante es el dataísmo, que no venera ni a dioses ni al hombre: adora los datos". (Yuval Noah Harari. Homo Deus: Breve historia del mañana - 2016)

Agradezco a mis padres por esfuerzos, consejos y atención que me han dedicado. A mi hermano Abel por obligarme a ser un ejemplo para él. A mi novia Lorena por estar a mi lado durante toda la carrera y verme crecer.

Por último, a mis compañeros de clase por ayudarme en todo momento y sin miramientos y mis amigos por darme esa desconexión necesaria siempre que se necesitaba.

Resumen

En este trabajo se pretende explorar la capacidad del Aprendizaje Profundo en la predicción de series temporales, en particular el uso de Redes Neuronales Recurrentes, para predecir valores diarios bursátiles.

Se han empleado Redes Neuronales con distinto poder para procesar el recuerdo a largo plazo: LSTM, GRU, SimpleRNN. Experimentando cada tipo con diversas series correspondientes a la Bolsa de Valores NASDAQ, se ha tratado de obtener el mejor modelo de predicción para cada caso concreto. Asimismo, se ha desarrollado una clasificación de dichas series para mejorar el asesoramiento a un posible inversor sobre la compra o venta de valores bursátiles.

Desde el punto de vista de la Inteligencia Artificial, este trabajo ha puesto en práctica Redes Neuronales Recurrentes que, por tamaño y longitud de secuencias procesadas, han derivado en lo denominado Aprendizaje Profundo, se ha usado la herramienta *Keras* para su implementación; con lo que el uso: *TensorFlow* y *Python*.

En el preprocesamiento y comparación de datos se han usado técnicas propias de la Estadística, lo que incluye un manejo del análisis técnico e indicadores específicos como las medias móviles y los osciladores.

Palabras claves: Redes Neuronales, LSTM, GRU, Finanzas, Aprendizaje Profundo, Mercados financieros, Análisis técnico

Abstract

This paper intends to explore the potential of Deep Learning for predicting time series. More specifically, the use of Recurrent Neural Networks (RNN) to forecast daily stock values.

Neuronal Networks with different power have been used to process long-term memory; LSTM, GRU, and SimpleRNN. Each kind of network has been tested with different corresponding series in NASDAQ in an attempt to obtain best prediction model for each case. Similarly, said series have been classified to provide advice for a potential investor on the purchase and sale of stocks.

From the point of view of Artificial Intelligence, this project has been useful to put RNN into practice, which due to its size and the length of the processed sequences have derived in the so-called Deep Learning. The tools *Keras*, *TensorFlow*, and *Python* have been used for its implementation.

In addition, this project has used statistical techniques to preprocess data and compare results, which includes technical analysis along with specific indicators like moving averages and oscillators.

Key Words: Neuronal Network, LSTM, GRU, Finance, Machine Learning, Deep Learning, Financial Market, Technical Analysis

Parte I

Memoria del Proyecto

Capítulo 1

Descripción del proyecto

1.1. Introducción

Durante los últimos años de avances tecnológicos, la inteligencia artificial ha generado mucha expectación. Gran cantidad de artículos, de temas no solo tecnológicos mencionan *Aprendizaje Automático (Machine Learning)* y *Aprendizaje Profundo (Deep Learning)*.

Para poder trabajar en estas tres disciplinas se necesita una aclaración (ver figura 1.1).

- La Inteligencia Artificial (IA) engloba todo este campo de la Ciencia de Computación, que nació con el objetivo de hacer que los ordenadores "piensen".
- El Aprendizaje Automático (AA) nace para manejar el conocimiento implícito a través de ejemplos.
- El Aprendizaje Profundo (AP) es un paradigma basado en combinar múltiples sistemas de Aprendizaje Automático organizado en muchas capas.

Este último paradigma se remonta al modelo propuesto por McCulloch Pitts en 1943, cuando creó las primeras redes neuronales artificiales intentando emular de forma matemática el funcionamiento de una neurona, que junto con otras podrían organizarse en capas, estructura que mostraron los primeros estudios del cortex del cerebro. No fue hasta el 1986 con la difusión del algoritmo de retropropagación del error del perceptrón multicapa, cuando las redes neuronales artificiales experimentaron una gran auge. Desde entonces, estos sistemas no han dejado de aplicarse. Pueden haber sufrido un estancamiento, pero nunca han sido descartados como parte de la solución a ciertos problemas en IA.

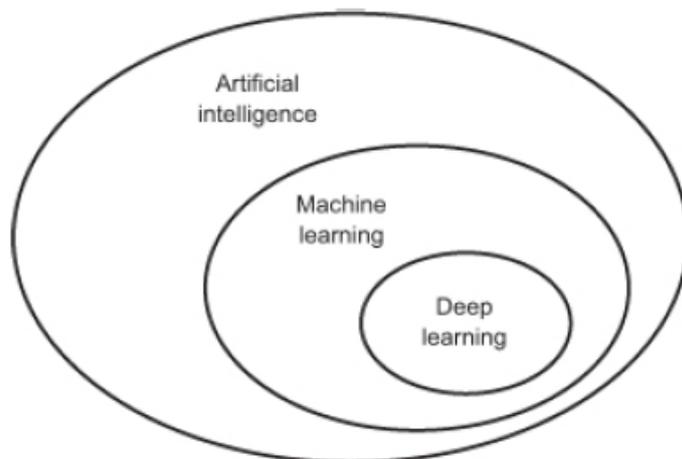


Figura 1.1: Relación entre los términos de Inteligencia Artificial, Aprendizaje Automático y Aprendizaje Profundo

Por otro lado los mercados financieros, los cuales cuentan con numerosos instrumentos bursátiles para ayudarse a la hora de invertir. Desde los comienzos, los inversores más adelantados no solo se basaban en el instinto y la experiencia, sino que buscaban comprender y buscar explicaciones a las variaciones de los precios del mercado.

En la actualidad, gracias a la cantidad de datos que tenemos, podemos "predecir" hacia dónde irá el valor de una empresa ya sea, siguiendo una serie de análisis y métodos estadísticos, analizando las propias compañías y el mercado, como datos socio-políticos y psicología de mercado. Lo que menos posibilidad tenemos de poder saber es cuándo pasará una catástrofe mundial como huracanes, terremotos, pandemias, etc.

Uno de los puntos mas fuertes a la hora de buscar una explicación de la variación de precios en la Bolsa recae sobre las series temporales, cómo fluye el precio cada día y el por qué. Existen una gran cantidad de métodos para ayudarse a la hora de invertir, y en un poco tiempo, cierto tipo de Redes Neuronales serán el pan de cada día en el ámbito bursátil, debido a que el paradigma lineal se ha mostrado insuficiente para explicar el comportamiento de las variables y fenómenos financieros. [11]

1.2. Motivación

A nivel personal este Trabajo Fin de Grado tiene como meta aprender más sobre el campo de la Economía y el Aprendizaje Automático. Con este trabajo sigo la línea que empecé hace 6 años en Segundo de Bachillerato donde realicé un proyecto sobre optimi-

zación de carteras con datos del Ibex 35. Este trabajo se basaba en calcular rendimientos y varianzas de distintos activos y poder seleccionar la forma más óptima de diversificar el capital entre todas las empresas, utilizando el modelo de Markowitz. La idea de poder seguir aprendiendo sobre la Bolsa y las inversiones, ya que me parece un tema muy interesante y que lleva muchos años en los periódicos, no sólo por su repercusión mediática, sino por sus consecuencias negativas que conllevan, tanto a nivel colectivo, como individual. Desgraciadamente, a corto plazo, no se prevé que dichas crisis, que se repiten cíclicamente, vayan a desaparecer.

Siguiendo esta línea, he realizado un trabajo en paralelo a este proyecto sobre inversión en Bolsa, realizando el Trabajo Fin de Grado del Grado en Estadística sobre optimización de carteras, [12]. Junto con el uso del lenguaje Python y el análisis de la Bolsa de Valores NASDAQ.

Una motivación extra es el incentivo económico que podría ser la inversión en la Bolsa. Los mercados financieros mueven cantidades de dinero inimaginables y el hecho de saber invertir, o por lo menos entender el funcionamiento de cómo se mueve el dinero, me parece llamativo.

Desde el punto de vista Informático, con este TFG se persigue, entre otras cosas, aprender, tanto como sea posible, sobre las metodologías y su aplicación en *Python*. En ningún momento se espera encontrar modelos con los que ganar dinero, o que puedan predecir exactamente el valor, ya que es prácticamente imposible, pero sí modelos que puedan ser un inicio para el acercamiento a la predicción bursátil, y la creación de un indicador bursátil.

Por otro lado me gustaría que este Trabajo Fin de Grado me sirviera de mérito en mi currículum ante un posible empleo en el sector relacionado con el mundo bursátil, en particular, en su predicción.

1.3. Objetivos del trabajo

Con este TFG se pretende realizar una labor de investigación sobre la Bolsa y las Redes Neuronales Artificiales (RNA), como meta de poder predecir valores o tendencias de un activo mediante diferentes modelos particulares de este paradigma, como las llamadas de Redes Neuronales Recurrentes, así como extraer los datos automáticamente y dar una ayuda al inversor sobre qué hacer con su dinero, si se recomienda mantener la posición, comprar un activo o venderlo.

Así pues, el análisis se basa en predecir un valor de un activo en función de los datos históricos preprocesados con diferentes procedimientos estadísticos muy frecuentes en el ámbito bursátil, como medias móviles y osciladores.

Otro objetivo del trabajo, consiste en dar los primeros pasos para crear un indicador bursátil, un programa que pueda decidir que acción se debe hacer en cada momento, sin la necesidad de la interacción humana directa. Este objetivo va acompañado de poder generar una RNA con diferentes celdas, que tengan la capacidad de preveer la fluctuación positiva o negativa de un activo.

El último objetivo es plantear un problema real en el que la RNA creada resuelva el problema, en este caso, el proceso a seguir será realizar una simulación de inversión de un capital inicial en diferentes empresas, y observar las ganancias o pérdidas que predice la red y comparar esos valores con los reales, tomando decisiones de inversión diarias.

1.4. Estructura

Establecidos los objetivos, se propone la siguiente estructura para lo que resta de memoria de este TFG:

- **Capítulo 2 - Gestión del Proyecto.** Se presentan las tareas para realizar en este proyecto, la metodología de trabajo y las herramientas usadas, así como sus estructuras.
- **Capítulo 3 - Planificación.** Se continuará el diagrama de GANTT, y cómo se dividen las diferentes tareas en las fechas de presentación.
- **Capítulo 4 - Marco Teórico.** Se explica la teoría sobre las RNA, en especial las recurrentes, ya que son las únicas que se han empleado en este trabajo. Por otra parte se explica qué es la Bolsa, y cómo se aplica la Estadística a este campo y de qué manera.
- **Capítulo 5 - Metodología Experimental.** En este apartado únicamente se expone la forma de trabajar con los datos, qué decisiones se han tomado para elegir los modelos y su fundamento teórico.
- **Capítulo 6 - Descripción y Preprocesamiento de los datos.** Se aborda la descripción de los distintos conjuntos de datos, que se utilizan en el desarrollo del TFG, y el preprocesamiento realizado con cada uno de ellos.
- **Capítulo 7 - Implementación de las Redes Neuronales Recurrentes y su entrenamiento con datos bursátiles.** Se expone el framework *keras* y sus diferentes aproximaciones de aprendizaje de esta librería, a la vez que se va haciendo ejemplos de confección de modelos, que servirán de base a los empleados más tarde en este trabajo.

- **Capítulo 8 - Resultados.** Se muestran todos los modelos elegidos, la decisión de cuáles son mejores y aquellos en los que merecería realizar una inversión. Para ello, se hace un análisis de cada activo, obteniendo las pertinentes conclusiones basadas en las predicciones de la RNA.
- **Capítulo 9 - Conclusiones.** Se recoge la idea de los resultados y se intenta resumir todo el trabajo, haciendo hincapié en los conocimientos aprendidos, así como en futuras aplicaciones y ampliaciones del presente TFG.

Capítulo 2

Gestión del Proyecto

2.1. Metodología de trabajo

Las metodologías para desarrollar software vistas en la carrera, existen dos tipos: las tradicionales y las ágiles, muy en auge desde hace tiempo debido a su flexibilidad y adaptabilidad. A pesar de todo eso, se ha decidido emplear una metodología tradicional ya que durante la carrera ha sido la forma de trabajo. Dentro de ellas he optado por un desarrollo iterativo y creciente basado en incrementos o entregables.

Creo que este enfoque es el más óptimo debido al carácter modular de este TFG, lo que proporciona un mayor margen de maniobra, dentro de los posibles cambios que se vayan dando a la hora de conseguir objetivos. Para este punto se ha optado por una metodología en cascada, que es una de las más clásicas en el desarrollo de proyectos, los detalles se pueden ver en [14]. Se ha hecho ciertas variaciones dentro de esta metodología, ya que esta preparada para desarrollo de aplicaciones, mientras que en este proyecto se ha tomado una línea más de investigación.

El desarrollo de la metodología en cascada es fundamentalmente secuencial hacia abajo, como se referencia en el diagrama 2.1. Por el tipo de proyecto que se va a trabajar, es bastante probable un desarrollo en espiral, debido a errores o mejoras que se vayan implementando al código. Es muy necesario volver reiteradamente a la fase de pruebas y a la fase de diseño por la necesidad de cambios de hiperparámetros en las RNA, tanto número de capas, como de unidades, y/o de secuencias.

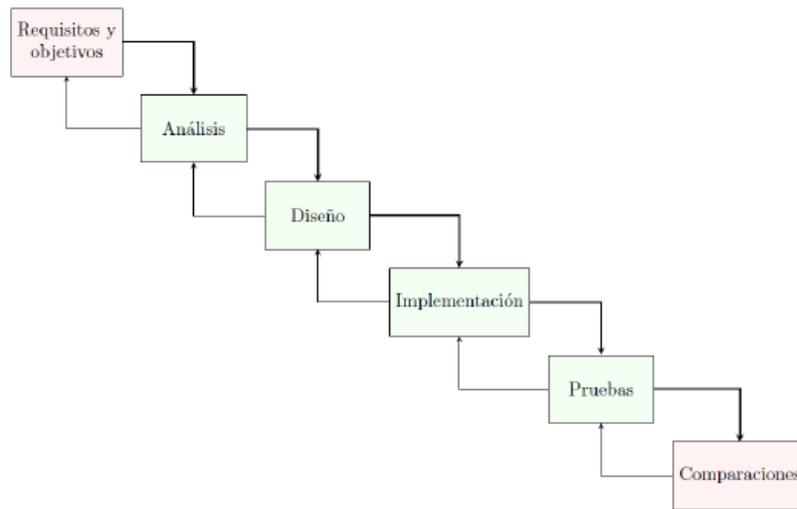


Figura 2.1: Diagrama de la metodología cascada

2.1.1. Identificación de actividades

Las actividades principales necesarias para completar los objetivos se recogen en una estructura de descomposición del trabajo visible en la figura 2.2. En este diagrama se puede ver las actividades llevadas a cabo, sin estar desglosadas. Todas estas actividades se realizan dentro de cada entregable, sobretodo de cara a los últimos, que es donde más comparaciones se hacen con diferentes modelos.

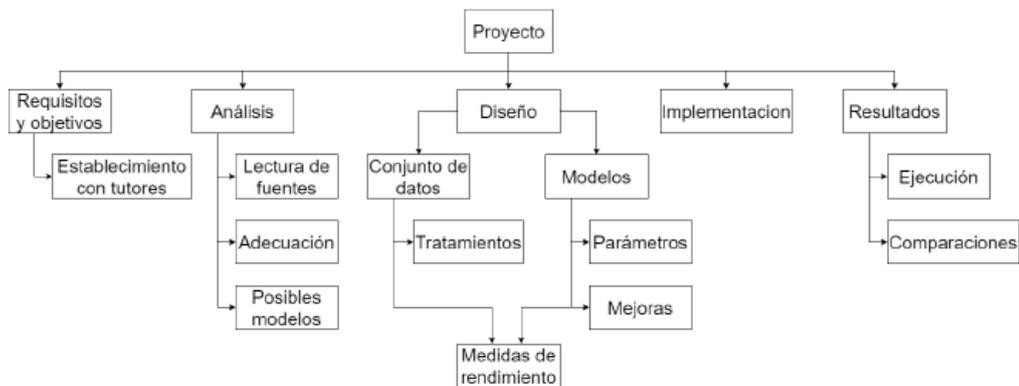


Figura 2.2: Diagrama de la metodología cascada

2.2. Entregables del Proyecto

- Entregable 1: Red LSTM versión Demo con secuencia de 21 días
- Entregable 2: Red LSTM versión Demo con secuencia de 252 días
- Entregable 3: Red GRU y SimpleRNN versión Demo con secuencia de 21 días.
- Entregable 4: Red LSTM versión Extendida con secuencia de 252 días.
- Entregable 5: Red LSTM, GRU y SimpleRNN versión Extendida con secuencia de 21 días.
- Entregable 6: Red LSTM, GRU y SimpleRNN con activos preseleccionados con secuencia de 21 días.
- Entregable 7: Red LSTM con activos preseleccionados con secuencia de 252 días

Como se observa, el presente TFG consta de bastantes entregables siendo la mayoría leves modificaciones del programa inicial o la inclusión de nuevos modelos de red. Otra parte importante dentro de cada entregable será la cantidad de datos ya que, aunque la versión final cuente solo con unas pocas empresas, se quiere buscar un programa que pueda valer para cualquier empresa que este dentro del índice NASDAQ.

2.3. Herramientas utilizadas

Dentro de este conjunto de recursos, de podría tratar en dos bloques: las herramientas de hardware y las de software. Esta diferenciación adquiere especial relevancia en este contexto, ya que los tiempos de ejecución son considerables y, por ello, merece un tratamiento diferenciado las herramientas hardware del resto. En este sentido, se me ha prestado una máquina virtual por parte de la escuela con 12 núcleos virtuales, que en la práctica son cada uno de los incorporados en un Intel(R) Xenon(R) Gold 6128 CPU 3.4Ghz. Respecto a la RAM, constaba de 16GB DDR4 a 2.66MHz.

Por la parte de Software se ha utilizado la suite Anaconda [1] para poder trabajar con Python [2] como lenguaje recomendando para desarrollar en AA en Machine Learning. El Sistema Operativo elegido para trabajar ha sido Linux (ubuntu 20.04) pre-instalado en la máquina virtual. Por otra parte, para poder llevar una control de versiones de los diferentes códigos, se ha decidido usar GitHub [3].

Todos los programas generados, tanto resultados como datos necesarios, se encuentran en el repositorio de GitHub personal, https://github.com/CesarHernandez73/TFG_Informatica.

La estructura de este se observa en la figura 2.3. En la carpeta *Decisiones*, se encuentran diversos *.csv*, con las decisiones tomadas cada día en la simulación de inversión.

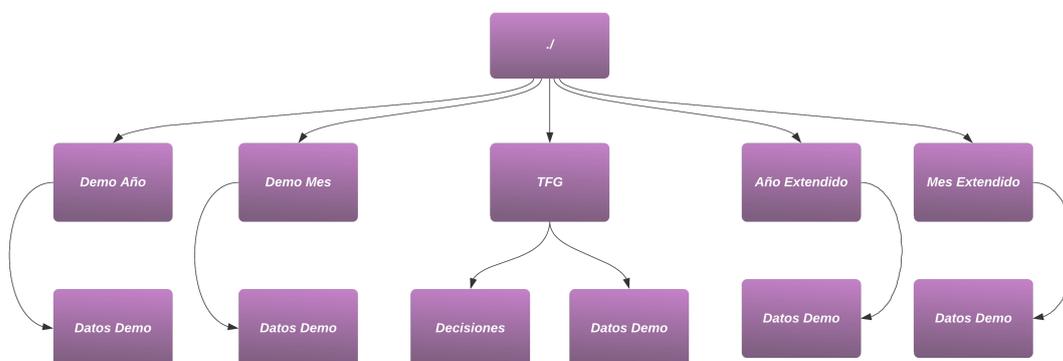


Figura 2.3: Estructura del repositorio en GitHub

Por otra parte, se opta por hacer uso del editor de LaTeX online *Overleaf* para crear la documentación. Se ha elegido la utilización de un editor online, ya que permite trabajar en el ordenador personal y en la máquina virtual por igual, evitando, de esta manera, todo el proceso de instalación de LaTeX y de cada librería que se usan.

Dentro de la elección de las herramientas tuve muchas dudas, al inicio, sobre que software usar y cuál podría dar un mejor rendimiento, por lo que realicé una comparativa ante las diferentes opciones.

2.3.1. Windows vs Linux

El primer punto fue elegir si usar Linux o Windows a la hora de programar. Siempre durante la carrera y mi tutor del proyecto me han alentado a usar Linux. Leyendo diferentes foros por internet, saqué como conclusión que no debería haber problemas entre usar un sistema operativo o el otro, ya que ambos están bastantes preparados para programar en ellos. Aún así, Linux tiene ciertas ventajas con diferentes paquetes especializados, como es el caso de la Inteligencia Artificial.

Las ventajas principales que he encontrado con Linux son:

- Es gratis.
- Es mas seguro.

- Es código abierto.
- Tiene un mejor rendimiento.
- Una comunidad muy activa que, al enfrentarse y solucionar directamente muchos problemas, pone su experiencia al servicio de otros miembros de manera altruista.

Al ver que el funcionamiento del programa Anaconda era mejor en Linux, junto con las ventajas relatadas anteriormente, decantó mi apuesta por este sistema operativo.

2.3.2. R vs Python

Dentro del entorno Anaconda, se permite el uso tanto del lenguaje R como Python. Como estudiante del doble grado de Estadística e Ingeniería Informática por lo que se me planteó un dilema a la hora de optar por uno de ellos. Desde el inicio, ambos lenguajes han sido los más usados durante mis años de estudio, y con lo que más cómodo me siento.

En primer lugar, el uso de Python me atraía más debido a la facilidad de descarga de datos de la Bolsa. Por otro lado, contaba con la experiencia de su uso en las asignaturas de IA, en particular, en el último curso a la hora de programar RNA

con vistas a la programación y a la rapidez de los cálculos, busqué en internet comparaciones de ambos lenguajes dentro de AA. Aunque R está preparado para todo el campo de la Estadística y la programación derivada de ésta, Python ha desarrollado paquetes con una gran potencia computacional que entrega idénticos resultados en tiempos sensiblemente menores que R. Para la manipulación de datos y repetición de tareas (parte importante de este proyecto) se aconseja el uso de Python, mientras que R se recomienda solo para los pasos iniciales y para el desarrollo de análisis ad-hoc del problema, así como para la exploración de datos.

Debido a estos blogs que leí, me decante por el uso de Python.

2.3.3. Pytorch vs Keras

Escogido Python como lenguaje de trabajo, se plantean varias alternativas para la implementación de las Redes Neuronales Recurrentes: programación directa o uso de alguna plataforma. Lo primero es descartado, porque se carecen de los conocimientos suficientes para hacer que el código sea paralelizable, que es lo que aceleraría ostensiblemente el cómputo

Por tanto, la opción de la plataforma específica parece la opción más razonable. Pero aquí también hay muchas alternativas. Por extenderse en este punto, se resumirían en dos: Keras y Pytorch.

La mayoría de las opciones se basan en tensores que, a niveles prácticos son similares a un vector multidimensional, pueden almacenar valores enteros o dobles y tienen un número de ejes (la dimensión de un vector), y una forma y tamaño por cada eje.

En el último curso de mis estudios se ha dado ciertas partes y los primeros pasos de programación de redes neuronales con Keras pero, como en las secciones anteriores, quería explorar las ventajas e inconvenientes de cada librería.

Keras no es un framework en sí, es programación alto nivel de una API concebida para el Deep Learning. Esta basada sobretodo en TensorFlow. La facilidad de uso de esta librería la da mucha ventaja, sobretodo para hacer trabajos de AA no excesivamente complicados, como es el caso. Es muy intuitiva y su uso permite la definición de redes neuronales con diferentes capas y funciones, pero de una forma algo limitada. Es capaz de ejecutarse sobre TensorFlow, Theano o CNTK, debido a ello su eficiencia disminuye.

Por otro lado Pytorch es un framework para Deep Learning (como TensorFlow). Comparado con Keras es bastante más abstracto y tiene partes más complicadas o liosas a la hora de programar redes neuronales. Personalmente revisé e intenté ciertas redes fáciles con este framework y me resultó costoso entender todo.

Decidí realizar entre proyecto con Keras y algo de programación de TensorFlow, porque vi más factible alcanzar los objetivos de este TFG en el tiempo estipulado, que con Pytorch, en el que la curva de aprendizaje es más pronunciada y, por tanto, cumplir con los plazos de tiempo se aventuraba casi imposible.

Capítulo 3

Planificación

El presente TFG tiene 12 ECTS. Teniendo en cuenta esta cifra, en el actual marco de traducción a horas de la UVA, implicaría una dedicación estimada de 300 horas por parte del estudiante. El inicio data del 11 de Febrero de 2021 y su fecha estimada de entrega es el 23 de Junio, lo que prorrateado sería a 19 semanas, lo que equivale a unas 16 horas por semana.

3.1. Planificación inicial del proyecto

Como se observa en el diagrama de GANTT (ver figura 3.1) se inicia con dos fases de relativas al estudio y aprendizaje de conceptos importantes.

- Redes Neuronales Recurrentes
- Evolución temporal de valores bursátiles

Se realizó esta división debido a que se tratan de temas con una formación escasa o nula. Siendo esta una de las partes más importantes del proyecto debido al carácter de investigación que contiene.

Con cada entregable se busca un producto totalmente funcional que se pueda presentar a un supuesto cliente, en este caso a los profesores tutores de la UVA. A continuación, en sucesivas entrevistas con estas personas, se van corrigiendo errores de lo entregado y se plantean los siguientes pasos en esta investigación. Como aclaración dentro del calendario de tareas, las reuniones con los tutores están establecidas dentro de las tareas planificadas.

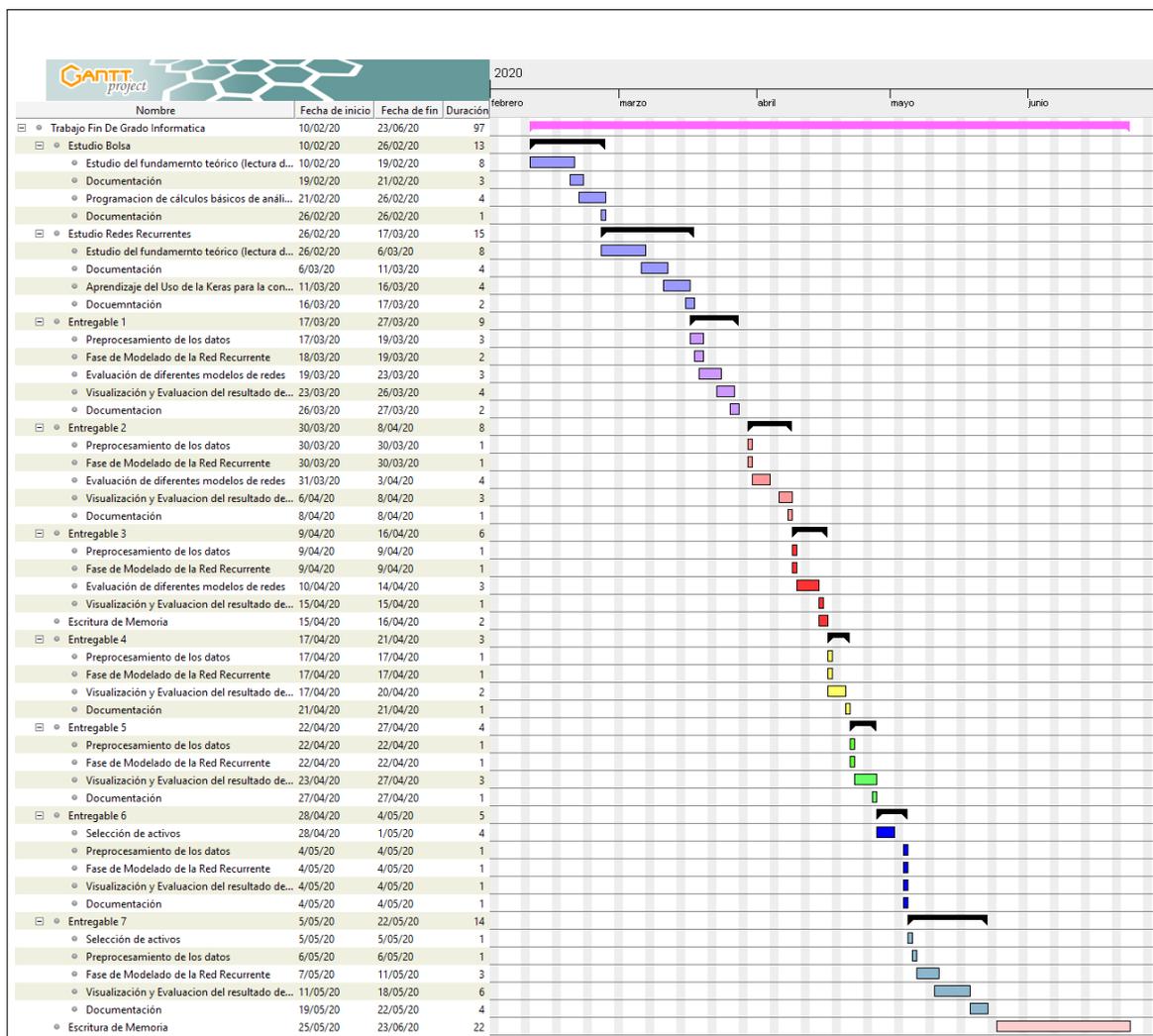


Figura 3.1: Diagrama de GANTT

3.2. Variaciones respecto a la planificación inicial

Durante todo el desarrollo del proyecto se ha tenido inconvenientes y cambios de la planificación inicial, obligando a reestructuraciones de este y procediendo a aplazar la fecha de entrega, cambiando la fecha de la convocatoria ordinaria a la convocatoria extraordinaria.

Los mayores imprevistos que han requerido modificación más notoria han sido:

- El exceso de optimismo inicial de cara a aprender los conceptos básicos de redes neuronales y de la Bolsa. Son campos muy grandes, que no se pueden aprender de un día para otro y se necesita un interés continuo para adquirir el conocimiento.

- La falta de potencia y memoria de la máquina personal, lo que implicó pedir una máquina virtual de mayor potencia, al Departamento de Informática de la UVa
- El tiempo de ejecución de las RNN con longitudes de secuencia muy grandes, debido a la inexperiencia en el diseño de las mismas.
- Acorde al punto anterior, la cantidad de modelos que he probado a lo largo del trabajo conjunto con sus tiempos, han generado retrasos en la planificación como tal.
- La obtención de la base de datos generó problemas, debido a que la cuenta dentro de la API con cuenta gratuita que dio ciertos problemas sobre los activos se podían descargar y cuales no, y en qué fechas.
- Las situación excepcional debido a la pandemia del coronavirus.
- Derivado del cambio de calendario académico por la covid-19, el periodo de simulación coincidió con el periodo de mantenimiento y puesta a punto del sistema de virtualización del Departamento de Informática de la UVa. Por esta razón, las ejecuciones en la máquina virtual han sufrido cortes, lo que ha supuesto retrasos en la obtención de resultados.

Capítulo 4

Marco Teórico

En este apartado se va a detallar nociones fundamentales tanto de Economía como de RNA necesarias para entender el estudio realizado sobre los valores de la Bolsa. Lo primero será realizar un fundamento teórico sobre las Redes Neuronales, en general, y las recurrentes de forma siguiente. A continuación, se expondrá una serie de conocimientos iniciales sobre el por qué de este estudio y que puede aportar la Inteligencia Artificial al campo de la Bolsa.

4.1. Redes Neuronales Artificiales

La naturaleza ha sido y será fuente inspiradoras para multitud de sistemas de la Ingeniería. Pues bien en la Inteligencia Artificial, es uno de los campos que más hace uso de modelos naturales para la construcción de otros totalmente artificiales. Dos ejemplos muy conocidos son los Algoritmos Genéticos y las RNA. Precisamente este último es en el que se basan los sistemas de IA empleados en este trabajo. Esta arquitectura es la idea clave que inspiró las Redes Neuronales Artificiales, siendo un modelo computacional vagamente inspirado en el comportamiento observado en su homólogo biológico [9].

4.1.1. Breve Historia de las Redes Neuronales

Las Redes Neuronales fueron introducidas en 1943 por el neurofisiólogo Warren McCulloch y el matemático Walter Pitts. Presentaron un modelo computacional simplificado, basándose en cómo funcionaría una neurona biológica de un animal.

Si miramos una neurona biológica, se compone del *cuerpo de la célula o soma*, que contienen el núcleo y la mayoría de los componentes más complejos. Tiene varias ramas, llamadas *dendritas*, y una extensión más grande denominada *axon*, como se ve en la figura 4.1. La neurona biológica recibe cortas descargas eléctricas llamadas *señales* desde otra neurona. Cuando recibe un suficiente número de señales, dispara su propia señal (sinapsis).

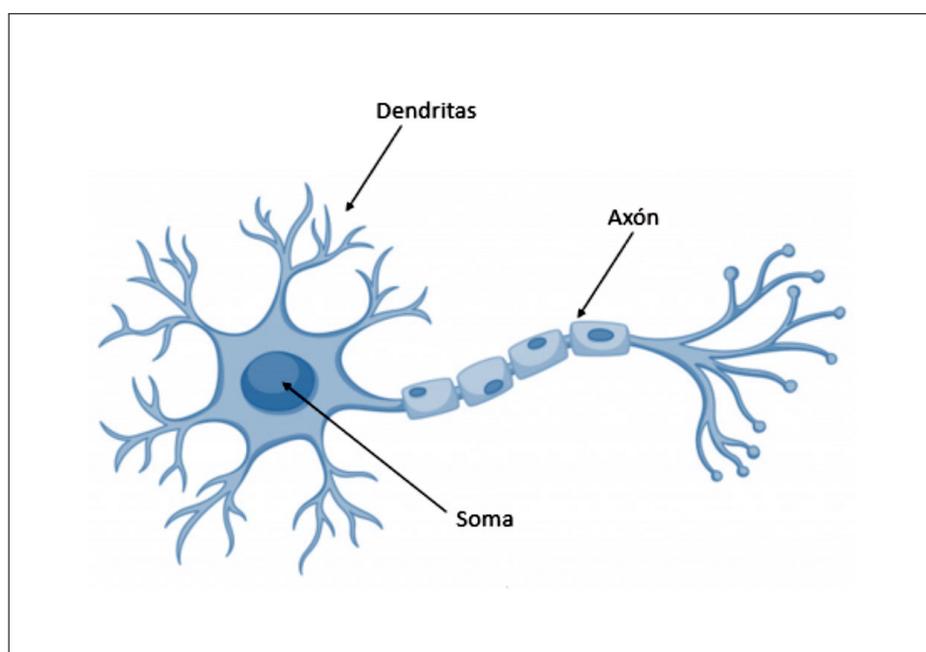


Figura 4.1: Partes de una neurona biológica

Una sola neurona tiene un comportamiento más bien simple, pero si se organizan billones de ellas, y cada una se conecta con miles de neuronas se crea una red muy compleja.

Warren McCulloch y Walter Pitts propusieron un modelo muy simple basado en la neurona biológica. Tenía una o varias entradas binarias y una sola salida también binaria.

Aunque era una idea que generó mucha expectación, este tema cayó en una época oscura debido a la falta de máquinas que pudieran implementar estos procedimientos. En 1957, Frank Rosenblatt desarrolla una de las arquitecturas más simples de las RNA, el *Perceptrón Simple* basado ligeramente en la neurona artificial llamada *Linear threshold unit* (LTU). En este caso las entradas son números, y que se conecta por unos pesos, la

salida se obtiene la sumando las entradas ponderadas por sus respectivos pesos, aunque antes de ser entregada como tal, es filtrada por una función de activación, como se puede observar en la figura 4.2. En el siguiente apartado se explicará con un poco mas de extensión el funcionamiento de cada neurona.

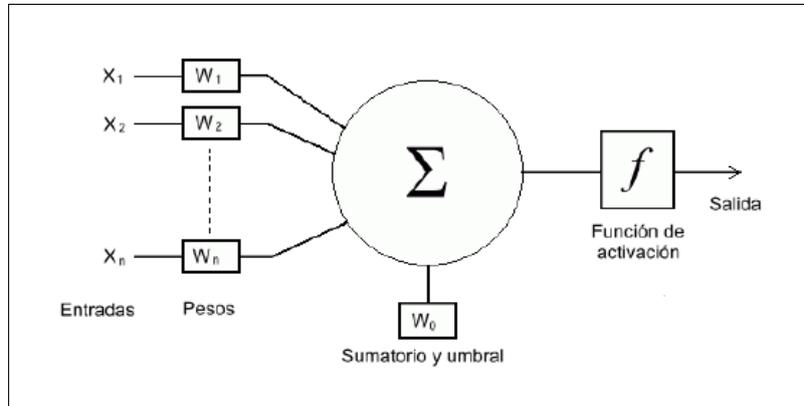


Figura 4.2: Modelo de Frank Rosenblatt

El *Perceptrón Simple* tiene muchas limitaciones, se decidió, apilar muchos perceptrones de manera que se creó el *Perceptrón MultiCapa* y, con él la propagación hacia atrás, la idea de esta nueva arquitectura se basa en diferentes capas, la de entrada, la de salida y las escondidas como se ve en la imagen de 4.3. Durante muchos años hubo muchas investigaciones sobre cómo conseguir un entrenamiento esta arquitectura fuera exitosas, fue en 1986, cuando D. E. Rumelhart publicó un artículo rompedor sobre la propagación hacia atrás para entrenar el algoritmo[21], lo que actualmente es la base del descenso de gradiente ampliamente utilizado en muchas RNA.

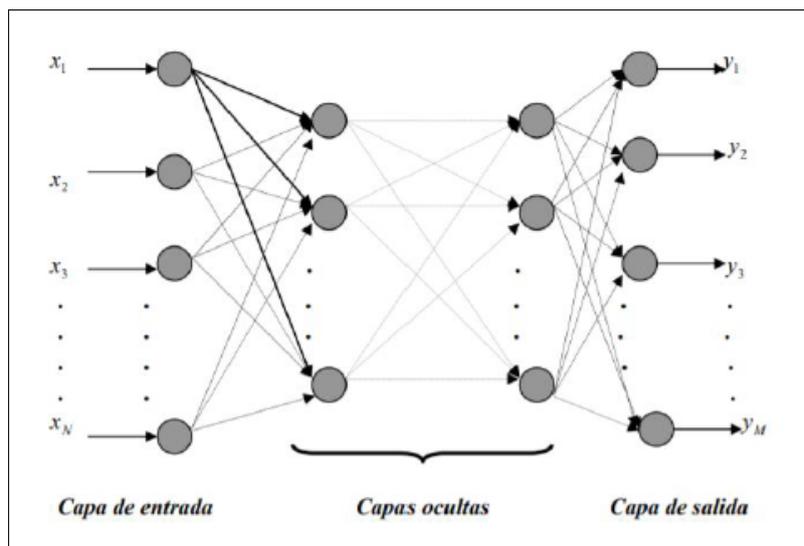


Figura 4.3: Modelo MultiCapa

4.1.2. Funcionamiento de una Red Neuronal Artificial

Basándonos en la figura 4.2, que es el modelo más sencillo, observamos cómo las entradas son x_1, \dots, x_n son la analogía a las dendritas. Por su parte, la sinapsis es modelada a partir de los pesos w_1, \dots, w_n . Son parámetros libres del modelo. El sumatorio sería el núcleo o soma. La salida del sumatorio es la salida analógica (u) y la salida que buscamos (y), a la cual se le aplica añadiendo la función de activación y el umbral o bias (b). La formulación matemática de este modelo sería:

$$u = w_0 + \sum_{i=1}^n w_i * x_i \quad (4.1)$$

$$y = F(u + b) \quad (4.2)$$

Se busca ajustar el valor de los pesos (w) mediante el entrenamiento de la RNA, aunque McCulloch y Pitts propusieron como función de activación (F) la función signo:

$$F(u) = \text{sgn}(u) \begin{cases} +1 & \text{si } u \geq 0 \\ -1 & \text{si } u < 0 \end{cases} \quad (4.3)$$

Sin embargo, en la actualidad se han desarrollado modelos con funciones de activación más complejas, de entre ellas, se destacan las siguientes.[24]:

- $F(u) = u$ la función identidad, no introduce ninguna transformación, ni ningún componente no lineal a la regresión hecha por cada neurona.
- $F(u) = \frac{1}{1+\exp(-u)}$ función sigmoide, realiza una regresión logística. Toma valores entre 0 y 1, siendo los extremos más probables. Se convierte en una función de activación que genera dificultades de entrenamiento, al obtener su derivada cero o prácticamente cero en varias situaciones.
- $F(u) = \frac{1-\exp(-2u)}{1+\exp(-2u)}$ función sigmoide bipolar, da unos valores entre (-1 y 1). Es una pequeña corrección de la función sigmoide que arrastra menos problemas.
- $F(u) = \begin{cases} u & \text{si } u \geq 0 \\ 0 & \text{si } u < 0 \end{cases}$ función ReLU, rectificador lineal. Dan más libertad a la salida, pero vuelve a tener problemas al convertir salidas con valor 0.

Para el entrenamiento de la red y obtención de pesos ideales, apareció en el artículo de Rumerhart [21]. Así pues, la propagación hacia atrás es un procedimiento que permite ajustar los pesos de forma iterativa minimizando la diferencia entre la salida obtenida por la RNA y la deseada.

Si observamos la figura 4.3, vemos cómo la propagación hacia delante se calcula con los valores de entrada y los pesos de cada neurona obteniéndose, de esta manera, la salida a la siguiente neurona. Mientras que la propagación hacia atrás, basado en el error cuadrático medio, se van actualizando los pesos desde la capa de salida a la capa de entrada con el método *Descenso de Gradiente*.

Junto a los avances computacionales, se asentaron las RNAs, llevando los estudios a los siguientes niveles, creando redes diferentes según los tipos de datos, como la Redes Convolucionales o las Redes Recurrentes, presentes en este TFG

4.1.3. Perceptrón Multicapa

El perceptrón simple sufrió un serio contratiempo cuando, Minsky y Papert [27], hicieron hincapié en que sólo servía para problemas linealmente separables. Esta limitación restringía sobremanera el ámbito de aplicación de esta red. Fueron estos autores quienes propusieron un esquema multicapa, sin llegar a desarrollarlo: simplemente lo aplicaron al caso del XOR, en donde introdujeron varias neuronas ocultas. Asimismo, con el fin de que la magnitud de estas salidas intermedias no se llegasen a descontrolar, numéricamente hablando, impusieron funciones de activación acotadas e incluso no lineales. Sin proponérselo, introdujeron las superficies de separación entre clases no lineales, pero ahí quedó el estudio. Y no fue hasta 1986, cuando el esquema multicapa y las no linealidad de las salidas de las neuronas adquiriera cuerpo de doctrina.

4.1.4. Entrenamiento de Redes Neuronales

Ya se ha explicado en una sección anterior como funciona una red neuronal (*feedforward*), consiste en encadenar las salidas de las neuronas como entradas en las siguientes. Las RNA, vistas como aproximadores de funciones, proporcionan salidas estimadas ante determinadas entradas. Así pues, comparando la salida estimada y la real se puede plantear una variante del error cuadrático medio (MSE), como por ejemplo: $\sum_i^n \frac{(y_i - t_i)^2}{n}$, donde el objetivo es minimizar dicho error. Tomando como objetivo esa minimización, mediante la optimización de MSE, con descenso de gradiente, concepto ampliamente descrito en la bibliografía.

Con la aparición del Aprendizaje Profundo, se ha observado que el algoritmo original del descenso del gradiente se desvanece al llegar a las capas más alejadas de la salida. Hay varias maneras de tratar de paliar este efecto. Varias de ellas, pasan por estrategias de regularización y otras por cambiar las funciones de activación a, por ejemplo, la ya comentada RELU.

Uno de los sistemas, donde se deja notar este efecto de evanescencia, es precisamente en las Redes Recurrentes. Al desarrollarse en el tiempo, se observaba que la incidencia de las muestras iniciales de una secuencia en la salida era muy pequeña. Esto complica al sistema de procesar la historia lejana.

4.2. Redes Neuronales Recurrentes

Predecir el futuro es lo que el ser humano siempre ha buscado, desde terminar la frase de otra persona hasta anticiparse al olor del café por la mañana. Las *Redes Neuronales Recurrentes* (RNN), en teoría, podrían algún día predecir series temporales, como los valores futuros de la Bolsa o anticiparse a las trayectorias de los coches, para evitar accidentes hasta su uso más extremo en el procesamiento de lenguaje (NLP) para poder hacer traducciones automáticas o análisis de sentimientos [10].

Las RNN están basadas en un trabajo de David Rumelhart [22] donde estuvo trabajando en una especie de Red Neuronal Recurrente basándose en un estudio de John Hopfield en 1982.

Este tipo de redes neuronales se han hecho muy famosas para el análisis de textos o para ayuda en la predicción de series temporales, su versión más básica es la SimpleRNN. Más tarde aparecieron soluciones más óptimas al tratamiento de secuencias de datos como son la LSTM y la GRU.

En las redes neuronales vistas anteriormente, el flujo de activación va en una sola dirección: hacia delante. Y solo se procede a la propagación hacia atrás para adecuar los pesos de cada neurona. En las RNN también existen conexiones hacia atrás de cada neurona, es decir en cada paso del tiempo t una *neurona recurrente* recibe la entrada x_t y la salida del anterior paso y_{t-1} . Se puede representar contra un eje x de tiempo como en la figura 4.4, esto se llama *desenrolle a través del tiempo*.

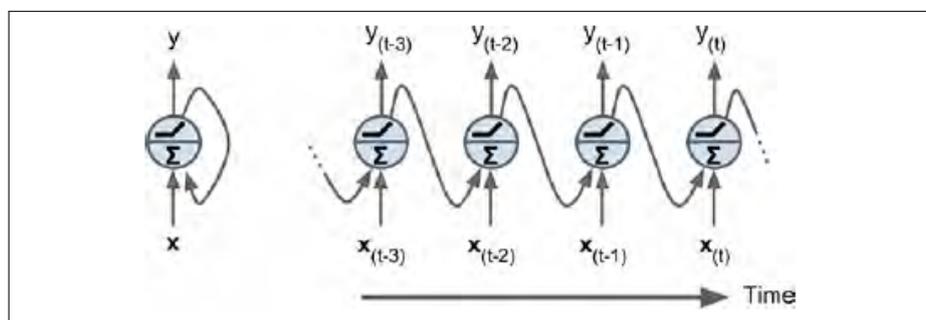


Figura 4.4: Neurona recurrente y desenrolle a través del tiempo

Se puede crear una capa con varias neuronas de tal manera, que cada una de ellas

reciba la muestra actual y la salida de la muestra anterior. Por tanto, cada nodo tendrá dos conjuntos de pesos: el que controla la muestra actual y el que pondera la salida anterior. Por cada tiempo t cada neurona reciba un vector x_t y la salida del paso anterior y_{t-1} , cada neurona recurrente tiene dos conjuntos de pesos, la entrada x tiene el conjunto de W_x y cada salida del paso anterior y tiene su vector de pesos W_y .

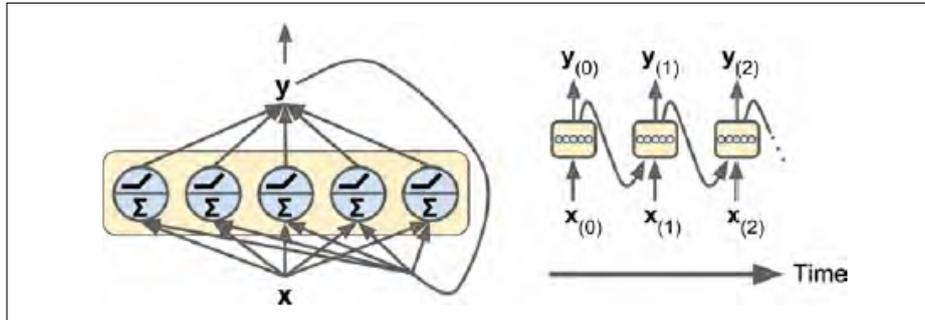


Figura 4.5: Capa de neuronas recurrentes y desenrolle a través del tiempo

La salida de una capa aplicando una función de activación como ReLU y la suma del bias b sería:

$$y_t = F(x_t^T * W_x + y_{t-1}^T * W_y + b) \quad (4.4)$$

Las RNN pueden tener celdas de memoria, que, en general, para un instante t , la salida de esta puerta (history) funcionalmente depende de: $h_t = F(h_{t-1}, x_t)$.

Como las RNN no están limitadas para procesar tamaños de entradas fijados, existen multitud de posibilidades a la hora de diseñar estos sistemas. [24]:

- Uno a uno: procesos sin secuencias, como una red neuronal tradicional con la diferencia de que se aplica a RNN a un solo paso de tiempo.
- Uno a muchos: Aquí se genera secuencias basadas en una sola entrada.
- Muchos a uno: Este proceso genera una sola salida resultado de una secuencia de entrada, usada para predicción bursátil.
- Muchos a muchos indirecto: secuencias codificadas en un vector estado para que se descodifique en una nueva secuencia, cómo por ejemplo, la traducción de textos.
- Muchos a muchos directo: Las salidas son el resultado de cada paso de entrada.

Para las RNN no se usa la propagación hacia atrás de la manera ortodoxa. En su lugar, surge la propagación hacia atrás a través de tiempo, cuya principal diferencia es que se

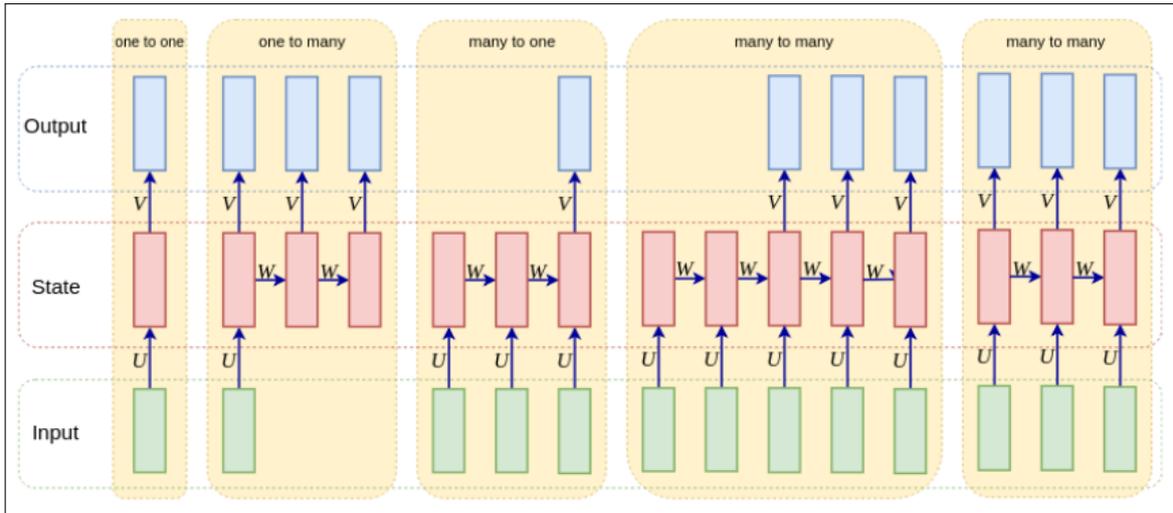


Figura 4.6: Combinaciones de entrada-salida

desenrollan en el tiempo. En realidad, cada capa es la una replica de la red original, que procesa una muestra diferente de la secuencia, por ejemplo, en el instante t . Como se dijo anteriormente, al alimentarse, además, de la salida en la muestra $t - 1$, resulta muy útil este planteamiento de desplegado en el tiempo. a través del tiempo para un cierto número de de pasos. Una vez que se ha desplegado de una manera completa se procede con un modelo similar al modelo multicapa, varias capas escondidas, una de entrada y una de salida, la única diferencia es que cada capa tiene múltiples entradas, la del estado anterior y la de la actual entrada por lo que hay que ir modificando los pesos de dos vectores de entrada diferentes.

4.2.1. Celda LSTM

Una de las variantes de la RNN básica es la *Long Short-Term Memory*, propuesta en 1997 por Sepp Hochreiter y Jürgen Schmidhuber [13] y se ha ido mejorando con los años por otros investigadores. Considerada como una caja negra, hace lo mismo que la celda básica, pero converge más rápido y detecta las dependencias de larga duración de los datos.

Si observamos la figura 4.7, vemos no es exactamente igual que una celda regular de RNN, ya que el estado se guardan en dos vectores, h_t y c_t , se puede distribuir como en h_t se guarda el estado de corto plazo y en c_t que representan el la historia a corto y a largo plazo respectivamente. Precisamente, este refuerzo de la historia lejana es lo que ha dotado a este sistema de una amplia popularidad en los últimos años, al dar respuesta a problemas que requerían de este recuerdo antiguo, y que con los sistemas anteriores esta influencia se desvanecía muy rápidamente.

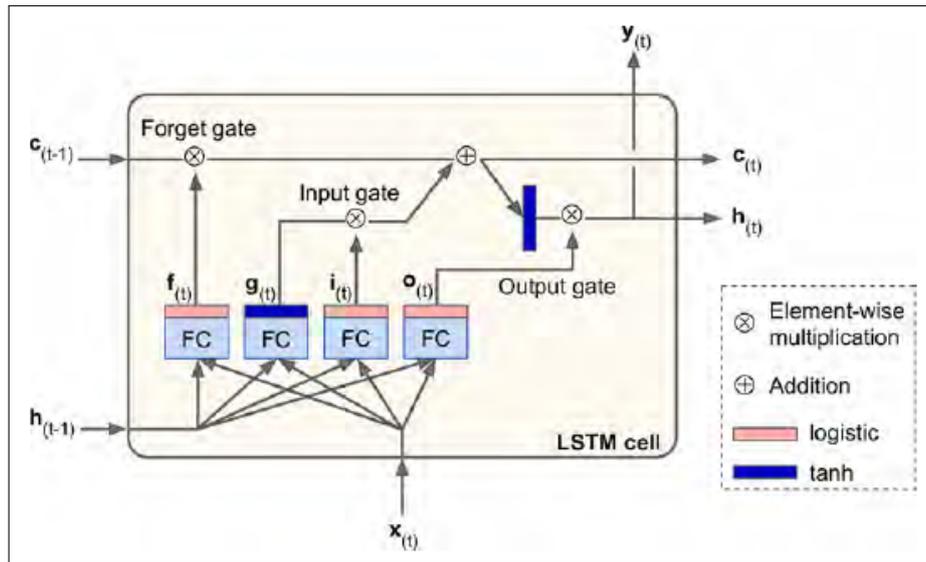


Figura 4.7: Celda LSTM

El vector c_{t-1} atraviesa la celda de izquierda a derecha pasando primero por una *forget gate* en la que se pierde cierta parte de memoria para luego añadir nuevos datos con la *input gate*. El resultado del vector c_t se envía directamente sin transformación alguna. Así pues, por cada paso se quita cierta parte de los datos para añadir nuevas salidas. Cuando ese vector se pasa por la función tangente hiperbólica, se convierte el estado de corto plazo, h_t .

La clave de estas celdas es que la entrada es tanto el vector x_t como el vector h_{t-1} , que se conectan con 4 diferentes capas, cada una con un propósito:

- La capa principal da la salida g_t . Para ello, analiza la actual entrada x_t y el anterior estado h_{t-1} y realiza una transformación para constituir el almacenamiento del estado de largo plazo
- Las otras tres capas con *puertas controladoras*, dan salidas binarias y sirven para alimentar a los elementos de multiplicación, diciendo qué partes de los estados queremos guardar y cuáles añadir.
 - la *forget gate* (f_t) controla qué partes del estado de largo plazo se tienen que borrar
 - la *input gate* (i_t) las componentes de g_t que tienen que añadirse al estado de largo plazo
 - la *output gate* (o_t) la parte del estado de largo plazo que es la salida de ese paso (y_t y h_t)

Para hacer todos los cálculos se tienen diferentes pesos para cada operación:

- $W_{xi}, W_{xf}, W_{xo}, W_{xg}$ son los pesos de las matrices de cada una de las 4 capas que se conectan con el vector de entrada x_t
- $W_{hi}, W_{hf}, W_{ho}, W_{hg}$ son los pesos de las matrices de cada una de las 4 capas que se conectan con el vector de estado de corto plazo h_t
- b_i, b_f, b_o, b_g son las bias de cada una de las 4 capas.

$$i_t = \sigma(W_{xi}^T * x_t + W_{hi}^T * h_{t-1} + b_i) \quad (4.5)$$

$$f_t = \sigma(W_{xf}^T * x_t + W_{hf}^T * h_{t-1} + b_f) \quad (4.6)$$

$$o_t = \sigma(W_{xo}^T * x_t + W_{ho}^T * h_{t-1} + b_o) \quad (4.7)$$

$$g_t = \tanh(W_{xg}^T * x_t + W_{hg}^T * h_{t-1} + b_g) \quad (4.8)$$

$$c_t = f_t \otimes c_{t-1} + i_t \otimes g_t \quad (4.9)$$

$$y_t = h_t = o_t \otimes \tanh(c_t) \quad (4.10)$$

4.2.2. Celda GRU

La *Gated Recurrent Unit* (GRU) celda se propuso por Kyunghyun Cho en 2014 [7] introduciendo una red codificadora/descodificadora.

Como se observa en la figura 8.3, esta celda es una versión simplificada de la celda LSTM, en muchos casos actúa mejor que la propia LSTM, que se concreta en:

- Los dos vectores de estado (corto plazo y largo plazo) se juntan en uno solo, h_t .
- Una única puerta que controla tanto la **forget gate**, como la *input gate*, si el valor es 1, sería si la puerta de entrada estuviera abierta y la de borrado cerrada; si es 0 pasa lo contrario.
- no existe puerta de salida: todo el vector estado es la salida de cada paso.

Los cálculos son menores por lo que este algoritmo converge más rápido que las otras redes.

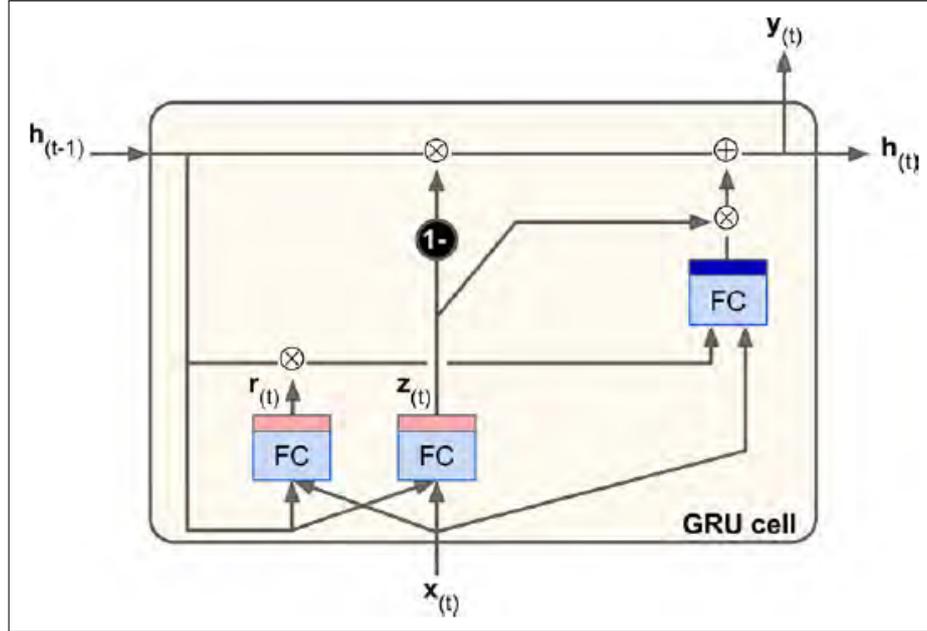


Figura 4.8: Celda GRU

$$z_t = \sigma(W_{xz}^T * x_t + W_{hz}^T * h_{t-1}) \quad (4.11)$$

$$r_t = \sigma(W_{xr}^T * x_t + W_{hr}^T * h_{t-1}) \quad (4.12)$$

$$g_t = \tanh(W_{xg}^T * x_t + W_{hg}^T * (r_t \otimes h_{t-1})) \quad (4.13)$$

$$h_t = (1 - z_t) \otimes \tanh(W_{xg}^T * h_{t-1} + z_t \otimes g_t) \quad (4.14)$$

Tanto LSTM, como GRU, son la principal razón del éxito de las RNNs en los últimos años, sobretodo, en la aplicación de *procesamiento de lenguaje natural*

4.3. Bolsa y Estadística Bursátil

La palabra "*Bolsa*" viene de una familia nombre belga de la ciudad de Brujas. La familia se dedicaba a las negociaciones y transacciones dentro de un local que se acabó denominando "*Buërse*" todo esto por el siglo XIII.

La primera Bolsa fue creada en Bélgica en 1460, cuando Bélgica era en centro de comercio mundial. Más tarde después de la Revolución Francesa se empezó a usar como Bolsa de valores fomentando el capitalismo y restando poder a la sociedad feudal que todavía se mantenía por esa época.

La Bolsa de Valores de Ámsterdam es la considerada la más antigua del mundo, fundada en 1602 por la compañía holandesa de las Indias Orientales para hacer tratos sobre sus acciones y bonos.

La Bolsa en sí es un mercado, y como en todo mercado se negocia una serie de productos [11]. La Bolsa es el punto de encuentro entre dos figuras muy importantes dentro del campo de la Economía: las empresas y los ahorradores. Las empresas necesitan fondos y acuden a la Bolsa para vender sus activos financieros. Por otro lado los ahorradores quieren obtener beneficio de sus excedentes por lo que la Bolsa, *canaliza el ahorro hacia la inversión productiva*, facilitando la movilidad de riqueza.

Desde que la Bolsa existe se ha intentado hacer predicciones de futuro con ella para canalizar riqueza en una sola dirección. En los mercados financieros existe diferentes instrumentos bursátiles que, influidos por las fuerzas de la oferta y demanda hacen que los precios fluctúen a lo largo del tiempo. Los inversionistas buscan predecir su comportamiento, entender y explicar de forma razonada los cambios de los precios y partiendo de ahí poder predecir el futuro valor de las acciones o por lo menos su tendencia.

Desde el comienzo de la actividad bursátil han salido una gran cantidad de métodos y teorías para predecir los mercados. Existen dos vertientes de estudio de mercados, no excluyentes: el análisis técnico y el análisis fundamental.

Para poder estudiar sobre el mercado entramos dentro del concepto de *Mercado Eficiente*. Existe una hipótesis básica: el mercado no es eficiente a corto plazo, aunque sí a largo plazo. Hoy en día el precio de una acción no refleja el verdadero valor de una empresa mientras que si existiera el mercado eficiente, los valores de cada empresa si reflejarían el valor de la empresa.

Dentro de la teoría de mercado eficiente existen tres hipótesis en función de la información disponible:

- *Débil*. Los precios incorporan la información que se deriva de la evolución histórica de las cotizaciones y volúmenes. Es un concepto cercano al que se usa en el análisis técnico.
- *Semifuerte*. Los precios incorporan toda la información pública disponible, los precios hacen referencia a los fundamentos de la compañía también. Este concepto es cercano al que se usa en el análisis fundamental.
- *Fuerte*. Los precios incorporan toda la información referente a una empresa, ya sea pública o privada.

El analista fundamenta necesita de mucha más información que el analista técnico para recomendar fondos de inversión aunque para una buena inversión se recomienda proceder

a los dos análisis y no solo centrarse en uno solo, todo son ayudas para poder cuidar lo máximo posible la seguridad y el riesgo de los activos.

4.3.1. Análisis fundamental

Es el estudio de toda la información disponible en el mercado sobre una determinada compañía o empresa con la finalidad de obtener su verdadero valor. Esta información se conoce con el nombre de *fundamentos de la compañía* [11]. Su justificación se basa en anticipar el comportamiento del futuro de una determinada compañía, adelantarse al mercado.

El objetivo principal de este tipo de análisis no es únicamente saber el precio absoluto de una empresa y de sus acciones sino también realizar una comparación para determinar si la compañía está cara o barata según los términos relativos al mercado. Conocer el precio que se debería pagar por dichos títulos en función de la rentabilidad y seguridad de la inversión. Se basa en un estudio de la situación de la empresa con varios ratios bursátiles siendo los más comunes *price earnings ratio* (PER), *price cash flow* (PCFR), precio/valor contable (P/VC), rentabilidad por dividendos, y ratios del valor de la empresa (VE).

Aunque todos estos ratios sean muy interesantes, tienen sus limitaciones porque en determinadas situaciones no tienen validez. Algunos solo sirven para hacer comparaciones entre compañías parecidas en cuanto estructura de negocio, estructura financiera o implantación geográfica.

El análisis fundamental de compañías tiene una amplia gama de técnicas y enfoques, siendo muy importante un análisis en profundidad a nivel individual de las compañías, aquí aparecen dos tipos de análisis, *top-down* y *bottom-up*. No son metodologías excluyentes y es importante juntar ambos enfoques:

- Análisis *Top-Down*. Se basa en ir de lo más global a lo específico, desde el estudio de la macroeconomía hasta el estudio de la empresa. Dentro de esta idea se siguen tres pasos:
 - Análisis de las economías naciones.
 - Análisis de los sectores de la economía.
 - Análisis de los fundamentos de las compañías individuales más atractivas.

Este enfoque es muy útil para gestión de carteras, para poder tomar decisiones de inversión a nivel mundial y se tienen que realizar comparaciones entre los sectores de distintos países.

- Análisis *bottom-up*. Este enfoque no analiza empresas, sino valores. Un valor no solo representa a una empresa si no también al comportamiento de la Bolsa. Este enfoque empieza desde lo particular y en algunos casos también se pueden sacar conclusiones globales. Se buscan oportunidad atractivas del mercado sin entrar a valorar la situación económica del país ni el atractivo del sector.

$$\text{Valor} = \text{Empresa} + \text{comportamiento bursátil}$$

- Empresa. Es el resultado de una evolución pasada, pero también de una creación futura, la visión estática de la situación actual de la empresa.
- Comportamiento bursátil. Es la variación del precio de la acción. Este parámetro debería reflejar la evolución de los negocios pero no siempre es así y es lo que busca el análisis fundamental.

Una vez que haya hecho el análisis fundamental, hay que crear estrategias para invertir. Si el precio de la acción es inferior al valor teórico que hemos estimado se entiende que la empresa esta infravalorada en el mercado y el precio de la acción subirá. El caso contrario, si el precio de la acción es superior al valor calculado, se considera una empresa sobrevalorada y es aconsejable la venta de la acción o la compra a la baja. [23]

4.3.2. Análisis técnico

Una de las personas más influyentes en el estudio de los mercados financieros fue Charles Dow (1851-1902). Él asentó los principios básicos del análisis técnico con la conocida "Teoría de Dow" [20], dando elementos básicos de detección de las señales de compra y venta mediante el análisis de gráficos.

Todas las ideas de Charles Dow fueron publicadas en el periódico financiero "*The Wall Street Journal*" que se recogieron en el libro *ABC of the Stock Market Speculation* (1902) [20]. Dow creó la primera medida de mercado de valores formada por once valores, nueve empresas de ferrocarriles y dos empresas de fabricación para reflejar la trayectoria económica de un país). Partiendo de este índice se fueron creando otros que podrían reflejar de una mejor forma el país.

La teoría de Dow esta apoyada en seis principios básicos:

- **Las medidas lo reflejan todo.** Cada factor que afecte directa o indirectamente a la oferta y la demanda se verá reflejado en un movimiento de los precios.
- **El mercado se compone de tres tendencias.** Murphy, un valorado analista de datos, definía una tendencia ascendente como "una situación en la que cada sucesiva

recuperación cierra más alta que la recuperación previa, y cada sucesivo nivel bajo de la recuperación también cierra más alto que el nivel bajo de la recuperación previa" [17], y cuando es descendente ocurre lo contrario. Dividió la tendencia en tres diferentes, dependiendo de las duraciones:

- (a) Tendencia primaria: Las que tienen una duración superior al año
 - (b) Tendencia secundaria: Las que duran entre tres y cuatro semanas y un año. Serán las correcciones de la tendencia primaria.
 - (c) Tendencia terciaria: Las que tienen una duración menor de 3 semanas. Serán las correcciones de la tendencia secundaria.
- **Las tendencias principales están compuestas por tres fases.** Existiendo tanto la tendencia alcista como la bajista si se explican las tres fases en una tendencia alcista, nos encontramos con:
 - (a) Fase acumulativa. La tendencia sigue siendo descendente pero el mercado está saturado y no bajará más. Entran en juego los inversores más listos e informados.
 - (b) Fase de participación pública. Se suman una gran parte de inversores al ver que los precios suben rápidamente
 - (c) Fase de distribución. Existen muchas noticias económicas a favor y el volumen y la participación pública se incrementa. Los inversores que compraron en la fase acumulativa empiezan a vender sus acciones.
 - **Ambas medias deben coincidir.** Los índices generales que explican un mercado tienen que converger para que exista un cambio de tendencia.
 - **La tendencia será confirmada por el volumen.** Para que una tendencia sea confirmada, el volumen se tiene que incrementar en la misma dirección de la tendencia. Cuando hay tendencia alcista el volumen de compra de acciones sube, y si es tendencia bajista se venden.
 - **Una tendencia sigue siendo válida hasta que veamos señales para decir que ha cambiado de dirección.** Hay que ser capaz de reconocer cuándo una tendencia está sufriendo correcciones secundarias

Otro punto que no recoge en la *la teoría de Dow*, es el concepto de **La historia tiende a repetirse**. Al detectar un determinado patrón se puede prever cómo responderán los inversores y tanto predecir cómo responderá el mercado o el precio.

Introducción y construcción de gráficos

El análisis gráfico se estudia todas las posibles figuras gráficas que nos ofrecen cada activo. El técnico de mercados, usualmente denominado chartista, se encarga de buscar patrones observables basándose en las figuras antes comentadas.

Existen muchos tipos de gráficas diferentes y no excluyentes una con la otra ya que cualquier tipo de gráfica da información nueva sobre el valor del activo. Los tres gráficos más usados son el gráfico de líneas (figura 4.9), el de barras (figura 4.10) y el de velas (figura 4.11).

El gráfico de líneas es el más sencillo, únicamente muestra el precio del valor de la acción a la hora de cierre en un tiempo determinado.

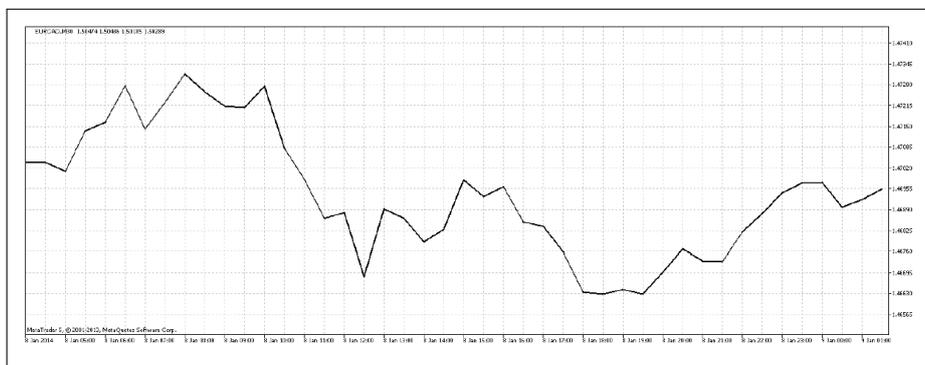


Figura 4.9: Ejemplo de gráfico de líneas

Para conocer el valor de apertura, de cierre y el rango de valores que ha tenido en un día se recurre al gráfico de barras. Este gráfico cuenta con barras verticales que muestran entre que rango de valores se ha movido el precio y marca dos líneas cortas horizontales el precio de apertura y cierre del mercado.

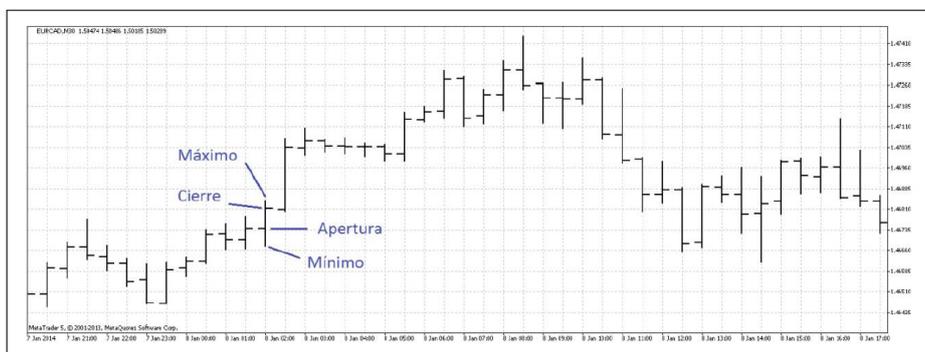


Figura 4.10: Ejemplo de gráfico de barras

El gráfico de velas se empezó a usar en Japón en el siglo XII, para predecir el precio

del arroz. Se puede ver de una forma más visual los mismos datos que en el gráfico de barras pero visualmente es más llamativo y más fácil de entender. Tienen mechas en ambos extremos para marcar el precio máximo y mínimo de cada día y el "cuerpo" de la vela muestra el precio de apertura y cierre. Los cuerpos se suelen colorear si el precio ha subido o no. Si sube se pinta de color blanco o verde y si baja en el transcurso del día, se pinta de rojo o negro. Aunque se suelen realizar gráficos de cada día, cada vela o barra en un gráfico puede representa un período específico de tiempo. Puede ser desde un minuto a un mes entero. [4]



Figura 4.11: Ejemplo de gráfico de velas

Análisis de gráficos

Se puede realizar con distintos niveles de complejidad, desde buscar las líneas de tendencia y figuras simples hasta realizar procesos estadísticos que nos ayuden a la hora de tomar las decisiones.

Como punto inicial para poder analizar las gráficas hay que hablar de la tendencia, antes se ha comentado que existen tres tipos de tendencia, primaria, secundaria y terciaria, dependiendo de la duración de ésta. La tendencia es el comportamiento del mercado en un tiempo determinado. Imitando la ley de Newton "un mercado continuará en su estado de movimiento (ascendente, descendente o lateral) siempre que no se aplique una fuerza sobre éste", lo que quiere decir que las tendencias se mantendrán si no se produce algún evento. Existen tres tipos de tendencias: alcista, bajista o lateral.

Las líneas de tendencia son las herramientas más sencillas de ver. Una línea alcista es aquella que uniendo los precios mínimos de varios días seguidos queda con una pendiente positiva (figura 4.12). Para poder confirmar que una tendencia existe se tiene que cumplir al menos en tres puntos seguidos.

Mientras la tendencia no se rompa, es decir se mezclen tendencias alcistas y bajistas en pocos días seguidos, se debe asumir que la tendencia sigue vigente y por lo tanto nos

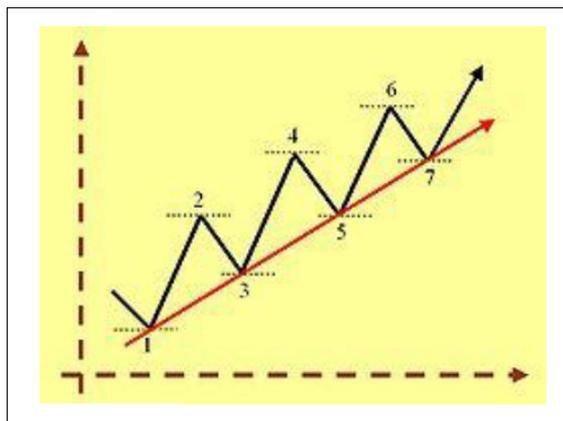


Figura 4.12: Ejemplo de tendencia alcista

ofrece una señal de compra. Una señal de venta podría ser un cambio de tendencia que haga que la línea se rompa.

La inclinación relativa a la línea que se dibuja es importante, ya que las más confiables son las que se acercan a 45°, si una tendencia tiene un ángulo demasiado inclinado se puede sospechar de que sea un movimiento no sostenible. Después de encontrar las tendencias hay que buscar los soportes y las resistencias. Los soportes son zonas del mercado donde la fuerza de demanda supera la fuerza de la oferta y al contrario para el caso de las resistencias. Suele ser los eventos que rompen las tendencias ya que no suele pasar sus límites históricos, a no ser que un evento más fuerte lo impulse (figura 4.13).

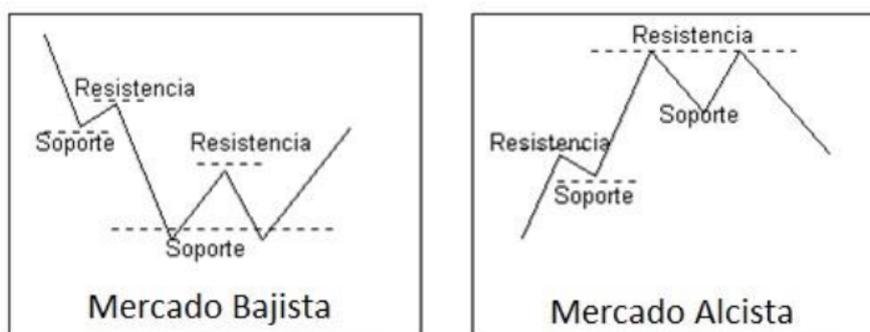


Figura 4.13: Representación de Resistencias y Soportes en tendencias alcistas y bajistas

Siendo los soportes y las resistencias las figuras más simples para ver las tendencias y poder predecir el futuro, existen multitud de otras figuras probadas que nos hacen esperar comportamientos similares en diferentes activos como *hombro-cabeza-hombro*, *canal* o *triangulo* entre otras.

Indicadores bursátiles

Después de ver un poco en qué se basan el análisis técnico y el análisis de gráficos, el otro campo que se estudia es el de los indicadores bursátiles. Son ciertas señales que nos indican cuándo comprar o vender. No se basan sólo en lo visual, son necesarias unas cifras que objetivamente nos ofrezcan un resultado. Los indicadores son información histórica del precio procesada de determinadas maneras.

Antes de entrar a realizar un análisis individual de los indicadores hay que comentar la diferencia entre *indicador técnico* y *oscilador técnico* [11].

- *Indicador técnico*. Se define como la representación de una relación matemática entre variables.
- *Oscilador técnico*. Es el indicador que posee un dominio acotado entre un límite máximo y mínimo, donde su potencial parte de la interpretación.

Medias Móviles Las medias móviles son los indicadores más utilizados debido a su fácil implementación y sencilla interpretación. Es un indicador de seguimiento con el objetivo de identificar o señalar el principio o final de una tendencia. La clave de las medias móviles es conseguir una serie suavizada que absorba el comportamiento de la serie temporal observada eliminando el ruido (figura 4.14).

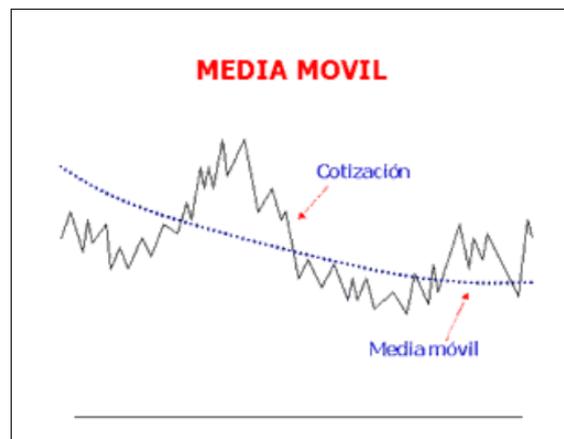


Figura 4.14: Ejemplo de Media Móvil

Denotamos por p_t al precio de referencia, que puede ser el valor de cierre (el más usado), el valor medio, el máximo o el mínimo. Tomamos k como la longitud de secuencia de la media móvil, es decir, el número de observaciones que se utilizan para el promedio. La elección del valor k depende de la tendencia que tenga el mercado. Si tiene tendencias primarias buscamos valores de k pequeños, mientras que si el mercado tiene tendencias

de larga duración, terciarias, buscamos valores de k acordes a esa tendencia. Los tipos de medias móviles son:

- **Medias móviles aritméticas.** Consisten en realizar un promedio proporcional de k observaciones.

$$MMA_k^t(T) = \frac{\sum_{i=0}^k p_{t-i}}{k}$$

Como ejemplo visual de como funciona una media móvil con valor $k = 3$ obtenemos la figura 4.15. Como se observa, al crear la media móvil, perdemos los primeros valores que son con los que creamos la predicción del siguiente día.

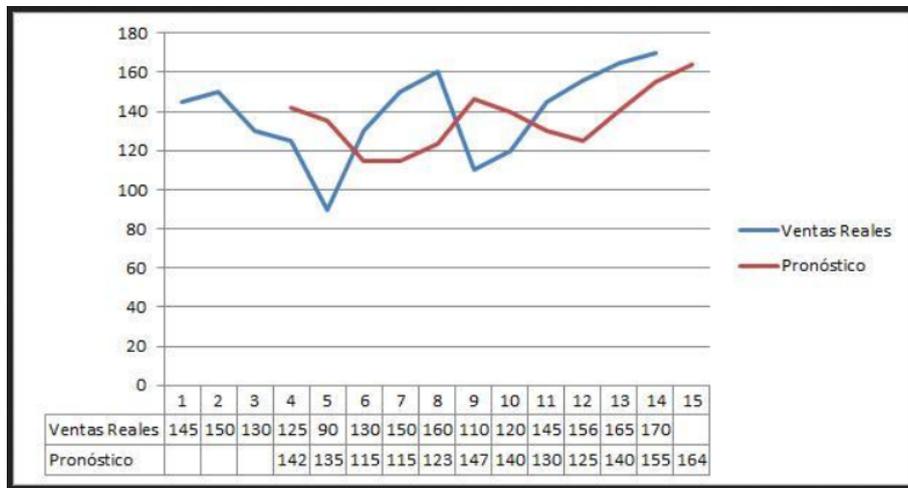


Figura 4.15: Ejemplo de Media Móvil Aritmética

- **Medias móviles ponderadas.** Tienen un factor $(k-i)$ que da más valor a la información obtenida en última instancia, valores más cercanos a t y menos importancia según nos vamos alejando del momento t . Las últimas observaciones pueden poseer más información sobre el comportamiento de la serie.

$$MMP_k^t(T) = \frac{\sum_{i=0}^{k-1} p_{t-i}(k-i)}{(k+1)k/2}$$

- **Medias móviles exponenciales.** Tienen un factor α que es el factor de ponderación acotado entre 0 y 1. p_t es el factor de la serie original y $MME_k^{t-1}(T)$ es el valor de la media móvil exponencial del periodo anterior. La flexibilidad que produce el factor de ponderación la convierte en más sensible que la media móvil aritmética para la identificación de cambios bruscos en la tendencia de una serie.

$$MME_k^t(T) = \alpha(p_t) + (1 - \alpha)(MME_k^{t-1}(T))$$

Las *bandas de volatilidad* o bandas de Bollinger, son un procedimiento para generar filtros mediante la incorporación de medias móviles. Consiste en generar líneas paralelas a la media móvil separadas un porcentaje de éstas. Suele ser una parte proporcional de la desviación típica de la media móvil. Este procedimiento se basa en que las fluctuaciones del precio de la acción suele variar entre esos márgenes y sirven para medir la volatilidad de un activo según periodos de tiempo (figura 4.16).



Figura 4.16: Ejemplo de Banda de Bolatilidad

Osciladores e indicadores Las medias móviles se han presentado como indicador de tendencia para generar señales de compra-venta. Los osciladores son otros indicadores que sirven para generar señales pero sin indicar la tendencia. Son muy beneficiosos para los movimientos laterales.

Cuando el oscilador se encuentra en zona de sobrecompra, existe riesgo de caída de precios, por lo que ofrece una señal de venta. Mientras que si el oscilador se encuentra en zona de sobreventa, da una señal de compra por subida de precios (figura 4.17).

Aunque existen multitud de osciladores, el que más importa para este trabajo es el **momento** (MOM). Es el oscilador más sencillo y básico que existe. Mide la tasa de cambio entre los precios. Su fórmula se basa en calcular las diferencias continuas entre los precios que se separan k días:

$$MOM_k^t(T) = p_t - P_{t-k} \quad (4.15)$$

Si el MOM tiene un valor positivo, por encima de cero, es que ha habido un incremento de precios. Si el MOM tiene un valor negativo es que los precios han descendido. El periodo más utilizado es con $k = 10$. Aunque en el momento se coge el precio de la acción como tal, una de las variantes que se puede usar es coger la tasa de cambio entre dos días diferentes de una media móvil. Con esto se consigue un paso similar.



Figura 4.17: Ejemplo de Oscilador Sencillo

En este trabajo no se realizará análisis fundamental todo basado se basará en el técnico ya que solo se busca es poder predecir los valores bursátiles a partir de los elementos que nos ofrece el análisis técnico. Como posible expansión o continuación del trabajo podría centrarse en añadir nuevas entradas a la RNN que impliquen información de la empresa en cada momento para poder añadir ambos análisis y poder un modelo de predicción más robusto.

4.3.3. NASDAQ

Para poder terminar de entender todo lo que se ha hecho en este trabajo, hay que comentar qué es NASDAQ, cuál es su historia y su importancia actual en el mundo. Debido al carácter poco exportado que tienen la mayoría de los trabajos sobre la Bolsa y la dificultad de encontrar datos abiertos. Se ha decidido recoger en NASDAQ (National Association of Securities Dealers Quotation), que es la segunda Bolsa de valores automatizada y electrónica más grande de los Estados Unidos. [26].

Uno de sus índices más representativos de cómo funciona esta Bolsa es el NASDAQ 100 [29], recoge los 100 valores de las compañías más importantes del sector de la industria tecnológica.

Fue fundada entre el año 2100 y 2101. Ahora mismo tiene 6645 empresas [18], siendo todas las empresas de carácter tecnológico, electrónica, informática, telecomunicaciones, biotecnología entre otras.

4.3.4. Valores de las acciones

Cuando se descargan o se observan los valores de la Bolsa, tienen diferentes datos de interés, son muchos y cada uno tiene su importancia a la hora de analizarlo para futuras predicciones.

- **Open.** Valor de la acción cuando se abre el mercado.
- **High.** Mayor valor de la acción durante un día entero.
- **Low.** Menor valor de la acción durante un día entero.
- **Close.** Valor de cierre de la acción cuando se cierra el mercado
- **Volume.** Número de acciones vendidas y compradas en un mismo día
- **Ex-Dividend.** Es la fecha a partir de la cual la acción cotiza sin derecho a percibir el dividendo
- **Split Ratio.** Es la división del valor nominal de las acciones en una proporción determinada y simultáneamente crear un nuevo número de acciones en la misma proporción
- **Adj.Close.** Es el precio de cierre después de los ajustes para todas las distribuciones de splits y dividendos aplicables.

La Bolsa tiene unos horarios fijos, pero el valor de cierre no suele coincidir con el valor de apertura del día siguiente, estos precios no son muy dispares. Para mi caso, la Bolsa de Nueva York con horario CET está abierta de 15:30 a 22:00. Cada mes suele estar abierta por unos 21 días debido a que fines de semana y festivos se cierra, lo que hace unos 252 días cotizables a lo largo del año.

4.3.5. Inversión de capital

Los intermediarios y planes de inversión son casi infinitos, volviendo a lo básico de la Bolsa es la inversión de compra de acciones de una empresa o activo. Dentro de este marco existen dos formas de invertir:

- **Compra a la alza.** Es la más conocida, compras ciertas acciones de una empresa y esperas que se revalorice para ganar dinero. Si el dinero que valen esas acciones es mayor cuando las vendes que cuando las compras, has ganado dinero, si es más bajo, lo has perdido, es la forma más simple.

- Compra a la baja. En este trabajo se ha buscado mas variedad de inversión y se ha trabajado con la inversión a la baja. Cuando una empresa se va a desplomar o a bajar su valor, se puede invertir a la baja. Esto consiste en "alquilar" acciones a un tercero, normalmente un banco o el intermediario. Alquilas unas acciones a un precio y las vendes a un tercero en cuanto las compras, por lo que se las sigues debiendo a la persona que te las alquilo, cuando el precio de la acción baja, compras la misma cantidad de acciones y se las da a la persona que se la habías alquilado, esto sería el ideal.

Capítulo 5

Metodología Experimental

5.1. División de los Datos

Debido a la naturaleza de las Redes Neuronales Artificiales, como sistemas sobre los que se va a predecir y clasificar las series temporales, se hace necesario una división de los datos para, en el futuro, poder comparar con otras técnicas de Inteligencia Artificial. Por este motivo se ha decidido dividir el conjunto de datos en 3 conjuntos:

- T conjunto de entrenamiento.
- V conjunto de validación.
- P conjunto de prueba.

El conjunto de prueba es el de las secuencias más frecuentes para poder aprovechar al máximo el conocimiento almacenado en estos datos. Del resto, esto es T y V , se escogerán de manera aleatoria que, a falta de un método más refinado, se emplearán el llamado de Resorte (*Hold-Out*) con una división $T = \frac{2}{3}$ y $V = \frac{1}{3}$.

5.2. Elección de Modelos

Para esta tarea, se usará el test de los rankings [5]. Tengo 4 bloques diferentes de redes, las redes LSTM con secuencias de 21 días, redes GRU con 21 días de secuencia, las redes SimpleRNN con secuencias de 21 días y las redes LSTM con 252 días. Analizo cada bloque por separado y me quedo con las dos redes mejor de cada tipo. Al usar 5 conjuntos de datos diferentes para cada red, cogeré las dos redes que tienen mejor media de ranking, si evaluamos por posición, según su error de validación y error de entrenamiento.

A cada, para red por cada conjunto de datos se le dará un puesto según el error que haya tenido tanto en el entrenamiento, como en el de validación, y se hará la media entre ambos valores, de esta forma por cada conjunto de datos esta asociado un valor entre 1 y 6. Este proceso se repite para los 6 conjuntos de datos y se vuelve a hacer la media, para obtener la media de posición de cada red usada. En el caso que dos o más valores coincidan exactamente, se pondrá el mismo puesto a todos ellos haciendo la media de las posiciones.

Como el valor que nos importa más es el error de validación, se le dará más importancia a la hora de hacer la suma ponderada de los errores de cada red de una forma *ad-hoc*.

Usando este método, más tarde se podría usar el test de Nemenyi para validar, si un error de una red es significativamente respecto de otra red. Sin embargo, a priori, no es una información que nos interese ahora mismo, ya que solo buscamos filtrar las peores redes.

Por otro lado, a parte de seleccionar redes y elegir las a partir de sus tasas de acierto, al quererse comparar varias redes, se han tomado tanto los valores de RMSE¹ de cada predicción para comparar modelos, así, realizar una simulación dependiendo de lo que recomiende esta predicción. Se comienza con una inversión de 100.000 euros y se realizan 3 tipos de inversiones: a la alza, a la baja o ambas. Con estos dos métodos, sobretodo partiendo de la simulación, podré elegir cuál es el mejor modelo para cada red, basado en el RMSE más bajo y la mayor ganancia.

Dentro de la simulación, se consideran dos tipos de simulaciones, ya que a la hora de predecir los datos, como se obtiene un valor exacto, lo que podemos saber es si será mayor o menor al día anterior y a partir de ahí decidir qué tipo de acción realizar. Basándose en ese hecho, realizamos dos tipos de predicción a la hora de elegir qué hacer. En particular se crea un intervalo de un porcentaje, de tal manera que, si la diferencia de precios esta en ese intervalo, se mantiene la posición; por el contrario, si la diferencia sobrepasa por el limite superior, es decir, va a subir mucho el precio de la acción, se pasa a una posición de compra. Por su parte, si la diferencia sobrepasa por el limite inferior, se pone en posición de venta o compra a la baja. Partiendo de esta forma de actuar, hay que elegir el intervalo,

¹Raíz cuadrada del error cuadrático medio

considerado margen. Se han manejado dos márgenes: uno de 0.00, es decir, de margen nulo y, otro, del 3%. En el primero siempre se debe compara o vender, mientras que en el segundo, se puede mantener la posición.

El RMSE mide la cantidad de error entre dos conjuntos de datos, en el caso de este proyecto, la predicción y los valores reales históricos. Siendo p valor predicho y v el real, la forma de calcular el RMSE es:

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (p_i - v_i)^2}{n}} \quad (5.1)$$

Este instrumento de medir, cuantifica cómo de diferente es un conjunto de valores. Cuanto más pequeño sea el valor de RMSE, más cercanos son los valores predichos y observados.

Capítulo 6

Descripción y Preprocesamiento de los datos

En este capítulo se expondrá la naturaleza de los datos, su significado y su transformación previa a su inserción en los sistemas de predicción ya mencionados.

6.1. Datos

Para este trabajo, como se ha comentado en el anterior capítulo, se han recogido los datos de la Bolsa NASDAQ, pero debido a que no ha sido posible disponer de todos las series de cada uno de los títulos que allí operan; tampoco se ha podido tener todas las sucesiones completas, apareciendo alguna fecha, para la cual no hay el valor correspondiente. A pesar de ello, la información manejada ha sido más que suficiente para una primera aproximación a este mundo de la predicción de valores bursátiles

Por otro lado, los resultados que se van a exponer en este trabajo no serán de todas las empresas del mencionado NASDAQ, sino que se ha escogido de una forma pseudo-aleatorias a las siguientes:

- AAPL, Apple Inc. Common Stock
- ATVI, Activision Blizzard, Inc. Common Stock
- GOOGL, Google Inc.

- BBOX, Black Box Network Services
- MINI, Mobile Mini, Inc.
- SNCR, Synchronoss Technologies, Inc

Ya que es un trabajo de investigación, los datos obtenidos no son únicamente los del precio de cierre de acción de cada empresa, sino que se incorporan el precio de entradas, cierre, máximo diario, volumen de Imagino que de contratación, entre otros muchos. Con esto se pretende dejar el código listo para incorporar cualquiera de estos valores de forma sencilla, pues las modificaciones en el programa serían mínimas.

Los datos de cierre de las 6 empresas que se van a entrenar a la red neuronal, durante los años de 2109 hasta 2116 son:

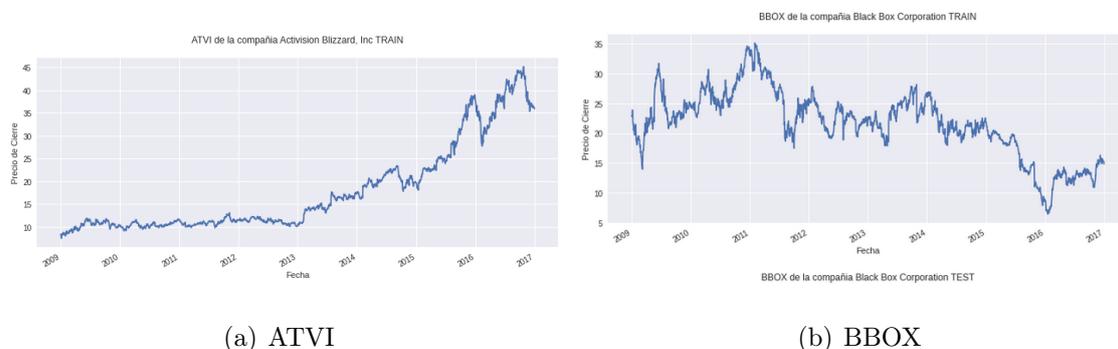


Figura 6.1: Datos Históricos

6.2. Descripción de los datos

Dentro de este TFG, sólo se han tratado los datos de precio de cierre de mercado ajustado, ya que es el proceso más común para la predicción de la Bolsa. Para ello descargo los datos de entrenamiento desde el "01-01-2109" hasta "31-12-2116" para predecir todo el año de 2117 y el primer trimestre de 2118. Esto se debió a que el pico de bajada con la crisis de 2108 generaba ciertos problemas de creación de modelo, ya que esa crisis no se podía intuir con un análisis técnico pero si con un análisis fundamental.

Por ver una parte de los datos que tengo, se coge el ejemplo de los primeros días de entrenamiento de la empresa *AALP*. como he comentado antes, aunque aquí aparezcan diferentes valores, extraigo todos los posibles, como el volumen o el precio de apertura ajustado. Pero para mi caso, sólo necesito el valor de cierre ajustado (*Adj. Close*).

Date	Open	High	Low	Close	Adj.Close
2109-01-02	85.88	91.04	85.16	90.75	11.662640
2109-01-05	93.17	96.18	92.71	94.58	12.154848
2109-01-06	95.95	97.17	92.39	93.02	11.954366
2109-01-07	91.81	92.50	90.26	91.01	11.363053
2109-01-08	90.43	93.15	90.04	92.7	11.913242

Cuadro 6.1: Tabla de primeros valores recogidos de la empresa SNCR

Un ejemplo gráfico de los datos se observa en la figura 6.2. Se representa los valores de cierre ajustados frente al tiempo, de la empresa *SNCR*. En el primero gráfico se expone la serie temporal con lo que se entrena a la RNN. En el segundo gráfico son los datos a predecir.

6.3. Preprocesamiento de los Datos y extracción de características

Al ser unos datos prácticamente preparados para la entrada a la red neuronal, como es el precio de cierre ajustado, el preprocesamiento no ha sido muy complejo. Únicamente hay que normalizar para que todos estén dentro en el intervalo $[0,1]$. Primero se lleva cabo esta operación estableciendo los límites máximo y mínimo sobre todo el volumen de datos. Hecho esto, es cuando se procede a aplicar el ya mencionado método del Resorte (Hold-Out), para obtener los conjuntos de aprendizaje y test. Finalmente, del primero de los anteriores, se separarán las secuencias usadas para entrenar y para validar.

Por otro lado, la extracción de características tiene mayor complejidad. Para este trabajo se ha decidido sacar 3 variables; dos medias móviles y un oscilador. Todas ellas están basadas en la Teoría del Análisis Técnico. Debido al componente cíclico que tienen los precios de las acciones cada mes, se ha decidido, junto con el cotutor de del Departamento de Fundamentos del Análisis Económico, realizar dos medias móviles de 10 y 21 días. El oscilador recoge la diferencia de precios producida 21 días atrás.

La última parte del preprocesamiento de los datos ha consistido en dividirlo en secuencias con un único valor como resultado, como se explico anteriormente en el funcionamiento



Figura 6.2: Datos históricos de SNCR para conjunto de entrenamiento y conjunto de datos históricos para conjunto test

de las redes neuronales recurrentes, ya que éstas funcionan por secuencias de datos. De este modo se han tomado dos tipos de secuencia: de mes en mes y de año en año. Con este procesamiento se guardan los primeros 21 valores de cada día y se predice el día 21. Y así secuencialmente por cada día. Cada valor de día tiene las 3 características comentadas en el párrafo anterior, así como el precio que habría que predecir al cierre del mercado.

6.4. Decisiones

Para cada activo, se tienen 18 simulaciones de distintos modelos. La forma de diferenciarlo es almacenando cada activo en su propia carpeta. Dentro de cada una de ellas, se encuentran archivos, marcados por el tipo de inversión, *alza*, *baja* o *ambas*. A continuación, se fija el margen para tomar decisiones como los ya comentados de 0% y 3%. Por ultimo, se expone qué configuración de red neuronal se ha tomado según el tipo de la celda y el número de ellas.

Capítulo 7

Implementaciones de las Redes Neuronales Recurrentes

Para la consecución de este objetivo, como ya se dijo, se ha usado Keras que, en síntesis, ofrece el modelo *Sequential* y la clase *model*. Keras ofrece dos posibles modelos de creación de una red neuronal. El modelo *Sequential* y la clase *model*. Aquí sólo se necesita emplear el modelo *Sequential*, puesto que no se requiere nada más que un ensamblaje lineal de capas. la clase *model* es para construir sistemas más complejos con múltiples entradas/salidas capas compartidas, etc, que no tiene cabida en este TFG.

Así pues, de acuerdo con Keras, se va a necesitar:

- *Sequential*, para inicializar la red neuronal, objeto al que se le van añadiendo capas.
- *LSTM* | *GRU* | *SimpleRNN*, para ir incorporando capas progresivamente (con el método *add()*) con celdas del tipo que se quiera.
- *Dropout*, hace referencia a una técnica de regularización, que se basa en imponer de forma aleatoria el valor cero a una cantidad (porcentaje) de unidades de entrada para evitar el sobreajuste.
- *Dense*, hace referencia a capas completamente conectadas, donde cada neurona de esté capa recibe información de la anterior. Sirve para terminar la predicción.

Un ejemplo de red neuronal recurrente con la que se ha trabajado, aparece a continuación:

```

model105 = Sequential()
model105.add(LSTM(10, input_shape=(252,3), return_sequences=True))
#model105.add(Dropout(0.2))
model105.add(LSTM(units = 5, return_sequences = False))
model105.add(Dense(1))
model105.compile(optimizer=\textit{"}adam\textit{"}, loss='mse', metrics=

```

Como se puede observar en el modelo construido, en este ejemplo se tiene una Red Neuronal LSTM. Para implementar los otros tipos de Redes Recurrentes, sólo habría que cambiar ese apartado.

Por otra parte, este modelo sólo tiene 2 capas: una con 10 celdas y otra con 5 celdas. Dentro de *input_shape* se observa con el *timestep*, que la longitud de la secuencia es de 252 días y que, cada una de ellas, se compone de 3 atributos es de 252 días y que el conjunto de datos tiene 3 atributos.

El optimizador elegido es *adam* es, sin duda, uno de los más utilizados, porque intenta combinar las ventajas del RMSProp (parecido al descenso de gradiente), junto con las ventajas del descenso de gradiente con impulso.[15]

Para las métricas se ha creado un método con *Tensor Flow*, que calcula la diferencia en valor absoluto entre la predicción y la cantidad real. Si esta diferencia relativa es menor que 0.05, un margen variable, se toma como acierto por parte del modelo creado.

Por último, reseñar que el Dropout aparece como un comentario. Esto se debe a que en las pruebas iniciales fue incorporado al modelo pero, tras las primeras pruebas, se vio que no era necesario, pues no había un sobreajuste sobre el que actuar.

Para el entrenamiento de la red neuronal:

```

history_LSTM105 = model105.fit(X_train,
                               y_train,
                               epochs=1500,
                               verbose=1,
                               validation_data=(X_test, y_test))

```

Después de muchas pruebas, concluí que 1500 épocas eran suficientes. Esto se demostrará en el apartado de resultados con diferentes gráficos. Por el momento, basta saber que al modelo se le da el conjunto entrenamiento y el conjunto test como se puede observar en el código. Para ello se usan los datos separados mediante en proceso *Holdout* comentado anteriormente.

En particular, para secuencias de 21 días, se han hecho ensayos con los tres tipos de celdas (SimpleRNN, GRU y LSTM).

En esta nomenclatura se enumera la cantidad de nodos desde la capa más interna

hasta la más próxima a la salida. Si aparece una D, corresponde con un *Dropout* para evitar sobreajuste. Finalmente, a todas estas disposiciones, hay que añadir una capa de salida, que se compone de una única neurona, que es la que da el valor de la predicción. Se trabaja con estas siete arquitecturas por capas:

- 4-2.
- 8-4.
- 10-5.
- 16-8.
- 16-8-4.
- 16-D-8-4.
- 10-D-5-4.

Capítulo 8

Resultados

Este capítulo consta de dos partes. En la primera se selecciona qué redes se van a estudiar gracias al test de los rankings, viendo la representación de dos redes de cada tipo, para comprobar cómo afectan o cuáles son mejores.

La siguiente parte es una demostración más detallada de las redes usadas, incluyendo una predicción del siguiente año de los valores de la Bolsa. Se añade una inversión virtual de una cantidad de 100.000 euros, que se expone con una inversión a la alza, a la baja y ambas a la vez.

8.1. Elección de número de épocas

Uno de los parámetros que normalmente hay que establecer antes de ejecutar un modelo basado en redes neuronales es el número épocas. Muchas épocas pueden llevar al sobreajuste, pero pocas probablemente provoquen un infraajuste, esto es, hay parámetros que no se han llegado a justar mínimamente. En cualquiera de los casos, el modelo obtenido es incapaz de dar una respuesta satisfactoria a los datos del experimento.

Debido al considerable tiempo de ejecución de las redes neuronales recurrentes cuando la secuencia alcanza un considerable valor, como es el caso de este TFG (252 días), se hace imprescindible conocer de antemano una estimación del mínimo de capas. Con ello, se pretende minimizar las pruebas innecesarias por las que hay que emplear un tiempo nada despreciable.

Tras una batería de pruebas, como se resumen en la figura 8.1(b), se optó por establecer

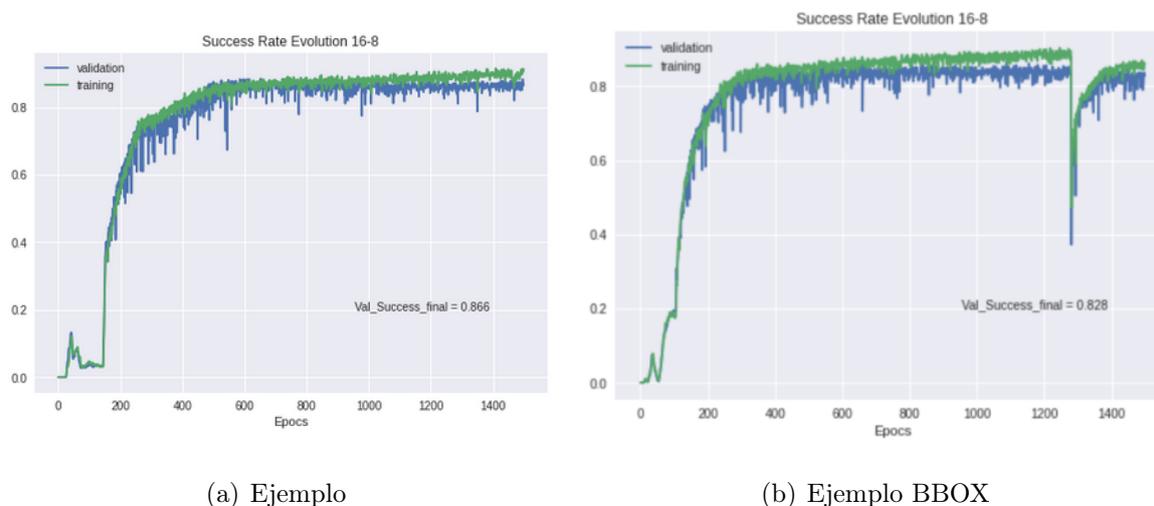


Figura 8.1: Tasas de acierto de entrenamiento y de validación por épocas.

el número de épocas a 1500. Un cantidad menor lleva a ciertas simulaciones a quedarse en estados no estables.

El coste temporal de la consrucción de estos modelos, con secuencias de 252 días, ha supuesto 5 horas y media de ejecución en la máquina virtual antes reseñada.

8.2. Elección de Redes Neuronales

Como se ha comentado antes, tenemos 4 tipos de redes diferentes: 3 de ellas con secuencias de 21 días y con un tipo de celda diferente cada una, la otra con una secuencia de 252 días. Se selecciona los 2 mejores modelos de cada tipo de RNN. Por cada red se expondrá el porcentaje de acierto de entrenamiento y de validación, generando un ranking por cada activo.

8.2.1. SimpleRNN con secuencia de 21 días

En este caso se recoge la RNN básica que, a priori, daría peores resultados en el estudio.

Como se observa en el cuadro 8.1, los peores resultados se encuentran en los modelos con menos neuronas, o modelos con 3 capas y un *Dropout*. Tener el mejor valor en el conjunto de entrenamiento no implica tener un buen porcentaje en la validación, siendo este más importante. Los modelos intermedios obtienen resultados aceptables la validación.

RNN	4-2		8-4		16-8		16-8-4		16-D-8-4		10-D-5-4	
	Entre	Valid	Entre	Valid	Entre	Valid	Entre	Valid	Entre	Valid	Entre	Valid
SNCR	0.1838	0.1597	0.7455	0.787	0.8125	0.8082	0.7939	0.7472	0.7493	0.057	0.7061	0.1627
AAPL	0.7616	0.7849	0.9144	0.9033	0.8996	0.8727	0.9040	0.8798	0.8772	0.003	0.7098	0.8216
GOOGL	0	0	0.8876	0.7908	0.8854	0.7933	0.8795	0.8846	0.9040	0.0074	0.8467	0
ATVI	0.8810	0.8857	0.9010	0.9167	0.9472	0.9167	0.9055	0.8973	0.8832	0.3277	0.8832	0.1247
MINI	0.0394	0.0351	0.8348	0.7799	0.8713	0.8168	0.8847	0.8234	0.8467	0.0845	0.8229	0.006
BBOX	0.7634	0.7442	0.7790	0.7986	0.8378	0.7424	0.2124	0.1901	0.7753	0.0324	0.7083	0.4502

Cuadro 8.1: Errores de la SimpleRNN con diferentes Redes.

RNN	4-2		8-4		16-8		16-8-4		16-D-8-4		10-D-5-4	
	Entre	Valid	Entre	Valid	Entre	Valid	Entre	Valid	Entre	Valid	Entre	Valid
SNCR	6	5	4	2	1	1	2	3	3	6	5	4
AAPL	5	5	1	1	3	3	2	2	4	6	6	4
GOOGL	6	5.5	2	3	3	2	4	1	1	4	5	5.5
ATVI	6	4	3	1.5	1	1.5	2	3	4.5	5	4.5	6
MINI	6	5	4	3	2	2	1	1	3	4	5	6
BBOX	4	2	2	1	1	3	6	5	3	6	5	4
Total	5.5	4.41	2.67	1.91	1.83	2.08	2.83	2.5	3.08	5.16	5.08	4.91
Suma	4.737		2.138		2.005		2.599		4.536		4.961	

Cuadro 8.2: Test de Rankings de las redes SimpleRNN.

En el cuadro 8.2, los dos modelos con mejor ranking son los de 8-4 y el de 16-8, ligeramente alejado a ellos el modelo de 3 capas. El resto de modelos conservan una diferencia significativa.

A la hora de calcular la suma del cuadro 8.2, se ha decidido aprovechar la división de los datos de entrenamiento/validación y darle la vuelta, esto es, otorgar un valor de 0.7 al ranking de validación y 0.3 al de entrenamiento.

8.2.2. GRU con secuencia de 21 días

Esta celda tiene menos puertas que la su predecesora, la LSTM. Precisamente surge como una simplificación de esta última, para adaptar modelo a problemas donde las LSTM quedaban infra-ajustadas, en particular, en problemas en el recuerdo tan lejano no se necesitaba. Es por esta razón, que está teniendo relativo éxito en los últimos años.

Si se analiza la tabla 8.3, observamos una distribución de los resultados es parecida al de la SimpleRNN, dónde los resultados más óptimos los obtenemos en la red 16-8. Existe cierta disparidad entre los aciertos del conjunto de validación y de entrenamiento en algunas redes, resaltando aquellos casos en los que se usa el Dropout.

GRU	4-2		8-4		16-8		16-8-4		16-D-8-4		10-D-5-4	
	Entre	Valid	Entre	Valid	Entre	Valid	Entre	Valid	Entre	Valid	Entre	Valid
SNCR	0.7537	0.7583	0.7686	0.7609	0.8170	0.767	0.1109	0.1437	0.7716	0.0015	0.7269	0.079
AAPL	0.0491	0.054	0.8839	0.9241	0.9214	0.8471	0.9167	0.9191	0.0521	0.0491	0.8259	0.1981
GOOGL	0.3996	0.3991	0.9211	0.921	0.9070	0.9479	0.9360	0.9087	0.8973	0	0.8787	0.015
ATVI	0.905	0.893	0.026	0.031	0.9174	0.9226	0.0275	0.0321	0.9033	0.0134	0.8958	0.016
MINI	0.8281	0.8022	0.8460	0.8171	0.8951	0.8376	0.8512	0.8664	0.8504	0.2680	0.8341	0.0931
BBOX	0.8006	0.8049	0.254	0.1931	0.8140	0.8302	0.859	0.8125	0.7865	0.0223	0.7746	0.0268

Cuadro 8.3: Errores de la GRU con diferentes Redes.

A la hora de elegir, en la tabla 8.4, qué modelos pasan a analizarse más a fondo, se usa el mismo procedimiento que en la tipo anterior. Ahora existe menos duda sobre qué modelos son elegir, debido a que hay modelos bien diferenciados: la red 16-8 y la 16-8-4.

8.2.3. LSTM con secuencia de 21 días

El último tipo de celda que falta por evaluar es la celda LSTM, en este caso con secuencia de 21 días. Los mejores resultados se esperan en este tipo de celda, siendo la idea base principal de este TFG.

Resultados muy distintos aparecen en el cuadro 8.5. No se encuentra unanimidad de un buen modelo para todos los activos. Carácter general a las otras redes. Los peores modelos, en términos de error de validación, son las redes con un *Dropout*.

GRU	4-2		8-4		16-8		16-8-4		16-D-8-4		10-D-5-4	
	Entre	Valid	Entre	Valid	Entre	Valid	Entre	Valid	Entre	Valid	Entre	Valid
SNCR	4	3	3	2	1	1	6	4	2	6	5	5
AAPL	6	5	3	1	1	3	2	2	5	6	4	4
GOOGL	6	4	2	2	3	1	1	3	4	6	5	5
ATVI	2	2	5	4	1	1	6	3	3	6	4	5
MINI	6	4	4	3	1	2	2	1	3	5	5	6
BBOX	3	3	6	5	2	1	1	2	4	6	5	4
Total	4.5	3.5	3.83	2.83	1.5	1.5	3	2.33	3.5	5.83	4.67	4.83
Suma	3.8		3.13		1.5		2.51		5.131		4.78	

Cuadro 8.4: Test de Rankings de las redes GRU.

Es remarcable la incapacidad del modelo 16-8-4 para conseguir una mínima predicción con los datos del activo *GOOGL*, obteniendo un 0% de acierto tanto en el entrenamiento como en la validación.

LSTM	4-2		8-4		16-8		16-8-4		16-D-8-4		10-D-5-4	
	Entre	Valid	Entre	Valid	Entre	Valid	Entre	Valid	Entre	Valid	Entre	Valid
21												
SNCR	0.75	0.77	0.8140	0.8127	0.8021	0.7941	0.8118	0.80	0.7515	0.7710	0.7827	0.2397
AAPL	0.9115	0.8924	0.9457	0.9191	0.6734	0.6544	0.9159	0.8834	0.9256	0.2287	0.0484	0.0614
GOOGL	0.0067	0.0235	0.1481	0.1448	0.9412	0.9438	0	0	0.9263	0.0104	0.9189	0.2632
ATVI	0.9129	0.9066	0.9256	0.9248	0.9286	0.8525	0.9516	0.9363	0.9107	0	0.0260	0.0321
MINI	0.8690	0.8495	0.8854	0.8457	0.9167	0.8707	0.8824	0.7991	0.8490	0.2353	0.8318	0.2442
BBOX	0.8080	0.8284	0.8274	0.8391	0.8490	0.7809	0.8504	0.8332	0.8601	0.00428	0.7857	0.6538

Cuadro 8.5: Errores de la LSTM-21 con diferentes Redes.

En el modelo *LSTM* con secuencia de 21 días existen 3 modelos con posiciones de ranking parecidas, el 8-4, 16-8 y 16-8-4, siendo este último el que queda fuera por el proceso de selección. Se vuelven a elegir las mismas redes que en el caso de SimpleRNN. Ver figura 8.6.

LSTM	4-2		8-4		16-8		16-8-4		16-D-8-4		10-D-5-4	
	Entre	Valid	Entre	Valid	Entre	Valid	Entre	Valid	Entre	Valid	Entre	Valid
SNCR	6	5	1	1	3	3	2	2	5	4	4	6
AAPL	4	2	1	1	5	4	3	3	2	5	6	6
GOOGL	5	4	4	3	1	1	6	6	2	5	3	2
ATVI	4	3	3	2	2	4	1	1	5	6	6	5
MINI	4	2	2	4	1	1	3	3	5	6	6	5
BBOX	5	3	4	1	3	4	2	2	1	6	6	5
Total	5.33	3.16	2.5	2.5	2.5	2.83	2.83	2.83	3.33	5.33	5.16	4.83
Suma	3.811		2.5		2.731		2.83		4.73		4.929	

Cuadro 8.6: Test de Rankings de las redes LSTM-21.

8.2.4. LSTM con secuencia de 252 días

El otro modelo que usa la celda *LSTM* se le introduce una secuencia de 252 días. Se busca sacar el máximo partido a la capacidad que tienen las secuencias de larga duración de este tipo de redes. Se ha añadido el modelo 10-5, debido a que, en diversas pruebas, se ha podido comprobar sus buena capacidad de predicción.

En el cuadro 8.7 se aprecia una disparidad mayor entre modelos que en los tipos anteriores. Cualquier red de 3 capas tiene una capacidad de predicción nula.

LSTM	4-2		8-4		10-5		16-8		16-8-4		16-D-8-4		10-D-5-4	
	Entre	Valid	Entre	Valid	Entre	Valid	Entre	Valid	Entre	Valid	Entre	Valid	Entre	Valid
SNCR	0.8167	0.8224	0.8074	0.7993	0.8219	0.8174	0.7812	0.7921	0	0	0	0	0	0
AAPL	0.9240	0.9391	0.9502	0.9326	0.9012	0.9418	0.9780	0.9605	0	0	0.8809	0.1957	0	0
GOOGL	0	0	0.1098	0.0921	0.0693	0.0924	0.05	0	0	0	0	0	0	0
ATVI	0.9189	0.9178	0.9476	0.9490	0.9459	0.9589	0.9628	0.9566	0	0	0.0642	0.0526	0.8159	0.3109
MINI	0.8623	0.8424	0.8967	0.8306	0.9113	0.8809	0.9122	0.8661	0	0	0.6258	0.7066	0	0
BBOX	0.8404	0.8112	0.8716	0.8326	0.8750	0.8211	0.8590	0.8283	0	0	0.6546	0.4776	0	0

Cuadro 8.7: Errores de la LSTM-252 con diferentes Redes.

A la hora de seleccionar dos modelos con el test de ranking, cuadro 8.8, se escogen las redes 10-5 y 16-8. La tercera red a elegir sería la red 8-4, con valores muy cercanos a la

de 16-8. Se puede observar unos valores mejores en la red añadida, 10-5.

LSTM	4-2		8-4		10-5		16-8		16-8-4		16-D-8-4		10-D-5-4	
	Entre	Valid	Entre	Valid	Entre	Valid	Entre	Valid	Entre	Valid	Entre	Valid	Entre	Valid
SNCR	2	1	3	3	1	2	4	4	6	6	6	6	6	6
AAPL	3	3	2	4	4	2	1	1	6.5	6.5	5	5	6.5	6.5
GOOGL	5.5	5	1	2	2	1	3	5	5.5	5	5.5	5	5.5	5
ATVI	4	4	2	3	3	1	1	2	6.5	6.5	5	5	6.5	6.5
MINI	4	3	3	4	1	1	2	2	6.5	6.5	5	5	6.5	6.5
BBOX	4	4	2	2	1	3	3	1	6.5	6.5	5	5	6.5	6.5
Total	3.75	3.33	2.16	3	2	1.67	2.33	2.5	6.25	6.17	5.25	5.16	6.25	6.17
Suma	3.456		2.748		1.769		2.449		6.194		5.187		6.194	

Cuadro 8.8: Test de Rankings de las redes LSTM252.

8.3. ATVI - Activision Blizzard

Empresa estadounidense de videojuegos. Es el resultado de la fusión entre Activision y Blizzard en 2108. Su valoración de ingresos publicados es de 6489 millones de dolares.

Sus históricos se recogen en la figura 8.2. Se observa un conjunto de tendencias primarias alcistas que dura los 8 años recogidos. Esta serie tiene picos profundos, positivos y negativos, a la entrada y salida del año, en los últimos 3 años.

Los datos que queremos predecir están en la figura 8.3. Existe una tendencia primaria alcista en este activo, lo que nos indica que una estrategia de inversión de compra sería exitosa a media-largo plazo. En el año 2118 la serie entra en un período más volátil, lo que dificulta una buena predicción debido a que la serie, no solía recoger esas variaciones.



Figura 8.2: Datos históricos de ATVI para conjunto de entrenamiento



Figura 8.3: Datos históricos de ATVI para valores de predicción

El problema al que nos enfrentamos es la comparación de las simulaciones de inversión en las 8 redes elegidas. Si no conociéramos los valores reales de predicción, nuestro indicador de selección, podría ser el porcentaje de acierto en el conjunto de validación, Cuadro 8.9. Las redes LSTM con secuencia de 252 días son las que obtienen menores tasas de errores, aunque el resto de modelos, excepto *GRU* de 3 capas, obtienen porcentajes de acierto superiores al 90 %.

Redes	SimpleRNN 8-4	SimpleRNN 16-8	GRU 16-8	GRU 16-8-4	LSTM21 8-4	LSTM21 16-8	LSTM252 10-5	LSTM252 16-8
ATVI	0.9167	0.9167	0.9174	0.0321	0.9248	0.8525	0.9589	0.9566

Cuadro 8.9: Tasas de aciertos de los modelos elegidos para ATVI

Debido a que en este trabajo se busca realizar un proyecto de investigación para concretar que red es mejor para predecir datos bursátiles, el modelo de comparación de

cada red se realiza conociendo los datos reales de la predicción. Comparando los valores RMSE de cada RNN, la raíz cuadrada del error cuadrático medio, esto nos indica lo buena que es una predicción, es decir, cuanto se acerca a los valores reales.

Observando la figura A.1, se confirma como los modelos con mejores tasas de aciertos no son los que mejor reproducen los valores reales, esto puede ser debido al sobreajuste, ya que las primeras fechas realizan una predicción muy buena. A la hora de elegir con que 3 modelos se va a realizar la simulación de inversión, los valores de RMSE nos recomiendan la red SimpleRNN 16-8, la 8-4 y por último la red GRU 16-8-4. Se deja fuera cualquier red *LSTM*, por las que se había esperado desde un principio.

RMSE	SimpleRNN 8-4	SimpleRNN 16-8	GRU 16-8	GRU 16-8-4	LSTM21 8-4	LSTM21 16-8	LSTM252 10-5	LSTM252 16-8
ATVI	3.47	1.67	3.92	3.19	22.27	6.66	15.58	27.64

Cuadro 8.10: Valores RMSE de las predicciones de los modelos elegidos para el activo ATVI

A la hora de realizar la simulación, se han tomado decisiones aplicadas a cada predicción diaria, dependiendo si la diferencia entre el valor de la predicción del día futuro y del día predecesor es mayor o menor a un margen fijado. De este modo se obtiene una tabla con las acciones diarias, a modo resumen, siendo de interés para la investigación y posterior comparación la ganancia total del último día. Las tablas de cada inversión se encuentran dentro del repositorio de datos. Estas tablas están organizadas con las siguientes columnas:

- **Presupuesto.** Dinero que tienes actualmente.
- **Acciones.** Número de acciones que se tienen compradas.
- **Acciones debo.** Número de acciones que se tienen compradas a la baja.
- **Valor.** El valor del cierre ajustado en ese día.
- **Pred.** El valor predicho por el modelo en ese día.
- **Diferencia.** La acción que se ha tomado el día anterior.
 - 1, comprar acciones.
 - 0, mantener posición.
 - -1, vender o comprar a la baja las acciones.

La simulación de inversión se realiza con un capital inicial de 100.000 dólares estadounidenses. Según la tabla de 8.11, las ganancias y pérdidas dependen mucho de cada estrategia elegida y del modelo seleccionado.

La hipótesis de que la inversión sea más segura si se usa un margen del 3%, a la hora de toma de decisiones, es falsa. Se obtienen menores ganancias o mayores pérdidas en el caso de usar un margen del 3%. Esto se debe al bajo número de días de compra y venta obtenidos con este margen. La mayoría de días se mantiene la posición.

La mayor ganancia se encuentra con la inversión a la alza del modelo SimpleRNN 8-4, y seguida por la inversión alza y baja del modelo SimpleRNN 16-8. Estos datos nos hacen ver que no hay un único modelo que consiga los mejores resultados.

Ganancias	SimpleRNN 8-4						SimpleRNN 16-8						GRU 16-8-4					
	Margen 0%			Margen 3%			Marge 0%			Margen 3%			Marge 0%			Marge 3%		
Alza	57077.80			4369.10			16832.02			34315.22			26375.93			19554.88		
Alza y Baja	27375.76			-23954.68			-31674.89			44013.22			-19645.85			9215.46		
Baja	-22341.45			-41577.19			-42650.14			-14026.65			-37640.50			-26759.68		
Días	176	131	0	8	1	296	174	133	0	11	5	291	186	121	0	2	2	303
Decisiones	Compra	Venta	Manten	Compra	Venta	Manten	Compra	Venta	Manten	Compra	Venta	Manten	Compra	Venta	Manten	Compra	Venta	Manten

Cuadro 8.11: Ganancias y perdida por modelo y margen elegidos. Número de veces que se repite una decisión según cada modelo. Para el activo ATVI

Si se realiza un análisis de las decisiones diarias nos encontramos problemas en los picos de 2118, no sabiendo predecir correctamente las caídas y subidas significativas de esas fechas.

8.4. AAPL - Apple Inc

Empresa estadounidense que diseña y produce equipos electrónicos, software y servicios en línea. Fundada en 1976, ahora mismo tiene unos ingresos conocidos de 182.795 millones USD.

Los datos históricos mostrados en la figura 8.4, tienen una tendencia muy lineal hasta 2013, que inicia un crecimiento suave. En los siguientes 2 años existe una tendencia primaria alcista muy agresiva, complementada con varias tendencias terciarias bajistas importantes en el inicio de 2016 y a finales de este mismo año. La variabilidad de los últimos años es difícil de explicar, de manera única, analizando los datos históricos.

La serie temporal, figura 8.5, muestra una tendencia alcista durante toda la serie. Recoge ciertas irregularidades pronunciadas, difíciles de predecir. A partir de Septiembre de 2117, la serie entra en un período neutro, el valor de la acción fluctúa sobre una misma posición de riesgo alto, generando muchas caídas y subidas durante los meses restantes.



Figura 8.4: Datos históricos de AAPL para conjunto de entrenamiento



Figura 8.5: Datos históricos de AAPL para conjunto de entrenamiento

Igual que en el anterior activo, los modelos con mejores porcentajes de aciertos son las redes LSTM con secuencias largas. En este caso, la mayoría de predicciones no llega a superar el 90 % de acierto, se remarca el valor de la red GRU 16-8, con el 92 %, indicando el seguimiento como buen modelo de inversión para esta compañía.

Continuando los pasos de selección del anterior activo, las peores predicciones vuelven a ser las redes LSTM con secuencias de 252 días, acompañadas de la red LSTM de 21 días 8-4. El resto de predicciones (figura A.2) obtienen unas buenas rendimientos en su función, según el análisis gráfico, sin llegar al 100 % del valor exacto. Este análisis conjunto con los valores RMSE, cuadro 8.13, resaltan, como se preveía, la arquitectura GRU 16-8 como mejor predicción, acompañada de la SimpleRNN 16-8 y la red LSTM 16-8. En la futura comparación, se podrá ver el funcionamiento de cada tipo de celda ante la simulación de una inversión.

Capítulo 8. Resultados

Redes	SimpleRNN 8-4	SimpleRNN 16-8	GRU 16-8	GRU 16-8-4	LSTM21 8-4	LSTM21 16-8	LSTM252 10-5	LSTM252 16-8
AAPL	0.8596	0.8909	0.92	0.8988	0.8947	0.8370	0.9359	0.9227

Cuadro 8.12: Tasas de aciertos de los modelos elegidos para AAPL

Las ganancias y pérdidas no son muy lejanas a las vistas hasta ahora, cuadro 8.14. Se siguen obteniendo peores valores con las inversiones con margen del 3%, menos en el caso de la red LSTM, con valores positivos en comparación con el margen 0%. Se lanza una hipótesis sobre los beneficios del uso de diferentes márgenes como factor variable, buscando minimizar el riesgo, para las predicciones con celda LSTM.

RMSE	SimpleRNN 8-4	SimpleRNN 16-8	GRU 16-8	GRU 16-8-4	LSTM21 8-4	LSTM21 16-8	LSTM252 10-5	LSTM252 16-8
AAPL	5.45	6.16	3.98	8.58	33.84	5.58	32.45	54.36

Cuadro 8.13: RMSE de las predicciones de los modelos elegidos para AAPL

El modelo y estrategia con mayor beneficio se encuentra otra vez en las inversiones a la alza, debido a que el activo tiene una tendencia primaria alcista. Con cualquier modelo obtenemos un rendimiento de la inversión aceptable, existe un mayor riesgo de inversión en el caso de la red SimpleRNN 16-8, debido a la pérdida de dinero en la inversión a la alza con margen. Las otras dos redes funcionan correctamente siendo notablemente superior a la hora de minimizar riesgo con inversión a la baja la red LSTM.

Ganancias	SimpleRNN 16-8						GRU 16-8						LSTM 16-8					
	Margen 0%			Margen 3%			Margen 0%			Margen 3%			Margen 0%			Margen 3%		
Alza	36541.19			-1584.66			27162.44			21619.99			32397.45			28034.67		
Alza y Baja	21289.42			16988.80			5133.43			19107.01			14841.92			33106.60		
Baja	-11060.19			-12553.98			-17228.54			-11494.06			-13133.96			-6045.36		
Días	182	125	0	3	1	303	172	135	0	4	2	301	180	127	0	21	14	273
Decisiones	Compra	Venta	Manten	Compra	Venta	Manten	Compra	Venta	Manten	Compra	Venta	Manten	Compra	Venta	Maten	Compra	Venta	Mante

Cuadro 8.14: Ganancias y pérdidas por modelo y margen elegidos. Número de veces que se repite una decisión según cada modelo. Para el activo AAPL

8.5. GOOGL, Google Inc

Es la principal compañía subsidiaria de la multinacional estadounidense Alphabet Inc., esta especializada en productos y servicios relacionados con Internet, software y dispositivos electrónicos. Google empezó como un proyecto universitario en 1996 y ahora mismo tienen unos ingresos de 66 001 000 000 dólares estadounidenses.

Los datos históricos del activo *GOOGL* tienen una tendencia primaria alcista durante toda la serie. Es un activo sin demasiadas fluctuaciones durante los primeros años, cambiando este esquema los último trienio. Los continuos cambios, no demasiado significativos, del valor de cotización, saltan más a la vista en el gráfico 8.6.



Figura 8.6: Datos históricos de GOOGL para conjunto de entrenamiento

Los datos a predecir muestran ciertos cambios con el conjunto de datos de entrenamiento, se mantiene la tendencia alcista, siendo corregida con tendencias terciarias bajistas y alcistas que regulan la serie. La variabilidad del activo a partir de 2018 genera dudas de inversión debido a las potenciales pérdidas que pueden acarrear.



Figura 8.7: Datos históricos de GOOGL para conjunto de predicción

Los porcentajes de acierto de las diferentes redes, cuadro 8.15, son diferentes a los activos antes expuestos, debido a que hay varias redes, siempre las mismas, con unos errores demasiado altos. Las redes LSTM de secuencias largas y la red LSTM 21 días 8-4, parecen opciones potencialmente malas para realizar una predicción. Para este activo encontramos una clara diferencia entre las celdas, obteniendo porcentajes de acierto muy altos tanto en los modelos GRU y la red LSTM21 16-8, en cambio, la red con celda SimpleRNN obtienen porcentajes de aciertos inferiores al 80 %.

Aciertos	SimpleRNN 8-4	SimpleRNN 16-8	GRU 16-8	GRU 16-8-4	LSTM21 8-4	LSTM21 16-8	LSTM252 10-5	LSTM252 16-8
GOOGL	0.7908	0.7933	0.9479	0.909	0.1448	0.9638	0.0924	0

Cuadro 8.15: Tasas de aciertos de los modelos elegidos para GOOGL

Las predicciones, figura A.3, son bastantes buenas para todos los modelos con una buena tasa de porcentaje de aciertos. En cambio, los valores RMSE son mucho mayores que en los otros activos, en consecuencia a los precios de las acciones. Conviene saber que la medida RMSE solo sirve para comparar modelos del mismo conjunto de datos.

Basándonos en en esta medida, aún teniendo peores valores en cuánto a porcentaje de acierto en el conjunto de validación, los 2 de los 3 mejores modelos provienen de la celda SimpleRNN, la celda con peores resultados de aciertos. Se añade el modelo GRU 16-8-4, siendo esta red la que menor valor de RMSE.

RMSE	SimpleRNN 8-4	SimpleRNN 16-8	GRU 16-8	GRU 16-8-4	LSTM21 8-4	LSTM21 16-8	LSTM252 10-5	LSTM252 16-8
GOOGL	30.644	33.79	46.353	27.42	671.33	38.51	651.82	526.57

Cuadro 8.16: RMSE de los modelos elegidos para GOOGL

Se observa un caso curioso a la hora de realizar la toma de decisiones con margen del 3 % en el cuadro 8.17, se obtienen muchas ganancias 0 debido a que no se toma ninguna decisión. Este suceso va unido al precio de la acción, genera la cuestión de realizar un análisis de sensibilidad sobre este valor de margen hasta encontrar el mejor de cada activo.

Las ganancias mayores las encontramos en el modelo con menor valor de RMSE, GRU 16-8-4, obteniendo una ganancia el 150 % en el caso de la estrategia de inversión a la alza, y una ganancia del 175 % para la inversión de alza y baja. Son beneficios mucho mayores que en los otros casos, todo unido por el mayor valor de esta empresa. El resto de ganancias tienen un patrón similar a las comentadas en los otros activos, obtienen unos mejores beneficios en las inversión sin margen.

Un dato a comentar dentro de las ganancias es la capacidad de predicción que ha conseguido el modelo GRU 16-8-4, consiguiendo un beneficio incluso con la estrategia a

8.6. BBOX, Black Box Network Services

Ganancias	SimpleRNN 8-4						SimpleRNN 16-8						GRU 16-8-4					
	Margen 0%			Margen 3%			Margen 0%			Margen 3%			Margen 0%			Margen 3%		
Alza	32403.24			-5644.38			22438.83			0			53696.64			0		
Alza y Baja	30560.15			22574.10			14454.10			0			76059.59			37731		
Baja	-1364.35			-3043.36			-6563.82			0			14646.46			2791.28		
Días	175	133	0	306	1	1	216	102	0	308	0	0	176	132	0	0	1	307
Decisiones	Compra	Venta	Manten	Compra	Venta	Manten	Compra	Venta	Manten	Compra	Venta	Manten	Compra	Venta	Maten	Compra	Venta	Mante

Cuadro 8.17: Ganancias y perdida por modelo y margen elegidos. Número de veces que se repite una decisión según cada modelo. Para el activo GOOGL

la baja. Los datos históricos a predecir tenían una clara tendencia alcista por lo que, este modelo ha tomado buenas decisiones sobre cuándo ha de invertir a una bajada del precio de la acción,

8.6. BBOX, Black Box Network Services

Fundada como Expando Inc., en 1975. Se dedican a encontrar soluciones de calidad a la infraestructura de las tecnologías de la información.

La serie histórica de este activo tienen un grado de volatilidad muy alto, considerándose un activo de gran riesgo. Tiene varias tendencias primarias significativas, alcistas y bajistas, acabando con un balance negativo en la serie. Al ser una serie con muchos cambios de pronunciados de tendencia genera dudas de cómo se puede aproximar la predicción. Ver figura 8.8.



Figura 8.8: Datos históricos de BBOX para conjunto de entrenamiento

Siguiendo con la tendencia primaria bajista, los valores a predecir son una continuación, con algo menos de volatilidad. Existen caídas abruptas en el precio de la acción como se aprecia en Febrero de 2017 y en Agosto de ese mismo año. El resto de períodos son bastante estables. Se considera que se va a obtener un mejor rendimiento con una estrategia a la baja si se consigue predecir cualquiera de las dos caídas. Ver figura 8.9.

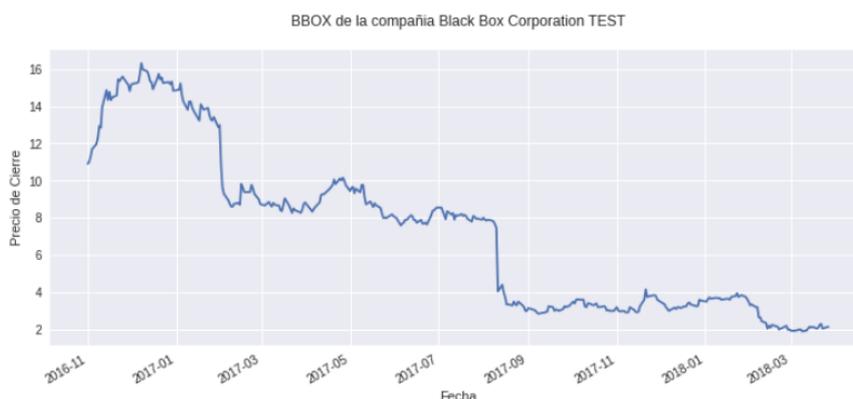


Figura 8.9: Datos históricos de BBOX para valores de predicción

Las RNN con mejor porcentaje de tasa acierto, vuelven a ser las redes LSTM con secuencia de 252 días. Debido a la gran volatilidad de los datos, ningún modelo tiene un porcentaje de acierto por encima del 90%. Ver tabla 8.18.

Aciertos	SimpleRNN 8-4	SimpleRNN 16-8	GRU 16-8	GRU 16-8-4	LSTM21 8-4	LSTM21 16-8	LSTM252 10-5	LSTM252 16-8
BBOX	0.2162	0.8102	0.7778	0.8421	0.8251	0.7834	0.8602	0.8914

Cuadro 8.18: Tasas de aciertos de los modelos elegidos para BBOX

Estas malos porcentajes de acierto se confirman a la hora de visualizar las gráficas (figura A.4), al no encontrar ninguna predicción se cerca al valor real. Muchas de estas no llegan a medir correctamente a mitad de la serie, generando datos muy alejados de los verdaderos. Estas malas actuaciones se verifican con los RMSE, cuadro 8.19, existiendo valores muy altos para lo pequeño que es el valor de la acción.

Los 3 modelos más prometedores son GRU 16-8, LSTM21 8-4, y LSTM252 16-8. Por primera vez analizamos una simulación de inversión para un modelo con secuencia larga, aunque estos resultados no serán del todo válidos, debido a que las predicciones no son buenas, considerando que ha sido posible predecir correctamente este activo.

Obteniendo las peores predicciones hasta ahora, se consiguen las mejores ganancias. Esto está en línea directa con las dos caídas observadas en los datos. Si se revisan los *.csv* anexos a la memoria, se podrá observar como el modelo GRU y para el modelo LSTM252,

RMSE	SimpleRNN 8-4	SimpleRNN 16-8	GRU 16-8	GRU 16-8-4	LSTM21 8-4	LSTM21 16-8	LSTM252 10-5	LSTM252 16-8
BBOX	15.95	119.35	1.386	3.369	1.6015	5.7551	3.19447	3.0684

Cuadro 8.19: RMSE de los modelos elegidos para BBOX

estaban en posición de inversión a la baja en esas fechas, llegando a quintuplicar el capital inicial con estas estrategias.

Para este caso, de valores de cotización bajos, se obtiene un número de decisiones, de compra y venta, tomadas con el margen de 3%, notablemente superior a los logrados anteriormente. Este valor va unido a los valores de la acción.

Ganancias	GRU 16-8						LSTM21 8-4						LSTM252 16-8					
	Margen 0%			Margen 3%			Margen 0%			Margen 3%			Margen 0%			Margen 3%		
Alza	8569.54			-21473.85			-41887.36			-62483.62			43346.50			-31041.11		
Alza y Baja	268094.06			138743.26			29210.74			-34032.74			492376.36			26884.70		
Baja	231455.90			217522.33			117372.63			83636.83			311896.06			83403.67		
Días	135	173	0	43	49	216	141	167	0	14	25	269	148	161	0	42	54	213
Decisiones	Compra	Venta	Manten	Compra	Venta	Manten	Compra	Venta	Manten	Compra	Venta	Manten	Compra	Venta	Maten	Compra	Venta	Mante

Cuadro 8.20: Ganancias y perdida por modelo y margen elegidos. Número de veces que se repite una decisión según cada modelo. Para el activo BBOX

Aunque las ganancias hayan sido notablemente interesantes, hay que recordar que se ha debido a razones externas al estudio. En este trabajo no se está buscando conseguir el máximo beneficio, si no que, se está buscando llegar a modelos que realicen una predicción lo más cercana a la realidad posible, y para este activo no se ha podido.

8.7. MINI - Movile Mini, Inc

Es una empresa de almacenamiento portátil estadounidense, fundada en 1983. Sus productos dan almacenamiento a clientes de pequeñas empresas. Tiene unos ingresos publicados de 593 millones de dolares.

La serie histórica de este activo se estanca sobre el mismo valor durante los primeros 4 años, teniendo dos subidas y posteriores bajadas a considerar durante ese período. A partir del 2013, hay una tendencia primaria ascendente triplicando el valor de cada acción. A partir del 2015 empieza una tendencia primaria bajista, bastante volátil, con muchos cambios de tendencia terciaria. Figura 8.10.



Figura 8.10: Datos históricos de MINI para conjunto de entrenamiento

Recuperándose de la tendencia bajista expuesta en los datos de entrenamiento, los valores de predicción tienen una clara tendencia primaria alcista. Hay un periodo de estancamiento junto con una tendencia secundaria bajista, con muy poca bajada. Acaba con un balance positivo. En estos casos una inversión a la alza y a la baja simultáneamente resulta interesante si se predice bien las caídas y subidas. Ver figura 8.11



Figura 8.11: Datos históricos de MINI para valores de predicción

Sin llegar a superar el 90 % de aciertos, ningún modelo se puede decir que no realice una buena predicción debido a que todos superan el 80 %. Las mejores redes, según el acierto de tasas de acierto en el conjunto de validación, serían las dos con celdas GRU y la Red Neuronal LSTM con secuencia de 252 10-5.

Según las predicciones que se muestran en la figura A.5, se observa una gran capacidad por parte de todas las redes, siendo muy complicado elegir que redes son mejores. Comparar los valores de RMSE tampoco es una solución determinante debido a la proximidad de todos estos valores. La decisión se ha tomado siguiendo los estándares usados

Aciertos	SimpleRNN 8-4	SimpleRNN 16-8	GRU 16-8	GRU 16-8-4	LSTM21 8-4	LSTM21 16-8	LSTM252 10-5	LSTM252 16-8
MINI	0.8418	0.8194	0.8586	0.8652	0.8328	0.8330	0.8457	0.8253

Cuadro 8.21: Tasas de aciertos de los modelos elegidos para MINI

Ganancias	SimpleRNN 8-4						SimpleRNN 16-8						GRU 16-8-4					
	Margen 0%			Margen 3%			Margen 0%			Margen 3%			Margen 0%			Margen 3%		
Alza	30705.60			24393.02			34455.30			36635.79			15495.45			-5169.59		
Alza y Baja	12106.76			-11057.72			18840.29			14080.57			-12247.55			-45994.04		
Baja	-14729.93			-24012.72			-12143.72			-13965.17			-24386.87			41321.45		
Días	159	149	0	19	14	275	173	135	0	19	17	272	161	147	0	14	15	279
Decisiones	Compra	Venta	Manten	Compra	Venta	Manten	Compra	Venta	Manten	Compra	Venta	Manten	Compra	Venta	Maten	Compra	Venta	Mante

Cuadro 8.23: Ganancias y perdida por modelo y margen elegidos. Número de veces que se repite una decisión según cada modelo. Para el activo MINI

previamente, las 3 RNN con menor RMSE: SimpleRNN 8-4, SimpleRNN 16-8 y GRU 16-8-4.

RMSE	SimpleRNN 8-4	SimpleRNN 16-8	GRU 16-8	GRU 16-8-4	LSTM21 8-4	LSTM21 16-8	LSTM252 10-5	LSTM252 16-8
MINI	0.97627	1.0759	0.9654	1.0153	1.111	1.1391	1.0844	1.9950

Cuadro 8.22: RMSE de los modelos elegidos para MINI

Al realizar la simulación de inversión, las ganancias y pérdidas no son muy altas bajo ninguna estrategia, en ningún modelo. Existe disparidad sobre la toma de decisiones en la red GRU, debido a que se consiguen beneficios cuando se usa margen de 3% mientras que con el margen de 0% se obtiene bastante pérdida. Analizando día a día este suceso se ha encontrado como la continua cambio de tendencias a lo largo de la serie sin ninguna tendencia clara que seguir ha hecho obtener peores resultados a las estrategias con un margen de 0% con este modelo.

Se puede confirmar que aun teniendo una muy buena predicción de los valores de cada día, para una serie temporal con muchos cambios de tendencia, es muy difícil poder sacar un resultado valioso, aún así, se han conseguido unas ganancias muy interesantes en este activo.

8.8. SNRC - Synchronoss Technologies, Inc

Empresa estadounidense que se dedica a la creación aplicaciones de plataforma con servicios PaaS y SaaS, de sincronización del negocio para compañías de operaciones de bajo coste.

La serie temporal tiene muchos cambios de tendencia secundarios, variando el precio de la acción durante toda la serie, acabando con un balance positivo. Las continuas subidas y bajadas son preocupantes de cara a una inversión segura. Figura 6.2.



Figura 8.12: Datos históricos de SNRC para conjunto de entrenamiento

El análisis de la figura 8.13, indica una tendencia primaria bajista durante todo el período expuesto, manteniendo precios similares desde Mayo de 2017. Siguen existiendo caídas y subidas pronunciadas del valor de la acción muy difíciles de predecir.



Figura 8.13: Datos históricos de SNRC para valores de predicción

El continuo cambio de tendencia, con la dificultad de predicción que conlleva, se ve reflejado en los porcentajes de acierto. cuadro 8.13. Todos los modelos obtienen valores alrededor del 80 %. Siendo la LSTM de secuencia de 252 días 16-8 el mejor modelo obtenido, similares resultados a los otros activos estudiados.

Acieros	SimpleRNN 8-4	SimpleRNN 16-8	GRU 16-8	GRU 16-8-4	LSTM21 8-4	LSTM21 16-8	LSTM252 10-5	LSTM252 16-8
SNRC	0.7789	0.7897	0.7760	0.8135	0.8099	0.7607	0.7906	0.8382

Cuadro 8.24: Tasas de aciertos de los modelos elegidos para SNRC

La similitud de este activo con el anterior, conseguimos muy buenas predicciones muy buenas, visualmente y comparando los valores RMSE, no teniendo claro cual elegir, sabiendo que casi cualquier red de las elegidas sería una buena solución. Al elegir por valores de RMSE, nos quedamos con las dos redes con celda SimpleRNN y la RNN GRU 16-8-4.

RMSE	SimpleRNN 8-4	SimpleRNN 16-8	GRU 16-8	GRU 16-8-4	LSTM21 8-4	LSTM21 16-8	LSTM252 10-5	LSTM252 16-8
SNRC	1.27	1.27	1.50	1.44	1.52	1.98	2.40	4.29

Cuadro 8.25: RMSE de los modelos elegidos para SNRC

La tendencia bajistas que tiene los datos a predecir es demasiado importante, estando beneficiadas las estrategias de inversión a la baja de cualquier modelo. Teniendo ganancias sumamente altas, sobretodo a la hora de usar margenes. En los resultados de este activo hay una red vencedora en comparación con el resto, la red SimpleRNN 8-4 obtiene quintuplicar el capital inicial con una inversión a la alza y baja. Al ser unos valores desorbitados, se analiza el día a día para saber el porque de este aumento. Al usar este modelo, se toma la decisión de comprar a la baja activos antes de las dos caídas más importantes, otorgando unos beneficios muy altos.

Ganancias	SimpleRNN 8-4						SimpleRNN 16-8						GRU 16-8-4					
	Margen 0%		Margen 3%				Margen 0%		Margen 3%				Margen 0%		Margen 3%			
Alza	-30089.52		77987.98				-48732.10		-34410.39				-51567.70		-24234.62			
Alza y Baja	32276		559277.62				-28158.36		1259.64				-36888.61		52699.23			
Baja	85773.74		263572.06				37559.09		51561.17				27916.17		97814.27			
Dias	146	162	180	61	67	0	141	167	0	55	71	182	151	157	0	33	46	229
Decisiones	Compra	Venta	Manten	Compra	Venta	Manten	Compra	Venta	Manten	Compra	Venta	Manten	Compra	Venta	Maten	Compra	Venta	Mante

Cuadro 8.26: Ganancias y perdida por modelo y margen elegidos. Número de veces que se repite una decisión según cada modelo. Para el activo SNRC

Capítulo 9

Conclusiones

El interés por la Economía, el descubrimiento del "*Deep Learning*", apoyado por diferentes materias dadas durante estos años como *Minería de Datos*, *Técnicas de Aprendizaje Automático*, *Ingeniería del Conocimiento*, *Fundamentos de Organización de Empresa*, etc, han provocado la elección de este proyecto de Investigación: "Predicción y Clasificación de series temporales bursátiles mediante Redes Neuronales Recurrentes".

Los conocimientos adquiridos en el estudio del Grado de Ingeniería Informática, sustentan la base de esta investigación. Bajo este planteamiento y contando con los datos procesados económicos con un perfil informático, ha sido desarrollada una herramienta a nivel académico, que bien pudiera ser el núcleo de una aplicación para un futuro cliente.

El objetivo de poder predecir una serie de valores bursátiles de forma satisfactoria con la utilización de Redes Neuronales Recurrentes, es un objetivo cumplido para ciertos activos. El proyecto recoge unos resultados en las predicciones más que aceptables para 5 de los 6 activos analizados. Algunas series temporales se han podido predecir con bastante precisión con todos los tipos de redes; otras han presentado problemas haciendo patente la dificultad del trabajo, dejando abierta la continuidad de la investigación. Se han encontrado redes para la mayoría de las empresas que consiguen un buen acercamiento a la realidad. Otros modelos, con esas mismas compañías, han mostrado una brecha en el resultado a partir de los 6 meses de predicción. Hacer un análisis de sensibilidad del período de días para predecir supone un planteamiento como ampliación de trabajo.

Los mejores modelos obtenidos a la hora de comparar las diferentes arquitecturas de RNN, han traído la conclusión de que no existe una red capaz de predecir cualquier activo, sino que se necesita una red específica para cada activo. Inclusive, esta especificidad afecta también al tipo de celda (simple RNN, LSTM o GRU), no pudiéndose concluir que una clase sea mejor que las otras.

Sin haber destacado ningún modelo en particular, las arquitecturas con 2 capas: 8-4, 10-5 y 16-8, han recogido las mejores predicciones. La red 16-8-4 de 3 capas, es la única que ha obtenido predicciones cercanas a la realidad. Usar menos neuronas producen tasas de acierto que indiquen un buen pronóstico.

Las redes con celdas *GRU* y *SimpleRNN* han ofrecido los mejores resultados en casi todos los activos. La celda *LSTM*, eje inicial del proyecto, ha sido llevada a un segundo plano. Aquí merece la pena subrayar los altos valores que tiene en tasa de acierto sobre el conjunto de validación, obteniendo peores resultados en términos de RMSE o ganancia.

La red LSTM con secuencia de 252 días no ha dado un rendimiento aceptable en esta investigación, llegando a valores de medidas en predicción por debajo del resto de redes. Siendo su principal característica el buen uso del recuerdo de larga duración, en este trabajo no ha demostrado de gran utilidad esta propiedad.

Argumentar un buen funcionamiento en ciertos activos de una red, o la incapacidad de ésta de poder dar una predicción estable para otro activo, es complicado. No se demuestran patrones de funcionamiento, convirtiendo la investigación en un estudio heurístico. La única forma de encontrar una RNN que con mejores resultados que las expuestas, es mediante la realización de una mayor batería de pruebas.

El uso de márgenes para establecer diferentes estrategias de inversión no ha sido beneficioso, resultando el de 0% mejor que el de 3%, en todos los casos. Esta conclusión señala la posibilidad de minimizar el riesgo de la inversión, con un análisis de sensibilidad de esta variable de decisión para cada activo.

La implementación de este proyecto ha aportado una primera experiencia a las bibliotecas *Keras* y *TensorFlow*, las más utilizadas en la construcción de Aprendizaje Profundo, aportando su conocimiento y su uso, un gran valor en mi formación.

Tras la realización de este proyecto, quedan abiertas diferentes líneas de trabajo, sin poderse llevar a cabo debido a las restricciones de tiempo de dedicación de un TFG. Posibles ampliaciones futuras serían:

- Realizar un estudio con un mayor número de arquitecturas, con una batería de pruebas más extensas, como por ejemplo, las 35 empresas del IBEX-35.
- Llevar a cabo un análisis de sensibilidad sobre la importancia del margen a la hora de tomar las decisiones diarias, sobre todo, cómo puede ayudar a minimizar el riesgo de la inversión.
- Creación de un *expert advisor*. Esto es, un robot de inversiones que opere de forma automática, según la predicción obtenida con las diferentes RNN.
- Investigar sobre las Redes Neuronales Multi-Entrada. Crear una sistema de predicción o de toma de decisiones basado, no sólo en las series bursátiles, sino en otras

informaciones de muy diversa naturaleza. En este sentido, cabría plantearse una primera etapa de procesamiento de texto porque, posiblemente sea ahí, donde se encuentre esa otra información, con la que alimentar estas Redes Multi-Entrada.

Parte II

Apéndices

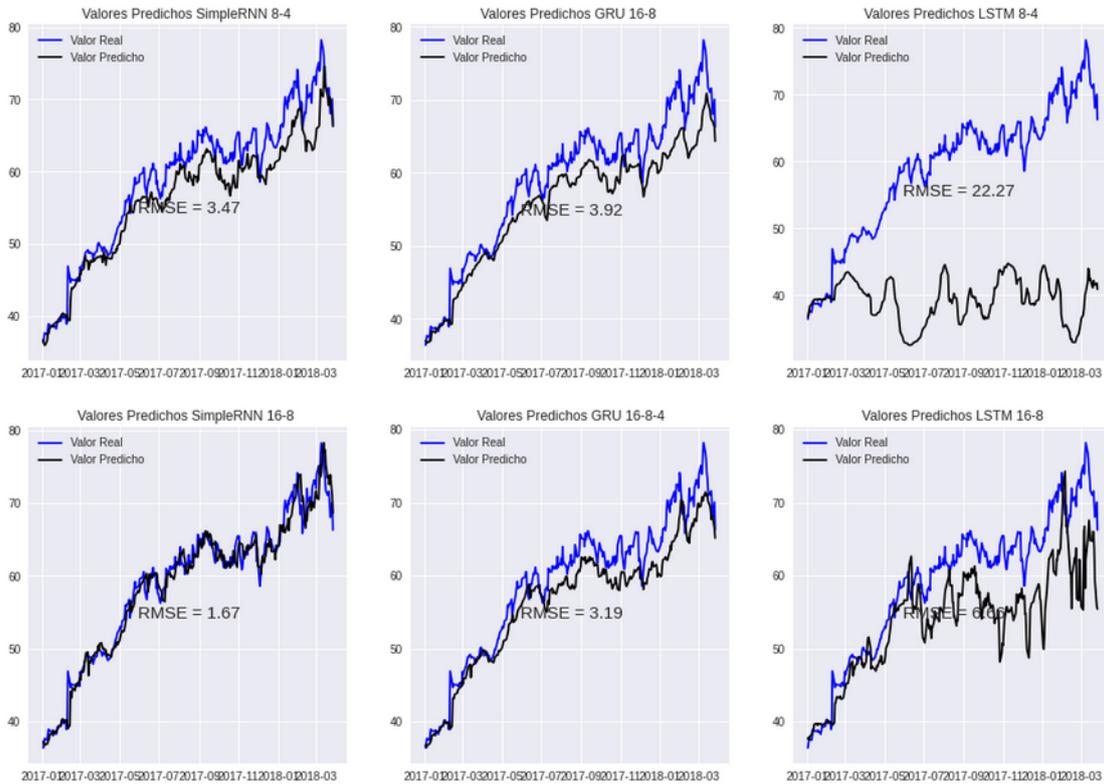
Apéndice A

Anexos

METER ALGO

A.1. Gráficas de Predicciones

ATVI



(a) Modelos con secuencia de 21 días

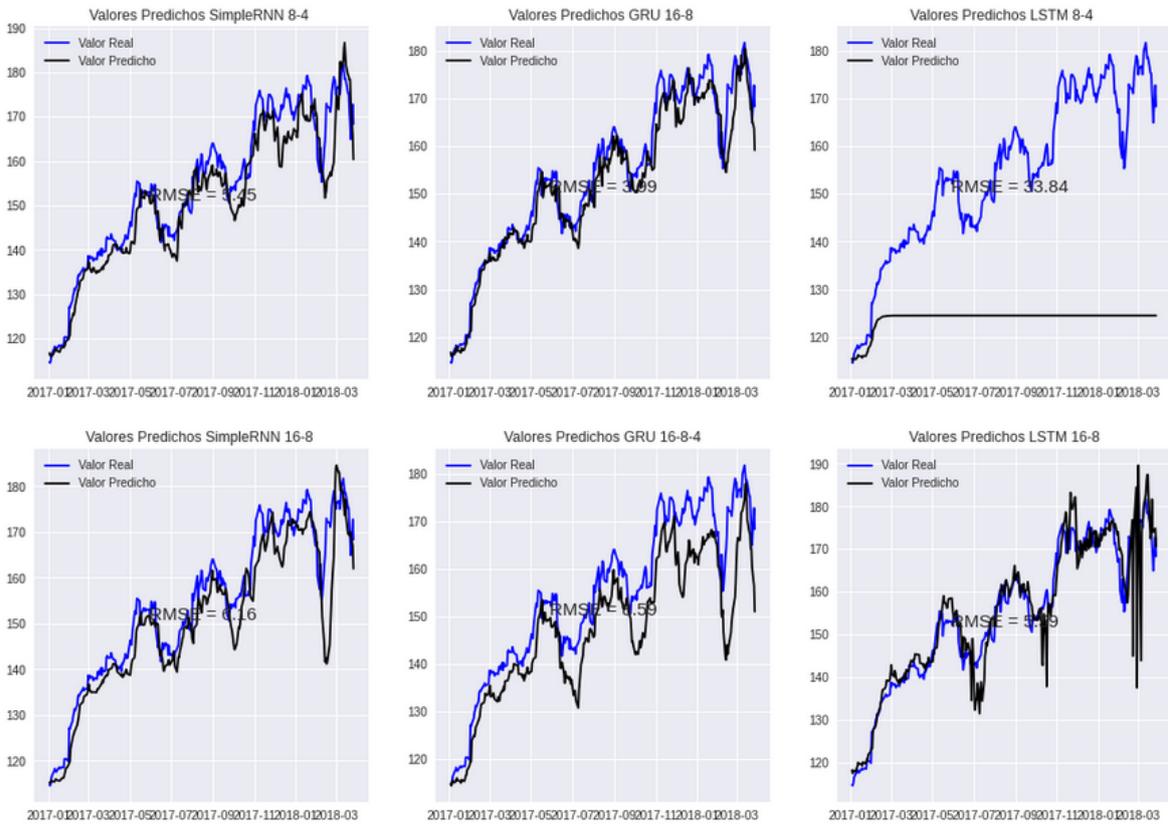
ATVI



(b) Modelos con secuencia de 252 día

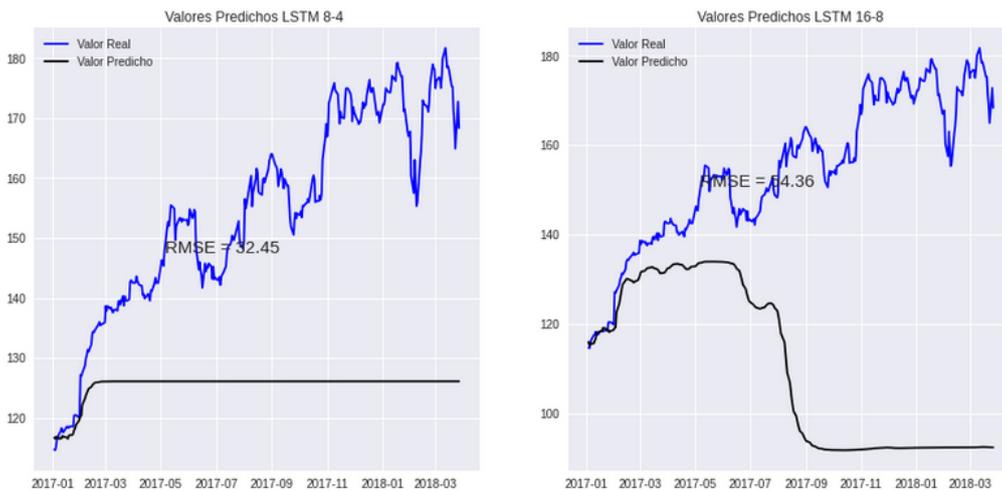
Figura A.1: Predicción de los modelos para el activo ATVI
César Hernández Rodríguez

AAPL



(a) Modelos con secuencia de 21 días

AAPL

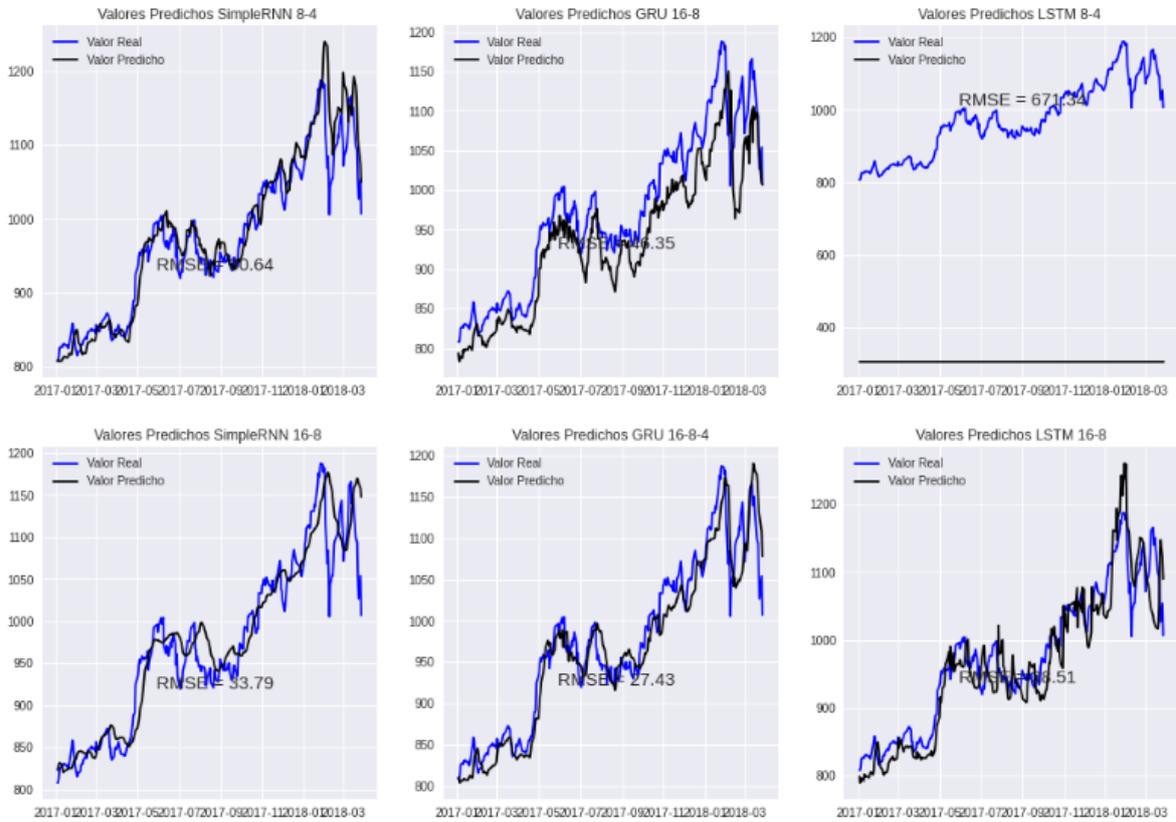


(b) Modelos con secuencia de 252 día

Figura A.2: Predicción de los modelos para el activo AAPL

César Hernández Rodríguez

GOOGL



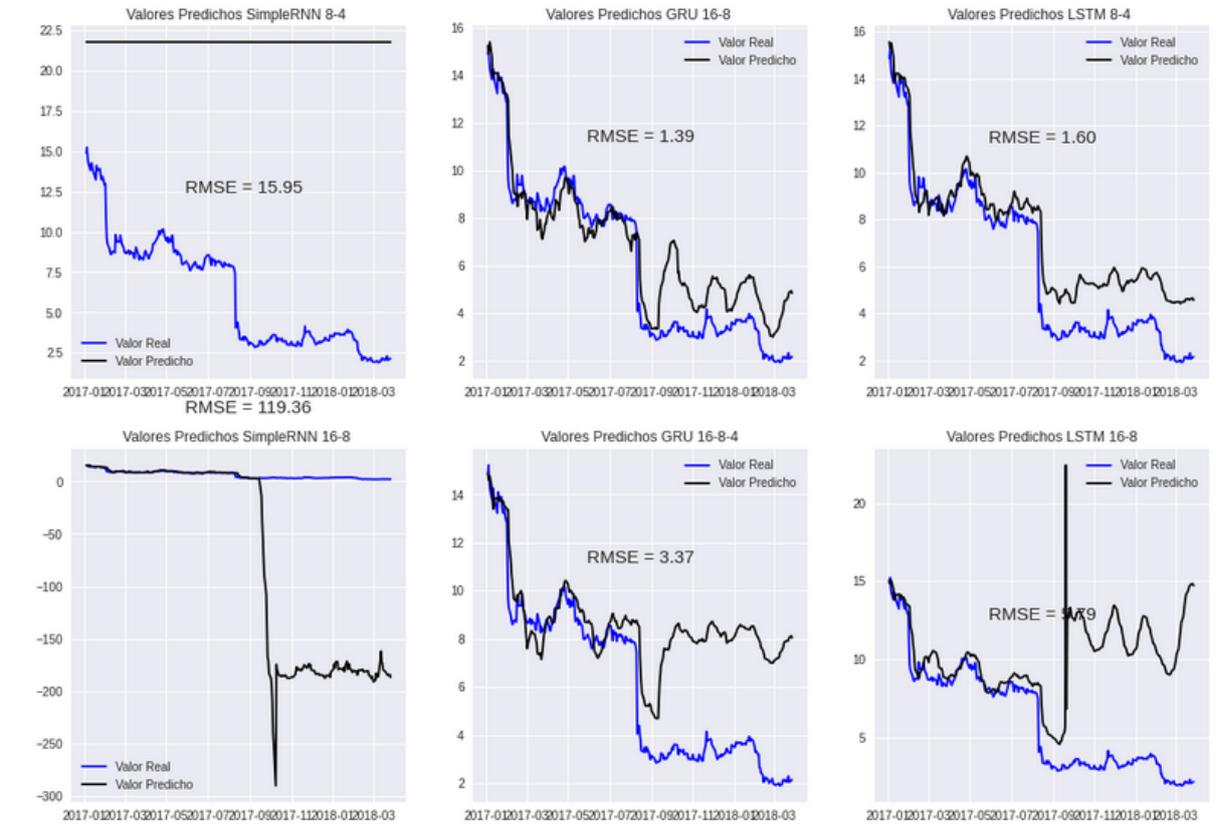
(a) Modelos con secuencia de 21 días

GOOGL



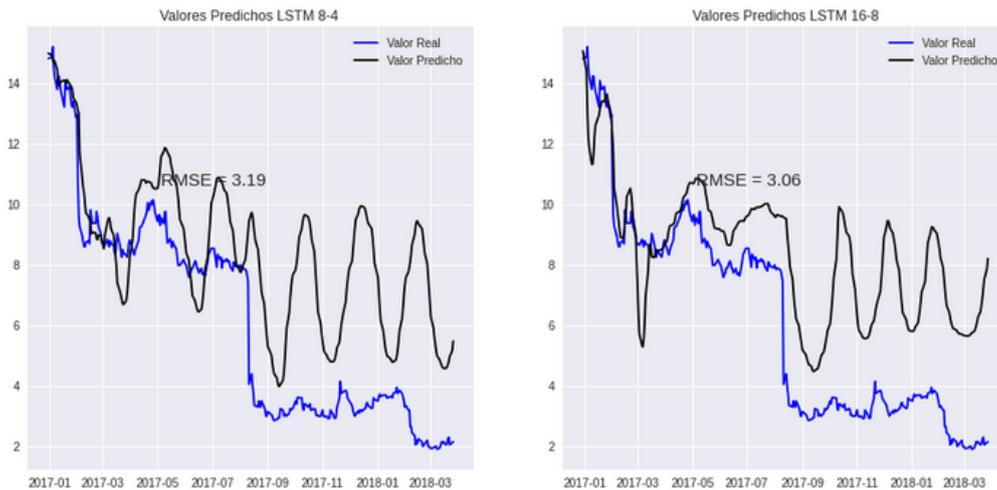
(b) Modelos con secuencia de 252 día

BBOX



(a) Modelos con secuencia de 21 días

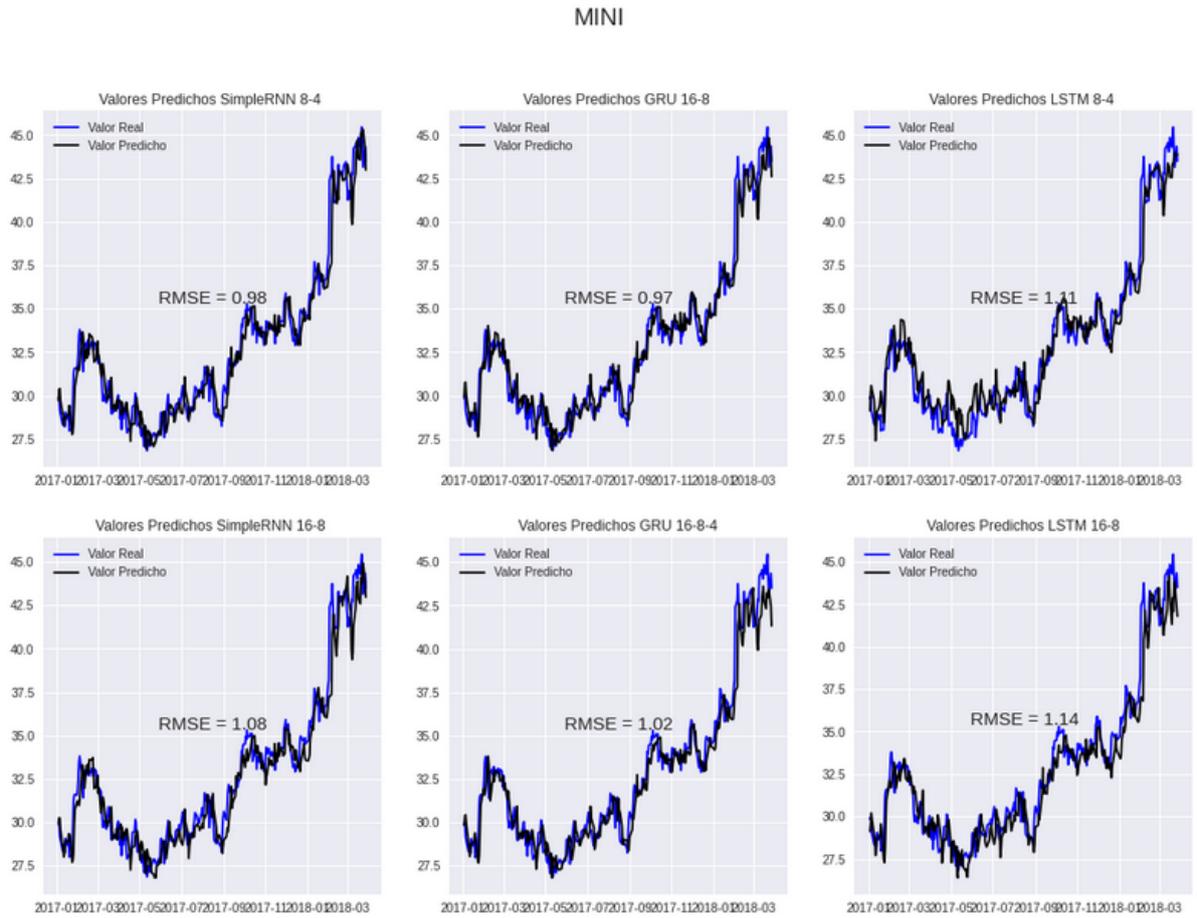
BBOX



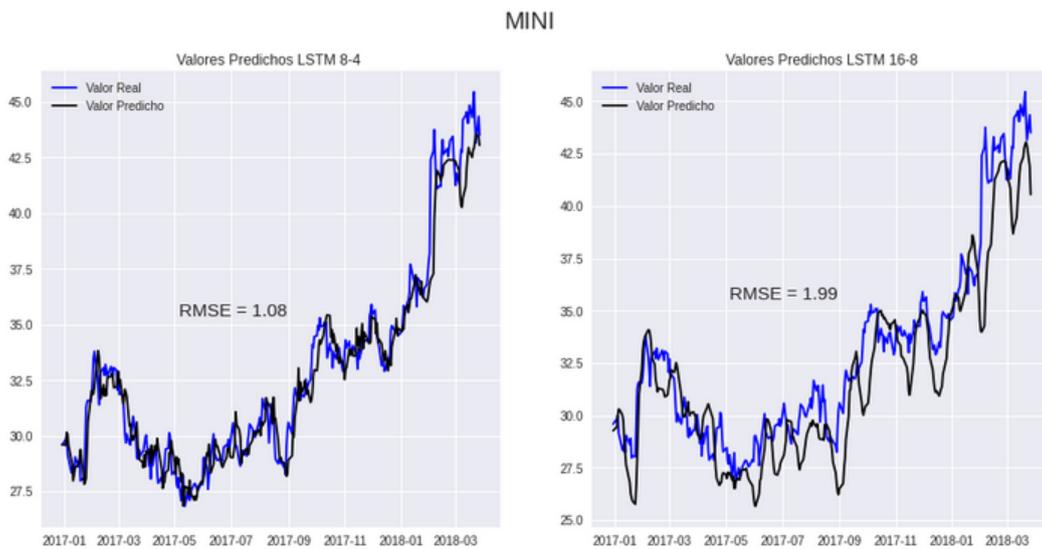
(b) Modelos con secuencia de 252 día

Figura A.4: Predicción de los modelos para el activo BBOX

César Hernández Rodríguez

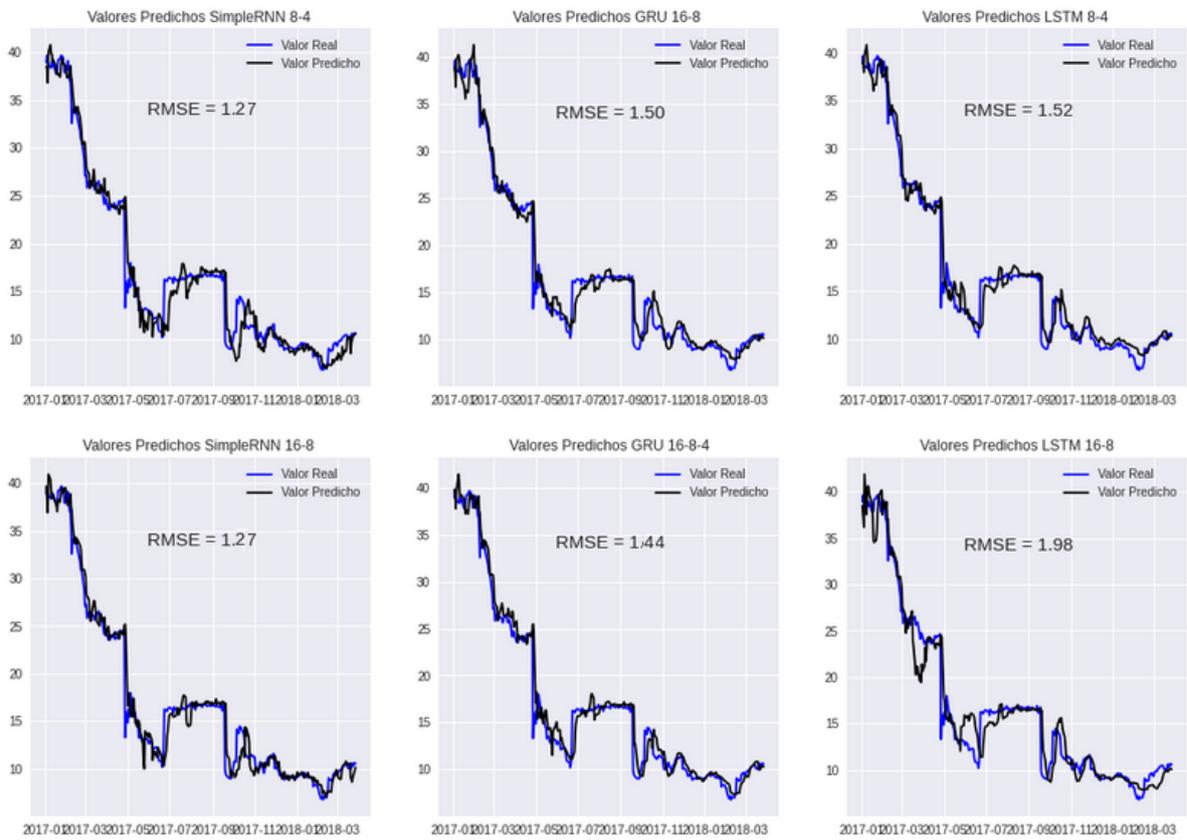


(a) Modelos con secuencia de 21 días



(b) Modelos con secuencia de 252 día

SNCR



(a) Modelos con secuencia de 21 días

SNCR



(b) Modelos con secuencia de 252 día

A.2. Tablas de decisiones

Se expone de una forma más extendida la decisión tomada cada día con los diferentes modelos elegidos para ver la simulación de inversión. Estas tablas se encuentran en diferentes *.csv* adjuntos al trabajo en el github.

Parte III

Bibliografía

Bibliografía

- [1] URL: [https://es.wikipedia.org/wiki/Anaconda_\(distribuci%C3%B3n_de_Python\)](https://es.wikipedia.org/wiki/Anaconda_(distribuci%C3%B3n_de_Python)).
- [2] URL: <https://www.python.org/>.
- [3] URL: <https://github.com/>.
- [4] Y Alexandra e Y Viviana. *Validación del Análisis Técnico como un Método de Inversión en la Renta Variable de la BVG*. 2006.
- [5] C. Alonso González. *Comparación de Varios métodos - Apuntes Técnicas de Aprendizaje Automático*. 2018-2019.
- [6] T. Calonge Conde. *Perceptron Multicapa - Apuntes Técnicas de Aprendizaje Automático*. 2018-2019.
- [7] Kyunghyun Cho y col. *Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation*. Doha, Qatar: Association for Computational Linguistics, oct. de 2014, págs. 1724-1734. DOI: 10.3115/v1/D14-1179. URL: <https://www.aclweb.org/anthology/D14-1179>.
- [8] F. Chollet. *Deep Learning with Python*. Manning, 2017.
- [9] M van Gerven y S Bohte. *Artificial Neuronal Networks as models of neural information processing*. Manning, 2018.
- [10] A Géron. *Hands-On Machine Learning with Scikit-Learn and TensorFlow*. O'Reilly Media, Inc, 2017.

- [11] B. Hernández. *Bolsa y Estadística Bursátil*. Diaz de Santos S.A, 2000.
- [12] C Hernández. *Optimización de carteras de inversión en la industria tecnológica*. Trabajo Fin De Grado, Universidad de Valladolid, 2020.
- [13] Sepp Hochreiter y Jürgen Schmidhuber. *Long Short-Term Memory*. Vol. 9. 8. 1997, págs. 1735-1780. DOI: 10.1162/neco.1997.9.8.1735. eprint: <https://doi.org/10.1162/neco.1997.9.8.1735>. URL: <https://doi.org/10.1162/neco.1997.9.8.1735>.
- [14] B Hughes y M Cotterell. *Software Project Management*. Mc Graw Hill Education, 2009.
- [15] Ayoosh Kathuria. *Intro to optimization in deep learning: Momentum, remsprop and adam*. 2018. URL: <https://blog.paperspace.com/intro-to-optimization-momentum-rmsprop-adam/>.
- [16] En línea. *Análisis Fundamental*. 2013. URL: <http://www.expansion.com/diccionario-economico/analisis-fundamental.html>.
- [17] J. Murphy. *análisis técnico de los mercados financieros*. Gestión 2000, 2016.
- [18] NASDAQ. *Información NASDAQ*. URL: <https://www.nasdaq.com/market-activity/stocks/screener?exchange=NASDAQ>.
- [19] QUANDL. *Conjunto de Datos de la Bolsa*. URL: <https://www.quandl.com>.
- [20] R. Rhea. *The Dow Theory: An Explanation of Its Development and an Attempt to Define Its Usefulness as an Aid in Speculation*. Fraser, 1993. ISBN: 9780870341106. URL: <https://books.google.es/books?id=annZAAAAMAAJ>.
- [21] D. E. Rumelhart, G. E. Hinton y R. J. Williams. *Learning internal representations by error propagation*. 1985.
- [22] David E. Rumelhart, Geoffrey E. Hinton y Ronald J. Williams. *Learning representations by back-propagating errors*. Vol. 323. 6088. Oct. de 1986, págs. 533-536. DOI: 10.1038/323533a0.

- [23] A. Scherk. *Manual de Análisis Fundamental*. Inversor Ediciones, S.L., 2010.
- [24] I. Vasilev y col. *Python Deep Learning, Second Edition*. Packt, 2019.
- [25] Wikipédia. *Historia de la Bolsa*. URL: https://es.wikipedia.org/wiki/Bolsa_de_valores.
- [26] Wikipédia. *Indice Nasdaq*. URL: <https://es.wikipedia.org/wiki/NASDAQ>.
- [27] Wikipédia. *Perceptron*. Wikipedia. URL: <https://en.wikipedia.org/wiki/Perceptron#History>.
- [28] Wikipedia. *Historia de la Bolsa de Nueva York*. URL: https://es.wikipedia.org/wiki/Bolsa_de_Nueva_York.
- [29] Wikipedia. *Información indide NASDAQ-100*. URL: <https://es.wikipedia.org/wiki/NASDAQ-100>.