

UNIVERSIDAD DE SALAMANCA  
MÁSTER UNIVERSITARIO EN MODELIZACIÓN MATEMÁTICA

# Modelización de los precios de las acciones de Apple, Microsoft, Amazon y Google utilizando redes neuronales LSTM

AUTORA: Esther Valverde Nieto

TUTORES: Juan Manuel Corchado Rodríguez

Javier Parra Dominguez

Roberto Casado Vara

Curso 2020-2021





VNiVERSiDAD  
D SALAMANCA

CAMPUS DE EXCELENCIA INTERNACIONAL

UNIVERSIDAD DE SALAMANCA  
MÁSTER UNIVERSITARIO EN MODELIZACIÓN MATEMÁTICA

# Modelización de los precios de las acciones de Apple, Microsoft, Amazon y Google utilizando redes neuronales LSTM

AUTORA: Esther Valverde Nieto

TUTORES: Juan Manuel Corchado Rodríguez

Javier Parra Dominguez

Roberto Casado Vara

Curso 2020-2021



# Certificado de los tutores

Los profesores Prof. Juan Manuel Corchado Rodríguez, Dr. Javier Parra Domínguez y Dr. Roberto Casado Vara, miembros de la Universidad de Salamanca y tutores de este trabajo de investigación.

Hacen constar:

Que el trabajo titulado “**Modelización de los precios de las acciones de Apple, Microsoft, Amazon y Google utilizando redes neuronales LSTM**”, que se presenta, ha sido realizado por D. Esther Valverde Nieto para la superación de la asignatura Proyecto de Fin de **Máster de a Titulación de Máster Universitario en Modelización Matemática**.

En Salamanca, a 12 de Julio de 2021

Firmado digitalmente por  
CORCHADO RODRIGUEZ JUAN  
MANUEL - 70978310B  
Fecha: 2021.07.12 14:20:18 +02'00'

**Prof. Juan M. Corchado Rodríguez**

Javier Parra  
Firmado digitalmente por  
Javier Parra  
Fecha: 2021.07.12  
13:05:00 +02'00'

**Dr. Javier Parra Domínguez**

CASADO VARA  
ROBERTO  
CARLOS -  
45686237A  
Firmado digitalmente por  
CASADO VARA  
ROBERTO CARLOS -  
45686237A  
Fecha: 2021.07.12  
11:36:18 +02'00'

**Dr. Roberto Casado Vara**

# Índice general

<b>1. Introducción</b>	<b>1</b>
1.1. Motivación . . . . .	1
1.2. Planteamiento del trabajo . . . . .	2
1.3. Estructura de la memoria . . . . .	3
<b>2. Contexto y Estado del arte</b>	<b>5</b>
2.1. El mercado de renta variable . . . . .	5
2.2. Análisis fundamental y análisis técnico del mercado bursátil . . . . .	7
2.3. Inteligencia Artificial para predicciones financieras . . . . .	9
2.4. Enfoque actual en la modelización de predicciones financieras: Long short-term memory . . . . .	12
<b>3. Conceptos técnicos</b>	<b>14</b>
3.1. Redes neuronales artificiales . . . . .	14
3.2. redes neuronales Long-Sort-Term-Memory . . . . .	15
3.3. Configuración de una red LSTM . . . . .	17
<b>4. Objetivos y metodología</b>	<b>21</b>
4.1. Objetivo general . . . . .	21
4.2. Objetivos específicos . . . . .	21
4.3. Metodología . . . . .	22
<b>5. Solución propuesta</b>	<b>23</b>
5.1. Análisis exploratorio de los datos . . . . .	23
5.2. Selección de las características del dataset . . . . .	30
5.3. Separación del dataset de entrenamiento y validación . . . . .	31
5.4. El vector de entrada o características . . . . .	33
5.5. El vector de salida . . . . .	33
5.6. Modelos propuestos basados en redes LSTM . . . . .	33
5.7. Validación del modelo en el resto de empresas . . . . .	38
<b>6. Resultados</b>	<b>39</b>
<b>7. Discusión</b>	<b>43</b>
7.1. Evaluación de los modelos . . . . .	43
7.1.1. Selección del modelo . . . . .	43
7.1.2. Validación del modelo 3 . . . . .	44

7.2. Discusión de los resultados . . . . .	44
<b>8. Conclusiones y trabajo futuro</b>	<b>46</b>
8.1. Conclusiones . . . . .	46
8.2. Líneas de trabajo futuro . . . . .	47
<b>Bibliografía</b>	<b>51</b>

# Índice de figuras

3.1. Capa RNN. Por autoría de Chrislb, bajo la licencia CC BY-SA 3.0. . . . .	15
3.2. Célula LSTM. Por autoría de fdeloche, bajo la licencia CC BY-SA 4.0. . . .	16
4.1. Esquema del proceso de modelización. . . . .	22
5.1. Precios de cierre (USD). . . . .	24
5.2. Precios de cierre en escala logarítmica (USD). . . . .	25
5.3. Datos financieros sobre la actividad económica de las empresas. . . . .	25
5.4. Medias móviles de 5 y 10 periodos (USD). . . . .	26
5.5. Autocorrelación (ACF) y autocorrelación parcial (PACF) de los precios de cierre diferenciado por empresas. . . . .	28
5.6. Diagrama de correlación de precios de cierre. . . . .	29
5.7. Rentabilidad - riesgo de las acciones. . . . .	30
5.8. Arquitectura del modelo 1 . . . . .	34
5.9. Arquitectura del modelo 2 . . . . .	35
5.10. Arquitectura del modelo 3 . . . . .	36
5.11. Arquitectura del modelo 4 . . . . .	37
5.12. Arquitectura del modelo 5 . . . . .	38
6.1. Validación de los modelos propuestos. . . . .	40
6.2. Función de pérdida y MAE. . . . .	41
6.3. Comparación de los modelos . . . . .	41
6.4. Validación del modelo 3 utilizando distintos datasets de entrenamiento . .	41
6.5. Validación del modelo 3. . . . .	42
8.1. Código para exportar los datos de cotización. . . . .	53
8.2. Estadísticos del rentimiento de las acciones. . . . .	53
8.3. Modelo 1 . . . . .	54
8.4. Modelo 2 . . . . .	55
8.5. Modelo 3 . . . . .	55
8.6. Modelo 4 . . . . .	56
8.7. Modelo 5 . . . . .	56



# Capítulo 1

## Introducción

### 1.1. Motivación

“El mercado es el mecanismo más eficiente del mundo para transferir riqueza de la gente impaciente a la gente paciente”, Warren Buffett.

La bolsa es un medio de financiación para las empresas, al mismo tiempo que un mecanismo para rentabilizar el dinero que depositan allí los inversores. Estos participantes de la bolsa se someten a sí mismos a un constante interrogatorio: ¿qué, cuándo y cuánto comprar o vender?.

Estas preguntas plantean al inversor una doble problemática. Por una parte, el inversor tendrá que hacer un esfuerzo por adivinar la tendencia del mercado bursátil para así poder decidir qué y cuándo comprar o vender. Por otra parte, el inversor deberá formar una estrategia o plan de acción, principalmente para limitar las pérdidas que pudiera tener del capital invertido inicialmente. La manera en la que el inversor se enfrenta estos dos problemas es muy distinta. Mientras que la formación de una estrategia es un asunto muy personal, centrado en el inversor, en sus intereses y en su exposición al riesgo, realizar un pronóstico sobre cómo evolucionarán los precios del mercado es un problema cuya resolución ha sido ampliamente estudiada. Por lo tanto, existen multitud de métodos, herramientas y procedimientos para predecir los precios de la bolsa.

La forma de predecir los precios del mercado de renta variable consiste fundamentalmente en analizar precios pasados. Sin embargo, la formación de los precios es muy compleja. Por una parte, los precios a los que cotizan las acciones de una empresa miden expectativas de futuro. Son una especie de indicadores sobre los beneficios que van a generar las empresas en el próximo semestre. Por otra parte, los fenómenos en masa y la debilidad del género humano determinan fuertemente la conducta de la bolsa. Los ciclos de la bolsa responden a un problema de psicología de masas [32]. A demás, los precios están sujetos a la legislación de cada país, y son muy sensibles a las catástrofes naturales y a otros fenómenos imprevisibles, como ocurrió en marzo con la crisis sanitaria a nivel mundial que desencadenó la COVID-19.

Sin embargo, en la ciencia económica la mayoría de los problemas que en principio

aparentan ser de resolución complicada se pueden reducir a un simple ajuste entre la oferta y la demanda. En este caso hablaremos de la oferta y demanda de acciones financieras, que se intercambian a un precio acordado mutuamente por el comprador y el vendedor. Al conjunto de operaciones de compra de acciones para venderlas a posteriori a un precio superior con el objetivo de lograr beneficios se le llama especulación.

Un inversor no puede caer en el espejismo de que la bolsa crea riqueza por sí misma, y por lo tanto debe saber que una acción que haya aumentado de precio un 20% en el último año no tiene por qué seguir creciendo a ese ritmo en el año siguiente. Como se suele decir en bolsa, *“rentabilidades pasadas no garantizan rendimientos futuros”*.

Durante los últimos años, la popularidad de la digitalización financiera y el vigoroso desarrollo de la Inteligencia Artificial (IA) también han impulsado la tendencia futura de las finanzas móviles y los nuevos tipos de tecnología financiera (FinTech). Con el avance de la tecnología de Internet y la creciente popularidad de los conocimientos financieros, la inversión en acciones se ha convertido en una parte de la inversión y la gestión en la vida. Todo esto ha condicionado los métodos de análisis de los mercados, que han ido incluyendo y perfeccionando progresivamente distintas técnicas y herramientas, con la reciente incorporación de la Inteligencia Artificial.

Aunque la Inteligencia Artificial no esté muy extendida en la práctica como soporte para la toma de decisiones de inversión, sí que está recibiendo un enorme interés en el ámbito científico. No es de extrañar que su capacidad para trabajar con cantidades masivas de datos, su capacidad de aprendizaje y adaptación a nuevos datos, y su capacidad de incorporar en un mismo análisis datos de tipo cuantitativo y cualitativo, como el procesamiento de textos, sean grandes fortalezas de esta tecnología y que sean incentivos suficientes para elegirla en el desempeño de predecir los precios futuros del mercado de renta variable.

## 1.2. Planteamiento del trabajo

Este trabajo toma el problema al que se enfrenta un inversor a la hora de analizar el mercado financiero para adivinar qué movimiento describirán los precios en el futuro incierto. A partir de dicho problema, se formula una respuesta a través de la modelización. Así, será necesario definir qué resultados que se esperan obtener, de toda la información disponible cuál es la que se tendrá en cuenta y, por supuesto, qué tipo de modelo es el más apropiado para resolver el problema.

La idea que subyace en el trabajo es que para poder crear un modelo que prediga su valor en el futuro hay que lograr antes una mejor comprensión sobre cómo se forman los precios. Sólo entonces se llega a la conclusión de que utilizando solamente precios pasados se puede pronosticar qué pasará con los precios futuros.

Para llevar a cabo tal propósito se han elegido cuatro gigantes tecnológicas multina-

cionales: Apple, Microsoft, Google y Amazon. Cotizan en Nasdaq, la segunda bolsa de valores más grande de Estados Unidos. Se trata de cuatro empresas de sobra conocidas, en boca de todo el mundo, punteras y con gran potencial de crecimiento. Elegir empresas que tengan estas características no es una decisión caprichosa, sino porque todas estas son razones por las que el tráfico de compras y ventas de sus acciones en un día normal es muy alto, lo que le da representatividad sus precios.

### **1.3. Estructura de la memoria**

La memoria se divide en 8 capítulos, del que el primero es la parte correspondiente a la introducción. El resto de capítulos continúan de la siguiente manera:

El Capítulo 2 tiene como fin exponer el contexto sobre el que se construye este proyecto. Este trabajo consiste en resolver un problema de modelización, por lo que para contextualizar el problema primero plantea las bases económicas sobre las que se establece el mercado, para conocer cómo funciona, quiénes lo forman y en qué situación se encuentra actualmente. Después, en el estado del arte se revisa la literatura precedente respecto a la construcción de modelos cuya función sea predecir los precios del mercado de renta variable, empezando por modelos más sencillos, hasta llegar a los modelos basados en Inteligencia Artificial.

El Capítulo 3 reúne un conjunto de definiciones y técnicas sobre algoritmos de Inteligencia Artificial que se utilizarán a lo largo del trabajo. Se trata de un capítulo introductorio a la IA, pero se ha procurado que sólo presente el contenido técnico indispensable para comprender el funcionamiento de la red neuronal Long-Sort-Term-Memory sobre la que se construirá el modelo.

En el Capítulo 4 se concreta el objetivo general que motiva este Trabajo de Fin de Máster, así como los objetivos específicos que han ido guiando el desarrollo de este proyecto. Además, en este capítulo se define la metodología que se ha seguido con el fin de alcanzar los objetivos marcados. Con este capítulo se da pié a la fase experimental de este trabajo, que pretende diseñar un modelo capaz de predecir precios futuros utilizando una red neuronal Long-Sort-Term-Memory.

El Capítulo 5 comprende todo el proceso de modelización. Se detalla en diferentes etapas el marco de aplicación del modelo, la selección de las características que se introducirán en el modelo, la construcción de cinco modelos alternativos y el método que se ha seguido para comparar y validar la actuación de los modelos.

El Capítulo 6 es una exposición de los resultados obtenidos en la fase experimental. Para ello se utilizan fundamentalmente gráficas y tablas.

En el Capítulo 7 se discuten y analizan en detalle los resultados publicados en el capítulo anterior. Se emiten una serie de comentarios sobre la comparación entre los modelos

propuestos, para luego elegir el modelo que según el criterio establecido con anterioridad sea el más adecuado para resolver el problema de modelización.

Para terminar, el Capítulo 8 recopila las conclusiones finales de todo el proceso llevado a cabo. Conjuntamente, se reconocen las limitaciones de este trabajo, para así poder dar paso al planteamiento de nuevas líneas de trabajo futuras que quedan pendientes para la continuación de este proyecto.

# Capítulo 2

## Contexto y Estado del arte

### 2.1. El mercado de renta variable

La primera pregunta que habría que plantearse a la hora de buscar el modelo más adecuado es cuál es la naturaleza de la bolsa o mercado de renta variable. Cada día, en el mercado de renta variable se negocian los precios de las acciones de aquellas empresas que optan por financiar parte de su capital a través de las aportaciones de dinero de millones de inversores que esperan obtener una rentabilidad de su aportación individual. La evolución de estas negociaciones para una acción concreta se presenta normalmente en una serie de datos, que son el precio máximo que alcanza en un día, el precio al que se compra la primera acción del día (precio de apertura), el precio al que se compra la última acción del día (precio de cierre), y el número de compras que se realizan de la misma acción a lo largo del día (volumen de operaciones).

Al igual que el mercado de divisas, el mercado de materias primas, el mercado de renta fija y cualquier otro tipo de mercado, está claro que el mercado de renta variable es un sistema dinámico, donde el precio de cotización de las acciones fluctúa en el tiempo a partir de ciertas variables explicativas. No obstante, no es sencillo determinar cuáles son estas variables ni en qué medida afectan al precio.

Normalmente los estudios en el ámbito de la economía política entienden el mercado como un sistema determinista. Así, se pretende explicar los precios a partir de ciertas variables macroeconómicas como el PIB nacional, el nivel de inflación o el nivel de desempleo, y un término aleatorio exógeno no explicado como respuesta a las anomalías y las irregularidades en los precios. Al menos es más fácil ver la idea contraria: cómo las tendencias de los precios de los activos, las mercancías, las monedas y cualquier otro tipo de producto financiero son importantes indicadores económicos [18].

Aún así, todavía hay muchos más factores que intervienen en la determinación de los precios. El mercado de renta variable, desde el punto de vista orgánico, es un sistema formado por un enorme número de agentes (personas, empresas, bancos, fondos de inversión y otros agentes financieros) que interactúan entre sí a través de transacciones de compra y venta, y justamente por sus acciones son éstos los actores responsables de los cambios en los precios [27].

Con todo esto, en el caso de que el precio de cotización del mercado sea una variable determinista, para estudiar cómo se determina su valor es imprescindible utilizar la hipótesis de la racionalidad no limitada (REH): todos los agentes que forman parte del sistema tienen información perfecta sobre el sistema mismo, es decir, sobre los precios, cómo se calculan los precios y sobre el resto de agentes que conforman el sistema [28]. Asumir esta hipótesis en la práctica es insostenible.

Ninguno de los agentes que operan en el mercado comprende del todo cómo funciona este sistema ni cómo se forman los precios. De hecho, lo que realmente diferencia un inversor que empieza por primera vez a realizar transacciones en el mercado, de un inversor con años de experiencia, no es el grado de acierto de la predicción de los precios, sino la vulnerabilidad de los primeros a las pérdidas de dinero debido a la opacidad del mercado. La formación financiera orientada a la operación en los mercados bursátiles es muy importante para poder establecer juicios sobre la desorbitante información que se transmite en el mercado. Para protegerse ante las pérdidas es necesario comprender las distintas escalas geopolíticas y temporales desde donde se puede observar el mercado, y conocer la gama de productos financieros disponibles para operar.

Ahora bien, aunque los operadores del mercado no puedan determinar las causas que mueven los precios por su racionalidad limitada, esto no supone una gran preocupación para ellos. La gran mayoría de las transacciones de compra y venta del mercado no se ejecutan en base a esmerados escrutinios sobre cuáles son las razones originales de que el precio vaya a subir o bajar. [19] no podía expresar con mejores palabras el razonamiento del inversor, que piensa que “si todo lo que afecta al precio de mercado queda reflejado definitivamente en dicho precio, entonces lo único que hace falta es estudiar ese precio de mercado o valor de cotización.”

Existe otra premisa muy relacionada con la anterior, que busca de la misma manera comprender los precios futuros estudiando los precios pasados: la historia se repite. Los inversores llevan cien años clasificando patrones gráficos que dibujan los precios. Se han establecido categorías de imágenes que revelan la psicología alcista o bajista del mercado, y dado que estos patrones han funcionado bien en el pasado, se asume que seguirá funcionando bien en el futuro. Estos ciclos de los precios están basados en el estudio de la psicología humana [19].

[3] expone claramente cómo estudiando el mercado como un sistema complejo se consigue liberarse de esta hipótesis, siendo el cambio de los precios del mercado el resultado de dinámicas endógenas complejas. Los agentes participantes del sistema son por lo tanto heterogéneos y de racionalidad limitada. A partir de aquí seguir perfilando la definición del mercado de renta variable resulta más complicado. Una parte de la comunidad científica se decanta por utilizar modelos de predicción que consideren que los precios de la bolsa se rigen por dinámicas estocásticas, mientras que otra buena parte, cada vez con mayor presencia, escoge crear modelos que consideren dinámicas caóticas (referencias).

Existen diferentes respuestas a la pregunta de si los precios son predecibles. [34] afir-

ma que con los datos e información histórica disponibles del mercado y de acceso público, en mayor o en menor grado los mercados, en particular los emergentes, no están total y bien organizados, y la predicción de los precios y rendimientos futuros de las acciones posiblemente arroje mejores resultados que la selección aleatoria. De nuevo, [12] sostiene que el mercado de valores es predecible hasta cierto punto si se mira desde la economía del comportamiento y la teoría socioeconómica de las finanzas.

## 2.2. Análisis fundamental y análisis técnico del mercado bursátil

Existen dos métodos fundamentales para elaborar predicciones de los precios del mercado bursátil: el análisis fundamental y el análisis técnico. El análisis fundamental se centra en la causa de la variación del precio. Tiene en cuenta variables macroeconómicas, indicadores del rendimiento de las empresas y la percepción de los participantes del mercado, por lo que emplea datos de informes de valoración financieros de las empresas y noticias de prensa. El análisis técnico está basado en la identificación de cambios de tendencia en los precios. Principalmente utiliza gráficos de velas para detectar tendencias alcistas, bajistas o laterales, soportes y resistencias y figuras de precios. El análisis fundamental y análisis técnico son dos enfoques de estudio complementarios.

Un inversor minorista que compre acciones por valor de 100 euros seguramente haya utilizado técnicas más rudimentarias para predecir los precios de cotización futuros de las acciones. Un inversor con más experiencia preferirá movilizar un capital mayor, y es probable que utilice algún sistema automático de trading como soporte en la toma de decisiones de inversión. Los sistemas automáticos de trading se basan en indicadores técnicos matemáticos o estadísticos sobre los precios pasados. Los más conocidos son las Medias Móviles (MA) en el análisis a corto plazo, y las Medias Móviles de Convergencia/Divergencia (MACD) y el Índice de Fuerza Relativa (RSI) en el largo plazo. La mayor parte del volumen de operaciones de inversión que se producen en el mercado de renta variable se realizan en base a las señales de compra y venta que generan éstos.

Pero el análisis del mercado bursátil se puede perfeccionar aún más para tratar grandes volúmenes de datos y así conseguir estimaciones más precisas a un coste de dinero y tiempo pequeños. Los grandes fondos de inversión y bancos implementan cada vez más nuevos algoritmos de trading, con la reciente incorporación de la Inteligencia Artificial para adaptar el análisis técnico a nuevos entornos que atraviese el mercado, aunque éstas continúan todavía en fase de desarrollo y aparecen sólo de manera prematura, como herramienta auxiliar [33].

La construcción de un modelo que elabore predicciones precisas se puede realizar desde muchos enfoques distintos atendiendo al mercado al que se dirija el estudio y al sujeto del análisis. De hecho, el análisis también cambia de matiz cuando se realiza desde el punto de vista de un inversor, cuyo principal objetivo es ganar dinero, a cuando se realiza desde el punto de vista científico, cuya motivación suele ir más encaminada a optimizar el nivel de

precisión del pronóstico. Para el inversor cobran mucha importancia en el análisis factores como la duración de la inversión, pues en ese tiempo no tiene disponibilidad de su dinero, y el riesgo que tiene de perder.

Desde el punto de vista de la investigación, una gran cantidad de trabajos que se vienen desarrollando en esta línea centran su atención en encontrar la técnica o método que mejor precisión consiga y del que mayores beneficios se podrían obtener si se hiciera caso a su pronóstico. Es por ello que la mayoría de artículos comparan la actuación de distintos modelos para un mismo problema de modelización muy concreto (por ejemplo, la predicción de precios de Bitcoin a dos años vista). Al final, la conclusión general que se obtiene es que no hay una técnica mejor que otra, y que su adecuación depende más del planteamiento del problema o la situación a modelar. Esto ha servido de motivación en el desarrollo de este trabajo para no perder de vista el contexto económico, e intentar definir cuál es el problema de modelización que se quiere resolver.

Los activos financieros para los que se elaboran más modelos predictivos de precios son las acciones, por encima de los índices de mercado y muy por encima de las divisas, aunque en el último año ha incrementado notablemente el número de modelos diseñados para predecir el precio de las criptomonedas. En menor medida se elaboran modelos para predecir los precios de materias primas y bonos del Estado [30]. Puede ser que estos últimos sean casos menos interesantes por estar sujetos a un gran efecto de estacionalidad el primero, y a una muy baja volatilidad el segundo.

Los modelos ARIMA son modelos predictivos estadísticos muy sencillos que se basan en explotar los patrones de autocorrelación presentes en una serie temporal, lo que implica que las observaciones no son independientes. En [1] se estima el precio de cierre de Nokia y Zenith como combinación lineal de precios de cierre anteriores y sus errores, por lo que el problema a resolver se reduce a encontrar un buen ajuste de parámetros de autorregresión y media móvil. El modelo ARIMA diseñado es bueno en tanto que los residuos son series de errores aleatorios. Dado que no hay picos significativos de ACFs y PACFs, significa que los residuos del modelo son ruido blanco, no quedan otros patrones significativos en la serie temporal.

En la literatura más reciente los modelos ARIMA se suelen utilizar dentro de modelos híbridos para la predicción de precios, ya que los modelos ARIMA por sí mismos no son capaces de captar las no linealidades de las series temporales [24],[35].

Una práctica muy común dentro del análisis fundamental que realizan las personas que operan en el mercado es elaborar predicciones sobre la volatilidad del mercado. En el ámbito de la investigación destaca el uso del modelo autorregresivo generalizado GARCH para estudiar la fluctuación y el riesgo de las criptomonedas. [20] compara distintas variantes del modelo GARCH que expliquen con el mínimo Error cuadrático medio (RMSE), error porcentual absoluto medio (MAPE) y error absoluto medio (MAE) el comportamiento volátil de las seis monedas nacionales y las seis criptomonedas más influyentes del mercado de divisas. Los resultados apuntan a que mientras que para estimar la volatilidad de las divisas el modelo óptimo es el modelo GARCH integrado (IGARCH), para capturar la



volatilidad de cada una de las criptomonedas hace falta construir modelos GARCH más rigurosos que incorporen efectos como el apalancamiento financiero o el distinto impacto entre una “buena noticia” y una “mala noticia”.

En ambos casos los modelos se ajustan de acuerdo a los patrones históricos de una serie temporal fija, por lo que aunque obtengan predicciones satisfactorias no garantizan que ampliando la serie temporal original los mismos modelos continúen siendo adecuados. El mercado de renta variable está continuamente expuesto a los estímulos de los agentes que lo forman, siempre cambiantes. Este hecho se hace aún más relevante en el mercado de criptomonedas, altamente volátil por la incertidumbre que generan la falta de regulación y su característica naturaleza virtual. Será más conveniente establecer modelos capaces de adaptarse a la entrada de nuevos datos, y es aquí donde tomarán el protagonismo de las técnicas de Inteligencia Artificial [33].

### 2.3. Inteligencia Artificial para predicciones financieras

La extracción de características de los precios y la captación de patrones y tendencias es una tarea extremadamente complicada. Sin embargo, las técnicas de Inteligencia Artificial se utilizan continuamente para la resolución de este tipo de problemas. La IA es capaz de aprender patrones de los datos e inferir soluciones a partir de nuevos datos desconocidos. Además, es capaz de trabajar con grandes conjuntos de datos. Esto mediante un conjunto de reglas internas, que serán halladas por el mismo agente, según sea el caso. El éxito de dichos sistemas es ser capaz de entregar la solución a un problema, con base en la generalización de características halladas.

El aprendizaje automático es una rama de la Inteligencia Artificial (IA), y es un proceso de aprendizaje que comienza con la identificación del dominio de aprendizaje y concluye con la prueba y emplear los resultados obtenidos del aprendizaje en la resolución de un problema [26]. En el caso del mercado de valores, los datos generados son enormes y muy poco lineales. Para modelar este tipo de datos dinámicos hay que buscar modelos que puedan analizar los patrones ocultos y la dinámica subyacente. Los algoritmos de aprendizaje profundo son capaces de identificar y explotar las interacciones y los patrones existentes en los datos mediante un proceso de autoaprendizaje.

Además, el aprendizaje profundo proporciona un marco muy potente a la hora de utilizar grandes conjuntos de datos en diversas tareas de predicción, ya que es capaz de modelar relaciones complejas y no lineales entre variables u observaciones. Debido a esto, se han desarrollado muchos algoritmos de aprendizaje automático y se han aplicado a la predicción bursátil [25].

[23] revela que los algoritmos de aprendizaje automático más utilizados en las predicciones bursátiles son el algoritmo Support Vector Machine y las redes Artificial Neural Networks (ANN), en español “redes neuronales Artificiales”. Además, afirma que la ma-

yoría de los trabajos revisados predijeron el movimiento del precio de las acciones, y pocos se dedicaron a predecir el precio real de las acciones futuras. Con esto anima a nuevos estudios que analicen las dificultades que presenta predecir el precio exacto de un activo financiero en comparación con si el precio va a crecer o disminuir en un horizonte de tiempo.

## redes neuronales

Para los inversores, la sustitución del análisis técnico del movimiento de los precios por sistemas basados en técnicas de Inteligencia Artificial todavía les suena a una realidad lejana. Las redes neuronales funcionan como un método de “*caja negra*”, y las reglas extraídas de los métodos no son fácilmente comprensibles. No hay que pasar por alto esta característica inherente de las redes neuronales, que crea una división entre los analistas financieros que están a favor de los sistemas automáticos de trading y los más conservadores, a quienes no les convence eliminar el elemento humano subjetivo de la transacción de compra y venta de acciones. Este último grupo mantiene el control de la interpretación de dichos indicadores así como el proceso de toma de decisiones.

En [15] el modelo híbrido propuesto utiliza un algoritmo genético de expectativas adaptativas para optimizar el sistema de inferencia difuso basado en la red adaptativa (ANFIS) para predecir las tendencias del precio de las acciones. Se trata de un tipo de modelo de uso muy extendido en la predicción a partir de series temporales. Es interesante ver cómo utiliza una metodología sencilla, pensada para ser utilizada fácilmente por los inversores. Al trasladar el planteamiento del problema al punto de vista del inversor, se da cuenta de que los inversores suelen tomar sus decisiones a corto plazo basándose en la información bursátil reciente, como las últimas noticias del mercado o las fluctuaciones de los precios en tiempo real. Para considerar las relaciones lineales entre los periodos recientes, aplica un modelo de expectativas adaptativas al modelo que propone inicialmente, con lo que consigue mejorar el rendimiento de las previsiones.

Llama también la atención la aplicación de inferencia difusa (FIS) para producir reglas a partir de los valores lingüísticos de los indicadores técnicos. También es utilizada en [14] por ventajas como su simplicidad y estructura natural, y por ser una alternativa a la clasificación de los resultados en grupos excluyentes, pues se basa en conjuntos analógicos o de razón que puede tomar cualquier valor entre cero y uno. Además se utiliza frecuentemente en la modelización de escenarios con incertidumbre, y es de fácil combinación con otros métodos de inteligencia para formar modelos híbridos.

Una posible aplicación de ANN en la predicción de precios podría ser la clasificación de los famosos gráficos de velas utilizando redes neuronales Convolucionales (CNN). Las CNN son un tipo de ANN cuyos datos de entrada son matrices bidimensionales. A diferencia de las MLP, es importante la disposición de los datos dentro de la matriz de entrada. Por esta razón se usa frecuentemente para la clasificación de imágenes, ya que lugar que ocupa cada píxel y la relación con sus vecinos se debe guardar siempre [6].

En [11] el modelo se entrena en el mercado de renta variable de Estados Unidos y es

capaz de elaborar predicciones a largo plazo (a 12 años vista) sobre los precios de los mercados de 31 países diferentes. Los autores remarcan de hecho la utilidad de que este modelo pueda entrenarse en un mercado maduro como el de Estados Unidos y consiga realizar buenas predicciones en mercados de países emergentes, que normalmente no tienen tantos datos como para entrenar el modelo basado en redes neuronales. Así, consideran como su modelo adecuado en el sentido de que la arquitectura del modelo, las características de entrada y el procedimiento de entrenamiento son satisfactorios, dando como resultado un modelo que no tiene por qué ser entrenado y probado en el mismo mercado. Además, este artículo explora la aplicabilidad del modelo acompañándola de una comparación entre las características de diferentes los mercados a los que se aplica, según su tamaño y su liquidez.

[11] diseña un modelo basado en CNN que transforma las series temporales en imágenes, que serán los inputs entrada de la red neuronal, y después se clasifican como “comprar”, “vender” o “mantener” en función de la tendencia alcista, bajista o lateral que reproduzca la serie temporal. Este modelo consigue obtener mayor rentabilidad de la inversión en ETFs y en el Dow Jones 30 que si simplemente se comprara una participación en el instante inicial y se vendiera en el instante final. Además, mejora la rentabilidad esperada a largo plazo (10 años vista) que de utilizar métodos de regresión como el RSI, SMA de 14 y 50 días, y otros modelos comparados, un modelo LSTM y un modelo Perceptrón Multicapa (MLP) [31].

A diferencia del MLP y las CNN, las redes neuronales Recurrentes (RNN) durante el aprendizaje almacenan en un estado interno (al que se suele denominar como “*memoria*”) información relativa a la transformación de un fragmento o secuencia de la serie temporal. Esta operación se repite para cada elemento de la serie temporal, y cada subsiguiente output estará determinado por el cálculo anterior. De este modo, durante el proceso de aprendizaje se incorpora la dependencia o características comunes entre secuencias de la serie temporal. En [17] se hace visible que el modelo RNN no es capaz de captar patrones en los precios movidos por la estacionalidad. En este caso los modelos basados en MLP y CNN predicen mejor el comportamiento del precio al extraer los datos de la serie temporal en ventanas de información.

Las redes neuronales generales, como las redes neuronales convolucionales, procesan muestras independientes e idénticamente distribuidas, y los problemas que resuelven, que pueden ser problemas de clasificación, problemas de regresión o problemas de expresión de características, todos ellos satisfacen la propiedad de independientes e idénticamente distribuidos [10]. Sin embargo, hay más problemas reales que no satisfacen estas propiedades, como la traducción de idiomas o la generación automática de textos. Se tratan de problemas en los que se trabaja con secuencias de datos, que incluyen series temporales y secuencias espaciales. En comparación con las redes CNN, los datos de secuencias o series temporales son más adecuados para ser procesados con redes neuronales Recurrentes (RNN) [29], [12].

## 2.4. Enfoque actual en la modelización de predicciones financieras: Long short-term memory

A la hora de manejar series temporales de gran longitud pueden aparecer problemas al utilizar RNN. Durante el entrenamiento de la red neuronal en cada paso de tiempo los datos de entrada sufren transformaciones, por lo que hay información que se pierde en cada paso. Después de un tiempo de entrenamiento, la memoria de la RNN no contiene prácticamente ningún rastro de las primeras entradas [6]. Para solventar este problema aparece la célula Long Short-Term Memory (LSTM), propuesta por primera vez en 1997 por Sepp Hochreiter y Jürgen Schmidhuber [7]. Es capaz de reconocer señales de entrada importantes y almacenarlas en la memoria por el tiempo que se necesite, así como de olvidar fluctuaciones y errores a corto plazo, lo que permite capturar dependencias no lineales muy lejanas en el tiempo dentro de la serie temporal.

Las redes neuronales que tienen estructura de LSTM tienen gran éxito en el ámbito de las aplicaciones de datos secuenciales, como la traducción secuencial de textos [8] y las series temporales [4], [21], [9]. Además, su incorporación al modelo y su diseño es bastante simple, lo que hace que dentro de los modelos de predicción de precios basados en RNN sea ampliamente utilizada por los investigadores [30].

Para el desarrollo de este trabajo se han utilizado como referencia los modelos que se citan a continuación, ya que los problemas de modelización que resuelven tienen muchos puntos en común con el problema de modelización que se plantea aquí. Utilizan redes LSTM para la predicción de precios de activos de renta variable. Su contribución ha sido especialmente importante en el problema que se presenta en el proceso de modelización a la hora de seleccionar el input que se introducirá en la red neuronal LSTM. [4] desarrolla un modelo que compara la precisión del pronóstico, la rentabilidad esperada y la volatilidad (medida por el Ratio de Sharpe) de carteras de inversión en 1, 5, 10, 100 y 200 empresas del índice S&P. Para ello utiliza como input los rendimientos históricos de las empresas del S&P.

[21] calcula también el rendimiento esperado de 5 activos del mercado brasileño, con la diferencia de que utiliza como input de la red neuronal 175 indicadores financieros, además de los precios históricos de los activos. En el mismo estudio se propone mejorar el modelo disminuyendo la dimensión del input, como hace [13] en su modelo. Éste analiza un ETF de Taiwan (un fondo de inversión cotizado) introduciendo como inputs precios históricos más los seis indicadores más relevantes en el análisis financiero, por lo tanto reduce el tamaño del input haciendo una selección de las características. Además, utiliza una ventana móvil de datos para generar una nueva red neuronal al final de cada día de negociación, técnica que se ha trasladado al diseño de los cuatro modelos.

Por último, [22] explora la predicción de precios de dos activos utilizando datos de precios históricos, datos macroeconómicos y datos de “Google trends index”, así como datos cualitativos obtenidos de twitter, webs de noticias financieras y foros de discusión. Al igual que en el artículo anterior el modelo conseguía mejores predicciones reduciendo

do el tamaño del input, mejor dicho, haciendo una selección manual de las variables de entrada, aquí para mejorar el modelo se utilizan técnicas de selección automática de los inputs a partir de los datos de cotización. El estudio concluye en que de esta forma los datos de tendencias extraídos de Google mejoran la precisión del modelo, y no introducen percepciones incorrectas que aumenten los errores de los resultados. Es por eso que la integración de técnicas de IA en la predicción de precios constituye una grande oportunidad para analizar ingentes cantidades de datos, cambiantes, diversos y complejos, aunque existe un gran desafío al momento de integrarlas dentro del problema de modelización que se esté estudiando.

# Capítulo 3

## Conceptos técnicos

### 3.1. Redes neuronales artificiales

Ya se ha expuesto la problemática de analizar los precios en el mercado de renta variable. La compra y venta de acciones de los agentes en el mercado genera cada día una cantidad masiva de precios que fluctúan y que son altamente no lineales. Con el propósito de modelar las relaciones complejas y no lineales existentes entre los precios de cierre de las acciones de Apple, dentro de los modelos propuestos en este trabajo se utilizarán redes neuronales Recurrentes (en concreto redes LSTM) por su extendido uso y excelente rendimiento en este tipo de tareas. Los algoritmos de aprendizaje profundo son capaces de identificar y explotar las interacciones y los patrones existentes en los datos mediante un proceso de autoaprendizaje. En esta sección se explicará brevemente en qué consiste una red LSTM y cómo funciona.

Una red neuronal Artificial es un modelo ejecutado por un ordenador formado por un conjunto de neuronas o células conectadas entre sí. Las neuronas que se encuentran conectadas se transmiten señales de unas a otras, como lo harían las neuronas de un cerebro. Para ilustrar cómo se transmite la información a través de la red se pone el ejemplo de una red sencilla formada por dos neuronas conectadas entre sí. Una de ellas será la neurona de entrada, es decir, aquella que recibe un estímulo externo a la red y la transmite al resto de la red, en este caso la segunda neurona. Esta será la neurona de salida de la red, que ante el estímulo que ha recibido genera una respuesta que envía al exterior de la red. Todo este proceso se mueve y dirige por el cómputo de una serie de operaciones matemáticas. El estímulo externo que se introduce a la red consiste en un conjunto de valores organizados en un vector. Este vector de entrada llega a la primera neurona y se multiplica por un peso, de tal manera que llega transformado a la neurona siguiente. En realidad lo que sucede cuando un vector de entrada atraviesa la red neuronal es que se multiplica por matrices de pesos, además de ser sujeto de otras operaciones matemáticas, produciendo unos valores de salida.

Cuando decimos que este sistema se está entrenando lo que ocurre es que se están buscando los valores óptimos de los parámetros de la red. Para ello la red utiliza el algoritmo de retropropagación [29]. Dicho rápidamente, la idea en la que se basa es que si se introduce en la red el vector de entrada y además se le indica cuál es el vector de salida que

se quiere que asocie a dicha entrada de datos, las conexiones entre las neuronas que están dentro de la red se configurarán de tal forma que, si se vuelve a introducir el mismo conjunto de datos, la propia red asocie esta entrada con la respuesta “correcta”. Los detalles del funcionamiento del algoritmo se retoman un poco más adelante en este mismo capítulo.

## 3.2. redes neuronales Long-Sort-Term-Memory

La información que se transmite a través de la red no tiene por qué fluir en una sola dirección, esto es, de la capa de entrada a la de salida. Una red neuronal Recurrente recibe entradas, produce una salida y además recurre a sí misma, es decir, envía su propia salida de vuelta a sí misma (ver figura 3.1). En cada paso de tiempo, cada neurona recurrente recibe diferentes entradas, así como su propia salida del paso de tiempo anterior, denotada en la figura como  $D$ . En el caso del primer instante de tiempo, dado que no existe vector de salida anterior generalmente se iguala  $D = 0$ .

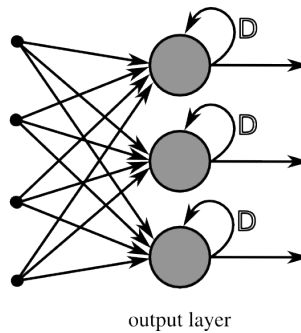


Figura 3.1: Capa RNN. Por autoría de Chrislb, bajo la licencia CC BY-SA 3.0.

En la figura 3.1 se disponen tres neuronas una encima de otra, es así como se representa una capa dentro de la red neuronal. A la hora de construir una red es posible organizar las neuronas en diferentes capas, de esta forma se distribuye el trabajo que debe realizar la red en distintas capas, una para cada función. Las redes neuronales Recurrentes se componen de una capa de entrada, una o varias capas ocultas intermedias y una capa de salida. El número de neuronas de la capa de entrada es igual al número de variables explicativas (espacio de características). El número de neuronas de la capa de salida se corresponde con el número de clases en las que se quiere clasificar el vector de salida, en este caso una neurona que indica el precio de cierre ajustado de la acción al día siguiente.

La principal característica de las redes LSTM, dentro de las Redes Recurrentes, se encuentra en las capas ocultas, formadas por las llamadas células de memoria. En la figura 3.2 se muestra un esquema de la célula LSTM. Una capa oculta en una red LSTM está formada por una célula conectada a sí misma (recurrente) que, comúnmente hablando, tiene un espacio de memoria a corto plazo  $h_t$  y otro a largo plazo  $c_t$ . Durante el entrenamiento, la célula aprende a reconocer un vector de entrada importante y es capaz de almacenar esta información en su memoria, así como de descartar aquella información que

no sea relevante.

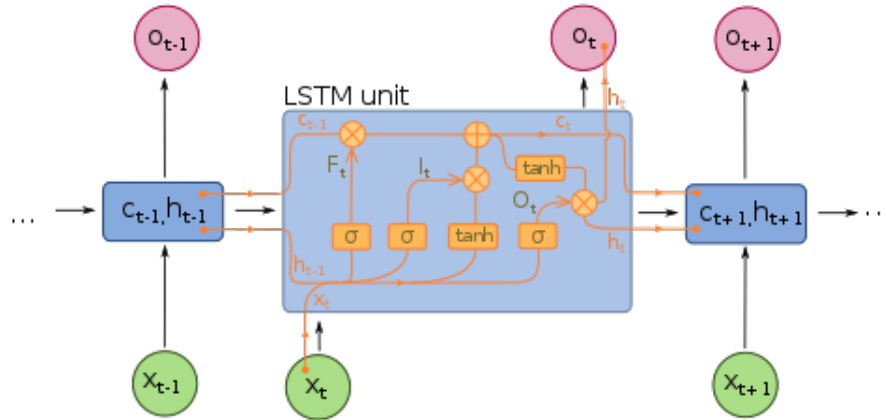


Figura 3.2: Célula LSTM. Por autoría de fdeloche, bajo la licencia CC BY-SA 4.0.

El funcionamiento de la célula es el siguiente: el vector de entrada  $x_t$  y el estado a corto plazo de la célula anterior  $h_{t-1}$  pasan a través de cuatro capas *fully-connected* diferentes. La capa principal, la que tiene por función de activación la tangente hiperbólica, es aquella que discrimina entre las dependencias más importantes y las poco significativas en el conjunto de datos. Así, la salida de esta capa se separará en dos caminos: la parte más importante se almacenará en el estado a largo plazo  $c_t$ , y el resto será desechado.

Las otras tres capas son controladores de puertas. Una célula LSTM puede aprender a reconocer una entrada importante (ese es el papel de la de la puerta de entrada  $I_t$ ), almacenarla en el estado a largo plazo, conservarla durante el tiempo que se necesite (esa es la función de la puerta del olvido  $F_t$ ), y extraerla cuando se necesite (puerta de salida  $O_t$ ). Las puertas utilizan la función de activación logística, de esta manera sus salidas están dentro del intervalo  $[0, 1]$ .

Las siguientes ecuaciones resumen cómo calcular el estado a largo plazo de la célula, su estado a corto plazo a corto plazo, y su salida en cada paso de tiempo para una sola instancia. Las ecuaciones que realmente se utilizan, teniendo en cuenta cada mini-batch, son por extensión muy similares.

$$\begin{aligned}
 I_t &= \sigma(W_{xI}^\top x_t + W_{hI}^\top h_{t-1} + b_I) \\
 F_t &= \sigma(W_{xF}^\top x_t + W_{hF}^\top h_{t-1} + b_F) \\
 O_t &= \sigma(W_{xO}^\top x_t + W_{hO}^\top h_{t-1} + b_O) \\
 G_t &= \tanh(W_{xG}^\top x_t + W_{hG}^\top h_{t-1} + b_G) \\
 c_t &= F_t \otimes c_{t-1} + I_t \otimes G_t \\
 o_t = h_t &= O_t \otimes \tanh(c_t).
 \end{aligned}$$

Donde la notación que se ha utilizado es la siguiente:



- $W_{xI}, W_{xF}, W_{xO}, W_{xG}$  son las matrices de pesos de cada una de las cuatro capas para su conexión con el vector de entrada  $x_t$ .
- $W_{hI}, W_{hF}, W_{hO}, W_{hG}$  son las matrices de pesos de cada una de las cuatro capas para su conexión con el estado anterior a corto plazo  $h(t-1)$ .
- $b_I, b_F, b_O, b_G$  son los términos de sesgo para cada una de las cuatro capas. Hay que recalcar que Tensor-
- Flow inicializa  $b_F$  con un vector lleno de 1s en lugar de 0s. Esto evita que se olvide todo al comienzo del entrenamiento.

### 3.3. Configuración de una red LSTM

#### Capas fully-connected

Las redes neuronales también pueden estar compuestas por diferentes tipos de capas según se establezcan los enlaces entre las neuronas de la capa anterior y posterior. En las redes que se han diseñado también se utilizan capas *fully connected*, en las que las neuronas de esa capa están conectadas a todas las neuronas de la capa anterior, esto es, sus neuronas de entrada.

#### Capas bidireccionales

También se han utilizado capas LSTM bidireccionales. En cada epoch, una capa recurrente normal sólo tiene en cuenta para el cómputo de los pesos de entrenamiento las entradas de datos pasados y presentes, con los que genera su salida. Dicho de otra forma, es “causal”, por lo que no tiene en cuenta los datos futuros del dataset. Cada nueva capa bidireccional creará un duplicado de la capa LSTM correspondiente, sólo que en sentido inverso, y ejecutará ambas y para concatenar luego sus salidas en cada epoch. Por ejemplo, aunque la capa LSTM del modelo 4 tiene 150 neuronas, la capa bidireccional dará el doble de valores por paso de tiempo al combinar sus salidas, es decir 300 (figura 5.11). Este tipo de RNN tiene sentido aplicarlas cuando se quiere realizar un pronóstico dentro de una serie temporal como la que estamos utilizando, pero para muchas otras tareas, como en el procesamiento del lenguaje natural para traducir textos automáticamente, a menudo es conveniente mirar al futuro las siguientes palabras antes de codificar una palabra determinada.

Dentro de la configuración de la arquitectura de una red neuronal también es necesario definir una serie de elementos para que funcione de manera correcta, como el número de epochs, las funciones de activación de las neuronas, la función de optimización del modelo y el tamaño del batch.

## Número de epochs

En la simulación de los modelos se ha fijado el número de epochs a 50. Un epoch es un recorrido por las muestras del conjunto de entrenamiento. Después de cada epoch la red neuronal predice los valores de las muestras no vistas del conjunto de validación y se calcula una pérdida de validación (MSE) de comparar los valores de predicción con los valores reales de la muestra.

## Función de activación

La función de activación aplicada a una neurona modifica su salida en función de la entrada que reciba, añadiéndole un sesgo. Ésta transformación de la salida se realiza con la finalidad de introducir no linealidad al entrenamiento. Gracias a este hiperparámetro se hará posible la retropropagación, debido a que los gradientes se suministran junto al error permitiendo la actualización de pesos y sesgos. Durante la programación del modelo se ha escrito el código de manera que el propio programa utilice la función de activación que considere más oportuna en cada caso.

## Algoritmo de retropropagación

Ahora sí, después de haber presentado estas pocas nociones técnicas, se retoma la explicación del algoritmo de retropropagación, es decir, cómo se entrena la red. El funcionamiento de este algoritmo es iterativo y se lleva a cabo en dos etapas para cada epoch:

- se recorre la red en el sentido normal, hacia la capa de salida, multiplicando el vector de entrada por los pesos de la red y aplicando las funciones de activación de cada neurona o capa de neuronas.
- se calcula el error al comparar las salidas de la red con los valores esperados. Luego, se propaga dicho valor hacia atrás para así poder ajustar los pesos de la red, con el fin de reducir el error hasta alcanzar un valor razonable.

La optimización de los pesos de la red se lleva a cabo mediante la técnica del descenso del gradiente en cada neurona. En los modelos que se han desarrollado se implementa la optimización de Adam, que es un método estocástico de descenso de gradiente que se basa en la estimación adaptativa de los momentos de primer y segundo orden de cada neurona.

No obstante, existen una serie de problemas relacionados con el descenso del gradiente. El algoritmo de retropropagación funciona yendo de la capa de salida a la de entrada, propagando el gradiente de error por el camino. Una vez que el algoritmo ha calculado el gradiente de la función de coste con respecto a cada parámetro de la red, utiliza estos gradientes para actualizar cada parámetro con un Paso de descenso de gradiente. Por desgracia, los gradientes suelen ser cada vez más pequeños a medida que el algoritmo avanza hacia las capas inferiores. Como resultado, la actualización de la pendiente de gradiente

deja los pesos de conexión de las capas inferiores prácticamente sin cambios, y el entrenamiento nunca converge a una buena solución. Esto es lo que llamamos el problema de los gradientes de fuga.

En algunos casos, puede ocurrir lo contrario puede ocurrir lo contrario: los gradientes pueden crecer más y más hasta que las capas grandes actualizaciones de peso y el algoritmo diverge. Este es el problema de los gradientes explosivos, que aparece en las redes neuronales Recurrentes. En términos más generales, las redes neuronales profundas sufren de gradientes inestables; diferentes capas pueden aprender a velocidades muy diferentes.

Para garantizar que los problemas de desaparición/explosión de gradientes no aparezcan durante el entrenamiento suele ser muy recomendable utilizar la técnica de Batch-normalization. Ésta técnica consiste en añadir una operación en el modelo justo antes o después de la función de activación de cada capa oculta. Esta operación simplemente centra y normaliza cada entrada, y luego escala y desplaza el resultado utilizando dos nuevos vectores de parámetros por capa: uno para la escala el otro para el desplazamiento. En otras palabras, la operación permite al modelo aprender la escala y la media óptimas de cada una de las entradas de la capa.

En muchos casos, si se añade una capa Batch-Normalization como primera capa de la red neuronal, no es necesario estandarizar el conjunto; la capa Batch-Normalization lo hará automáticamente (aunque de manera aproximada, ya que sólo mira un lote a la vez, y también puede reescalar y desplazar cada característica de entrada). Para centrar y normalizar las entradas, el algoritmo necesita estimar la media y la desviación estándar de cada entrada. Lo hace evaluando la media y la desviación estándar de la entrada sobre el mini-batch actual. Así que durante el entrenamiento, la capa Batch-Normalization estandariza sus entradas, y luego las reescala y compensa.

El momento de la prueba no resulta tan sencillo. De hecho, puede ser necesario hacer predicciones para instancias individuales en lugar de para lotes de instancias: en este caso, no se tendrá forma de calcular la media y la desviación estándar de cada entrada. Además, incluso teniendo un lote de instancias, puede que sea demasiado pequeño, o las instancias pueden no estar independiente e idénticamente distribuidas, por lo que el cálculo de las estadísticas sobre el lote del lote no sería fiable.

## Técnicas de regularización

En los modelos presentados en este trabajo se utilizan diferentes técnicas de regularización. Las redes neuronales tienen capacidad para entrenar millones de parámetros, de hecho en el modelo 5 propuesto se alcanzan los 1.150.053 parámetros. De esta forma, la red tiene gran libertad para ajustarse a una enorme variedad de conjuntos de datos complejos. No obstante, esta misma flexibilidad facilita también que la red sobreajuste el conjunto de datos de entrenamiento (lo que se conoce como *overfitting*).

Es aquí donde entra en juego la regularización. La técnica de regularización más conoci-

da es la función dropout. Ésta elimina un número aleatorio de inputs en cada actualización durante el tiempo de entrenamiento. Cuando se aplica una técnica de regularización como ésta, las neuronas no pueden coadaptarse con sus neuronas vecinas; tienen que ser lo más útiles posible por sí solas. Tampoco pueden depender excesivamente de unas pocas neuronas de entrada; deben prestar atención a cada una de sus neuronas de entrada. Acaban siendo menos sensibles a los pequeños cambios en las entradas. Al final, se consigue una red más robusta que generaliza mejor. También se utiliza en los modelos la regularización L2, que restringe los pesos de conexión de una red neuronal.

## **Tamaño del batch**

Por último, en muchas ocasiones resulta muy práctico dividir la cantidad de datos de entrada que tomará la red en pequeños lotes, pues disminuye el tiempo de entrenamiento del modelo. A esto se le conoce como fijar el tamaño del batch (el lote), y es un factor muy a tener en cuenta si se quiere que el modelo sea aplicable multitud de veces. Por ejemplo, para predecir precios intradía será más adecuado un modelo que tarde 1 minuto en entrenarse que otro que tarde 15 minutos, aunque consiga peores predicciones. Aún así, esta técnica hace que se generen más actualizaciones de parámetros internos dentro de la red, y al final empeora el aprendizaje en comparación con el caso en el que se entrenara el conjunto de entrenamiento directamente.

# Capítulo 4

## Objetivos y metodología

### 4.1. Objetivo general

En el presente trabajo se pretende diseñar un sistema capaz de predecir el precio de cotización exacto de las acciones de Apple, Google, Microsoft y Amazon. Este problema de modelización se abarca desde el punto de vista de la economía, pues desde es allí donde surge la inquietud que motiva a llevar a cabo este estudio. Por lo tanto, las mismas motivaciones, la delimitación del marco de aplicación del modelo y los resultados que se esperan obtener se definirán en términos económicos. Para ello se utilizarán técnicas de Inteligencia Artificial.

### 4.2. Objetivos específicos

Los puntos que se abarcan en este estudio son los siguientes:

- Revisión de la literatura existente en relación con las técnicas de análisis técnico y fundamental del mercado de renta variable y el desarrollo de modelos predictivos de precios.
- Preparación de un dataset pre-existente y exploración de la serie temporal
- Implementación de un primer modelo sencillo para la predicción de los precios de cierre de las acciones de Apple.
- A partir del modelo anterior, desarrollar distintos modelos y seleccionar aquél que minimice el error obtenido en la predicción.
- Implementar el mejor modelo para predecir los precios de las acciones de Amazon, Google y Microsoft.
- Comparar la aplicabilidad del modelo utilizando distintos datasets de entrenamiento y de validación.

### 4.3. Metodología

Para la realización de este trabajo se ha utilizado como punto de partida un script obtenido de Kaggle. En él se exportan los datos que se van a utilizar y se desarrolla un modelo muy simple, el que será el modelo 1 de este trabajo y del que derivarán los siguientes tres propuestos. Aclarar que la integridad del trabajo es obra original de la autora.

Las simulaciones de los modelos basados en redes LSTM se han programado desde la plataforma Google Colaboratory. Se trata de una extensión de Google donde se pueden programar trabajos de aprendizaje y desarrollo basados en Aprendizaje Automático, todo haciendo uso desde la nube. Se ha utilizado el acelerador de hardware GPU como entorno de ejecución del programa.

Las redes neuronales han sido programadas con la API secuencial Keras, de la versión 2.5.0 de TensorFlow. En el código, dentro de la construcción de la red neuronal no se han especificado las funciones de activación de ninguna de las capas internas. Tampoco en el entrenamiento de las redes se ha fijado ningún tamaño del batch. Esta configuración la determinará automáticamente Keras.

En la figura 4.1 se muestra un esquema del procedimiento que se ha seguido para modelizar la predicción de precios de las cuatro tecnológicas multinacionales. Primero se toman los datos de las cotizaciones de las acciones. De ahí a partir de un análisis fundamental se extraen las características que servirán como input del modelo. Después se hace una separación del input entre datos de validación y datos de entrenamiento. A continuación se formula el modelo, que consistirá en una red LSTM. Por último, se hará un test de la actuación del modelo, y se tomará como aceptable en función del error entre los precios que prediga el modelo y los precios del conjunto de validación.

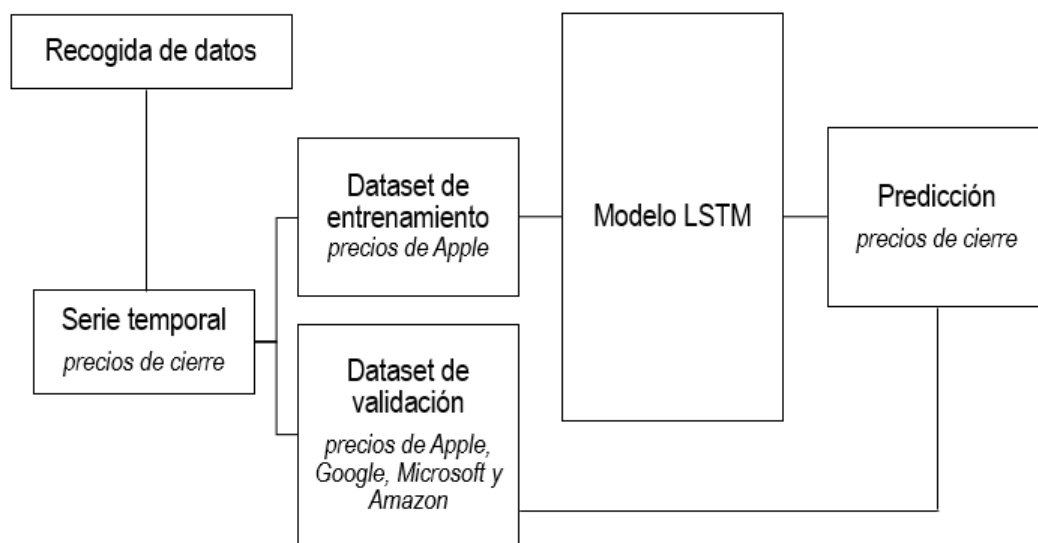


Figura 4.1: Esquema del proceso de modelización.

# Capítulo 5

## Solución propuesta

### 5.1. Análisis exploratorio de los datos

Si bien conocer cómo se ajustan los precios del mercado de renta variable y de qué depende su fluctuación en el tiempo es una cuestión nada sencilla, en este apartado se tratará de justificar por qué en sí mismos los precios responden a esta pregunta. De esta manera, cuando se diseñe el modelo en los siguientes pasos sólo se observará como variable en función del tiempo el precio de cierre diario de las acciones. Como en la gran mayoría de los estudios de naturaleza económica, este estudio se abarcará desde un punto de vista cualitativo.

Como bien hemos dicho ya, la muestra de trabajo son datos sobre las cotizaciones de las acciones de Apple, Google, Microsoft y Amazon, empresas tecnológicas que cotizan en el mercado de renta variable de Estados Unidos. De ahí se obtienen para cada día los precios máximos, mínimos, de apertura y de cierre de las acciones, medidos en dólares estadounidenses (USD), y también el volumen de operaciones. También se añade el precio de cierre ajustado, que incorpora al precio el valor contable de ciertas operaciones o actuaciones que realice la empresa y que influyen en el valor real de la acción, tales como el reparto de dividendos o la división de acciones.

Se obtienen y preparan los datos utilizando las librerías *pandas* y *numpy*. En la figura 5.1 se presentan los precios de cierre de las cuatro empresas desde 2010.

Amazon encabeza en precios sus acciones con una enorme ventaja, seguida con mucha distancia de Google. Microsoft y Amazon cotizan a precios muy inferiores. Esto no quiere decir que sea la mejor oportunidad de inversión, porque al fin y al cabo lo que se busca son inversiones cuyo precio prometa crecer. En la figura 5.2 se utiliza una escala logarítmica, así la amplitud de una variación del precio en el tiempo (en el eje vertical) representa el mismo porcentaje de variación para precios bajos como para precios altos del activo. Se puede decir entonces que todas las acciones han crecido a un ritmo parecido, y que desde el punto de vista de la inversión generan rendimientos semejantes.

A juzgar por las gráficas de los precios, uno podría pensar además que Amazon es la empresa con mayores beneficios de las cuatro. Sin embargo sucede todo lo contrario,

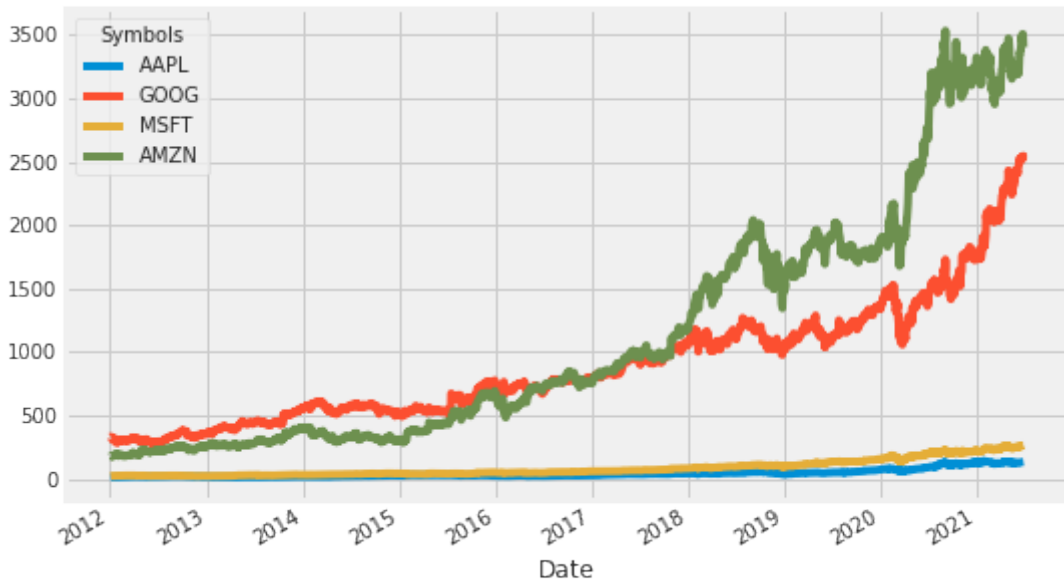


Figura 5.1: Precios de cierre (USD).

Amazon generó 21.331 millones de dólares de beneficios en el año 2020, frente a los 40.269 de Google, 44.281 de Microsoft y 57.411 de Apple. En la tabla 5.3 se resumen estos datos junto con la cifra del Activo de cada empresa, es decir, el valor contable del conjunto de bienes de los que son propietarias para el desarrollo de su actividad económica (maquinaria, equipos informáticos, terrenos, filiales, deudas a cobrar, dinero en circulación...). Sería necesario observar esta cantidad de manera desglosada para comprender en detalle cuáles son los puntos fuertes de la empresa, en qué basa su modelo de negocio y cómo de eficiente está desarrollando su actividad económica una empresa. No obstante, conocer la cifra del Activo sí basta para realizar la siguiente puntualización general: el tamaño de negocio de las cuatro empresas tecnologías y su capacidad de generar ganancias es muy similar.

En la tabla 5.3 también se recogen los incrementos porcentuales que ha experimentado el beneficio neto de cada empresa el 2020 comparado con el año anterior, y es aquí donde aparece otra pista fundamental de cómo se guían los precios en el mercado. Amazon alcanzó a finales de 2020 un beneficio un 84 % superior al que obtuvo en 2019, un sorprendente resultado comparado con las demás empresas. Por poner un ejemplo, si este año Amazon mantuviera este ritmo de crecimiento en el año 2021 generaría un beneficio neto de 39.249 millones de dólares, mientras que Google para incrementar también sus beneficios en un 84 % debería generar un beneficio neto de 74.095 millones de dólares. Volviendo a la afirmación de que los precios de cotización miden expectativas de crecimiento de las empresas, es razonable pensar que Amazon experimentará un mayor crecimiento en el precio que las demás empresas puesto que los beneficios anuales que liquida son bastante pequeños en comparación, y que lo más probable es que a Apple y Microsoft les cueste superar sus ganancias actuales, y en consecuencia se espera que su precio de cotización se mantenga estable.

Es importante tener en cuenta también que la crisis sanitaria originada por la pan-



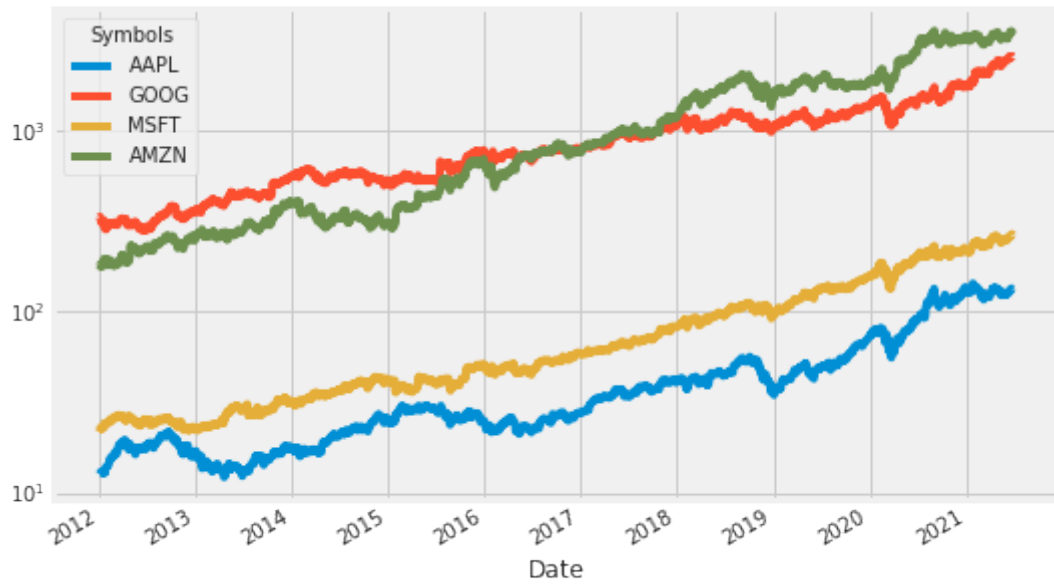


Figura 5.2: Precios de cierre en escala logarítmica (USD).

<i>Cifras en millones de dólares (\$)</i>	<b>Apple</b>	<b>Google</b>	<b>Microsoft</b>	<b>Amazon</b>
<b>Patrimonio</b>	323.888	319.616	301.311	321.195
<b>Beneficio neto 2020</b>	57.411	40.269	44.281	21.331
<b>Diferencia con el beneficio neto 2019</b>	4%	17%	13%	84%

Figura 5.3: Datos financieros sobre la actividad económica de las empresas.

demia del COVID-19 ha agitado las bolsas mundiales este último año. Por una parte, los efectos económicos que ha causado en las empresas han sido muy variados, se ha visto tanto empresas fuertes incapaces de impedir que sus ventas cayeran, como empresas que han captado nueva clientela y han tenido que duplicar su plantilla de trabajadores.

En particular, las empresas tecnológicas estudiadas no se han visto afectadas negativamente por la COVID-19, más bien todo lo contrario. En la figura 5.2 se detecta muy bien el inicio de la crisis sanitaria, al inicio del año 2020 se produce una gran caída simétrica del precio de las acciones de las tecnológicas, que se prolonga unos escasos meses. Enseguida los precios se recuperan e incluso siguen creciendo a mayor ritmo que antes. Destaca la asombrosa subida del precio de cotización de Amazon (figura 5.1), que posteriormente se estanca el resto del año.

Parte de esta extraordinaria recuperación se puede entender por el modelo de negocio de las empresas, que el mercado rápidamente interpretó como una apuesta segura ya que las empresas no verían gravemente interrumpidas sus ventas, al menos en comparación con el resto de empresas que cotizan en bolsa. También es muy importante la dimensión de las empresas, cuya actividad económica se desarrolla a escala multinacional. Es verdad que las empresas están muy expuestas al riesgo económico derivado de los tipos de cambio en los países en los que están presentes. Pero también es verdad que la situación crítica que se vivió en cada país no se dió con la misma gravedad de manera simultánea:

se produjo de manera escalonada.

Por otra parte, la alta volatilidad y la incertidumbre que tienen hoy todavía las bolsas es un escenario ideal para los inversores a corto plazo, que pueden llegar a reunir miles de euros en un solo día. Así, cada vez son más las personas que sin tener la formación necesaria se aventuran a invertir en bolsa con la esperanza de ganar dinero rápido. Por tanto, es razonable sostener que el comportamiento de los precios se encuentra en un momento extraordinario, y debería esperarse que las predicciones obtenidas representaran los cambios bruscos que están teniendo los precios.

Como aproximación a los resultados que se esperan obtener con el desarrollo del modelo, se han calculado en el último trimestre las medias móviles de los precios de cierre ajustados en periodos de 5 y 10 días (figura 5.4). Dentro del análisis técnico son muy utilizadas para identificar tendencias de los precios a corto plazo, que serán muy útiles partiendo de la premisa de que es más probable que una tendencia continúe a que retroceda [19]. En las cuatro empresas la media móvil de 5 periodos supera en algún punto la de 10 periodos, lo que indica un periodo de tendencia alcista, es decir, se estima que en los próximos 10 días el precio será superior al precio de la fecha actual. La desventaja de este indicador técnico es que el cambio de tendencia en el precio se produce siempre unos días antes a la señal que emite, es decir, aunque podemos identificar que en los últimos 15 días la tendencia del precio es alcista, los 15 días siguientes bien podrían no serlo. En un caso así, un analista financiero debería respaldar sus sospechas documentándose sobre las últimas noticias sobre el sector económico y las empresas afectadas, es decir, basándose en el análisis fundamental del mercado.

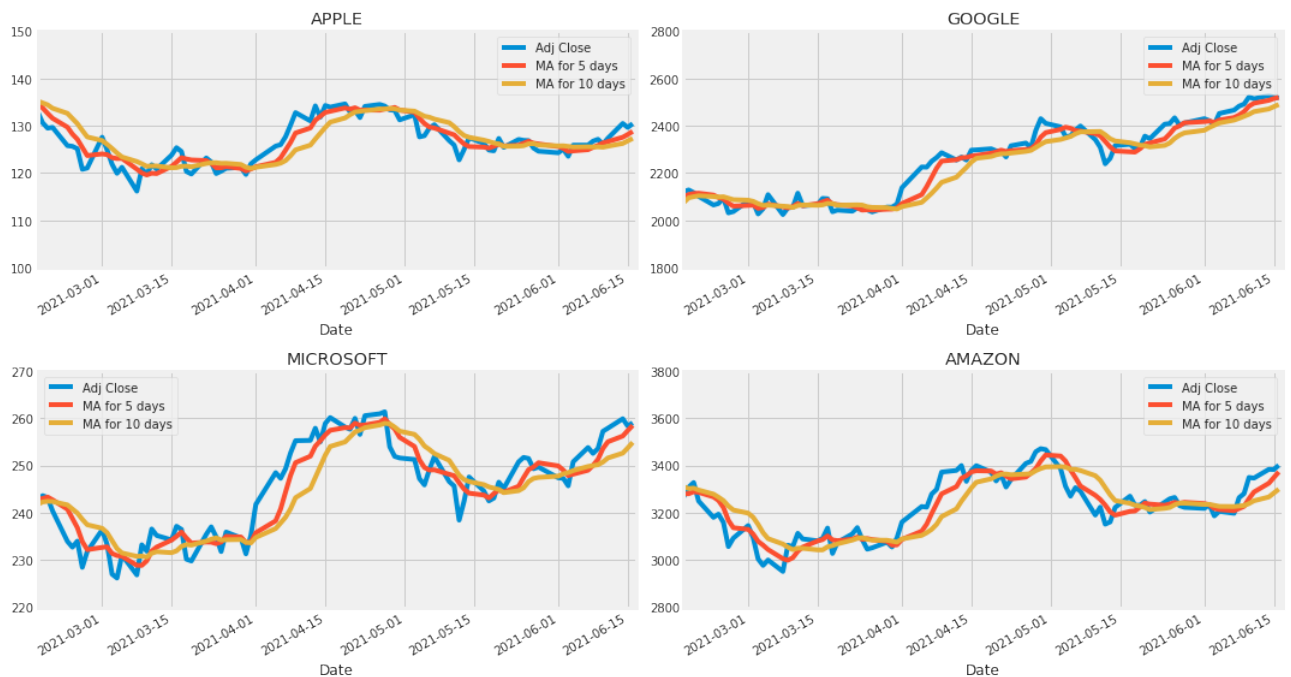


Figura 5.4: Medias móviles de 5 y 10 periodos (USD).

Para conocer más sobre la formación de los precios puede ser buena idea analizar si presentan estacionalidad (stationarity en inglés). Una serie temporal es estacionaria cuando sus propiedades no dependen del periodo en el que son observadas. Normalmente las series temporales que presentan tendencias, o que dependen de la época de año no son estacionarias. Los precios de cierre sí presentan una clara tendencia alcista. En la figura 5.5 se utilizan gráficos ACF y PACF para detectar si los precios de cierre de las acciones de las cuatro empresas presentan estacionalidad. Si las series temporales fueran estacionarias el valor de ACF caería a cero relativamente pronto. No es el caso que reflejan los gráficos, por lo que se concluye que los precios de cierre no son estacionarios, y parte de la fluctuación de su valor en el tiempo estará motivada por tendencias o la época de año.

Una pregunta que puede surgir a raíz de esta cuestión es si, a la hora de diseñar el modelo, puede ser un problema que los precios históricos que estamos observando estén todos dentro de una larga tendencia alcista. Quizás las predicciones sean demasiado optimistas. Podría ser que exista un exceso de optimismo en el mercado, fenómeno que recibe el nombre de “*apalancamiento financiero*”. Sin embargo, lo más normal es que el curso del precio siga un ritmo creciente, es decir, que exista inflación. La inflación es un aumento generalizado en los precios de los bienes y servicios de una economía durante un periodo de tiempo [2]. Se trata de otro fenómeno que ocurre en casi todos los países, de hecho, los bancos centrales siempre tratan de que en su país haya algo de inflación, normalmente entre el dos y el tres por ciento.

En análisis técnico se controla la dirección que lleva la inflación, puntos fuertes o débiles de la economía: crecimiento en los precios de las mercancías apuntan a una economía más fuerte y a una mayor presión inflacionaria. Unos precios descendientes de las mercancías habitualmente avisan que la economía se ralentiza junto con la inflación. También la dirección del dólar norteamericano y las monedas es una guía anticipada de la fuerza y debilidad de las respectivas economías globales. Lo que resulta más impresionante es el hecho de que las tendencias en estos mercados de futuros generalmente aparecen mucho antes de quedar reflejadas en los indicadores económicos tradicionales que se dan a conocer mensual o trimestralmente, y que nos informan sobre lo que ya ha ocurrido.

Dentro del capítulo 7 de discusión de los resultados se tendrá en cuenta otra variable que se obtiene como combinación lineal del precio de cierre: el rendimiento de la inversión. El rendimiento de la inversión, denotado por  $i_{t-n,t}$ , es el incremento porcentual que sufre el precio de la acción  $p_t$  comparado con su precio  $n$  días anterior, denotado por  $p_{t-n}$ .

$$i_{t-n,t} = \frac{p_t}{p_{t-n}} - 1.$$

El rendimiento o la rentabilidad diaria de una acción ( $i_{t-1,t}$ ) es el incremento porcentual del precio con respecto al precio del día anterior  $p_{t-1}$ . Se ha utilizado la librería pandas para su cálculo. La mayoría de los trabajos revisados de la bibliografía que utilizan modelos basados en IA utilizan esta variable como variable de entrenamiento (input) y también como vector de salida (output). Se discutirá este tema en este capítulo,

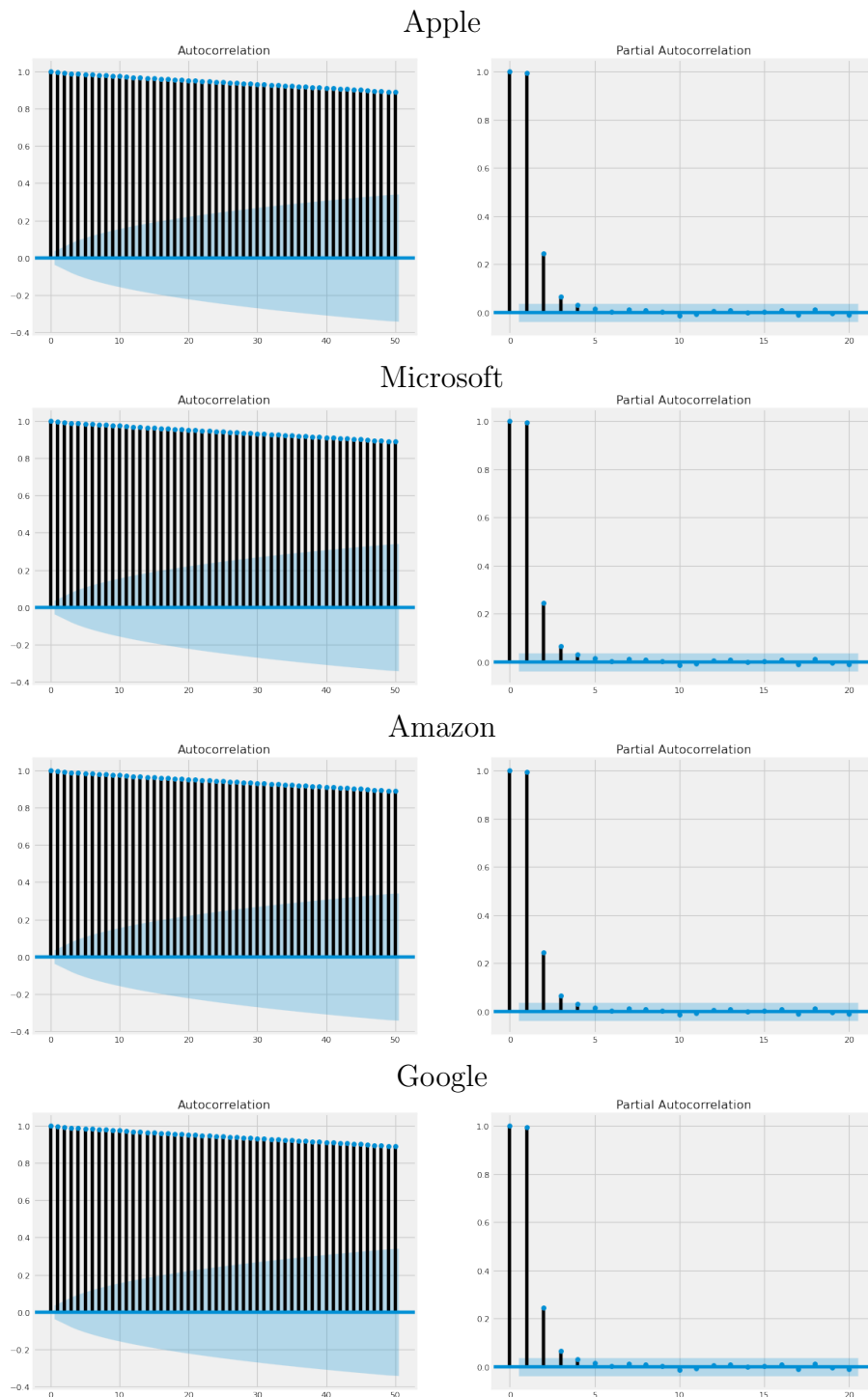


Figura 5.5: Autocorrelación (ACF) y autocorrelación parcial (PACF) de los precios de cierre diferenciado por empresas.

en la selección de las características del dataset.

En el anexo 8.2 se resumen los estadísticos de los rendimientos que han generado las cuatro acciones en el último año. Comparando entre las diferentes empresas, las acciones de Apple y Google crecen más rápido. Las razones pueden ser que las empresas tienen más ventas anuales, mayor margen de beneficio con respecto a sus costes, o una mejor gestión financiera de sus deudas patrimoniales. Esta superioridad puede deberse también a razones de especulación, si es que el mercado percibe que el precio de las acciones va a subir porque existan señales favorables sobre las empresas en la prensa, en las redes sociales o en cualquier otro medio de comunicación. Por otro lado, comparando las desviaciones típicas de las diferentes inversiones, también se extrae que Apple es la más volátil, y esto va a ser clave para elegirla como muestra para el diseño del modelo.

En la figura 5.6 se muestra un diagrama de correlación entre los rendimientos de las acciones de Apple, Google, Microsoft y Amazon en el último año. En todos los casos, la correlación entre los rendimientos diarios de los diferentes activos es positiva. Con esto lo que se quiere es buscar un rastro de la tendencia alcista de los precios que tienen las empresas tecnológicas, y buscar señales de interdependencias entre ellas. El diagrama muestra unas cifras razonables, las que cabría esperarse de un sector en crecimiento, en plena tendencia alcista y donde no hay competencia entre las empresas, sino que su actividad económica es lo suficientemente diferente para que se pueda dar la complementariedad entre sus negocios. No obstante, ni estas afirmaciones ni la existencia de correlación entre los rendimientos de las empresa implica necesariamente una relación de causalidad.

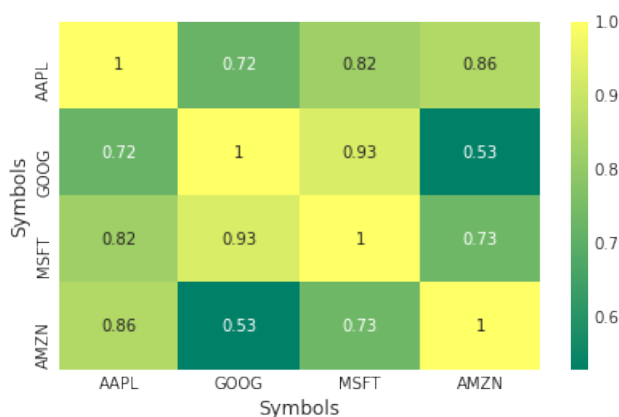


Figura 5.6: Diagrama de correlación de precios de cierre.

Para cerrar esta sección, se establece una comparación de las cuatro acciones teniendo en cuenta la rentabilidad esperada y el riesgo asociado a la inversión en el último año (figura 5.7). Será necesario cuantificar el riesgo de las acciones. En este caso para medir el riesgo se comparan los rendimientos diarios esperados con sus desviaciones estándar asociadas. Apple es la más volátil, como decíamos antes. Microsoft es la acción más segura. Google es la acción más atractiva desde el punto de vista riesgo-beneficio, pues es la que genera mayores ganancias de inversión y a un riesgo muy pequeño en comparación con

las demás, sólo superior al de las acciones de Microsoft.

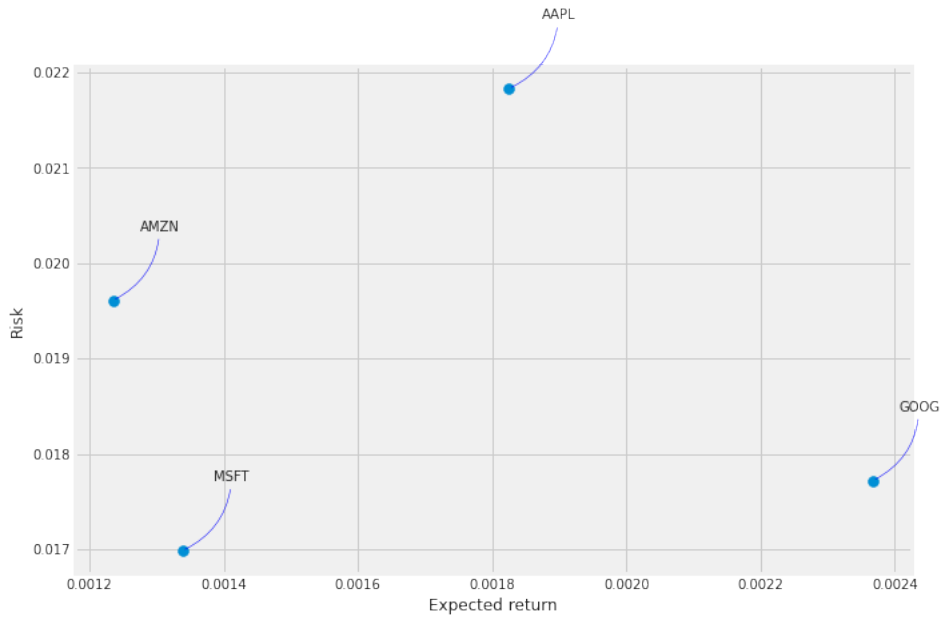


Figura 5.7: Rentabilidad - riesgo de las acciones.

## 5.2. Selección de las características del dataset

El dataset se compone de datos históricos desde el 01-01-2010 hasta la fecha 07-07-2021, extraídos de *Yahoo Finanzas* (ver anexo 8.1). Recoge los datos de cotización de 2899 días distintos, que sumados no llegan a los 8 años de duración. Para cada año el dataset contiene datos de menos días de los que tiene un año natural debido a que el mercado estadounidense Nasdaq cierra determinados días, es decir, no se realizan operaciones de compra y venta y, por lo tanto, los datos de cotización no cambian.

Las redes neuronales con aprendizaje profundo pueden tratar cantidades ingentes de datos, pero esto no tiene por qué ser una práctica recomendable. Desde el punto de vista de la computación, cuantos más larga sea la serie temporal que se le proporcione a la red neuronal para entrenar, mejor aprenderá. Sin embargo, tener en cuenta muchas variables distintas puede introducir consigo ruido blanco al modelo, algo no deseable ya que disminuirá la calidad de la predicción [13].

Como hemos dicho, el precio contiene información muy completa sobre la empresa. Por contra, los modelos [4], [8], [13], [16], [21], [22] basados en LSTM utilizan como entrada de la red neuronal los rendimientos diarios que generan las acciones, y predicen rendimientos futuros a corto plazo. La ventaja de utilizar los rendimientos para entrenar el modelo es que una vez realizada la predicción se pueden extraer estadísticos sobre los rendimientos que generará cada acción, y así comparar los beneficios que se consiguen con cada una. Se puede decir que el problema se mira desde el punto de vista del inversor, que busca

obtener beneficios. Esto es porque estos estudios tienen como propósito automatizar decisiones de inversión para optimizar las ganancias del inversor.

Sin embargo, las estrategias de inversión (los algoritmos) que proponen estos estudios son muy simples. Básicamente consisten en comprar cuando se prediga que al día siguiente va a subir el precio, y vender cuando se prediga una bajada del precio. La estrategia se define entonces en función del precio del día siguiente, pero esta no es la forma en la que operan los inversores a corto plazo, que lo que les interesa es localizar en un rango de tiempo de al menos dos semanas los máximos y los mínimos locales del precio y estimar qué valor tendrá al finalizar el periodo considerado. En principio, un inversor racional para optimizar la rentabilidad de su inversión compraría cuando el precio caiga a un mínimo local y vendería cuando alcanzara un máximo local.

Pero es que además nunca hay garantía completa de que el precio se vaya a comportar de una manera, y la estrategia de inversión tiene que incorporar otros aspectos que contemplen el riesgo. Un inversor normalmente basa su estrategia en la cantidad de ganancias acumuladas y en la cantidad de pérdidas que esté dispuesto a asumir de acuerdo a su perfil de riesgo, como se suele decir. La planificación de una estrategia de inversión es muy personal, y habrá personas que se encuentren cómodas asumiendo un 20 % de riesgo de perder su inversión inicial y habrá otras con mayor aversión al riesgo que busquen activos con menos de un 2 % de riesgo. Así, un inversor compra una acción cuando espera que vaya a subir de precio hasta cierto valor, y mantiene la inversión y sólo vende cuando alcance ese precio o cuando el precio caiga por debajo de cierto valor umbral, limitando sus pérdidas. Sabiendo la complicación de esta tarea, nos conformamos con conocer la tendencia del precio minimizando el error de estimación.

Aclarar que se utilizarán los precios de cierre, como bien se habrían podido utilizar de manera indistinta los precios de apertura del mercado.

### **5.3. Separación del dataset de entrenamiento y validación**

Para la selección del modelo se utilizarán solamente datos de Apple durante el entrenamiento. La razón reside en que la acción más volátil, es decir, sus precios de cotización fluctúan de manera más acelerada y con mayor apuntamiento. Parece razonable pensar que los precios de Apple son más sensibles, o dependen de más variables, y que la inferencia del aprendizaje de la red resultará más efectiva en el siguiente orden: entrenar la red neuronal con precios más cambiantes y extrapolar lo aprendido a precios menos cambiantes. Es decir, será más fácil pasar de precios más cambiantes a menos, que de precios más estables a precios más bursátiles. De todas formas, una vez seleccionado el mejor modelo se probará a entrenarlo con diferentes datasets. Hay que tener en cuenta además que Apple es, de las cuatro empresas, aquella cuya acción cotiza a menor precio.

Se hace una división del dataset en un bloque de entrenamiento, con el primer 85 %

de los datos (2464 días), y un bloque de validación con el 15 % restante (435 días, desde el 16 de octubre de 2019 hasta el 7 de julio de 2021). El primer conjunto se utiliza para entrenar la red y para ajustar iterativamente sus parámetros de forma que la función de pérdida se minimice. Hay que tener presente que el conjunto de validación contendrá todo el periodo de tiempo durante el que aconteció (y continúa transcurriendo) el episodio de la crisis sanitaria y la pandemia del COVID-19.

Bien es cierto que para comprobar la actuación de un modelo se suele utilizar proporciones como 80 % de datos de entrenamiento y 20 % de datos de validación. Ésto se hace para evitar que el modelo tenga problemas de *overfitting*, esto es, que el modelo aprenda muy bien las características de entrenamiento pero la *IA* no sea capaz de extrapolar su actuación a nuevos entornos fuera de los datos de entrenamiento, en nuestro caso nuevas tendencias de precios. En definitiva, pierde capacidad de generalización. No obstante, se realizará una simulación del modelo con una configuración 95/5 para conseguir una predicción a corto-medio plazo y comparar los resultados con la prueba anterior. Utilizar un dataset de validación del 5 % equivale a predecir el precio de casi cinco meses en adelante. Además, existe el problema de que los errores son acumulativos, lo que se explicará en los párrafos siguientes. Por estos motivos se probará la configuración 95/5 solamente con la intención de aclarar este punto.

Durante el entrenamiento, y de forma similar a las Redes Recurrentes, en la red LSTM los pesos y los términos de sesgo se ajustan de forma que minimicen la pérdida de la función objetivo especificada en las muestras de entrenamiento. La función de pérdida será el error cuadrático medio (MSE). Por comparación, se obtendrá también el error absoluto medio (MAE) y se representará en una gráfica la evolución de las dos métricas en el capítulo de resultados. Además, para cada modelo se extrae al final del entrenamiento el valor de la raíz del error cuadrático medio (RMSE). Este índice de rendimiento mostrará una estimación del residuo entre el valor real ( $y_i$ ) y el valor predicho por el modelo ( $o_i$ ), como se indica en la ecuación siguiente

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - o_i)^2}$$

donde  $n$  es el número de datos que tiene la muestra.

Por otro lado, y de acuerdo con los resultados obtenidos, el método de optimización que mejor funciona es el *Adam*, que se ha comparado con el optimizador *SGD*.

En la parte de inferencia de los resultados del modelo se utiliza una ventana móvil de datos de 60 días para predecir el precio del día siguiente en cada paso de tiempo, por lo que la validación del modelo finaliza una vez que la ventana haya recorrido el dataset de entrenamiento en 2200 pasos de tiempo aproximadamente. Esto quiere decir que genera una nueva red neuronal al final de cada día de negociación, por lo que se define un nuevo conjunto de pesos utilizando un nuevo conjunto de datos de entrenamiento y validación. En cada ventana se predice el precio de la acción del primer día de la ventana



siguiente. Esto implica que en cada paso de tiempo se utilizará el precio estimado el día anterior para realizar la nueva predicción del precio del día siguiente. Por lo tanto, se perderá precisión cuantos más pasos de tiempo se produzcan en la predicción, ya que los errores son acumulativos.

No obstante, utilizar la técnica de ventanas móviles también puede ser ventajosa, en tanto que la función pérdida contendrá un término para la salida de la RNN en todos y cada uno de los pasos de tiempo, no sólo la salida en el último paso de tiempo. Esto significa que habrá muchos más gradientes de error fluyendo a través del modelo, y no tendrán que fluir sólo a través del tiempo; también fluirán desde la salida de cada paso de tiempo. Esto estabilizará y acelerará el entrenamiento.

## 5.4. El vector de entrada o características

Como entrada del modelo se utilizan los precios de cierre no ajustados históricos, denotados por  $x_t$  (ver figura 3.2). Para que el modelo se pueda entrenar con cualquier vector de precios de cierre  $x_t$ , no se le especificará ningún tamaño de dimensión. Para unificar los datos, se han normalizado los precios en un rango de 0 a 1.

## 5.5. El vector de salida

El vector de salida de la red neuronal, denotado por  $o_t$ , reunirá el conjunto de predicciones del precio de cierre del día siguiente a la fecha actual (ver figura 3.2). Cada elemento del vector se utilizará después dentro del vector de entrada  $x_{t+1}$  como último precio de cierre de la siguiente ventana de datos.

Merece la pena recalcar que el vector es de una dimensión, porque lo que interesa es conocer el valor numérico de los precios. Es muy frecuente que a la hora de modelizar un problema utilizando redes neuronales se caracterice al vector de salida con más de una dimensión, planteando el modelo como herramienta que da solución a un problema de clasificación. Por ejemplo, se podría construir un modelo que clasificara la salida en dos clases: si el precio del día siguiente es mayor al precio del día actual el output es de clase 1, y si el precio es menor el output es de clase 2 [4], [21], [22]. También podría crearse una tercera clase donde se clasificaran los precios que no han cambiado de un día para otro, aunque es una situación más que poco probable [13].

## 5.6. Modelos propuestos basados en redes LSTM

En esta sección se presentan los modelos para la predicción de los precios de cierre de Apple. Se toma como punto de partida el modelo 1, y de la aplicación de una serie de modificaciones se formulan tres variantes del modelo inicial que mejoran la predicción

de los precios. Cada descripción comienza con un esquema de la arquitectura del modelo correspondiente (figuras 5.8, 5.9, 5.10, 5.11 y 5.12).

Antes de nada, hay que aclarar un denominador común de todos los modelos. Durante la programación se ha implementado en una línea de código una inicialización aleatoria de los datos fijando una semilla de 51. Así, cada vez que se ejecute el programa se generará el mismo conjunto de pruebas. Esto permite que al ejecutar el código se reproduzcan los mismos conjuntos de pruebas para todos los modelos. Así se asegura que los resultados que se obtengan de ejecutar el código sean convergentes a los que refleja el capítulo 6.

## Modelo 1

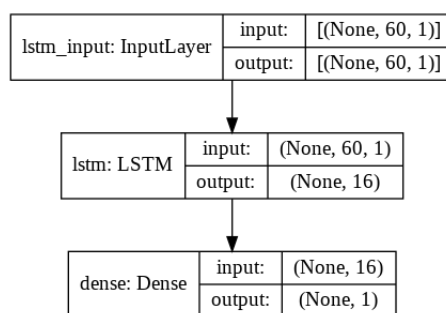


Figura 5.8: Arquitectura del modelo 1

La topología del modelo de partida es la siguiente:

- Primera capa LSTM con 16 neuronas, donde el vector de entrada a la capa es de dimensión (2404, 60, 1).
- Una capa de salida fully-connected con una neurona, que es el número de clases en las que se quiere clasificar el precio de cierre ajustado: su valor estimado.

Esta configuración produce 1.169 parámetros para la LSTM, lo que no nos da ni siquiera a guardar la relación de 2 ejemplos de entrenamiento por cada parámetro. En [5] se utiliza una proporción de 93 ejemplos de entrenamiento por parámetro. Un número elevado de observaciones por parámetro permite obtener estimaciones más robustas en caso de que los datos de entrenamiento sean tan ruidosos, y reduce el riesgo de sobreajuste u overfitting. En los modelos consecutivos se aumentará progresivamente el número de neuronas de las capas LSTM para abordar esta cuestión.

Las predicciones de este modelo son aceptables puesto que captan los días en los que la variación del precio cambia de ser creciente a decreciente, y viceversa. Es decir, la predicción logra localizar en la serie temporal los puntos de inflexión de la curva del precio. Sin embargo, aparecen tres problemas:

- los precios estimados quedan siempre por debajo de los precios de cierre reales

- a medida que se avanza en la predicción aumenta el error
- la predicción penaliza los picos a la alza y a la baja a los que llega el precio, que es quizás la parte más interesante de la predicción, es decir, las mejores oportunidades para comprar o vender acciones.

Sin embargo, la función de pérdida es muy baja desde el momento inicial y converge a 0 en el epoch 1 de entrenamiento (figura 6.2). Es muy probable que este modelo no se haya entrenado lo suficiente y los resultados no sean realistas, en el sentido de que el modelo no sepa extrapolar las características del dataset a nuevos datos. Viendo ésto, y con la pretensión de mejorar el entrenamiento, se propone un segundo modelo.

## Modelo 2

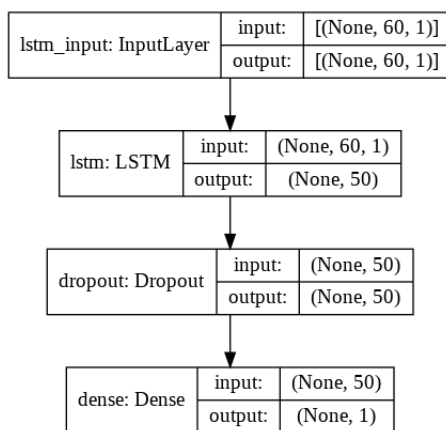


Figura 5.9: Arquitectura del modelo 2

En el modelo 2 se realizan dos modificaciones en la capa LSTM:

- Se incrementa el número de neuronas a 50.
- Se se aplica la regularización dropout con un factor de regularización de 0.25, lo que implica que se eliminará el 25 % de los inputs de entrada de la capa para cada ventana móvil de datos.

El objetivo de este ajuste es generalizar más la aplicabilidad del modelo a nuevos datos. Se ha observado que para valores mayores de dropout derivan en un empeoramiento de la actuación del modelo.

Después de estas modificaciones, el modelo soluciona ligeramente los problemas que presentaba el modelo 1 . Merece la pena destacar la potencia que tiene esta herramienta: aunque los cambios han sido muy ligeros, en este modelo la red entrena 10.451 parámetros distintos, casi diez veces más que la red del modelo inicial.

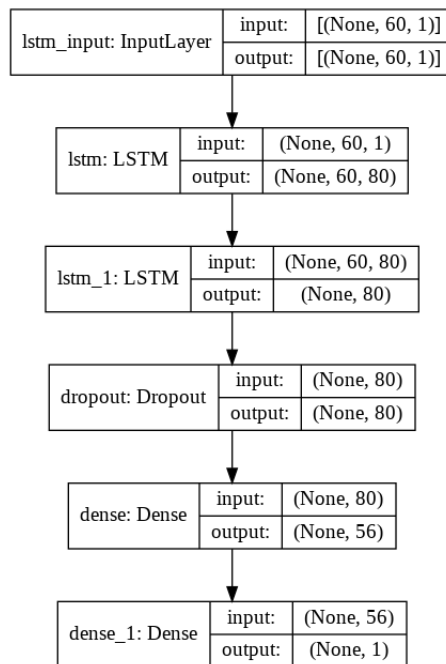


Figura 5.10: Arquitectura del modelo 3

### Modelo 3

En este modelo se añaden dos capas nuevas. En primer lugar, se añade una nueva capa LSTM después de la primera y se aumenta a 80 el número de neuronas de las dos capas LSTM. En segundo lugar, después de la activación dropout y antes de la última capa fully-connected se añade otra capa de este mismo tipo, con una configuración de 56 neuronas.

El modelo pronostica precios futuros con menor error que los modelos anteriores, aunque las predicciones todavía no convergen los máximos y los mínimos locales. El siguiente paso será elaborar un modelo aún más complejo que consiga predicciones con menor error.

### Modelo 4

En este modelo las capas LSTM se configuran como capas bidireccionales de 150 neuronas cada una, por lo que habrá 300 valores de salida en cada capa bidireccional. En este caso, se quiso explorar la característica del precio como medida de expectativas futuras de rentabilidad. El primer mandamiento que debe seguir todo inversor es no dejarse eclipsar por las rentabilidades que haya generado una acción en el pasado, de hecho la primera lección que se enseña a los inversores novicios es la frase que se citaba en la introducción, que dice que “rentabilidades pasadas no garantizan rendimientos futuros”. Además de los cambios anteriores, se incrementa a 64 el número de neuronas de la primera capa fully-connected.

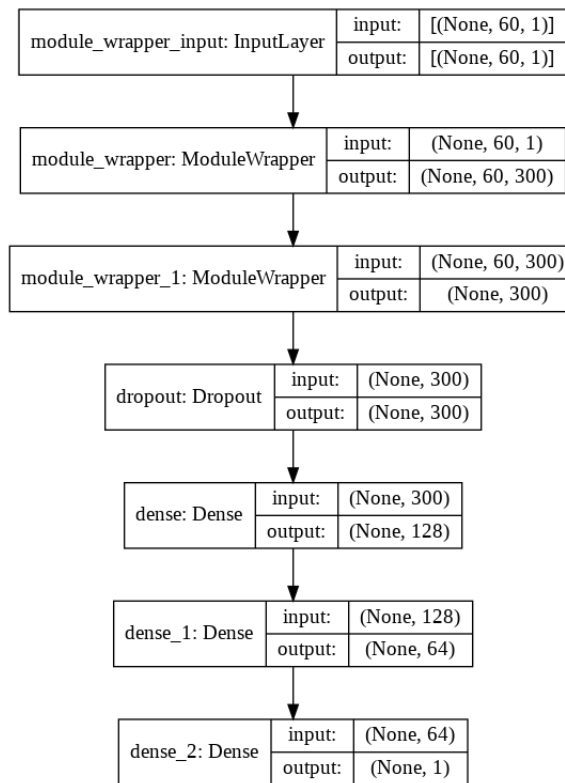


Figura 5.11: Arquitectura del modelo 4

## Modelo 5

En este último modelo se han utilizado nuevas técnicas de regularización. La estructura del modelo se vuelve más complicada que la de los modelos anteriores. Todas las modificaciones se realizan con el propósito de solucionar problemas relacionados con el overfitting. El modelo se estructura de la siguiente manera:

- Se prepara la entrada de la red LSTM utilizando la técnica de batch normalization.
- Primera capa LSTM bidireccional con 180 neuronas.
- Segunda capa LSTM bidireccional con 180 neuronas.
- Se aplica una regularización dropout de 0.25.
- Primera capa fully-connected con 256 neuronas.
- Segunda capa fully-connected con 64 neuronas.
- Se aplica la técnica de regularización L2.
- Última capa fully-connected con 1 neurona.

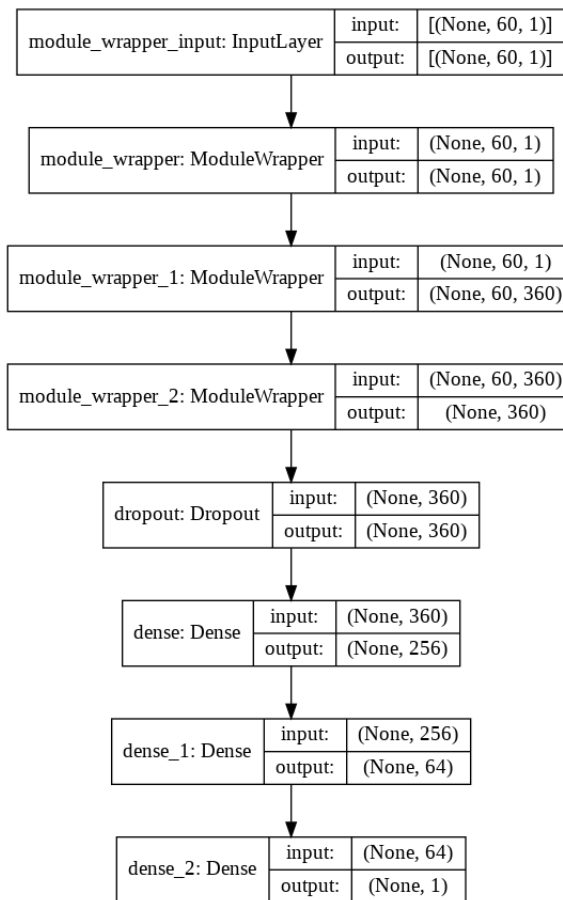


Figura 5.12: Arquitectura del modelo 5

## 5.7. Validación del modelo en el resto de empresas

Una vez seleccionado el modelo 3 como el que mejor predice los precios futuros de las acciones de Apple, se ha probado el mismo modelo para predecir los precios de las acciones de Microsoft, Amazon y Google. Para esto, se utilizará en cada caso los precios de cierre históricos de las acciones de la correspondiente empresa. También se ha querido probar si entrenando el modelo con los datos relativos a los precios de Apple hubiera sido de aplicación para las demás empresas, punto que tienen a su favor el trabajo [17]. En él diseña una red CNN con la que fue capaz de predecir los precios de cinco empresas diferentes de la Bolsa de Nueva York habiendo utilizado como dataset de entrenamiento los precios de una sola empresa de la Bolsa de la India. Esto fue posible porque ambos mercados de valores comparten algunas dinámicas internas comunes que el modelo consigue captar.

# Capítulo 6

## Resultados

En esta sección se muestran los resultados obtenidos en este estudio. Primero se hace una comparativa entre los cinco modelos planteados en el capítulo anterior con el propósito de seleccionar el más adecuado. En los gráficos de la figura 6.1 se representan las predicciones que realiza cada modelo de los precios de cierre de las acciones de Apple, comparadas con los precios de validación. En la figura 6.2 se representan las funciones de pérdida (medidas en SME) y el error medio absoluto (MAE) que se consiguen con cada uno. En la figura 6.3 se amplía la comparativa entre los modelos. Por un lado, la tabla refleja para cada modelo las mediciones del RMSE, el rendimiento estimado de la acción desde la primera fecha del dataset de validación hasta la última fecha, y el error de este rendimiento estimado comparado con el rendimiento real que tiene la acción durante el mismo periodo, estimación. Por otro lado, estas medidas se toman para una división del dataset de precios de las acciones de Apple de 85/15 (la que se ha utilizado en los modelos) y, a modo de comparación, se toman también para una división 95/5.

Una vez se ha escogido el modelo 3 como el que mejor aproxima el problema, se aplica al precio de las acciones de las otras tres empresas para ver los resultados de la validación. En la figura 6.4 se compara el RMSE que se obtiene en las predicciones con el modelo 3 pudiendo aplicarse de dos formas: o utilizando únicamente los precios de cierre de Apple como dataset de entrenamiento, o utilizando el mismo dataset de entrenamiento que de validación, es decir, utilizar precios pasados de una empresa para predecir los precios futuros de la misma empresa. Por último, la figura 6.5 refleja las predicciones finales de los precios de cierre de las acciones de Apple, Microsoft, Amazon y Google, utilizando como dataset de entrenamiento los precios históricos de Apple.

Hace falta destacar que las gráficas de la figura 6.5 tienen diferentes escalas de precios en el eje vertical. Por ejemplo, aunque parezca que el modelo predice mejor los precios de Google que los de Microsoft, los precios de Google fluctúan dentro de un intervalo de 1500 y 2500 dólares (\$), frente a los precios de Microsoft, que fluctúan dentro de un intervalo de 150 y 250 dólares (\$). Esto queda corroborado con los RMSE de la figura 6.4

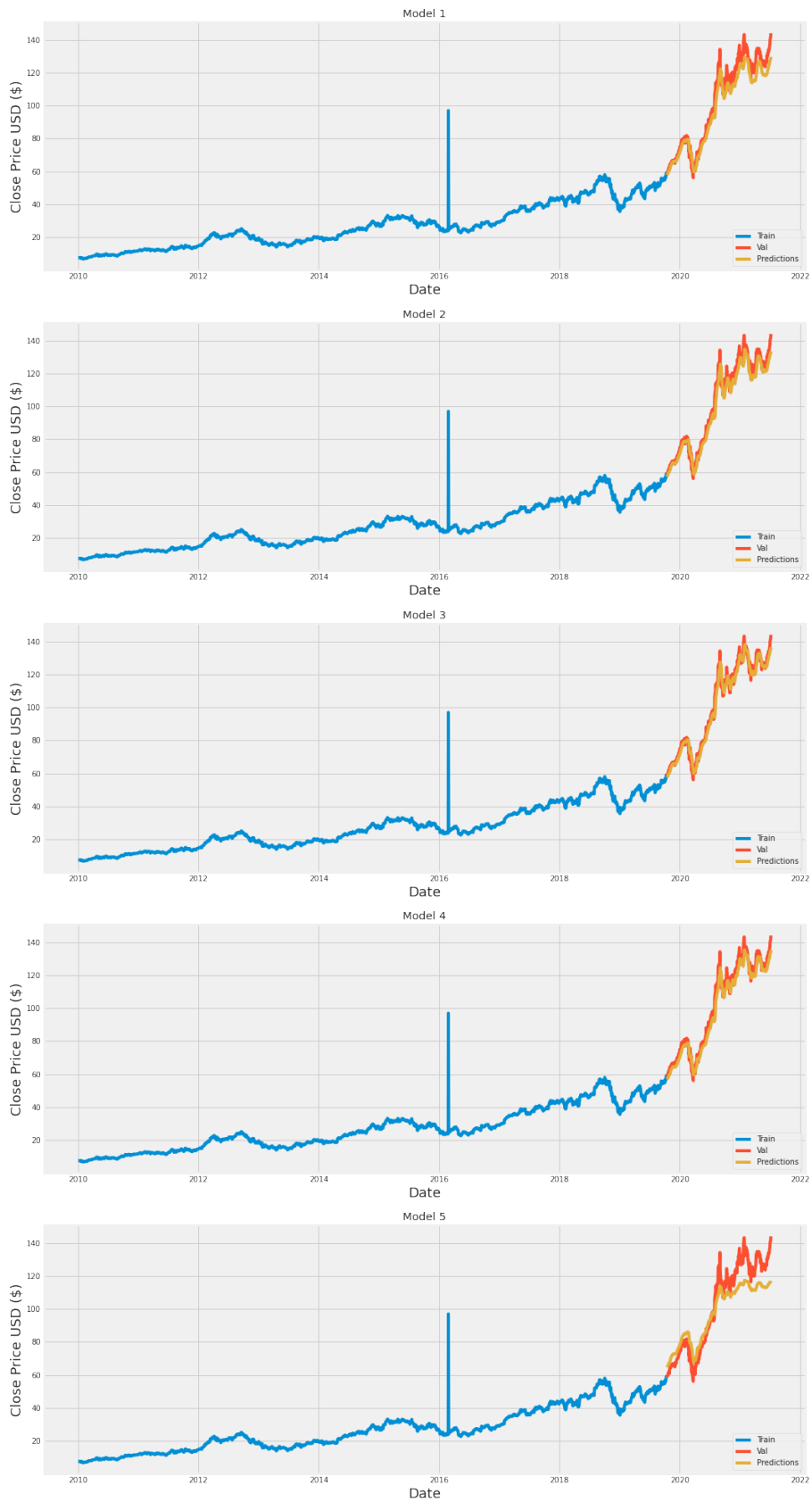


Figura 6.1: Validación de los modelos propuestos.



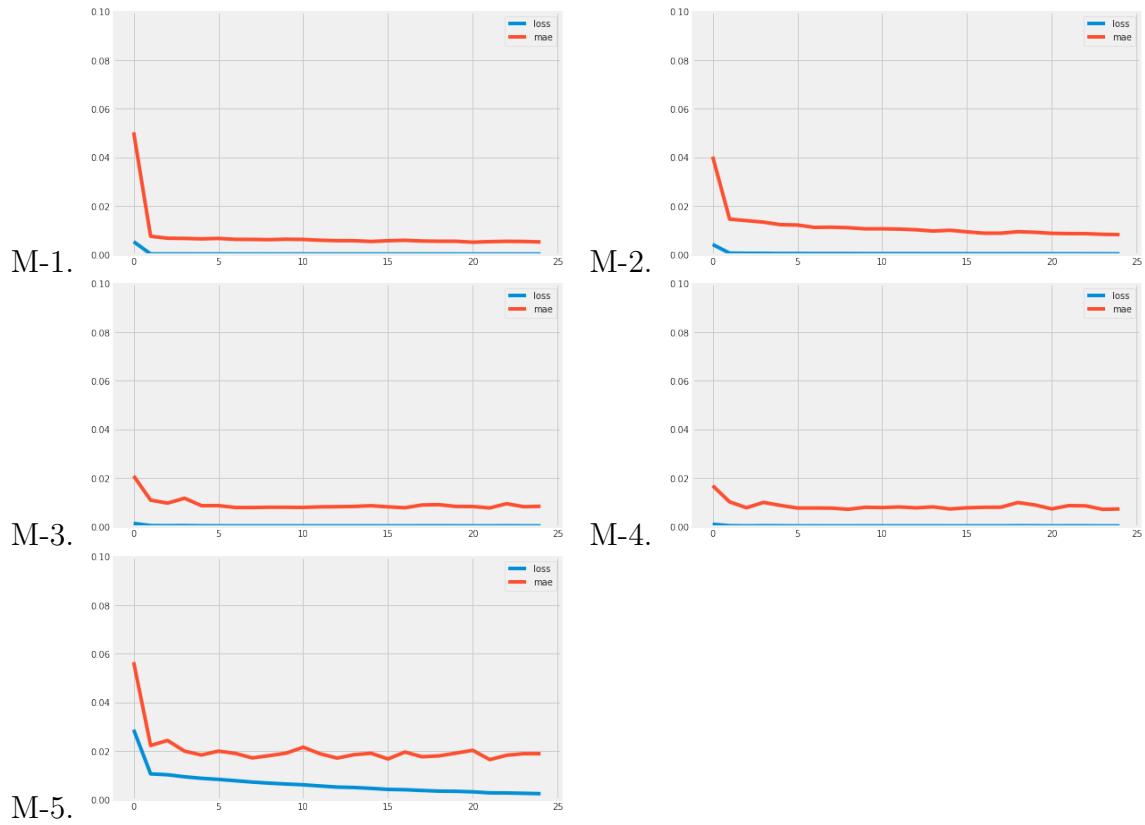


Figura 6.2: Función de pérdida y MAE.

<b>Dataset 85/15</b> 2019-10-16 a 2021-07-07	<b>M-1</b>	<b>M-2</b>	<b>M-3</b>	<b>M-4</b>	<b>M-5</b>
- RMSE	6.2538	4.8081	3.6591	4.4134	10.7750
- Rendimiento estimado	1.2420	1.3313	1.3704	1.3770	0.8107
- Error de estimación	-0.2149	-0.1256	-0.0864	-0.0799	-0.6462
<b>Dataset 95/5</b> 2020-12-09 a 2021-07-07	<b>M-1</b>	<b>M-2</b>	<b>M-3</b>	<b>M-4</b>	<b>M-5</b>
- RMSE	3.3584	3.6388	6.1581	3.0177	10.2295
- Rendimiento estimado	0.1212	0.1236	0.1212	0.1285	0.0848
- Error de estimación	-0.0618	-0.0594	-0.0618	-0.0545	-0.0982

Figura 6.3: Comparación de los modelos

<b>RMSE</b>	<b>Dataset de validación</b>			
<b>Dataset de entrenamiento</b>	<b>Apple</b>	<b>Google</b>	<b>Microsoft</b>	<b>Amazon</b>
- Apple	3.6776	54.3083	5.9623	89.6472
- Mismo que el Dataset de validación	3.6776	52.2671	9.7991	138.5008

Figura 6.4: Validación del modelo 3 utilizando distintos datasets de entrenamiento

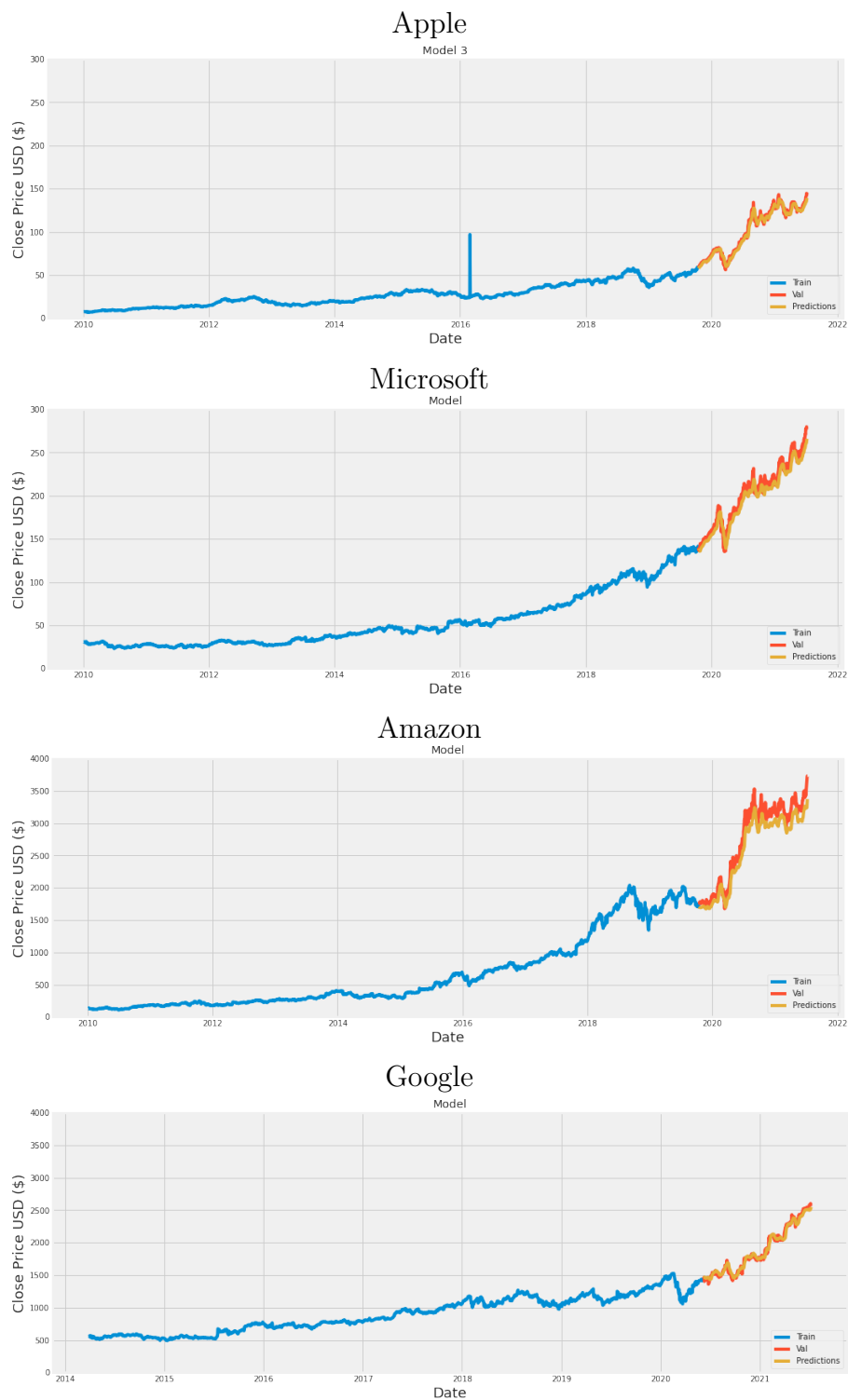


Figura 6.5: Validación del modelo 3.

# Capítulo 7

## Discusión

### 7.1. Evaluación de los modelos

#### 7.1.1. Selección del modelo

El modelo 3 es el mejor de los cinco propuestos. Consigue un RMSE mínimo de 3.6591, y también es mínimo el error en la estimación del rendimiento que generaría una inversión en acciones de Apple, desde el 16 de octubre de 2019 al 07 de julio de 2021, cuantificado en -0.0864. Además de estos datos, en la figura 6.1 se observa directamente que resuelve de mejor manera los problemas que planteaba el modelo inicial, esto es, que los precios estimados infravaloran levemente los precios de cierre reales; que la predicción empeora a medida que nos alejamos de la fecha en la que comienza la predicción (16 de octubre de 2019); y que la predicción no captura los máximos y los mínimos locales de los precios.

Dentro de las modificaciones realizadas en la arquitectura de la red neuronal Recurrente, se pueden destacar varios aspectos:

1) La regulación dropout ha mejorado las predicciones, en términos de disminuir el error en cada estimación del precio.

2) Aumentar el número de neuronas ha contribuido a mejorar la calidad de la predicción, así como la introducción de nuevas capas LSTM y fully-connected, aunque no se conoce hasta qué grado de implicación.

3) Las capas bidireccionales no producen el efecto esperado, pues no mejoran la actuación del modelo 3. Sí es verdad que utilizando la configuración del dataset 95/5, es decir, para una validación de menor número de días, el modelo 4 es el que obtiene mejores resultados, mejores incluso que los del modelo 3 (figura 6.3).

4) La adición de las técnicas de Batch-normalization y la regularización L2 en el modelo 5 han empeorado la calidad de las predicciones (basta ver la figura 6.1), obteniendo los peores datos de RMSE y de error del rendimiento estimado (figura 6.3). Aplicar una regularización estricta limita el espacio de soluciones, y tal vez no sea el efecto apropiado para el modelo propuesto porque el vector de entrada de la red neuronal (que contiene

los precios de cierre históricos de Apple) es muy simple.

Las funciones de pérdida que se representan en la figura 6.2 convergen a 0 muy pronto, prácticamente en el primer epoch en los primeros cuatro modelos. Añadiéndole que las curvas loss y mae son prácticamente lineales antes del epoch 5, hace pensar que hay presencia de overfitting. En este punto los modelos 3 y 4 son mejores que los modelos 1 y 2, pues la curva mae presenta algunas oscilaciones. Las curvas del modelo 5 describen un comportamiento más afín al esperado, donde además la convergencia a 0 de la curva loss no llega hasta el epoch 24 aproximadamente.

### 7.1.2. Validación del modelo 3

De los resultados mostrados en la figura 6.4 que comparan el RMSE del modelo 3 usando diferentes datasets de entrenamiento se pueden extraer una serie de características. Para empezar, los datos de RMSE obtenidos al entrenar la red neuronal con los precios de las acciones de Apple son siempre mejores, salvo en el caso de Google, donde por escasa diferencia (2 dólares estadounidenses) el RMSE en la validación es mayor. En otras palabras, es mejor utilizar como dataset de entrenamiento los precios de las acciones de Apple que utilizar como dataset de entrenamiento los precios correspondientes a la empresa para la que se pretenda predecir los precios.

Esto se hace muy evidente en la predicción de los precios de Amazon, donde la diferencia de RMSE utilizando como dataset de entrenamiento los precios de cierre de Apple es de 48.85 dólares menos, lo que implica una gran mejora.

## 7.2. Discusión de los resultados

La predicción de los valores futuros exactos del precio de las acciones es una tarea ambiciosa, por eso los estudios suelen enfocarse como un problema de clasificación: el precio futuro sube, baja o se mantiene constante. Aún así, con el modelo 3 se han obtenido unos valores con un RMSE aceptable. Su arquitectura no es especialmente sofisticada, lo que es sorprendente dados los resultados finales. Además, es muy positivo la escasa cantidad de datos que se ha tenido que utilizar. Sin embargo, no está del todo claro si el modelo presenta overfitting.

Dentro de las características de la predicción, en todas las simulaciones los valores de los precios estimados son siempre inferiores que los precios futuros reales: las estimaciones infravaloran los precios. También son pesimistas los rendimientos estimados que generarían las acciones. Sería deseable que los errores en la predicción tuvieran una distribución uniforme. Que no se cumpla esto es síntoma de que el modelo puede mejorarse. Puede ser que exista un condicionante externo reciente y con mucha trascendencia que haya afectado anormalmente a los precios, superando los valores que cabría esperar de ellos, como por ejemplo, la pandemia del COVID-19.

Por otra parte, las predicciones que más se acercan a la realidad son las de los precios de cotización de Apple y Microsoft, frente a las predicciones de los precios de cotización de Google y Amazon, para las que el RMSE asciende. Puede que el principal motivo de esta diferencia sea la magnitud de los precios mismos, cuyas cuantías a 16 de octubre de 2019 eran de 58.59\$, 140.41\$, 1438.4\$ y 1777.43\$, respectivamente. Se podrían extraer otras características de las empresas y compararlas con la precisión de las predicciones para ver las propiedades del modelo, pero realmente no se ha estudiado si existe una relación de causalidad. Por ejemplo, puede que el modelo realice mejores predicciones sobre los activos menos volátiles 5.7. Sin embargo, las correlaciones entre los precios de las empresas de la figura 5.6 no acompañan a estos resultados.

# Capítulo 8

## Conclusiones y trabajo futuro

### 8.1. Conclusiones

En este trabajo se ha elaborado un modelo que consigue predecir los precios de las acciones de Apple, Amazon, Google y Microsoft con un error de precisión aceptable. La modelización de este problema se ha realizado desde el punto de vista económico, pues es donde tiene sentido realizarse las preguntas pertinentes para motivar el desarrollo del estudio y comprender los resultados conseguidos.

La utilización de la Inteligencia Artificial en el análisis de los mercados bursátiles está recibiendo mucha atención últimamente, sobretodo por parte de los investigadores. Su capacidad para extraer automáticamente características de series temporales y capturar relaciones altamente no lineales, y para adaptarse a nuevos datos a través del aprendizaje la hacen una candidata excelente para el estudio del mercado, regido por un comportamiento extremadamente dinámico y complejo. En concreto, el modelo está basado en una red neuronal Long-Sort-Term-Memory, un algoritmo de Inteligencia Artificial que destaca porque permite considerar dependencias de contexto muy largas en los datos analizados, mejorando así el poder de predicción.

La programación se ha ideado de forma que para que el modelo funcione debe recibir un vector con precios de cierre históricos de cualquiera de las cuatro empresas. El 85 % de los precios del vector se utilizará para entrenar el modelo y el 15 % restante se utilizará para la validación. Los resultados muestran que el modelo final arroja buenas predicciones, incluso utilizando los precios de las acciones de diferentes empresas para el entrenamiento y la validación. De alguna manera, esto mide la capacidad del modelo de extrapolar características y patrones en los precios que pasarían desapercibidos de estudiar el problema de modelización con otras técnicas que no utilicen el aprendizaje profundo.

Para la elección del modelo final se han desarrollado 5 prototipos de modelos, que se han evaluado en función del error que producen al realizar las precciones de los precios de Apple. Por último, una vez escogido el modelo 3 como el más adecuado, se ha validado nuevamente su actuación comparando el error que produce al predecir los precios de las Amazon, Google y Microsoft.

## 8.2. Líneas de trabajo futuro

De acuerdo con los resultados, el modelo final es capaz de predecir satisfactoriamente los precios de las acciones de Apple y Microsoft, y en menor medida los precios de las acciones de Google y Amazon. Puede que en vez de diseñar un modelo de aplicación general, sea más adecuado diseñar distintos modelos para predecir los precios de cada acción en particular.

Sin embargo, parece más interesante encaminar la investigación en cómo mejorar el modelo para generalizar su aplicación. Una opción sencilla podría ser comprobar la actuación del modelo utilizando un dataset de precios de cierre de un índice bursátil, en lugar de una acción aislada, pues su precio es una combinación lineal de los precios de varias acciones diferentes. Otra opción podría ser mejorar la entrada del modelo utilizando técnicas de selección automática de los inputs. Seguramente al introducir más datos en el análisis habría que remodelar la red, dándole una estructura más compleja.

En cuanto a la integración de la Inteligencia Artificial en los sistemas automáticos de trading, todavía existe una gran brecha entre la teoría y la práctica. Esto se debe principalmente a que en el ámbito científico no se le presta la suficiente atención a una parte fundamental de la operativa en el mercado bursátil: la gestión de estrategias de inversión. Bien podría ser útil, aprovechando la contextualización económica que se hace en este trabajo y los buenos resultados conseguidos en las predicciones, mejorar el modelo para que sea capaz de emitir automáticamente señales de compra y de venta.





# Bibliografía

- [1] Adebiyi A. Ariyo, Adewumi O. Adewumi y Charles K. Ayo. «Stock Price Prediction Using the ARIMA Model». En: *2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation*. IEEE, mar. de 2014. ISBN: 978-1-4799-4922-9. DOI: [10.1109/UKSim.2014.67](https://doi.org/10.1109/UKSim.2014.67).
- [2] Blanchard, Amighini y Giavazzi. *Macroeconomics: A European Perspective*. 1.<sup>a</sup> ed. Pearson Education Limited, 2010.
- [3] Daniela Federici y Giancarlo Gandolfo. «Chaos in Economics». En: *Journal of Economics and Development Studies* 2 (1 mar. de 2014), págs. 51-79.
- [4] T. Fischer y C. Krauss. «Deep learning with long short-term memory networks for financial market predictions». En: *European Journal of Operational Research* 270(2) (2018).
- [5] Thomas Fischer y Christopher Krauss. «Deep learning with long short-term memory networks for financial market predictions». En: *European Journal of Operational Research* 270 (2 oct. de 2018). ISSN: 03772217. DOI: [10.1016/j.ejor.2017.11.054](https://doi.org/10.1016/j.ejor.2017.11.054).
- [6] Aurélien Géron. *Hands-on Machine Learning with Scikit-Learn, Keras & TensorFlow*. Ed. por Rachel Roumeliotis y Nicole Tache. 2.<sup>a</sup> ed. Gravenstein Highway North, Sebastopol: O'Reilly, 2019.
- [7] Sepp Hochreiter y Jürgen Schmidhuber. «Long Short-Term Memory». En: *Neural Computation* 9.8 (nov. de 1997). ISSN: 0899-7667. DOI: [10.1162/neco.1997.9.8.1735](https://doi.org/10.1162/neco.1997.9.8.1735).
- [8] M. Kraus y S. Feuerriegel. «Decision support from financial disclosures with deep neural networks and transfer learning». En: *Decision Support Systems* 104 (2017).
- [9] Mathias Kraus y Stefan Feuerriegel. «Decision support from financial disclosures with deep neural networks and transfer learning». En: *Decision Support Systems* 104 (dic. de 2017). ISSN: 01679236. DOI: [10.1016/j.dss.2017.10.001](https://doi.org/10.1016/j.dss.2017.10.001).
- [10] Y. Lecun y col. «Gradient-based learning applied to document recognition». En: *Proceedings of the IEEE* 86.11 (1998). ISSN: 00189219. DOI: [10.1109/5.726791](https://doi.org/10.1109/5.726791).
- [11] Jinho Lee y col. «Global Stock Market Prediction Based on Stock Chart Images Using Deep Q-Network». En: *IEEE Access* 7 (2019). ISSN: 2169-3536. DOI: [10.1109/ACCESS.2019.2953542](https://doi.org/10.1109/ACCESS.2019.2953542).
- [12] Ming-Che Lee y col. «Exploring the effectiveness of deep neural networks with technical analysis applied to stock market prediction». En: *Computer Science and Information Systems* 18.2 (2021). ISSN: 1820-0214. DOI: [10.2298/CSIS200301002L](https://doi.org/10.2298/CSIS200301002L).

- [13] Ming-Che Lee y col. «Exploring the effectiveness of deep neural networks with technical analysis applied to stock market prediction». En: *Computer Science and Information Systems* 18 (2 2021). ISSN: 1820-0214. DOI: [10.2298/CSIS200301002L](https://doi.org/10.2298/CSIS200301002L).
- [14] Raymond S. T. Lee. «Chaotic Interval Type-2 Fuzzy Neuro-oscillatory Network (CIT2-FNON) for Worldwide 129 Financial Products Prediction». En: *International Journal of Fuzzy Systems* 21.7 (oct. de 2019). ISSN: 1562-2479. DOI: [10.1007/s40815-019-00688-w](https://doi.org/10.1007/s40815-019-00688-w).
- [15] Wei Liang-Ying. «A hybrid model based on ANFIS and adaptive expectation genetic algorithm to forecast TAIEX». En: *Economic Modelling* 33 (2013), págs. 893-899. URL: <https://doi.org/10.1016/j.econmod.2013.06.009>.
- [16] «LSTM: A Search Space Odyssey». En: ().
- [17] Hiransha M y col. «NSE Stock Market Prediction Using Deep-Learning Models». En: *Procedia Computer Science* 132 (2018). ISSN: 18770509. DOI: [10.1016/j.procs.2018.05.050](https://doi.org/10.1016/j.procs.2018.05.050).
- [18] G.H. Moore. *Leading Indicators for the 1990s*. Homewood, IL, USA: Dow Jones-Irwin, 1990.
- [19] John J. Murphy. *Análisis técnico de los mercados financieros*. Ed. por Carlos Gnanzinelli. Barcelona: Ediciones Gestión 2000, 2000, undefined-546.
- [20] Viviane Naimy y col. «The predictive capacity of GARCH-type models in measuring the volatility of crypto and world currencies». En: *PLOS ONE* 16 (1 ene. de 2021). ISSN: 1932-6203. DOI: [10.1371/journal.pone.0245904](https://doi.org/10.1371/journal.pone.0245904).
- [21] David M. Q. Nelson, Adriano C. M. Pereira y Renato A. de Oliveira. «Stock market's price movement prediction with LSTM neural networks». En: IEEE, mayo de 2017. ISBN: 978-1-5090-6182-2. DOI: [10.1109/IJCNN.2017.7966019](https://doi.org/10.1109/IJCNN.2017.7966019).
- [22] Isaac Kofi Nti, Adebayo Felix Adekoya y Benjamin Asubam Weyori. «A novel multi-source information-fusion predictive framework based on deep neural networks for accuracy enhancement in stock market prediction». En: *Journal of Big Data* 8 (1 dic. de 2021). ISSN: 2196-1115. DOI: [10.1186/s40537-020-00400-y](https://doi.org/10.1186/s40537-020-00400-y).
- [23] Isaac Kofi Nti, Adebayo Felix Adekoya y Benjamin Asubam Weyori. «A systematic review of fundamental and technical analysis of stock market predictions». En: *Artificial Intelligence Review* 53.4 (abr. de 2020). ISSN: 0269-2821. DOI: [10.1007/s10462-019-09754-z](https://doi.org/10.1007/s10462-019-09754-z).
- [24] Ping-Feng Pai y Chih-Sheng Lin. «A hybrid ARIMA and support vector machines model in stock price forecasting». En: *Omega* 33 (6 dic. de 2005). ISSN: 03050483. DOI: [10.1016/j.omega.2004.07.024](https://doi.org/10.1016/j.omega.2004.07.024).
- [25] P Paik y B Kumari. «Stock Market Prediction Using ANN SVM ELM: A Review». En: *International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)* (2017), págs. 88-94.
- [26] Yusuf Perwej y Asif Perwej. «Prediction of the Bombay Stock Exchange (BSE) Market Returns Using Artificial Neural Network and Genetic Algorithm». En: *Journal of Intelligent Learning Systems and Applications* 04.02 (2012). ISSN: 2150-8402. DOI: [10.4236/jilsa.2012.42010](https://doi.org/10.4236/jilsa.2012.42010).

- [27] Robert S. Pindyck y Daniel L. Rubinfeld. *Microeconomics*. 7.<sup>a</sup> ed. Prentice Hall, 1995.
- [28] Robert S. Pindyck y Daniel L. Rubinfeld. *Microeconomics*. 7.<sup>a</sup> ed. Englewood Cliffs, NJ: Prentice Hall, 1995.
- [29] David E. Rumelhart, Geoffrey E. Hinton y Ronald J. Williams. «Learning representations by back-propagating errors». En: *Nature* 323.6088 (oct. de 1986). ISSN: 0028-0836. DOI: [10.1038/323533a0](https://doi.org/10.1038/323533a0).
- [30] Omer Berat Sezer, Mehmet Ugur Gudelek y Ahmet Murat Ozbayoglu. «Financial time series forecasting with deep learning : A systematic literature review: 2005–2019». En: *Applied Soft Computing* 90 (mayo de 2020). ISSN: 15684946. DOI: [10.1016/j.asoc.2020.106181](https://doi.org/10.1016/j.asoc.2020.106181).
- [31] Omer Berat Sezer, Mehmet Ugur Gudelek y Ahmet Murat Ozbayoglu. «Financial time series forecasting with deep learning : A systematic literature review: 2005–2019». En: *Applied Soft Computing* 90 (mayo de 2020). ISSN: 15684946. DOI: [10.1016/j.asoc.2020.106181](https://doi.org/10.1016/j.asoc.2020.106181).
- [32] Richard H. Thaler. «Behavioral Economics: Past, Present, and Future». En: *American Economic Review* 106.7 (jul. de 2016). ISSN: 0002-8282. DOI: [10.1257/aer.106.7.1577](https://doi.org/10.1257/aer.106.7.1577).
- [33] Vishal Dineshkumar Soni. «Prediction of Stock Market Values using Artificial Intelligence». En: *International Journal of Advanced Research in Electrical Electronics and Instrumentation Engineering* 7.4 (2018), págs. 1844-1848.
- [34] Ju-Jie Wang y col. «Stock index forecasting based on a hybrid model». En: *Omega* 40.6 (dic. de 2012). ISSN: 03050483. DOI: [10.1016/j.omega.2011.07.008](https://doi.org/10.1016/j.omega.2011.07.008).
- [35] Ju-Jie Wang y col. «Stock index forecasting based on a hybrid model». En: *Omega* 40 (6 dic. de 2012). ISSN: 03050483. DOI: [10.1016/j.omega.2011.07.008](https://doi.org/10.1016/j.omega.2011.07.008).

Otras referencias consultadas:

APPLE (2021). Informe financiero anual: Ejercicio 2020. Recuperado el 25 de junio 2021, de [https://investor.apple.com/investor-relations/default.aspx#tabs\\_content--2021](https://investor.apple.com/investor-relations/default.aspx#tabs_content--2021)

AMAZON (2021). Informe financiero anual: Ejercicio 2020. Recuperado el 25 de junio 2021, de <https://ir.aboutamazon.com/annual-reports-proxies-and-shareholder-letters/default.aspx>

MICROSOFT (2021). Informe financiero anual: Ejercicio 2020. Recuperado el 25 de junio 2021, de <https://www.microsoft.com/investor/reports/ar20/index.html>

GOOGLE (2021). Informe financiero anual: Ejercicio 2020. Recuperado el 25 de junio 2021, de <https://abc.xyz/investor/>

Célula LSTM. “File:Long Short-Term Memory.svg” por autoría de fdeloche, bajo la licencia CC BY-SA 4.0. Para ver una copia de esta licencia, visite <https://creativecommons.org/licenses/by-sa/4.0/>.

[org/licenses/by-sa/4.0](http://creativecommons.org/licenses/by-sa/4.0).

Capa RNN. “File:RecurrentLayerNeuralNetwork english.png” por autoría de Chrislb, bajo la licencia CC BY-SA 3.0. Para ver una copia de esta licencia, visite <http://creativecommons.org/licenses/by-sa/3.0/>

Datos de Yahoo Finance:

<https://finance.yahoo.com/quote/AAPL/history?period1=1593662400&period2=1625284799&interval=1d&frequency=1d&filter=history>

# Anexo

Figura 8.1: Código para exportar los datos de cotización.

```
!pip install investpy
import investpy
# Se crea un vector con los activos de Apple, Google, Microsoft y Amazon
tech_list = ['AAPL', 'GOOG', 'MSFT', 'AMZN']
# Se fijan las fechas de inicio y fin de toma de datos (1 año)
end = datetime.now()
start = datetime(end.year - 1, end.month, end.day)

# Exportación de los datos de cotización desde Yahoo Finance
for stock in tech_list:
    df = investpy.get_stock_historical_data(stock=stock,
                                           country='United States',
                                           from_date=start.strftime('%d/%m/%Y'),
                                           to_date=end.strftime('%d/%m/%Y'))

    df["Adj Close"] = df["Close"]
# Se crea el DataFrame
globals()[stock] = df
```

Figura 8.2: Estadísticos del rendimiento de las acciones.

Symbols	AAPL	GOOG	MSFT	AMZN
<b>count</b>	251.000000	251.000000	251.000000	251.000000
<b>mean</b>	0.001792	0.002381	0.001260	0.001178
<b>std</b>	0.021843	0.017735	0.016963	0.019668
<b>min</b>	-0.080061	-0.056496	-0.061947	-0.054456
<b>25%</b>	-0.009838	-0.006331	-0.008309	-0.010687
<b>50%</b>	0.001097	0.002880	0.001115	0.001415
<b>75%</b>	0.014805	0.012123	0.012493	0.011631
<b>max</b>	0.104689	0.073961	0.056241	0.079295

Figura 8.3: Modelo 1

```
tf.keras.backend.clear_session()
tf.random.set_seed(51)
np.random.seed(51)

# Diseño del modelo LSTM 1
MODEL1 = Sequential()
MODEL1.add(LSTM(16, input_shape= (x_train.shape[1], 1)))
MODEL1.add(Dense(1))

# Compilación del modelo
MODEL1.compile(optimizer='adam', loss='mean_squared_error', metrics=['mae'])

MODEL1.summary()

Model: "sequential"
-----
Layer (type)                Output Shape              Param #
-----
lstm (LSTM)                  (None, 16)                1152
-----
dense (Dense)                (None, 1)                  17
-----
Total params: 1,169
Trainable params: 1,169
Non-trainable params: 0
-----

# Entrenamiento del modelo
history1 = MODEL1.fit(x_train, y_train, epochs=25)
```

Figura 8.4: Modelo 2

```
tf.random.set_seed(51)
np.random.seed(51)

# Diseño del modelo LSTM 2 - añade regularización para reducir overfitting.
MODEL2 = Sequential()
MODEL2.add(LSTM(50, input_shape= (x2_train.shape[1], 1)))
MODEL2.add(Dropout(0.25)) # regularización dropout de 0.25
MODEL2.add(Dense(1))

# Compilación del modelo
MODEL2.compile(optimizer='adam', loss='mean_squared_error', metrics=['mae'])

MODEL2.summary()

Model: "sequential"
-----
Layer (type)                Output Shape              Param #
-----
lstm (LSTM)                  (None, 50)                10400
dropout (Dropout)           (None, 50)                0
dense (Dense)                (None, 1)                 51
-----
Total params: 10,451
Trainable params: 10,451
Non-trainable params: 0
-----

# Entrenamiento del modelo
history2 = MODEL2.fit(x2_train, y2_train, epochs=25)
```

Figura 8.5: Modelo 3

```
tf.random.set_seed(51)
np.random.seed(51)

# Diseño del modelo LSTM 3
MODEL3 = Sequential()
MODEL3.add(LSTM(80, return_sequences=True, input_shape= (x3_train.shape[1], 1)))
MODEL3.add(LSTM(80))
MODEL3.add(Dropout(0.25))
MODEL3.add(Dense(56)) # FC-1
MODEL3.add(Dense(1)) # FC-2

# Compilación del modelo
MODEL3.compile(optimizer='adam', loss='mean_squared_error', metrics=['mae'])

# Entrenamiento del modelo
history3 = MODEL3.fit(x3_train, y3_train, epochs=25)
```

Figura 8.6: Modelo 4

```
tf.random.set_seed(51)
np.random.seed(51)

# Diseño del modelo LSTM 4
MODEL4 = Sequential()
MODEL4.add(
    tf.keras.layers.Bidirectional(
        tf.keras.layers.LSTM(
            150, return_sequences=True, input_shape= (x4_train.shape[1], 1)))
)
MODEL4.add(tf.keras.layers.Bidirectional(tf.keras.layers.LSTM(150)))
MODEL4.add(Dropout(0.25))
MODEL4.add(Dense(128))
MODEL4.add(Dense(64))
MODEL4.add(Dense(1))

# Compilación del modelo
MODEL4.compile(optimizer='adam', loss='mean_squared_error', metrics=['mae'])

# Entrenamiento del modelo
history4 = MODEL4.fit(x4_train, y4_train, epochs=25)
```

Figura 8.7: Modelo 5

```
tf.random.set_seed(51)
np.random.seed(51)

# Diseño del modelo LSTM 5
MODEL5 = Sequential()
MODEL5.add(tf.keras.layers.BatchNormalization())
MODEL5.add(
    tf.keras.layers.Bidirectional(
        tf.keras.layers.LSTM(
            180, return_sequences=True, input_shape= (x3_train.shape[1], 1)))
)
MODEL5.add(tf.keras.layers.Bidirectional(tf.keras.layers.LSTM(180)))
MODEL5.add(Dropout(0.25))
MODEL5.add(Dense(256))
MODEL5.add(Dense(64, kernel_regularizer=tf.keras.regularizers.l2(0.0001)))
MODEL5.add(Dense(1))

# Compilación del modelo
MODEL5.compile(optimizer='adam', loss='mean_squared_error', metrics=['mae'])

# Entrenamiento del modelo
history5 = MODEL5.fit(x5_train, y5_train, epochs=25)
```