UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA

Capacidad de obtener beneficio de reglas técnicas de contratación basadas en algoritmos de minería de datos.

Evidencia práctica para la bolsa de Madrid.

Carlos Santana Vega

Las Palmas de Gran Canaria, 07 de Julio de 2015

Tutores:

Dr. Fernando Fernández Rodríguez
Dr. Javier Lorenzo Navarro
Dr. Juan Méndez Rodríguez

Trabajo de Fin de Grado

Título: Capacidad de obtener beneficio de reglas técnicas de contratación basadas en algoritmos de minería de datos. Evidencia práctica para la bolsa de Madrid.

Nombre del alumno: Carlos Santana Vega.
Fecha: 07 de Julio 2015

Tutores:

Dr. Fernando Fernández Rodríguez

Profesor de la Universidad de Las Palmas de Gran Canaria.

Área de interés: Finanzas Cuantitativas, Gestión del Riesgo, Mercados Financieros, Econometría Financiera, Finanzas Internacionales, Economía Computacional.

Dr. José Javier Lorenzo Navarro

Profesor de la Universidad de Las Palmas de Gran Canaria.

Área de conocimiento: Ciencias de la Computación e Inteligencia Artificial.

Dr. Juan Méndez Rodríguez

Profesor de la Universidad de Las Palmas de Gran Canaria.

Área de conocimiento: Ciencias de la Computación e Inteligencia Artificial.

Agradecimientos

A todos aquellos que hicieron posible mis días de universidad, a quienes compartieron mis tardes tumbado en el césped y a quienes me acompañaron en mis noches y madrugadas de estudio.

A todos los de aquí, que hicieron de estos años universitarios una aventura, y a todos los de allí, que hicieron de una aventura, el mejor recuerdo.

A todos, infinitas gracias. De verdad.

Índice general

- Capítulo 1 -
Presentación del proyecto.

En este capítulo se pretende exponer el propósito y los objetivos perseguidos con la realización de este proyecto, así como la justificación de las competencias adquiridas.

1.1 Introducción.

"La información es poder."

-- Anónimo. –

La producción de datos en volúmenes masivos es uno de los hechos fundamentales de nuestro tiempo. Esta afirmación es fácilmente comprobable cuándo se observa que actualmente, en un minuto, *Google* registra 4.1 millones de búsquedas, que *Youtube,* ve incrementado sus contenidos en más de 100 horas de video y, que durante ese mismo minuto, en *Facebook* se han enviado una media de 6,9 millones de mensajes entre sus usuarios. En términos cuantitativos, en Internet se generan aproximadamente 26.000GB [1] de datos cada segundo, cifra que si bien es abrumadora, se prevé que quedara eclipsada ante el pronóstico de un futuro donde Internet cobrará un carácter omnipresente en todo tipo de objetos cotidianos que nos rodee (conceptualmente llamado *El Internet de las cosas)*, con las implicaciones que esto supone en la masiva generación de datos.

Este fenómeno, es el que consecuentemente ha pasado a llamarse Big Data, y en el que, si bien la generación y almacenamiento de datos es una parte fundamental de todo su proceso, es realmente, la fase de procesamiento y análisis de dichos datos, la que nos permitirá obtener información de gran valor a partir de estos. El campo que engloba al conjunto de herramientas y técnicas de análisis que van a formar parte de esta parte del proceso, es conocida como Minería de Datos.

Nos encontramos, por tanto, ante una disciplina cuyo campo de actuación son las grandes bases de datos que almacenan toda la información que generamos diariamente, y cuyo objetivo será el de descubrir el conocimiento oculto en dichos datos. Para ello, la minería de datos se nutre de los conocimientos y técnicas de un

amplio rango de disciplinas, tales como la Inteligencia Artificial, la Estadística, el Aprendizaje Automático o los Sistemas de Bases de Datos, entre otras muchas.

En la actualidad, podemos encontrar numerosos ejemplos que confirman el auge de la minería de datos como una tendencia que va cobrando un papel cada vez más importante en los sistemas de información de las organizaciones.

En el ámbito privado, vemos como se incrementa el número de empresas que comprenden el gran valor de minar datos como una fuente de ventaja competitiva, y esto se evidencia al comprobar que en 2014, la red social profesional *LinkedIn,* registraba "Análisis estadístico y Minería de Datos" como la habilidad más demandada por las empresas a los nuevos profesionales [2]. En el ámbito público, podemos examinar el ejemplo de como el gabinete del presidente Barack Obama, durante su campaña electoral de 2008, analizó grandes volúmenes de datos proveniente de numerosas fuentes diferentes, con el fin de ser capaz de predecir en qué medida sus actos en campaña electoral ejercían influencia sobre el voto de la gente, dotándole de una fundamental ventaja competitiva para poder ganar las elecciones [3]. Finalmente, en el ámbito académico, se puede observar también como la minería de datos es un tema bastante recurrente en las publicaciones científicas, dónde el estudio, mejora y aplicación de las diferentes técnicas que la componen, permite una rápida evolución de esta disciplina.

De los diferentes ámbitos que pueden verse beneficiados por la aplicación de técnicas de minería de datos, centraremos nuestra atención para el desarrollo de este proyecto en el sector financiero, y más específicamente, en los mercados bursátiles. El atractivo de centrarnos en este ámbito viene dado por el alto número de factores que influyen directa o indirectamente en los mercados, generando un elevado volumen de datos que pueden ser utilizados como materia prima para la búsqueda de conocimiento. Además, los casos de estudio que se muestren favorables, dónde la aplicación de estas técnicas contribuya a la obtención de información útil que pueda ser incorporada en nuestra operativa bursátil, son de elevado valor por los rendimientos económicos que su uso supone.

Concretamente, el objetivo se propone con la elaboración de este proyecto, es el de investigar y validar el rendimiento aportado por este tipo de técnicas en la tarea de predecir información financiera. Para ello, plantearemos una hipótesis de trabajo que requiera el uso de este tipo de herramientas, y se diseñará e implementará una solución que ejecute todas las fases del proceso de descubrimiento de conocimiento, desde la obtención y preparación de los datos, hasta el análisis y obtención de resultados. La definición de los pasos a seguir para el correcto cumplimiento del proceso, estarán validados por el uso de la metodología *CRISP-DM*, la cual se erige como la metodología más utilizada para el desarrollo de proyectos de análisis, descubrimiento de conocimiento y minería de datos [4].

A lo largo de los próximos capítulos, se presentarán todos los materiales teóricos, técnicos y prácticos que han sido de utilidad para la elaboración de este proyecto y que han permitido, además de alcanzar los objetivos propuestos, obtener un mayor conocimiento sobre el apasionante campo de la minería de datos aplicada a la predicción bursátil.

1.2 Estructura de la memoria.

La memoria de este proyecto pretende ser el documento dónde se presente toda la información obtenida, tanto del proceso de documentación previa como la generada durante la resolución del propio proyecto, de forma que sirva como guía y registro para la compresión del trabajo realizado.

Este capítulo sirve como presentación del proyecto, introduciendo al lector en el tema que vamos a tratar, y justificando su realización con las competencias correspondientes a ambos títulos que han sido adquiridas durante su elaboración.

A continuación, se presentará la metodología CRISP-DM, cuya aportación ha servido para dotar de una estructura coherente tanto a nivel de desarrollo del proceso de minería de datos, como en la elaboración de esta memoria.

En los capítulos 3 y 4 se presentarán todos los conocimientos teóricos necesarios sobre las herramientas y técnicas que han sido fundamentales para la resolución de este proyecto.

Tras la teoría, en los capítulos 5 y 6 se presentará el problema a solucionar y la solución propuesta, para posteriormente hacer un recorrido detallado sobre el diseño e implementación de dicha solución.

Finalmente, en los capítulos 7 y 8, realizaremos un análisis de los resultados obtenidos haciendo uso del sistema implementado, y de dicho análisis extraeremos las conclusiones buscadas como objetivo de este proyecto. Además, se presentará un vía de desarrollo futuro sobre la cuál poder continuar el trabajo iniciado con este proyecto de fin de grado.

1.3 Competencias específicas cubiertas.

Dada la naturaleza multidisciplinar de la Doble Titulación de Ingeniería Informática y Administración de Empresas, ámbito de este proyecto de fin de grado, las competencias cubiertas con su desarrollo competen a ambos campos de estudios.

A continuación se presentan un listado de todas las competencias cubiertas con la realización de este proyecto, correspondiente a cada título, y su justificación.

Grado Ingeniería Informática - Competencias del título.

T1. Capacidad para concebir, redactar, organizar, planificar, desarrollar y firmar proyectos en el ámbito de la ingeniería en informática que tengan por objeto, de acuerdo con los conocimientos adquiridos según lo establecido en apartado 5 de la resolución indicada, la concepción, el desarrollo o la explotación de sistemas, servicios y aplicaciones informáticas.

Se justifica con la realización de este proyecto de fin de grado.

T2. Capacidad para dirigir las actividades objeto de los proyectos del ámbito de la informática, de acuerdo con los conocimientos adquiridos según lo establecido en apartado 5 de la resolución indicada.

Se justifica con la realización de este proyecto de fin de grado.

T5. Capacidad para concebir, desarrollar y mantener sistemas, servicios y aplicaciones informáticas empleando los métodos de la ingeniería del software como instrumento para el aseguramiento de su calidad, de acuerdo con los conocimientos adquiridos según lo establecido en apartado 5 de la resolución indicada.

Se justifica con el desarrollo de la herramienta software implementada para este proyecto de fin de grado (Ver epígrafe 6, Implementación del sistema).

T8. Conocimiento de las materias básicas y tecnologías, que capaciten para el aprendizaje y desarrollo de nuevos métodos y tecnologías, así como las que les doten de una gran versatilidad para adaptarse a nuevas situaciones.

Se justifica con la utilización de conocimientos teóricos y sobre tecnologías expuestas en este proyecto, aplicados acertadamente en la resolución del problema planteado (Ver epígrafe 3, Recursos y tecnologías; epígrafe 4, Base teórica y estado del arte).

CII06. Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.

Se justifica con la aplicación de las herramientas algorítmicas adecuadas aplicados acertadamente en la resolución del problema planteado en este proyecto. (Ver epígrafe 4.4, Redes neuronales artificiales, *ANNs*; epígrafe 6.4, Módulo *dataAnalyzer*).

CII07. Conocimiento, diseño y utilización de forma eficiente los tipos y estructuras de datos más adecuados a la resolución de un problema.

Se justifica con la utilización de las estructuras de datos adecuadas en cada uno de los módulos implementados, aplicadas correctamente en la resolución del problema planteado en este proyecto. (Ver epígrafe 6, Implementación del sistema).

CII08. Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.

Se justifica con la utilización del lenguaje de programación más adecuado (lenguaje R) para la implementación de los módulos del sistema, así como el seguimiento de principios de modularidad para la arquitectura del sistema. (Ver epígrafe 3, Recursos y tecnologías; epígrafe 5.2, Diseño de la solución).

CII013. Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los Sistemas de información, incluidos los basados en web.

Se justifica con la utilización de los Sistemas de Información web de *Yahoo Finance* para la obtención de los datos utilizados en el análisis, así como la herramienta *Yahoo YQL* para la extracción de dichos datos. (Ver epígrafe 3.3, *Yahoo YQL*)

CII015. Conocimiento y aplicación de los principios fundamentales y técnicas básicas de los sistemas inteligentes y su aplicación práctica.

Se justifica con el conocimiento y la utilización de las diferentes técnicas de minería de datos correspondientes al campo del aprendizaje automático y la inteligencia artificial. (Ver epígrafe 4.1, Minería de datos; epígrafe 4.4 Redes neuronales artificiales, *ANNs;* epígrafe, 4.5 El problema del *overfitting*; epígrafe 6.4, Módulo *dataAnalyzer*).

Grado en Administración y Dirección de Empresa - Competencias del título.

CE1.- Capacidad de aplicar los conocimientos en la práctica.

Se justifica con la aplicación de los conocimientos teóricos sobre finanzas, economía, econometría y estadística aplicados para la resolución de este proyecto.

CE3.- Habilidad de transmisión de conocimientos.

Se justifica con la realización de este proyecto de fin de grado, que da soporte a la divulgación de conocimiento sobre el tema desarrollado, así como la defensa oral del mismo.

CE5.- Poseer y comprender conocimientos acerca de las instituciones económicas como resultado y aplicación de representaciones teóricas o formales acerca de cómo funciona la economía.

Se justifica mediante la comprensión del funcionamiento a nivel teórico de los mercados bursátiles y sus interrelaciones (Ver epígrafe 4.2, Algoritmos aplicados a los mercados bursátiles; epígrafe 5.1, Presentación del problema).

CE9.- Identificar la generalidad de los problemas económicos que se plantean en las empresas y saber utilizar los principales instrumentos existentes para su resolución.

Se justifica mediante la aplicación de herramientas de carácter estadístico para la resolución del problema del análisis de existencia de correlaciones entre mercados (Ver epígrafe 5.1, Presentación del problema; epígrafe 5.2 Diseño de la solución; epígrafe 6, Implementación del sistema).

CE18.- Entender las instituciones económicas como resultado y aplicación de representaciones teóricas o formales acerca de cómo funciona la economía.

Se justifica mediante la comprensión del funcionamiento a nivel teórico de los mercados bursátiles y sus interrelaciones (Ver epígrafe 4.2, Algoritmos aplicados a los mercados bursátiles; epígrafe 5.1, Presentación del problema).

CE19.- Derivar de los datos información relevante imposible de reconocer por no profesionales.

Se justifica mediante la realización del proceso de minería de datos aplicado en este proyecto para la obtención de información relevante que pudiera ser de utilidad en la toma de decisiones financieras (ver epígrafe 7, Análisis y resultados).

- Capítulo 2 –
Metodología de trabajo.

Siguiendo la tendencia mostrada por la industria, hemos hecho uso del enfoque planteado por la metodología CRISP-DM para la organización de las tareas a realizar en proceso de minería de datos realizado. En este capítulo, presentaremos dicha metodología y justificaremos el uso de ella mostrando su aplicación.

2.1 Cross Industry Standard Process for Data Mining, CRISP-DM.

A la hora de enfrentarnos al reto de realizar un proyecto de minería de datos, la metodología CRISP-DM nos asiste en la descomposición del proceso de descubrimiento de conocimiento a realizar, definiendo las diferentes tareas que deberán ser seguidos para alcanzar el objetivo establecido. El grado de descomposición de las tareas propuestas variará en cada uno de los cuatro niveles de abstracción que CRISP-DM propone. Estos niveles, de más genérico a más específico, dividen el proceso en *fases, tareas genéricas, tareas especializadas* e *instancias de los procesos*. (Ver *Figura 1*) [5].

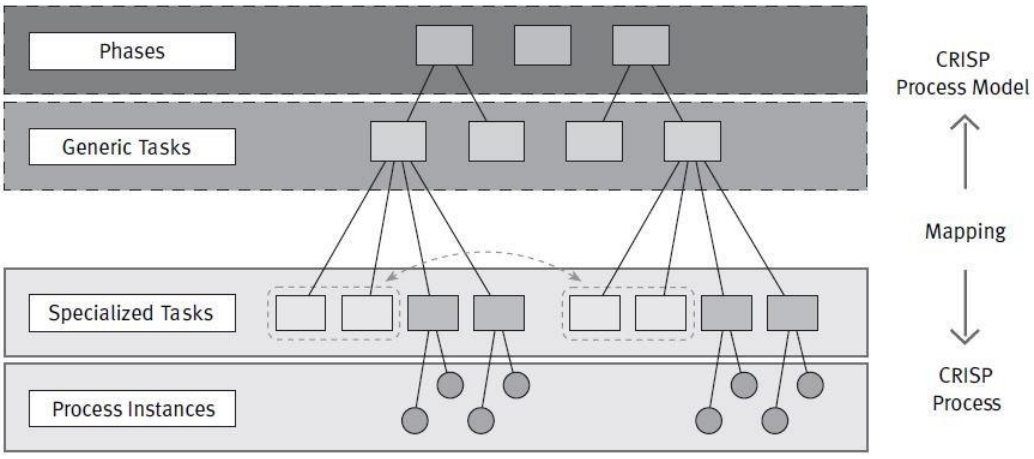

Figura 1: Los cuatro niveles de descomposición de la metodología CRISP-DM

Dado que el fin de este proyecto no es el de validar la utilización de esta metodología, no haremos uso de los niveles con mayor grado de especificación, pues su uso está más orientado a ser aplicado en grandes proyectos empresariales de minería de datos. Para las necesidades de nuestro proyecto, nos hemos bastado con los primeros dos

niveles de abstracción dónde se definen las *fases* y *tareas genéricas* que deberemos cumplir. Concretando, la descomposición que plantea CRISP-DM es la siguiente [6]:

1. **Comprensión del negocio.**

   Se busca entender y definir aquellos objetivos que se quieren alcanzar tanto desde la vertiente empresarial como desde la vertiente del analista de datos. Los objetivos del proceso son establecidos, así como los pasos y recursos que serán necesarios para poder alcanzarlos satisfactoriamente.

2. **Comprensión de datos.**

   Esta fase recoge el proceso de recopilación, integración, entendimiento y verificación de la calidad de los datos. Con esto, se pretende obtener datos que sean apropiados para la satisfacción del objetivo final, y obtener las pautas que nos guiarán a la hora de decidir cuál herramienta de minería de datos será más adecuado para ello.

3. **Preparación de datos.**

   Sobre el conjunto de datos obtenidos, seleccionaremos aquellos que vayan a ser usados durante el análisis. Estos datos, serán limpiados y formateados para elevar su calidad al nivel requerido por las técnicas de minería seleccionadas.

4. **Modelado.**

   Será en este punto del proceso dónde definiremos con precisión la/las técnica/as a utilizar para analizar y obtener información relevante de los datos suministrados. Además, se crearán los mecanismos necesarios para estimar la calidad del modelo, lo cual normalmente, es resuelto mediante la descomposición del conjunto de datos en dos subconjuntos: un subconjunto de entrenamiento sobre el cuál estimaremos el modelo y un subconjunto de prueba sobre el cuál se probará la validez de lo estimado.

5. **Evaluación.**

   Esta fase busca evaluar si los resultados obtenidos cumplen con las necesidades de negocio establecidas en la primera fase del proceso. Igualmente, se revisará si el proceso de minería de datos se ha realizado de forma correcta cumpliendo

con los requisitos analíticos propuestos. En caso afirmativo, se decidirá la evolución del desarrollo del proyecto en el futuro.

6. **Despliegue.**

   Finalmente, con los modelos estimados y validados, en esta fase se realizará la explotación de la información obtenida de la manera que el cliente requiera.

2.2 Aplicación práctica de la metodología.

Tal y como se ha precisado anteriormente, en este proyecto no se pretende demostrar la validez de la metodología CRISP-DM, sin embargo, se buscará con su aplicación el seguir su coherente definición de las diferentes etapas, de acuerdo con la industria, durante el proceso de minería de datos. Por ello, las 6 fases presentadas en el punto anterior han servido para estructurar y agrupar las diferentes tareas realizadas en la resolución de este proyecto.

Con el fin de justificar la aplicación de la metodología, a continuación se incluye una relación de las tareas realizadas y la fase a la que se corresponde dentro de la metodología CRISP-DM:

Fase 1: Comprensión del negocio.

- Búsqueda de información relacionada.
- Definición de la hipótesis de trabajo.
- Búsqueda de la fuente de los datos a utilizar.
- Establecimiento de las herramientas a utilizar.
- Planificación del proyecto.

Fase 2: Comprensión de datos.

- Análisis de las fuentes de datos disponibles.
- Implementación del módulo *dataCrawler.*

Fase 3: Preparación de datos.

- Planificación de datos requeridos.
- Análisis de las características de los datos adquiridos.
- Implementación del módulo *dataBuilder.*

Fase 4: Modelado.

- Selección de la técnica de minería a utilizar.
- Implementación del módulo *dataAnalyzer.*

Fase 5: Evaluación.

- Implementación del módulo *modelValidator.*
- Obtención de los resultados.
- Análisis de los resultados.
- Representación de los resultados.

Fase 6: Despliegue.

- Obtención de conclusiones.
- Redacción de la memoria del trabajo.

Como se puede comprobar, la distribución de las tareas realizadas durante el proyecto sigue la filosofía planteada por la metodología CRISP-DM. En los próximos capítulos, según se vayan presentando las diferentes fases con mayor detalle, se volverán a revisar las implicaciones del uso de esta metodología.

- Capítulo 3 –
Recursos y tecnologías.

En este capítulo se presentarán las herramientas y tecnologías que han sido fundamentales para el desarrollo de este proyecto.

3.1 Lenguaje R.

Lenguaje de programación que ofrece una amplia gama de herramientas para el análisis estadístico y cálculo numérico. Gran parte de las funcionalidades del lenguaje se encuentran implementadas utilizando el propio lenguaje R, aunque también si fuera necesario un mayor rendimiento, se puede hacer uso de lenguajes de más bajo nivel como C, C++ o Fortran. Además, el lenguaje permite a la comunidad de desarrolladores extender sus funciones mediante la creación de paquetes que pueden ser fácilmente compilados y utilizados en el código. Fue inicialmente desarrollado por Robert Gentleman y Ross Ihaka, actualmente se encuentra bajo la responsabilidad del *R Development Core Team.* Durante 2014, R se situó como el lenguaje más utilizado en el campo de la Minería de Datos [7].

Para el desarrollo de nuestro proyecto, ha sido indispensable la utilización de los siguientes paquetes:

- **Paquete *HTTR.*** [8]

  Conjunto de herramientas y métodos que facilita la realización de peticiones HTTP implementando los diferentes métodos soportados por el protocolo (GET, POST, PUT, PATCH, etc.). Mediante la especificación de una dirección URL se podrá obtener una *HTTP Response* con la información devuelta por dicha llamada. Además, este paquete incluye un amplio conjunto de funciones de configuración que permiten un mayor control de las peticiones que se realizan (i.e. autentificación, *cookies*, manipulación de *headers*, etc.).

- **Paquete *JsonLite.*** [9]

  Conjunto de herramientas robustas y de gran rendimiento para la creación e interpretación del lenguaje JSON. La traducción ofrecida por este paquete

permite tanto la interpretación de datos en notación JSON a estructuras de datos propias del lenguaje R, como el proceso inverso.

- **Paquete NeuralNet.** [10]

  Conjunto de herramientas de gran utilidad para la creación y entrenamiento de redes neuronales artificiales. Mediante el uso de este paquete se pueden generar diferentes configuraciones de redes neuronales, dónde se podrá definir su topología o las funciones de activación y propagación de sus neuronas internas. Una vez creada la red neuronal, este paquete aporta herramientas para su entrenamiento y posterior uso, pudiendo introducir nuevos valores con el fin de computar y realizar predicciones. A este paquete le acompañan también funciones para la representación de las redes neuronales, haciendo uso de las capacidades gráficas del lenguaje de programación R.

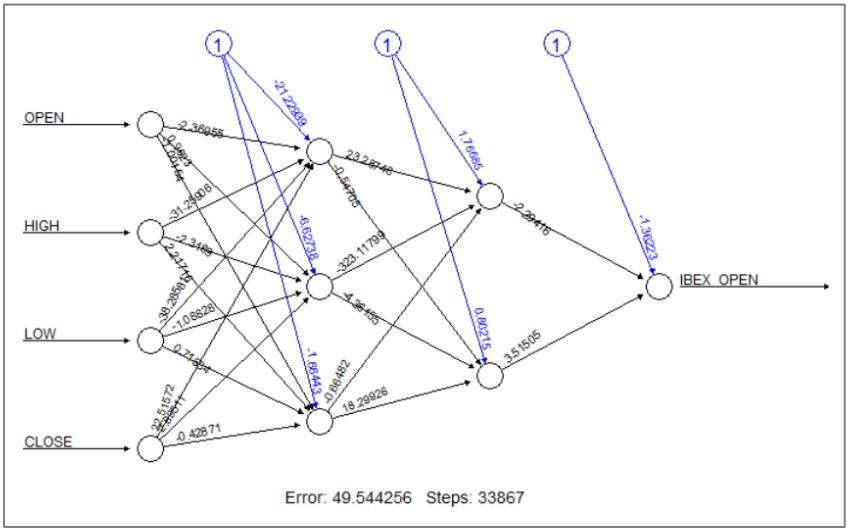

Figura 2 Ejemplo de representación de una red neuronal artificial, paquete *neuralnet.*

- **Paquete Quantmod.** [11]

  Conjunto de herramientas diseñadas para el desarrollo y prueba de modelos financieros. Con el uso de sus funciones, podremos trabajar con datos financieros, de los cuales podremos calcular índices, hacer representaciones gráficas, etc. Además, también nos permitirá la generación de reglas de contratación que podrán ser validadas con datos históricos del mercado, obteniendo estadísticos que nos indiquen el rendimiento de utilizar dicha estrategia en nuestra operativa.

3.2 *RStudio*. [12]

Entorno de desarrollo integrado (*IDE*) para el lenguaje de programación R. Alguna de las funcionalidades más importantes que ofrece son herramientas de resaltado de sintaxis, autocompletado del código, depuración interactiva, gestión de directorios, así como otro conjunto de herramientas que facilitan el desarrollo en R.

Se trata de software con licencia *open source* y como versión comercial y se encuentra disponible para la mayor parte de los sistemas operativos e incluso navegadores webs conectados a su servicio RStudio Server, que ofrece acceso centralizado a las capacidades de R.

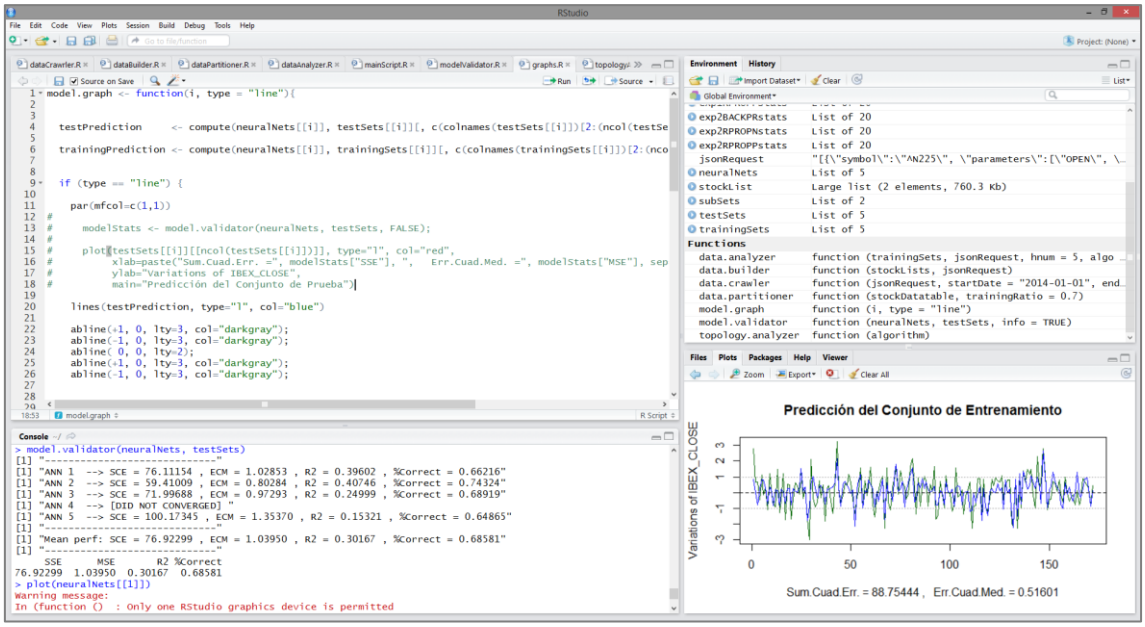

Figura 3: Captura de pantalla del IDE RStudio.

3.3 *Yahoo* YQL

Para la extracción de los datos, se ha optado por utilizar la herramienta *Yahoo YQL (Yahoo Query Language)*. YQL, es una plataforma diseñada por la compañía *Yahoo* que facilita enormemente el acceso a las APIs públicas que están alojadas en Internet. Mediante el uso de un lenguaje similar a SQL (*Structured Query Language*), permite realizar peticiones HTTP para obtener, filtrar y combinar datos desde diferentes fuentes de internet, escondiendo la complejidad de las APIs web. Además, los resultados obtenidos son transformados al lenguaje XML y JSON, lo cual permite una mayor interoperabilidad entre diferentes aplicaciones.

La forma de extraer los datos mediante sentencias YQL es muy sencilla. La estructura de su sintaxis básica es como se presenta a continuación:

```
SELECT qué FROM tabla WHERE filtro | función
```

De forma muy parecida al lenguaje SQL, la cláusula *qué* indica aquellos campos, (o columnas, dada la distribución de los datos en tablas) a los que queremos acceder. Con la cláusula *tabla,* especificaremos el nombre de la fuente a partir de la cual queremos extraer los datos. YQL, facilita a la comunidad de desarrolladores la extensibilidad del sistema permitiendo la creación personalizada de este tipo de tablas. Para su generación, se hace uso de archivos XML que definen la forma en la que se van a mapear las sentencia YQL con los servicios webs a los que se pretende acceder. Esto permite generar tablas propias dónde, en el proceso de mapeado, se puede combinar y procesar datos de numerosas fuentes diferentes, accedidas a través de la misma interfaz unificada.

Del conjunto de columnas ya seleccionado, con la cláusula *filtro*, podremos establecer un criterio que nos permita seleccionar aquellas filas que sean de nuestro interés. Finalmente, YQL ofrece un repertorio de funciones para operar sobre el conjunto de filas seleccionadas, permitiendo operaciones de ordenado, inversión, o eliminación de repeticiones, entre otras.

A modo ilustrativo, si quisiéramos realizar una consulta para obtener una lista de las fotografías de gatos alojadas en la web de Flickr, ejecutaríamos la siguiente sentencia:

```
SELECT * FROM flickr.photos.search WHERE text="cat";
```

Esta petición nos devolverá un resultado JSON o XML con los datos y metadatos de las imágenes alojadas en el servicio *Flickr.* En la plataforma online de YQL [13], se encuentra el listado completo de todas las tablas a las que se pueden hacer peticiones. De todas ellas, para nuestro proyecto, hemos usado la tabla *yahoo.finance.historicalData,* que conecta directamente con la base de datos de la plataforma *Yahoo Finance*, donde poder obtener el histórico de datos bursátiles que necesitaremos para nuestro análisis.

La licencia de uso de la plataforma YQL permite su uso gratuito para cualquier tipo de proyecto, sea comercial o no, quedando fuera de su ámbito las licencias que cada una de las APIs accedidas pudiera tener. En el caso de nuestro proyecto, el acceso a la API de *Yahoo Finance* es abierto, y no requiere de ninguna *API Key* para su uso, si bien es

cierto que existen algunas limitaciones respecto a la cantidad de datos y al número de peticiones que se pueden realizar al sistema.

- Capítulo 4 –
Base teórica y estado del arte.

En este capítulo se pretende abordar la base teórica que servirá para contextualizar el proyecto a desarrollar. Se explicarán las técnicas más comunes de minería de datos y se procederá a presentar la relación existente entre estos algoritmos y el sector financiero.

4.1 Minería de datos.

"Una importante cadena de supermercados posee los datos del conjunto de productos comprados por sus clientes en cada compra, y a partir de esos datos, le gustaría obtener información que pueda servir de ayuda para definir su próxima campaña de marketing."

"El departamento de recursos humanos de una gran empresa cuenta con bastantes datos sobre sus trabajadores. Entendiendo al capital humano como un recurso clave generador de ventaja competitiva, buscan automatizar el proceso de búsqueda de talento entre sus trabajadores para poder incentivarlos."

"La complejidad de los mercados bursátiles, dado de la gran cantidad de factores de los que dependen, hace compleja la tarea de encontrar reglas de contratación que permitan obtener un rendimiento económico ventajoso."

¿Qué tienen en común estos tres hipotéticos casos? La respuesta es que todos ellos han sido demostrados ser resueltos con gran rendimiento mediante la aplicación de técnicas de minería de datos. Tras un primer análisis, encontramos que todos ellos plantean un patrón similar, dónde se desea extraer información relevante que pueda asistir en la futura toma de decisiones a partir de un gran conjunto de datos. Este proceso es el que antiguamente, en los años 90, se conocía como KDD (*Knowledge Discovery in Databases*) y cuya fase de análisis de los datos era denominada Minería de Datos (ver *Figura 4*), aunque en los últimos años de forma común, este nombre ha pasado a englobar todo el proceso, desde la selección y preparación de los datos hasta la interpretación y obtención de conocimiento.

El aumento de interés en la Minería de Datos se justifica por el fenómeno denominado *Big Data*. Esto es, la acumulación de grandes cantidades de datos, que se produce fundamentalmente por el incremento en el uso de dispositivos capaces de registrar con sus sensores datos en gran volumen y variedad (móviles, *tablets*, *wearables*, etc.), así como por el constante abaratamiento de los sistemas de almacenamiento. Además, otra de las razones se da en el incremento en las capacidades de cómputo, lo cual ha permitido el desarrollo de algoritmos cada vez más complejos capaces de tratar estas grandes bases de datos, así como de crear conocimiento de mayor abstracción y complejidad. También cabría añadir que a estos cambios se le suma una transformación en la mentalidad de las organizaciones, las cuales comienzan a tomar conciencia del valor de los datos almacenados en sus sistemas, hasta hace pocos años infravalorados. Esto permite no solo que las propias organizaciones comiencen a aplicar el proceso de minería para explotar sus conjuntos de datos, sino que también comiencen a realizar un proceso de apertura e integración con otras organizaciones para acabar formando parte de un ecosistema basado en la información y el conocimiento.

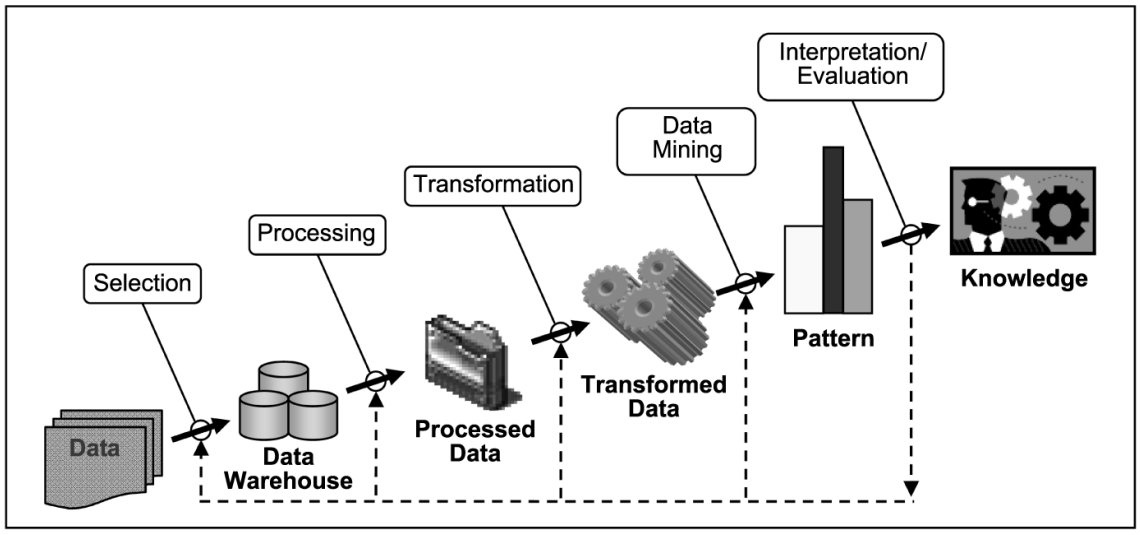

Figura 4: Proceso de Minería de Datos.

Una definición más teórica de la minería de datos, nos diría que ésta hace referencia al proceso analítico que intenta descubrir patrones en grandes volúmenes de conjuntos de datos, mediante la aplicación de técnicas de disciplinas tales como la inteligencia artificial, el aprendizaje automático, la estadística y sistemas de bases de datos, entre otras. Esta implicación de tantas disciplinas diferentes, en ocasiones genera un mal uso de los conceptos, dónde por ejemplo, se confunde aprendizaje automático o estadística con el propio proceso de minería de datos. Con el fin de esclarecer dudas, diremos que las disciplinas mencionadas anteriormente son las que crean las herramientas de procesamiento y análisis, y la minería de datos es la fase de

explotación de los datos que hace uso de dichas herramientas: El diseño de un algoritmo clasificador, compete al campo de estudio del aprendizaje automático. En cambio, el uso de dicho algoritmo clasificador con el fin de tratar un conjunto de datos para la obtención de conocimiento, define a un proceso de minería de datos.

Será la naturaleza de la técnica utilizada la que defina el tipo de proceso de minería que estemos realizando, y por tanto, el tipo de información que se obtendrá. De forma genérica, encontramos la siguiente clasificación [14], aunque es importante señalar que, a pesar de ser posible su separación en categorías individuales, en la práctica, todas ellas se encuentra muy interrelacionadas unas con otras. Entre estas técnicas encontramos:

- **Técnicas de clasificación:**

  Este tipo de algoritmos buscan la categorización de elementos a partir de sus atributos. Normalmente, este tipo de técnicas requiere de un proceso de aprendizaje previo en el que el algoritmo aprende la relación existente entre los atributos y la categoría del objeto. Es por ello que desde la vertiente de la disciplina del aprendizaje automático, esto sería considerado un algoritmo de aprendizaje automático supervisado (i.e. el algoritmo aprende a partir de definir un output esperado para un input determinado).

- **Técnicas de *clustering*:**

  Estas técnicas buscan la generación de conjuntos de objetos a partir de la búsqueda de regularidades entre los objetos que conforman dichos conjuntos. De alguna manera, esto podría ser confundido con los algoritmos de clasificación, si bien tienen una funcionalidad bastante diferente. Un algoritmo de *clustering* no busca la clasificación de objetos per se, sino la búsqueda de los atributos que segmentan dichos datos en grupos heterogéneos entre conjuntos, y homogéneos en el conjunto. Por tanto, es común encontrar el uso combinado de este tipo de algoritmos, dónde inicialmente un algoritmo de *clustering* identifica los atributos que posteriormente serán utilizados por el algoritmo clasificador. En este caso, en el contexto del aprendizaje automático, esto se trataría de un algoritmo no supervisado (i.e. El algoritmo aprende solo a partir del input, sin necesidad de indicarle un output esperada).

- **Técnicas de asociación:**

  Este tipo de técnicas busca generar reglas de asociación del tipo $\{A, B, \dots\} \rightarrow C$, dónde el estado de los elementos en un conjunto permite afirma con un nivel

de probabilidad establecido previamente, cuál será el estado de otro elemento individual. Las reglas identificadas por estos algoritmos pueden ser utilizadas posteriormente para realizar predicciones sobre el mismo conjunto de elementos.

- **Técnicas de predicción:**

  Las técnicas de predicción son aquellas que hacen uso del análisis de tendencia, modelos de regresión, clasificación, reconocimiento de patrones y relaciones sobre eventos pasados con el fin de poder hacer predicciones de dicho evento en el futuro.

- **Técnicas de patrones secuenciales:**

  Finalmente tenemos las técnicas de patrones secuenciales, las cuales son comúnmente aplicadas sobre datos a largo plazo, en las que poder identificar tendencias y ocurrencias de eventos similares que son repetidas cíclicamente en el tiempo.

Por tanto ya conociendo de forma genérica cuáles son las técnicas de minería de datos más comunes, podríamos hacer un primer acercamiento sobre cómo dar solución a los casos presentados al comienzo del capítulo.

En el primer caso, nos encontrábamos con la base de datos de un supermercado que contiene la información de cada una de las compras realizada por sus clientes, así como los productos que la conforman, y de ello, se quiere obtener información relevante sobre los hábitos de consumo de los clientes para así poder concretar una campaña de marketing más efectiva. Este ejemplo es típicamente conocido como el Problema de la Cesta de la Compra, y para resolverlo, se hacen uso de las **técnicas de asociación** anteriormente vistas.

Como se puede ver en la *Figura 5*, partiendo de los productos que integran cada compra, se pueden aplicar técnicas de asociación que evaluarán la cantidad de veces que se produce la aparición conjunta de subgrupos de productos. Del análisis de esto podremos obtener como resultado asociaciones del tipo, *cuando un cliente compra cerveza, a su vez, también compra nachos*, con una probabilidad de que esto suceda superior a un valor establecido previamente a la ejecución del análisis. Con esta asociación en su poder, el gerente del supermercado contará con información valiosa que le permitirá tomar decisiones tales como reubicar el lineal de *nachos* cerca del punto de venta de las *cervezas*.

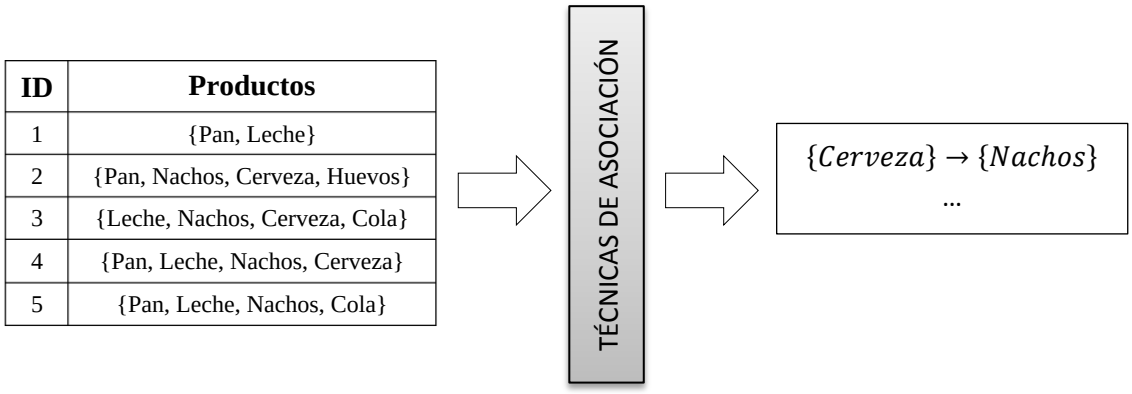

Figura 5: Proceso de Minería de Datos para el Problema de la Cesta de la Compra. [15]

En el segundo caso, nos situamos dentro del departamento de recursos humanos de una gran compañía que quiere localizar a aquellos trabajadores brillantes dentro de la organización con el fin de incentivarlos y situarlos en una posición adecuada a sus cualidades, siendo esto uno de los mayores desafíos en la gestión de recursos humanos a día de hoy. Para hallar una solución, se cuenta con un gran volumen de información sobre los trabajadores (datos personales, puesto de trabajo, rendimiento del trabajo, años en la compañía, habilidades y conocimientos, etc.) sobre el cual podemos aplicar diferentes técnicas de minería de datos en función de los resultados que quisiéramos obtener. Por ejemplo, podríamos utilizar técnicas de *clustering* para identificar patrones que agrupen a empleados con similares características de rendimiento, habilidades, etc. Si utilizáramos las técnicas de asociación, previamente vistas, podríamos ser capaces de identificar asociaciones que nos señalen qué características de los trabajadores son las más idóneas para según qué puestos de trabajos, basándonos en los rendimientos obtenidos en ocasiones previas. También, mediante técnicas de predicción y clasificación, podríamos conseguir predecir cuál será la proyección de progreso del trabajador en el medio y largo plazo, así como también podríamos asignarle un perfil determinado basándonos en los resultados de la clasificación. Por tanto, es evidente la utilidad que nos aporta el uso de las diferentes herramientas de minería de datos para su aplicación en el campo de la gestión de recursos humanos.

Una solución planteada por Hamidah Jantan (2010) [16] resuelve este problema mediante el uso de técnicas de árboles de decisión, utilizando un tipo específico de algoritmo denominado C4.5, con el que se obtiene un árbol clasificador, cuyo uso permite asistir si se debe promocionar a un trabajador. Este árbol ha mostrado un alto rendimiento en la ejecución de la tarea de promocionar a aquellos trabajadores brillantes, generando un alto valor para la organización que ahora cuenta con un sistema de incentivos más eficiente. Además, su uso es bastante sencillo, pudiendo ser

interpretado por cualquier persona de la organización, sin ser necesario conocimientos en el campo del análisis de datos.

Finalmente, en el tercer caso hablábamos de los mercados bursátiles, en los que dada la complejidad de sus estructuras, la cantidad de agentes que participan y las variables que afectan al valor de los productos que cotizan en ellos, hacen bastante compleja la tarea de poder realizar predicciones que nos puedan beneficiar en el proceso de gestionar nuestras operaciones bursátiles. Cabría preguntarse si en esta ocasión las herramientas de minería de datos podrán ofrecernos un resultado notable en esta materia. Este es el objetivo que se ha propuesto este proyecto, y que se seguirá desarrollando en los próximos capítulos.

4.2 Algoritmos aplicados a los mercados bursátiles.

Hace tiempo que los parqués de los principales mercados bursátiles dejaron de mostrarse como espacios abarrotados de *brokers* que, a gritos, intentaban hacer oír sus opciones de compra y venta. A día de hoy, el sonido que acompaña a gran parte del volumen de las operaciones que se realizan en los mercados, es el de los ventiladores que refrigeran a los servidores alojados en los grandes centros de procesamiento encargados de ejecutar avanzados algoritmos de *trading*. Nos encontramos, por tanto, frente al efecto de la utilización de algoritmos informáticos y modelos matemáticos aplicados a los mercados bursátiles, o tal y como se conoce, trading algorítmico. Es tal la presencia de estos sistemas computacionales en los procesos de inversión y especulación, que se estimó que un tercio de las operaciones producidas los mercados de la Unión Europea y Estados Unidos en 2006 fueron llevadas a cabo por estos algoritmos [17].

La ventaja que los sistemas automáticos de trading pueden aportar es evidente, pues permiten automatizar y procesar toda la información necesaria en el mercado para generar una gran cantidad de señales de compra o venta de forma mucho más rápida y eficiente que un agente humano. De hecho, en términos de rapidez, cabe mencionar que la mayor parte de las operaciones llevadas por algoritmos que se realizan a día de hoy, pertenecen a los que se conoce como negociación de alta frecuencia o HFT (siglas de su nombre en inglés, *High-Frequency Trading).* Una modalidad de *trading* que busca obtener beneficios mediante la colocación de un gran número de operaciones a muy alta velocidad (en cuestión de milisegundos) en múltiples mercados de forma simultánea.

En cualquier caso, operar a alta velocidad no es la única ventaja que los *traders* encuentran en la utilización de estos sistemas. Por ejemplo, su uso beneficia a aquellos inversores a medio y largo plazo que buscan comprar o vender en grandes volúmenes,

sin querer que sus operaciones influencien a los precios en el mercado. El objetivo de los algoritmos que implementan este tipo de estrategias es el de descomponer un gran pedido en pequeñas órdenes para después de forma sistemática, ejecutarlas en el mercado de forma pasiva, sin afectar al precio, basándose en datos históricos del volumen (estrategias VWAP) o en periodos de tiempo constantes (estrategias TWAP). La ventaja de usar este tipo de estrategias es que permite al *trader* profesional el poder realizar operaciones de gran volumen a un precio cercano al verdadero precio de mercado, sin que pudiera alterarlo y perjudicarle en su operativa.

Otro ejemplo de la utilización de algoritmos es la implementación de estrategias que explotan oportunidades de arbitraje. Este tipo de oportunidades se producen cuándo hay una diferencia de precios sustancial entre mercados correlacionados. Esta divergencia que se produce, permite realizar una operación de compra por el precio más bajo y una venta por el más alto de forma simultánea. Como es de prever, el trabajo de análisis en tiempo real de tantas variables operando de forma simultánea es una tarea que un computador puede ejecutar muy eficientemente.

Estos ejemplos de uso acompañan a otros tantos, que demuestran la gran introducción y aplicación que ha tenido el *trading algorítmico* a lo largo de las últimas dos décadas. Sin embargo, y como ya se ha adelantado previamente, la principal ventaja que aporta la utilización de los sistemas informáticos es la alta velocidad de cómputo que estos ofrecen. Las velocidades que se manejan superan considerablemente la barrera de tiempo sobre la cual un ser humano puede tener reacción, y por tanto, control de las acciones llevadas a cabo por los algoritmos. Hablamos de escalas basadas antes en milisegundos, ahora en microsegundos, y en un futuro cercano, en nanosegundos (la operación bursátil más rápida jamás registrada es de 740 nanosegundos [18]). Los mercados ya no son operados manualmente por personas, sino que hemos dejado el control a máquinas que realizan este trabajo de una forma que es imperceptible para nosotros. Las grandes empresas dedicadas al HFT, invierten grandes cantidades de dinero en localizar físicamente sus centros de procesamiento lo más cerca posible de los principales mercados bursátiles, con el único fin de reducir la latencia de sus operaciones y permitir que los algoritmos sean más eficientes que los de su competencia, y en consecuencia, generen un mayor beneficio.

Es evidente que las innegables ventajas que ofrece la utilización de estos sistemas en los mercados bursátiles justifican su uso. Pero por otro lado, habría que preguntarse hasta qué punto estamos dispuestos a ceder el control a las máquinas en la especulación del precio de productos básicos que pueden ser tan esenciales como el agua, el gas o el petróleo. Les permitimos operar a velocidades inaccesibles para nosotros, ejecutando patrones cada vez más complejos que hacen que su funcionamiento sea cada vez más incomprensible. El funcionamiento de estos algoritmos no es perfecto, en muchas ocasiones presentan inestabilidad y acaban

fallando. Un ejemplo de ello es lo que se conoce como el *Flash Crash* de las 02:45: La segunda mayor caída porcentual de la bolsa de Wall Street, sólo superada por la caída producida en Octubre de 2008, al inicio de la crisis financiera.

Figura 6: Gráfica diaría del *Flash Crash* de 2010.

La quiebra de billones de dólares producida durante el *Flash Crash* en 2010 en el mercado estadounidense, no sería de nuestro interés, sino fuera porque el intervalo de tiempo en el que se experimentó su colapso y posterior recuperación fue de 5 minutos. Es decir, un colapso de los principales mercados financieros del mundo que se produjo en un muy corto intervalo de tiempo y que a día de hoy, si bien se sabe que se produjo por la desestabilización de los algoritmos de alta frecuencia que operaban en dichos mercados, todavía se desconoce cuáles fueron las circunstancias que hicieron que el mercado acabará por colapsar.

Por tanto, es importante reflexionar hasta qué punto estamos dispuestos a explotar la potencia que nos ofrece el *trading* algorítmico, dado el riesgo que supone el dejar la especulación de nuestros mercados, en manos de máquinas que se alejan cada vez más de nuestro control. Al margen de estas inquietudes, queda demostrada la existencia de una realidad dónde la tecnología de la computación y los algoritmos informáticos cada vez juegan un papel más importante en la operativa producida en todos los mercados bursátiles a nivel mundial.

4.3 Estado del arte.

Ser capaz de predecir el valor del precio de cualquier activo cotizado en un mercado bursátil ha sido un tema de gran interés en el sector económico y financiero, y en términos generales, de carácter global, dado el gran peso que tienen estos campos en

las sociedades modernas. A lo largo del siglo pasado, este objetivo ha supuesto un gran reto para los investigadores que han centrado sus esfuerzos en explicar el comportamiento de la estructura del precio para poder así, predecir su valor futuro. La respuesta a este desafío ha acabado por dividir en tres grandes enfoques las opiniones de los estudiosos, que de forma antagónica, plantean como solución al problema de la predicción de precios el análisis fundamental, el análisis técnico y la hipótesis de los mercados eficientes [19].

El enfoque planteado por los defensores del análisis fundamental considera que el valor que puede tomar un activo queda determinado exclusivamente por los factores económicos relativos, tanto internos de la empresa, como externos del entorno económico en general, del activo cotizado. Por tanto, un enfoque fundamentalista hará uso de toda la información económica y financiera que pudiera influir sobre el mercado a la hora de determinar el punto de equilibrio entre la oferta y la demanda. Toda esta información permitirá calcular el valor esencial o fundamental de dicho activo, es decir, su valor real, y en las diferencias con su valor de mercado se encontrarán las bases para la predicción futura del precio, confiando en que en un futuro dicha diferencia sea equilibrada por el mercado. La vanguardia en la aplicación de las tecnologías de la información para el análisis fundamental, persigue la creación de sistemas inteligentes capaces de comprender semánticamente las noticias sobre eventos que pudieran influir en el sector financiero [20], para así poder estimar de manera inmediata su relación con los activos cotizados, y traducir su impacto en reglas de contratación eficientes.

Por el contrario, el enfoque basado en el análisis técnico no busca estimar el valor real del activo financiero, sino su evolución histórica en el mercado, haciendo uso principalmente de gráficas e indicadores estadísticos sobre el comportamiento del precio. Este enfoque no descarta la existencia de factores fundamentales, sino que los considera integrados dentro de la estructura del precio, y por tanto, estudia sus consecuencias en vez de dichos factores. El análisis técnico hace uso del análisis de patrones y tendencias en las gráficas del precio, así como la de los indicadores calculados a partir de otras variables características (volumen, interés abierto, etc.) para intentar predecir su evolución futura. Las ventajas obtenidas por el uso de herramientas informáticas son evidentes, como ya se vio en el epígrafe anterior, aportando velocidad y potencia de análisis en la búsqueda de patrones, correlaciones, tendencias, etc.

Antepuesta a los dos enfoques anteriores, y por tanto a la naturaleza de este proyecto, la hipótesis de los mercados eficientes descarta la posibilidad de predicción futura de los precios mediante el uso del análisis fundamental o técnico. Esto se explica basándose en la eficiencia de los mercados, que absorben instantáneamente cualquier información nueva, incorporándola al precio, el cual, representa el valor correcto del

activo cotizado. Las modificaciones en el precio se consideran perturbaciones aleatorias independientes, que son imposibles de predecir. Este enfoque, por tanto, descarta la posibilidad de tener éxito en la obtención de grandes beneficios de forma prolongada en el tiempo, mediante la predicción de precios.

Sin embargo, ha surgido un intenso debate en la literatura financiera acerca de la capacidad predictiva del análisis técnico, al ponerse de manifiesto que las rentabilidades obtenidas por las reglas de contratación, utilizadas comúnmente por el enfoque técnico, no pueden ser explicadas por los modelos econométricos más usados en finanzas.

En la actualidad, esta visión negativa en la capacidad predictiva de cualquier tipo de técnicas sobre el mercado, se ha visto empíricamente refutada en cierto grado, al demostrarse con nuevos enfoques y metodologías la posibilidad de poder obtener rendimientos económicos a partir de la predicción de los valores bursátiles, siendo la utilización de técnicas de minería de datos un gran ejemplo de ello. En la intersección dónde se cruzan el mundo de las finanzas y la minería de datos, es extensa la cantidad de publicaciones y estudios dónde se hace uso de alguna de estas técnicas para la labor de predicción, haciendo uso del gran volumen de información generada por el sector financiero.

En este sentido existe una amplia experiencia empírica en la que, a modo ilustrativo y entre muchos otros, cabría citar algunos artículos. En el trabajo de Ruxanda *et al*. (2013) [21], en el que hicieron uso de redes neuronales artificiales (ANN) entrenadas con datos históricos de la bolsa de Rumanía para su posterior testeo en la bolsa Croata. Algunas de las conclusiones que llegaron con su estudio indican que si bien el uso de ANNs no permite una predicción muy precisa del precio, sí se puede obtener un valor estimado cercano al valor real, en comparación con otras técnicas.

Un trabajo pionero en el uso de redes neuronales artificiales para la predicción de precios al cierre se encuentra en Fernández-Rodríguez *et al.* (1999) [22], dónde se investiga el rendimiento de reglas de contratación basadas en la información proporcionada por *ANNs*. Los resultados obtenidos demuestran que en ausencia de costes, el rendimiento obtenido durante periodos bajistas del mercado al utilizar el predictor artificial es mayor a estrategias del tipo *buy-and-hold*, dónde una opción de compra es mantenida a lo largo del tiempo. No ofrece sin embargo rendimientos mayores cuándo las características del mercado son contrarias y nos encontramos ante un mercado alcista.

Otro ejemplo de uso individual de la minería de datos, es el trabajo de Al-Radaideh *et al.* (2013) [23], en el que hicieron uso de árboles de decisión para crear un clasificador a partir de datos históricos de precios, que les permitiera decidir cuál era el mejor momento para comprar o vender acciones en el mercado. Los resultados obtenidos

demostraron un bajo rendimiento en la utilización exclusiva de árboles de decisión, obteniendo una tasa de acierto media poco significativa, cercana al 50%.

Es por eso que, analizando la arquitectura de los sistemas más avanzados de *trading* algorítmico utilizados hoy en día, se demuestra que las ventajas que se pueden obtener al aplicar técnicas de minería de datos surgen cuándo estas son combinadas de manera sinérgica, y no a partir de su utilización de forma aislada.

Un ejemplo de este tipo de sistemas, es el presentado por Huang y Lin (2014) [24]. Un sistema que, integrando varias técnicas de minería de datos, ha demostrado unos resultados bastante prometedores en la predicción de los índices del mercado, obteniendo en sus dos casos de estudios unos rendimientos en la inversión realizada del 53.6% y 128.4%.

Por tanto, el estado del arte en el estudio sobre los procesos de minería de datos para la predicción bursátil, se orienta actualmente hacia lo que se conoce como Minería de Datos Híbrida (*Hybrid Data Mining, HDM).* Esto es, la integración de diversas técnicas funcionando conjuntamente, con el fin de conformar complejos sistemas avanzados de *trading,* que normalmente acaban por incorporarse a las herramientas utilizadas por las grandes empresas financieras.

4.4 Redes neuronales artificiales, *ANNs*.

Dentro del campo del aprendizaje automático y las ciencias computacionales, nos encontramos con las redes neuronales artificiales (*Artificial Neural Networks, ANNs*) como una de las técnicas más representativas dentro de su género. Mediante su utilización, se facilita el descubrimiento automático de modelos, generados a partir de un conjunto básico de ejemplos de entrada, lo cual podría suponer una tarea de bastante complejidad si se hiciera mediante el uso de otro tipo de herramientas más convencionales. Su aplicación, actualmente, beneficia a numerosos campos de estudios tales como el de la percepción artificial (ej. detección de caras, reconocimiento de escritura y voz, visión artificial, etc.), la medicina (ej. diagnóstico médico, detección de cáncer, análisis bioquímicos, etc.) [25][26], procesos industriales (control de maquinaria, diagnóstico de defectos, ajustes de temperatura, etc.), o el campo de la economía y las finanzas (ej. sistemas automáticos de *trading*, calificación de deuda, segmentación de mercado, etc.), entre otros muchos ejemplos. Su gran interés surge, por tanto, dado el buen rendimiento que demuestra esta tecnología en la resolución de problemas que son bastantes complejos de definir algorítmicamente.

Tal y como sugiere su nombre, la estructura de una red neural artificial está inspirada en cómo funciona el sistema nervioso de los animales, dónde el desempeño de una tarea se ejecuta mediante la descomposición y tratamiento de subtareas más simple

por parte de conjuntos de neuronas que se encuentran interconectadas y que se comunican entre ellas. Al igual que un cerebro animal, las redes neuronales se especializan en el procesamiento paralelo y aprendizaje de reconocimiento de patrones, que llevado a mayor escala, permite la resolución mediante técnicas computacionales de tareas más complejas. Su descubrimiento se remonta a 1958 cuándo el psicólogo Frank Rosenblatt diseñó la primera red neuronal artificial, llamada *perceptrón*, con la intención de simular cómo el cerebro humano procesaba la información obtenida a través de la visión y como aprendía a reconocer objetos. Precisamente, la aplicación de los conocimientos de la psicología y la neurología en el campo de la computación, ha permitido no solo el desarrollo y uso de estas herramientas, sino también el poder obtener un mayor entendimiento de cómo funciona el sistema neuronal biológico y la cognición humana.

Una red neuronal artificial de tipo *feed-forward* opera mediante la creación de interconexiones entre unidades de procesamiento simple, denominadas *neuronas*. Como se puede ver en la *Figura 7,* la distribución de cada una de estas neuronas puede organizarse en diferentes niveles de procesamiento o capas, dónde cada neurona perteneciente a una capa determinada está interconectada con todas las neuronas de la capa anterior y/o la capa posterior.

La primera capa, es denominada la *capa de entrada*, y es la que recibirá el vector con los valores de los parámetros a ser procesados, sobre los cuáles queremos generar el modelo. Las capas intermedias de la red neuronal son denominadas *capas ocultas*, que serán las encargadas de procesar y memorizar la información llegada de las neuronas de otras capas. Finalmente, la última capa es denominada la *capa de salida*, y será la encargada de, una vez se han realizado los cómputos a través de todos los niveles de la arquitectura, devolver el vector resultado.

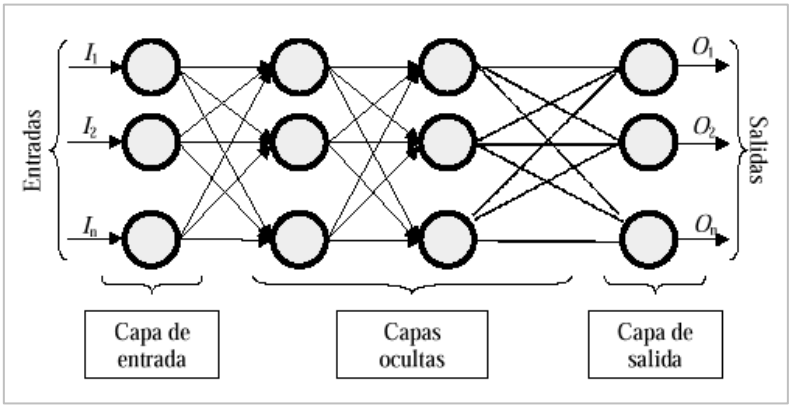

Figura 7: Esquema general de una red neuronal.

La composición interna de una neurona, está conformada por un conjunto de funciones matemáticas encargadas de realizar cálculos a partir de los datos de entradas recibidos por la neurona y generando un valor de salida determinado (*ver Figura 8)*. Para ello, a cada valor de entrada se le otorga un *peso* determinado en función de cual sea su importancia en el proceso de cálculo de esa neurona. Inicialmente el valor de estos pesos se asigna de manera aleatoria, y posteriormente según se ejecuta la fase de entrenamiento, estos valores se van reajustando iterativamente con el fin de minimizar el criterio de coste establecido. Estos parámetros son los que en términos finales dotan de flexibilidad de aprendizaje a la red neuronal. El cálculo realizado sobre los valores de entrada ponderados según los pesos asociados a cada una de las uniones entre neuronas (sinapsis) es lo que se conoce como la *función de propagación* de la neurona, normalmente implementado como el sumatorio de las entradas ponderadas. El valor de salida generado por la *función de propagación,* servirá como valor de entrada para computar lo que se denomina la *función de activación*, que generará una salida específica según el tipo de cálculo matemático realizado. La función de activación más común, es la denominada función sigmoidea [27], la cual está asociada con muchos procesos naturales, siendo uno de ellos la curva de aprendizaje en procesos complejos. Esta función acotará los resultados de la neurona a valores continuos entre 0 y 1, con una evolución logística en forma de S. Otro tipo de funciones también utilizadas son capaces de acotar los resultados a valores continuos entre -1 y 1 (función de tangente hiperbólica [28]) o la generación de salidas binarias.

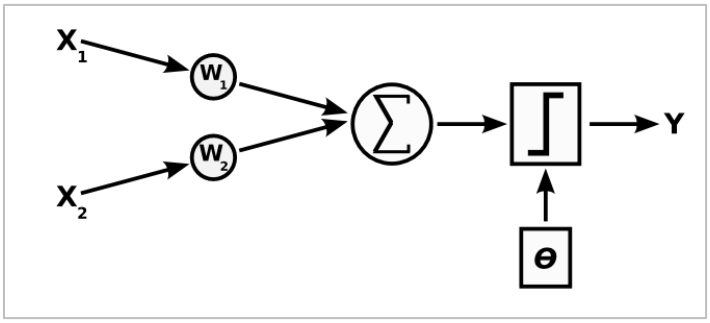

Figura 8: Esquema general de una neurona artificial.

La posibilidad de utilizar diferentes combinaciones de funciones de propagación y activación, así como los diferentes diseños de arquitecturas y algoritmos de aprendizaje existentes, hacen que de cara a un análisis se pueda elegir entre un amplio surtido de modelos de redes neuronales, que nos aportarán rendimientos diferentes, según la naturaleza de los datos y de la información que queramos obtener. De todos los modelos, el más utilizado es el que se conoce como Perceptrón Multicapa (*Multilayer Perceptron,* MLP) [29], que cumple con la estructura general que se ha presentado en este capítulo: múltiples capas (c. de entrada, c. ocultas, c. de salida),

conexiones que no forman ciclos (red neuronal *feed-forward)*, función de activación no lineal (ej. sigmoidea, hiperbólica, etc.) y aprendizaje mediante la técnica de retropropagación o *backpropagation.*

Esta última técnica se trata de un algoritmo de aprendizaje, y permite a la red neuronal reajustar los pesos utilizados para ponderar las entradas de cada una de las neuronas, con el fin de minimizar la suma de los errores al cuadrado entre la salida esperada y la proporcionada por la red, realizando por tanto, un proceso de memorizado y aprendizaje. Se denomina retropropagación porque este entrenamiento se realiza en dos fases: una primera, dónde se calcula el valor de salida a partir de los valores computados en cada neurona para cada valor de entrada de la red neuronal, y una segunda, dónde se compara el valor obtenido con el valor esperado y se calcula una señal de error para cada una de las salidas, que es propagada hacia atrás en la topología de la red, y que reajusta los valores de los pesos. La ejecución repetida de estas fases permite a la red neuronal reajustarse hasta poder minimizar los errores de predicción sobre el conjunto de datos. Sin embargo, es importante señalar que el mínimo alcanzado se trata de un mínimo local y no absoluto, y que por tanto, no representa la configuración óptima de pesos que pudiera alcanzar la red neuronal. El sesgo en los resultados que pudiera derivarse de alcanzar siempre un mismo mínimo local, se evita mediante la inicialización aleatoria de los pesos al comienzo del entrenamiento. Gracias a esta técnica de entrenamiento, mediante su ejecución iterativa, acabamos por obtener una red neuronal entrenada que nos servirá como modelo sobre el cual comenzar a realizar predicciones.

Las redes neuronales artificiales son, por tanto, potentes herramientas de aprendizaje automático que nos permiten a través de la realización de numerosos cálculos interconectados, la estimación de modelos basados en formas funcionales complejas, sobre los cual realizar predicciones. Si bien la complejidad de las funciones estimadas hace que sean herramientas difíciles de comprender internamente, su utilización es amplia y variada en numerosos sectores, demostrando un rendimiento que justifica su uso.

4.5 El problema del *overfitting*.

En el proceso de análisis de los datos, en ocasiones se hace uso de técnicas de aprendizaje automático (ej. Redes neuronales), dónde los algoritmos que se utilizan son capaces de realizar predicciones a partir del aprendizaje adquirido tras una fase de entrenamiento. El funcionamiento de estos algoritmos, permite que en la fase de entrenamiento, estos, puedan reajustar sus parámetros de funcionamiento con el fin

de maximizar el resultado de predicción sobre el conjunto de datos con el que son entrenados.

El problema del sobreajuste surge cuándo el algoritmo utilizado es sobreentrenado o se le otorga un alto grado de libertad, lo que le permite la generación de formas funcionales muy complejas que no solo se ajustan a la tendencia de los datos, sino que también recogen las posibles perturbaciones aleatorias que estos pudieran contener. Cuando esto sucede, normalmente se acaba obteniendo un modelo completamente optimizado para el conjunto de datos de entrenamiento, pero que sin embargo, tiene una pobre capacidad de predicción cuándo se hace uso de datos diferentes a los utilizados en el entrenamiento. Un ejemplo de una función sobreajustada sería la que se presenta en la *Figura 9.*

Como podemos ver, contamos con una nube de puntos que tiene una evidente tendencia lineal decreciente, dónde la mayor parte de los datos podrían ser explicados mediante un modelo de regresión lineal simple. Sin embargo, tras la aplicación de un algoritmo de aprendizaje automático para la estimación del modelo, la función estimada no consiste en una línea simple, sino que ha generado una curva de mayor complejidad capaz de ajustarse exactamente al conjunto de datos, no solo a su tendencia regresiva, sino también a los errores aleatorios que estos pudieran tener. Diremos entonces, que la función estimada es una función sobreajustada.

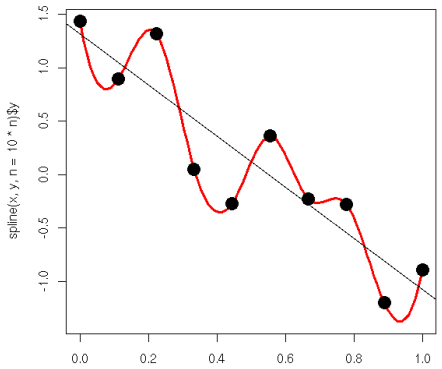

Figura 9: Ejemplo de una función sobreajustada.

Por tanto, para garantizar la calidad del modelo que se va a generar, será necesario atender a la posible aparición del sobreajuste. Para ello, y como ya se ha adelantado, la estrategia que se va a utilizar será la de generar dos subconjuntos de datos diferentes, uno de entrenamiento y otro de prueba. La finalidad de esto, es poder detectar la existencia de *overfitting* mediante el análisis de los errores de predicción de ambos subconjuntos. Un modelo que presente un problema de sobreajuste, tenderá a maximizar su capacidad predicción y por tanto a minimizar la suma de sus errores, calculados estos como la diferencia entre el valor estimado y el valor real de cada muestra. Cuándo se analizan los errores de ambos subconjuntos y se observa que el

error acumulado del subconjunto de prueba es notablemente superior al del subconjunto de entrenamiento, podremos deducir la existencia de un problema de sobreajuste, y por tanto, podremos iniciar estrategias para reducir su efecto.

Podemos ver en la *Figura 10* un caso de sobreajuste producido en un modelo estimado a partir de los datos bursátiles con los que estamos trabajando. La gráfica roja (gráfico superior) y verde (gráfico inferior) representan los resultados reales, o esperados, del valor de cierre del IBEX35 para el subconjunto de prueba y entrenamiento, respectivamente. Por otra parte, la gráfica azul representa el valor estimado por la red neuronal artificial para cada uno de los subconjuntos, tras haber sido entrenada.

Como se puede observar, la predicción que se realiza sobre el subconjunto de datos de entrenamiento se ajusta perfectamente a los valores reales del cierre de la bolsa, con un error cuadrático medio despreciable (ECM = 0.00828). Por el contrario, la función de predicción estimada sobre el conjunto de datos prueba, los cuales no fueron utilizados durante la fase de entrenamiento de la red, muestran como existe una notable diferencia entre los valores reales y los estimados, con un elevado error asociado (ECM = 2.4936). Sabiendo esto, podemos deducir la existencia de un problema de sobreajuste, que en este caso concreto se está produciendo al dotar a la red neuronal artificial de excesiva libertad en la creación de funciones complejas.

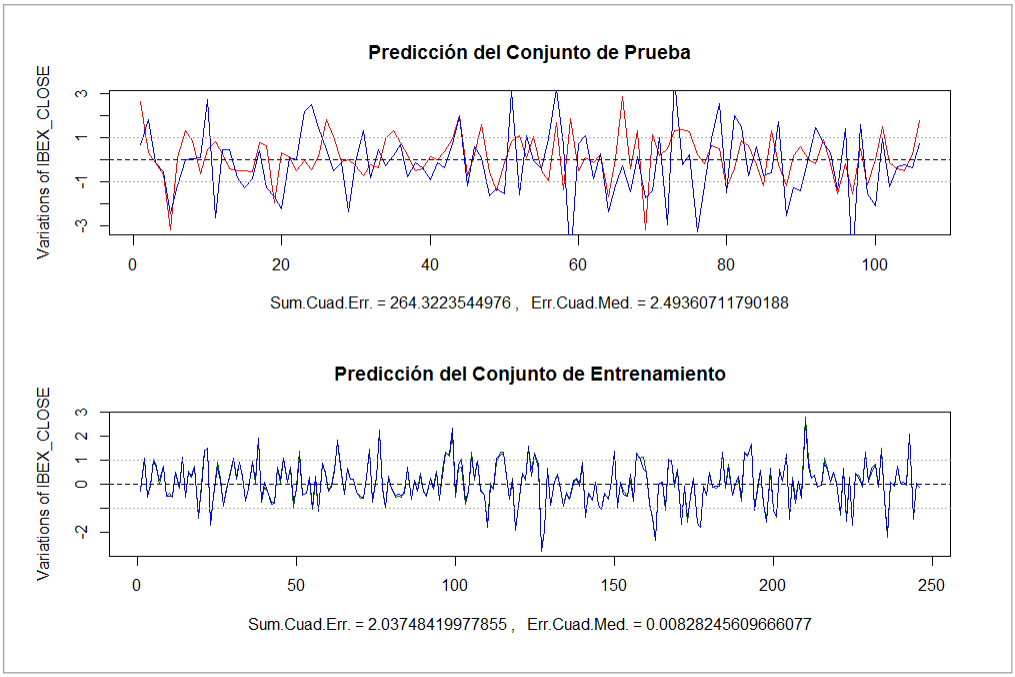

Figura 10: Ejemplo de problema de sobreajuste sobre el modelo bursátil estimado.

Si quisiéramos, por tanto, reducir el problema de *overfitting* tendríamos que reajustar los parámetros de la red neuronal y realizar un nuevo proceso de entrenamiento que nos genere un nuevo modelo. Como podemos ver en la *Figura 11,* este nuevo modelo,

dónde se ha limitado los grados de libertad de la red neuronal, recoge un mejor ajuste de la función de predicción en el subconjunto de prueba (ECM = 0.69581) en detrimento del subconjunto de entrenamiento. (ECM = 0.36301).

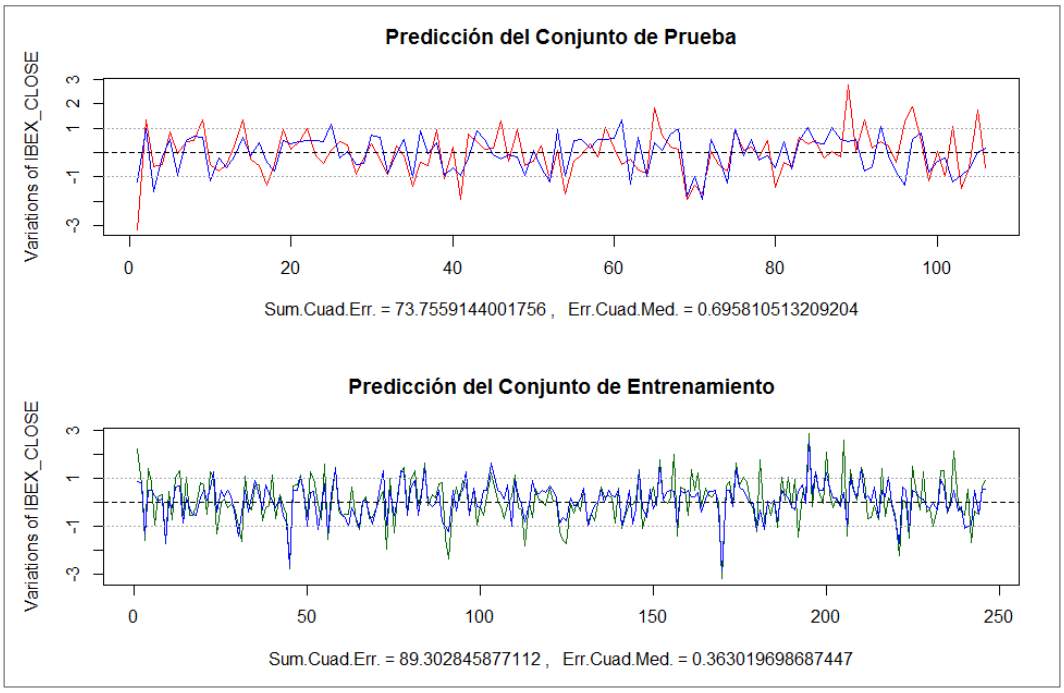

Figura 11: Ejemplo de resolución del problema de sobreajuste.

4.6. El ratio de *Sharpe*.

Es importante a la hora de gestionar nuestra cartera de inversión y en la evaluación de nuestras estrategias de *trading,* contar con los indicadores adecuados para la valoración del rendimiento que estas pudieran tener. Dicho rendimiento puede ser valorado siguiendo diferentes criterios, siendo uno de los más habituales el que hace uso de las potenciales ganancias, o rentabilidad de la inversión.

Sin embargo, si bien la rentabilidad que podamos obtener de nuestras inversiones es de carácter necesario para la valoración del rendimiento de nuestra estrategia, no se puede obviar otra variable de también gran importancia: el riesgo.

Por ilustrar esto, supongamos que iniciamos nuestra operativa con un balance de 1000€, y decidimos invertir sobre un fondo que nos promete una rentabilidad del 10% anual. A priori pudiera parecer una opción de bastante interés, a partir de la cual, dentro de un año obtendremos unos beneficios de 100€. Sin embargo, quizás no suponga una opción tan interesante al descubrir que la volatilidad de este producto financiero tiene una volatilidad del 50%, lo que implica asumir unos niveles de riesgo bastante elevados.

Por tanto, es evidente que a la hora de analizar el rendimiento de nuestras estrategias será importante no contar para elaborar nuestro indicador solo con el posible beneficio que obtendremos, sino también con el posible riesgo en el que vamos a incurrir.

Un indicador capaz de hacer uso de ambos criterios, es el conocido ratio de Sharpe, el cual es capaz de medir la rentabilidad obtenida por cada unidad de riesgo soportado en una inversión. Gracias a su uso, podremos realizar comparaciones en igualdad de riesgo entre diferentes estrategias bursátiles.

Lo computaremos a partir de la siguiente expresión: [30]

$$\textbf{Ratio de Sharpe} = \frac{\boldsymbol{R_i - R_f}}{\boldsymbol{\sigma_i}}$$

$\boldsymbol{R_i} = \textit{Rentabilidad de la inversión.}$
$\boldsymbol{R_f} = \textit{Rentabilidad de un activo sin riesgo.}$
$\boldsymbol{\sigma_i} = \textit{Volatilidad de la inversión}$

A la hora de interpretar este ratio, se observa que entre mayor sea su valor mejor es la rentabilidad del producto comparado directamente con la cantidad de riesgo que se ha asumido en la inversión. Sin embargo, un valor inferior a la unidad implica que el rendimiento del activo es inferior al riesgo que se asume, pudiendo ser tanto por una baja rentabilidad o por una alta asunción de riesgo.

- Capítulo 5 –
Hipótesis de trabajo.

Como ya se ha adelantado, el objetivo de este trabajo es el de realizar un proceso de minería de datos aplicado a los mercados bursátiles. Es materia de este capítulo el presentar la hipótesis de trabajo elegida para la ejecución de este proceso.

5.1 Presentación del problema.

De manera general, el objetivo propuesto para desarrollar de forma satisfactoria el proceso de descubrimiento de conocimiento, es el de la obtención de información relevante que nos pueda asistir durante la negociación bursátil. Para poder alcanzar este objetivo, se ha planteado una hipótesis de trabajo que se buscará demostrar mediante la aplicación de las adecuadas herramientas de minería de datos.

Para plantear la hipótesis, tomamos como punto de partida el contexto globalizado en el que cotizan los mercados bursátiles, cuyos productos financieros, son sensibles ante cualquier información que se genere en cualquier zona del planeta. Por ejemplo, es común el escenario en el que, ante la publicación de unos resultados macroeconómicos en EE.UU, el efecto de dicha noticia, no solo afecte a los mercados de Wall Street, sino que esto acabe afectando al resto de mercados y economías del mundo.

Por tanto, es bien conocida en la literatura la existencia de correlaciones entre los mercados de los diferentes países, dónde el valor de cotización de un índice en un mercado puede estar correlacionado con el valor de uno o varios índices cotizados previamente en otros mercados. En este sentido se plantea como hipótesis de trabajo de esta memoria la existencia de causalidad, posiblemente de naturaleza no lineal, desde los diferentes índices bursátiles hacia el IBEX35. La existencia de dicha causalidad podría permitir la mejorar de la capacidad predictiva del modelo de redes neuronales artificiales introduciendo la información adicional procedente de los otros índices cotizados previamente en otros mercados.

La evaluación de la mejora de la capacidad predictiva de las cotizaciones al cierre del IBEX35 que trae consigo la introducción de información procedente de otros índices bursátiles puede evaluarse de dos formas diferentes. En primer lugar se puede realizar una evaluación desde el punto de vista estadístico, siendo el criterio del menor error cuadrático medio uno de los procedimientos más usuales en dicha evaluación. En

segundo lugar puede realizarse una evaluación desde el punto de vista de la rentabilidad económica de la posible mejora predictiva; para ello habrá que transformar las predicciones en señales de una regla técnica de contratación y comparar las rentabilidades obtenidas en la predicción del IBEX35 con y sin información adicional sobre el resto de los mercados. Con el fin de considerar el balance entre rentabilidad y riesgo de las rentabilidades obtenidas el estadístico más empleado en este tipo de comparaciones es la ratio de Sharpe, que proporciona la rentabilidad por unidad de riesgo.

5.2 Diseño de la solución.

Una vez tenemos planteada la hipótesis sobre la que queremos trabajar, se procede a describir la solución que nos va a permitir demostrar o rechazar dicha hipótesis, así como el diseño del proceso de minería de datos que se llevará a cabo para ello.

Puesto que nuestro interés se centra en encontrar la existencia de causalidad, posiblemente de naturaleza no lineal, entre la cotización del mercado español y la actividad del resto de mercados, será necesario identificar aquellos mercados que por proximidad temporal, pudieran tener un mayor efecto en el IBEX35. Para ello, se ha procedido a representar en un gráfico los horarios de apertura y cierre de los, según su volumen de capitalización, principales mercados del mundo [31]. (Ver *Figura* 12).

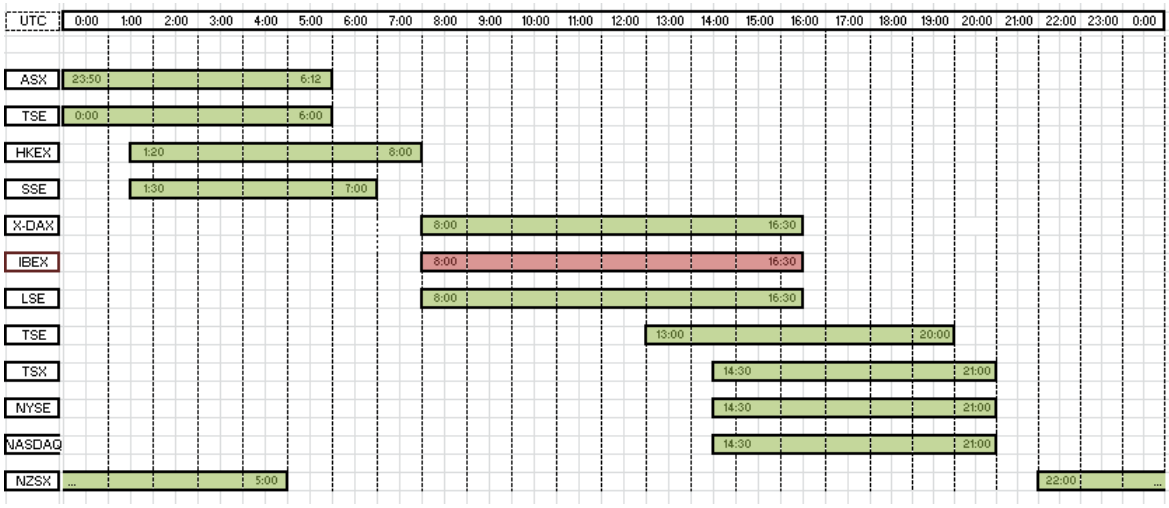

Figura 12: Franjas de apertura y cierre de los principales mercados mundiales.

Tras analizar el gráfico, se observa que los mercados con mayor relevancia y proximidad al horario de la bolsa española son, la Bolsa de Tokio (*Nikkei 225, NKY),* la Bolsa de Shanghái (*Shanghai Stock Exchange*, SSE), la Bolsa de Hong Kong (*Hong Kong Stock Exchange*, HKEX) y la Bolsa de Alemania (*Deutscher Aktienindex, G*DAX). Para la

realización del proceso de análisis, de los índices asiáticos expuestos se ha preferido utilizar el índice NKY frente al HKEX o al SSE, atendiendo a su mayor capitalización y a su mayor influencia global, que aumenta la probabilidad de encontrar mayores correlaciones con los eventos de la bolsa Española. Por tanto, una parte de la información utilizada para la predicción, vendrá dada por el índice Nikkei 225, cuyo cierre se produce dos horas previas a la apertura de la bolsa española, y por lo cual, nos podrá aportar datos de los valores de apertura, máximo, mínimo y cierre de la sesión que finaliza. Por otro lado, también haremos uso del índice alemán, del que podremos contar con su información de apertura, pero no de máximo, mínimo o cierre, puesto que el final de dicha sesión coincide con el cierre de la sesión española.

Por tanto, mediante la información que estas variables nos pueda aportar elaboraremos diferentes modelos de predicción. Un primer modelo a analizar será uno que nos permita predecir el valor de apertura del IBEX35 mediante el uso de las variables del índice bursátil japonés (Ver *Figura 13)*. Posteriormente, realizaremos un segundo modelo dónde el valor a predecir será el valor de cierre (respecto al valor de apertura de dicha sesión) a partir de la información de apertura, tanto del índice alemán como el propio IBEX35, y nuevamente la información aportada por el índice japonés (Ver *Figura 14)*. Las predicciones que nos pudiera aportar este segundo modelo son de bastante interés económico, pues nos podría aportar información de valor a ser utilizada durante la operativa de dicha sesión. En cualquier caso, para ambos modelos, generaremos estrategias de contratación que nos permita evaluar económicamente, en función del beneficio y el riesgo, el rendimiento de nuestro proceso de minería de datos.

	NKY225				IBEX35
DATE	OPEN	CLOSE	HIGH	LOW	OPEN
…	…	…	…	…	¿?
…	…	…	…	…	¿?
…	…	…	…	…	¿?

Figura 13: Modelo del experimento 1.

	NKY225				DAX	IBEX35	
DATE	OPEN	CLOSE	HIGH	LOW	OPEN	OPEN	CLOSE
…	…	…	…	…	…	…	¿?
…	…	…	…	…	…	…	¿?
…	…	…	…	…	…	…	¿?

Figura 14: Modelo del experimento 2.

Para la generación y validación de estos modelos, se ha diseñado un sistema que funcionalmente cumplirá todas las etapas del proceso de minería de datos en concordancia con la metodología CRISP-DM.

El sistema, consta de diferentes módulos implementados en el lenguaje R, dónde cada uno cumple con una funcionalidad específica en el proceso. Los módulos implementados, sobre los cuales se profundizará en los siguientes capítulos, son los siguientes:

- **Módulo *dataCrawler*.**

  Es el módulo encargado de capturar de Internet todos los datos históricos de los valores bursátiles necesarios para realizar el proceso de análisis. Estos datos son conseguidos mediante llamadas HTTP a la API que *Yahoo Finance* pone a disposición del público. Una vez este módulo es ejecutado, dispondremos de una lista con los datos financieros históricos del periodo que se haya especificado.

- **Módulo *dataBuilder*.**

  Este módulo es el encargado de dar el formato necesario a los datos históricos que hasta el momento tenemos almacenados en listas, con el fin de maximizar su rendimiento para la fase de análisis. Se calculará las variaciones porcentuales diarias producidas entre los valores significativos de cada mercado, y se almacenarán en tablas, asignándole a cada columna el nombre identificativo correspondiente.

- **Módulo *dataPartitioner*.**

  Previa a la fase de análisis, será necesario dividir el conjunto de datos en dos subconjuntos: un conjunto de entrenamiento, que será utilizado para la creación del modelo en la fase de análisis, y un conjunto de prueba, dónde se testeará el rendimiento del modelo generado. La generación de estos dos subconjuntos es la funcionalidad implementada por este módulo. Además, con el fin de obtener valores estadísticos fiables, generaremos para cada uno de los dos subconjuntos, cinco versiones aleatorias diferentes, que serán analizadas y validadas posteriormente.

- **Módulo *dataAnalyzer*.**

  Este es el módulo de mayor importancia en todo el proceso, pues es el que implementa el proceso de minería de datos en sí mismo. De las diferentes técnicas disponibles, se ha optado por utilizar una red neuronal artificial no lineal para el proceso de predicción de los valores del IBEX35. A partir de los datos extraídos y formateados anteriormente, será este módulo el que se encargará de generar, configurar y entrenar a la red neuronal artificial que se usará posteriormente para realizar predicciones.

- **Módulo *modelValidator*.**

  Una vez se dispone de un modelo entrenado, haremos uso de este módulo para comprobar la exactitud del modelo, utilizando para ello los datos del subconjunto de prueba. A partir de la validación de las predicciones estimadas con estos datos, podremos obtener valores estadísticos (error cuadrático absoluto, error cuadrático medio, coeficiente de determinación, etc.) que nos indiquen el grado de acierto en las predicciones.

La principal ventaja que aporta una arquitectura basada en un diseño modular, es que permite la sustitución de los módulos del sistema con facilidad. Así por ejemplo, si en un futuro se quisiera continuar estudiando el rendimiento de las diferentes técnicas de minería de datos, se podría sustituir sin mayor problema el módulo *dataAnalyzer* actual, basado en redes neuronales, por otro que estuviera basado en árboles de decisión, u otra técnica. De igual forma, el diseño modular también permite de forma sencilla la extensión de las funcionalidades actuales del sistema. Así por ejemplo, se podrían generar nuevos módulos que implementaran otro tipo de técnicas y encadenarlos unos con otros con el fin de conseguir un análisis mucho más complejo de los datos.

La coordinación entre módulos se consigue mediante la utilización de una *string* de configuración que es enviada a cada uno de los módulos. El contenido de este parámetro de configuración sirve para informar a los módulos de cuáles son los índices bursátiles y qué variables significativas utilizaremos para el modelado. Esta *string,* hace uso de la sintaxis JSON para su definición, facilitando su integración con otros sistemas y dejando abierta la posibilidad de distribuir la herramienta generada como un servicio web (*Software as a service*). A modo ilustrativo, la *string* de configuración que se deberá utilizar para la generación del modelo que se ha planteado al comienzo de este epígrafe, será la siguiente:

```
'[{"symbol":"^N225",  "parameters":["OPEN" , "CLOSE", "HIGH", "LOW"]},
  {"symbol":"^GDAXI", "parameters":["OPEN"]},
  {"symbol":"^IBEX" , "parameters":["OPEN", "CLOSE"]}]'
```

Figura 15: Ejemplo de una *string* de configuración.

Como se puede comprobar, se trata de un *array* JSON en el que cada elemento representa a un índice bursátil diferente, designado por el atributo *symbol*. El símbolo que identifica a cada uno de los índices, se corresponde con el proporcionado por *Yahoo Finance* en su página online [32]. Después, el atributo *parameters* será el que defina qué valores significativos de los índices seleccionados utilizaremos para nuestro modelo: *OPEN* designa a los valores de apertura, *CLOSE* designa a los valores de cierre, *HIGH* para los valores máximos, *LOW* para los valores mínimos y *VOLUME* para el volumen de operaciones de cada sesión. Además, se incluye una nueva variable que hemos denominado SESVAR, que representa a la tasa de variación producida durante una sesión diaria, calculada respecto al valor de apertura de dicha sesión.

De cara a los modelos que se van a predecir, el último valor significativo del último índice especificado (en nuestro ejemplo, parámetro *CLOSE del símbolo ^IBEX)*, será la variable que se pretenderá predecir a partir del resto de variables explicativas.

Las bondades del diseño de este sistema es que permite de manera sencilla y con gran flexibilidad la creación y validación de diferentes modelos, mediante una simple modificación de la *string* de configuración, a partir de los cuales poder estimar resultados bursátiles y medir su rendimiento.

- Capítulo 6 –
Implementación del sistema.

En este capítulo se presentará la implementación de los diferentes módulos que conforman el sistema que se ha planteado en el capítulo anterior, y que utilizaremos para la ejecución del proceso de minería de datos. Para cada uno de los módulos, plantearemos los diferentes pasos llevados a cabo así como la justificación de las decisiones tomadas en su diseño e implementación.

6.1 Módulo *dataCrawler*.

Como ya se introdujo en el Capítulo 2, en la segunda fase de la metodología CRISP-DM se encuentran las tareas dedicadas a la extracción y recopilación de los datos. En este punto inicial del proceso de minería, lo que se busca es la automatización de las tareas de extracción de todos los datos históricos bursátiles que vayan a ser utilizados durante el análisis. El módulo encargado de la implementación de estas funcionalidades es el módulo *dataCrawler.*

Este módulo, inicialmente recibe como parámetros la *string* de configuración y la fecha inicial y final que delimitan el periodo del cual se quieren extraer los datos. Una vez la *string* de configuración es procesada, se extraen cada uno de los símbolos que representan a los índices bursátiles de los que se quieren obtener sus históricos, y se ejecutan las peticiones a la fuente de datos *Yahoo Finance*. Estas peticiones, se realizan mediante la ejecución de llamadas HTTP GET a una dirección URL que representa a una petición realizada a la plataforma *Yahoo Query Language*. Mediante la ejecución de estas llamadas obtendremos una respuesta JSON con los datos que hayamos solicitado. Un ejemplo de este tipo de sentencias sería la siguiente, dónde se solicita el histórico de datos del IBEX35 del año 2015.

```
SELECT * FROM yahoo.finance.historicalData WHERE symbol = "^IBEX"
    AND startDate = "2015-01-01" AND endDate = "2015-31-01";
```

Las limitaciones que *Yahoo* establece sobre la cantidad de datos que se pueden enviar en cada petición (1.5mb por llamada), no hace posible el poder extraer los datos históricos de periodos más largos a un año. Por tanto, se ha tenido que solucionar esto

diseñando la lógica del módulo para que en caso de solicitar periodos superiores a un año, se realicen tantas llamadas como años se estén solicitando, uniendo posteriormente los conjuntos de datos obtenidos en una única lista. Finalmente, el módulo también se encarga de la conversión de los datos del formato JSON a un objeto que pueda ser tratado por el lenguaje R, preparándolo para las siguientes fases del proceso.

Una vez este módulo se ha ejecutado, el resultado obtenido es una lista dónde se almacena por cada día e índice bursátil, un elemento con toda la información significativa de dicha sesión. La estructura de esta lista es como se muestra a continuación.

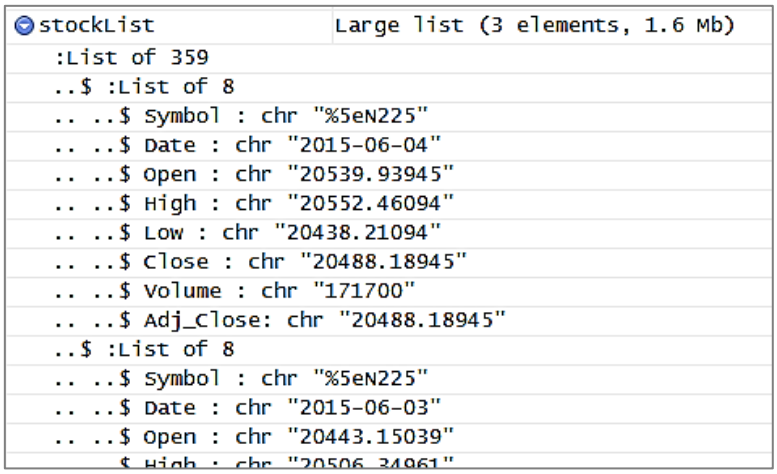

Figura 16: Estructura de la lista generada por el módulo dataCrawler.

6.2 Módulo *dataBuilder.*

La siguiente fase en nuestro proceso será la correspondiente a la selección, construcción y formateo de los datos ya extraídos. Siguiendo la línea de trabajo propuesta por la metodología CRISP-DM, esta se correspondería con la fase de *Preparación de los datos.* Es importante hacer notar que será en este punto dónde se tomen las decisiones que más afecten al rendimiento del análisis que se realizará posteriormente. Para ello, es importante dedicar tiempo al entendimiento de los datos y a la naturaleza que representan, atendiendo también a los requisitos y características de las herramientas de minería de datos que vayamos a utilizar.

Nuestro conjunto de datos (o *dataset)*, se conforma, para cada índice bursátil, por el conjunto de valores significativos registrados durante todas las sesiones diarias del periodo que estamos analizando. Estos valores son: el valor de apertura (*OPEN*), cierre (*CLOSE*) y cierre ajustado (*ADJ_CLOSE*), así como el valor máximo (*HIGH*) y mínimo (*LOW*) de la sesión. La información que proporcionan estos cinco valores, es con

respecto al precio, es decir, representan estados significativos que el precio de cotización ha tomado durante la sesión. Por otra parte, también se incluye el volumen (*VOLUME)* de la sesión, que en este caso no representa a un precio, sino a la cantidad o volumen de contratos negociados durante el periodo de operatividad del mercado. Además, para cada registro diario (cada fila en nuestra tabla), esta información se complementa incluyendo el símbolo del índice bursátil y la fecha a la que se refiere.

Puesto que la herramienta de minería de datos que hemos seleccionado para el análisis, se trata de una red neuronal artificial (*ANN,* por sus siglas en inglés), deberemos de dar el formato adecuado a los datos con el fin de maximizar su rendimiento para el análisis. Como ya se presentó en el epígrafe 4.4, una ANN realiza un proceso de entrenamiento en el que se le muestra cuáles serán los valores de salida esperados cuándo las variables de entrada toman un estado determinado. Mediante este entrenamiento, la red neuronal aprende automáticamente a predecir cuál será el valor de la variable de salida en función del estado de las variables de entrada.

Por tanto, una primera observación con respecto al tratamiento de los datos, es que no deberemos trabajar con los valores absolutos de los precios, sino que en cambio, deberemos utilizar las variaciones diarias con respecto a la sesión anterior. A modo ilustrativo, imaginemos dos situaciones en las que cuándo el precio de apertura del índice GDAX se ha incrementado en 10 unidades, el precio de cierre del IBEX35 se ha visto también incrementado en 5 unidades. Si estuviésemos trabajando con los datos en valores absolutos, la red neuronal realizaría un aprendizaje interpretando estos dos casos como situaciones diferentes, creando al final, un modelo mucho más complejo y sobre el que difícilmente se podrán hacer predicciones fiables. Por el contrario, utilizando valores derivados, permitiremos enseñar a nuestra red neuronal a realizar predicciones en base los incrementos o decrementos que se produzcan en las variables explicativas.

Una segunda observación que se ha realizado, es con respecto a las diferencias entre los valores de capitalización de los diferentes índices. Es trivial comprender que, un incremento de 100 unidades en el valor de capitalización, no tendrá igual relevancia en un mercado valorado en 400 unidades que en un mercado valorado en 1.000.000 unidades. En el primer caso, dicha variación supone un incremento porcentual del 25% del valor de mercado, mientras que en el segundo, dicha variación representa un incremento poco significativo del 0.01%. Por tanto, es de nuestro interés que, de cara al aprendizaje que pueda realizar nuestra red neuronal, sepa interpretar cuándo una variación del precio es significativa con respecto al valor del mercado, y cuando no, para así conseguir un análisis más cercano a la realidad económica de los mercados que estamos estudiando.

Para solucionar estas dos ineficiencias, deberemos de computar a partir de los valores absolutos iniciales, la siguiente tasa de variación (*formula 1)*. Mediante el cálculo de

esta fórmula, para cada uno de los valores significativos (OPEN, CLOSE, HIGH, LOW), tendremos resuelta las dos deficiencias que hemos observado. Por tanto, ya disponemos de nuestros datos con el formato adecuado para poder realizar un análisis óptimo mediante el uso de redes neuronales.

$$v_t = \frac{p_t - p_{t-1}}{p_{t-1}} * 100$$

v_t $= Tasa\ de\ variación\ del\ precio\ entre\ sesiones\ consecutivas.$
p_t $= Valor\ del\ precio\ de\ cotización\ de\ la\ sesión\ en\ t$
$p_{t-1} = Valor\ del\ precio\ de\ cotización\ de\ la\ sesión\ anterior\ a\ t.$

Además, para nuestro conjunto de datos, ahora conformado por las variaciones diarias de los datos históricos bursátiles, será necesario generar un nuevo tipo de variable significativa que también podrá ser utilizada en la fase de análisis. Esta nueva variable, representará las variaciones producidas por el precio en su valor de cierre, pero no con respecto al valor de cierre de la sesión anterior (tal y como representa el valor *CLOSE*), sino con respecto al valor de apertura de dicha sesión. Por tanto, la información contenida por esta nueva variable, que denominaremos con la etiqueta (SESVAR), representará las variaciones producidas en el precio de cotización durante una sesión diaria. Contando con la participación de esta nueva variable en la fase de análisis, si consiguiéramos generar un modelo capaz de predecir con cierta facilidad el valor futuro de esta, contaremos con una información de gran relevancia que nos podría aportar un gran valor económico en nuestra operativa.

Para su cómputo, calcularemos nuevamente tasas de variación, pero utilizando en esta ocasión las siguientes variables:

$$SESVAR_t = \frac{pC_t - pA_t}{pA_t} * 100$$

$SESVAR_t$ $= Tasa\ de\ variación\ del\ precio\ en\ la\ sesión\ en\ t.$
$pC_t = Valor\ del\ precio\ de\ cotización\ de\ la\ sesión\ en\ t$
$pA_t = Valor\ del\ precio\ de\ cotización\ de\ la\ sesión\ anterior\ a\ t.$

El módulo *dataBuilder* será por tanto el encargado de realizar todas las funciones de procesamiento y formateo adecuado de los datos. A partir de la información proporcionada por la *string* de configuración, se obtiene la información de cuáles son las variables significativas que queremos de cada uno de los índices bursátiles.

Posteriormente, los datos que habíamos obtenido tras la ejecución del módulo *dataCrawler* y que hemos almacenado en listas, son transformados a un formato matricial, que nos permitirá realizar un trabajo mucho más flexible. Cada fila de esta matriz representará a una sesión diaria de bolsa, dónde cada columna representará la información que se haya especificado en la *string* de configuración. Para cada una de estas columnas, se le asignará una etiqueta compuesta por el nombre del índice, un carácter separador y el valor significativo al que representa (ej. A la columna que contiene los valores de apertura del IBEX35, se le asignará el título "IBEX_OPEN"). Finalmente, para la obtención de un mayor rendimiento en el análisis, se calculará la tasa de variación diaria de los datos, tal y como se ha presentado en el epígrafe anterior. Por tanto, tras la ejecución de este segundo módulo, el resultado que obtendremos será el de una tabla con las variaciones porcentuales diarias de cada una de las variables significativas que hemos seleccionado. A partir de este conjunto de datos, podremos comenzar a realizar nuestras primeras operaciones de análisis.

DATE	N225_OPEN	N225_HIGH	N225_LOW	N225_CLOSE	GDAXI_OPEN	IBEX_OPEN	IBEX_SESVAR
20140107	-1.932987205767	-1.2915947925717	-0.4527013522782	-0.539265084218	0.284619560994	1.47516075542	2.6855000756620
20140108	0.683717888791	1.1511999849902	0.6905383928418	1.759693619554	0.704406923382	2.80353483985	0.6201838551488
20140109	0.371308230963	-0.7335294012111	-0.4294502683572	-1.381716242207	-0.216022900913	0.64079723558	-0.2096395135505
20140110	-1.360564433351	-0.5112438459510	-0.5305303387871	0.199445166683	-0.193527729463	0.36954487046	-0.0291443237930
20140113	0.803978269807	-0.0629825391690	0.9892728093704	0.000000000000	0.253854071282	0.41482277942	0.2825014323392
20140114	-1.601674339383	-1.5723364732041	-3.3359922167539	-3.037314714726	-1.120876590604	-0.60176462732	1.0502174896716
20140115	-0.051924245068	0.9239564957209	1.6051075480335	2.432758488661	2.120121385033	1.03853573608	1.3891048638713
20140116	1.252982286428	0.8450521005023	0.4623495511461	-0.386689565416	1.581132759230	1.45846555992	-0.7282447501939
20140117	-0.944708168154	-0.9976130677239	-0.5742435041967	-0.080066530319	-0.246450919207	-0.74533334485	0.1147918897247
20140120	0.182726005299	-0.3519858800373	-0.3042184766278	-0.601594243748	-0.051449855937	-0.13105593947	0.1350537260536
20140121	-0.084265341676	1.0607362100676	0.8377352330968	0.975918169410	0.399253317356	0.28543874521	-1.0735583878722
20140122	0.242635005745	-0.1535513201874	-0.4764475856116	0.158758380477	0.001331862799	-0.67145319213	-1.1519765854128
20140123	0.962727938155	0.5539577187358	0.3790880012821	-0.794881510455	-0.631664160644	-1.39814790134	-0.1257976272113
20140124	-2.685802852732	-2.9838987031558	-2.5603844620485	-1.945635533206	-0.452601318807	-0.38423084915	-3.3853152554518
20140127	-2.469501926803	-2.3519626505959	-2.2696001413783	-2.442580900133	-2.669117865412	-3.30602160174	-1.2007655057204
20140128	-0.349936860392	-0.1392285683388	0.1228790299857	-0.161623018218	0.174247043518	-0.38776759137	0.4106173905018
20140129	0.835848181366	1.0576195201207	1.3545624357272	2.572320501803	1.082547533681	1.49815942227	0.0002455164675

Figura 17: Estructura de la tabla generada por el módulo *dataBuilder* para el experimento2.

6.3 Módulo *dataPartiotioner.*

Cuándo ya contamos con el conjunto de datos preparados, antes de iniciar la fase de análisis, será necesario configurar dos subconjuntos de datos diferentes: un primer subconjunto denominado *conjunto de entrenamiento,* que será el que contenga los datos utilizados para el entrenamiento de nuestra red neuronal, y un segundo conjunto, el *conjunto de prueba,* que será el utilizado para probar la validez de nuestro modelo generado. El motivo que hay detrás de la utilización de estos dos subconjuntos es el de detectar el problema de *overfitting* o sobreajuste (ver epígrafe 4.5) en el que nuestra red neuronal pudiera incurrir.

La lógica encargada de generar los diferentes subconjuntos de datos a partir de los cuales seremos capaces de detectar y eliminar la aparición de sobreajuste en nuestro

modelo, será la contenida en el módulo *dataPartitioner*. Su implementación, está encargada no solo de generar estos dos subconjuntos, sino que para cada uno de ellos, generará varias muestras diferentes que contengan una selección aleatoria de los datos. El motivo de esto será el de prevenir la posible generación de subconjuntos que puedan representar estados sesgados de los datos, y que puedan representar una realidad ficticia de cara a su análisis (i.e. situación de mejor/peor caso).

Por defecto, el módulo *dataPartitioner* generará 5 muestras aleatorias diferentes, cada una de ellas con sus respectivos subconjuntos de prueba y entrenamiento. La proporción utilizada para distribuir cuantitativamente la cantidad de datos que habremos obtenidos en las etapas anteriores, será del 70% para el subconjunto de entrenamiento y del 30% para subconjunto de prueba. Esta distribución de los datos es la mayoritariamente aceptada para los procesos de minería de datos, pues destina la mayor parte al entrenamiento del sistema inteligente, permitiendo un mejor rendimiento de este, pero sin eliminar la posibilidad de poder calcular unos indicadores fiables de su rendimiento utilizando los datos restantes.

Una vez se ha ejecutado este módulo, si contábamos inicialmente con un único conjunto de datos, el resultado a devolver será una lista formada por dos sub-listas, que contendrán a las 5 instancias diferentes de cada uno de los subconjuntos, que serán posteriormente analizadas por una red neuronal específica para cada una de ellas, en la fase de análisis (ver *Figura 18).*

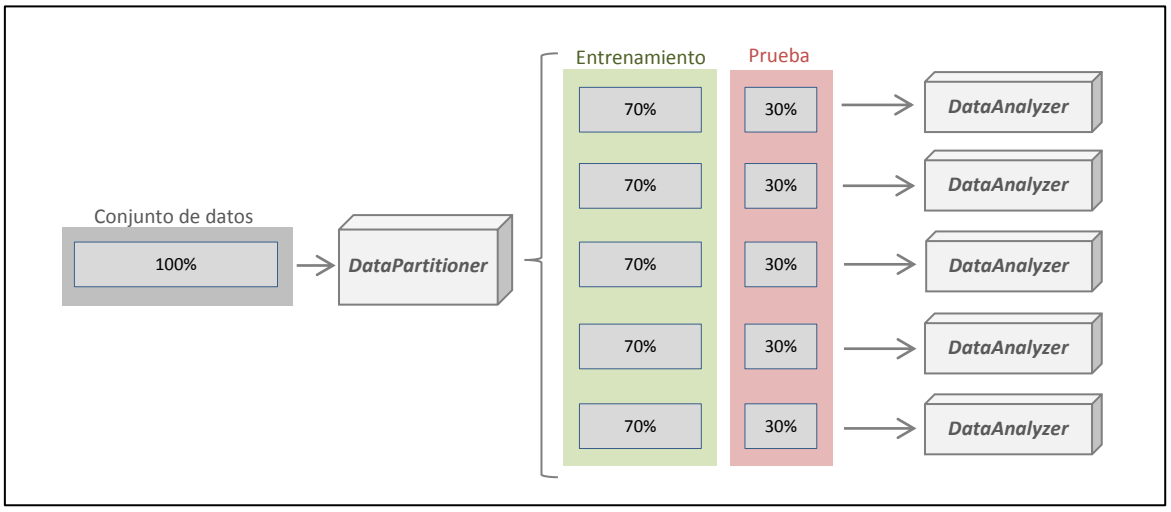

Figura 18: Diagrama de funcionamiento del módulo DataPartitioner

6.4 Módulo *dataAnalyzer.*

Una vez alcanzado este punto del proceso, ya contamos con las estructuras de datos necesarias para poder iniciar un análisis satisfactorio. Esta fase, la cuarta en la

metodología CRISP-DM, será la encargada de aplicar las técnicas de minería de datos seleccionadas, con el fin de extraer toda la información relevante del conjunto de datos. La correcta configuración de los parámetros de los cuales depende el funcionamiento de la técnica a utilizar, será un paso clave para obtener un mayor rendimiento en esta fase.

De las diferentes técnicas de minería de datos existentes, nos hemos centrado en la utilización de técnicas de predicción, y más específicamente en la utilización de redes neuronales artificiales. Mediante su uso, tras la fase de entrenamiento inicial, obtendremos un modelo sobre el cuál podremos realizar predicciones sobre las variaciones de los valores del IBEX35, si bien el sistema implementado permite fácilmente la generación de modelos diferentes dónde poder predecir y utilizar valores de otros mercados.

Para el funcionamiento de este módulo será necesario que se le proporcione como parámetro la lista que contendrá las diferentes muestras aleatorias generadas por el módulo *dataPartitioner*. A partir de la estructura de las tablas, la lógica del módulo se encargará de generar la *string* requerida por la función *neuralnet* del paquete de R que estamos utilizando para crear *ANNs*, y que servirá para definir la estructura de los parámetros de entrada y salida de la red neuronal. La sintaxis de esta *string* es bastante simple de generar, estando su estructura definida de la siguiente forma:

```
OUTPUT~INPUT_1+INPUT_2+...+INPUT_N
```

El resto de los parámetros aceptados por la función *neuralnet* servirán para configurar las características del análisis. A partir de ellos, podremos definir su tipología, definiendo la cantidad de capas ocultas de la red, así como el número de neuronas que conformará cada una de las capas. También, se podrá elegir el tipo de algoritmo que se ejecutará durante el análisis, pudiendo elegir entre diferentes variantes del algoritmo de retropropagación.

El resto de parámetros nos permitirán definir aspectos tales como la cantidad de repeticiones que queremos que se realicen para cada análisis, la cantidad de pasos máximos que podrá ejecutar el algoritmo encargado de minimizar el error del predictor, el valor inicial de los pesos de los parámetros de entrada, o definir cuál será la función utilizada para el cálculo del error. Para estos últimos parámetros, haremos uso de la configuración que viene por defecto dentro del paquete.

Finalmente, puesto que el objetivo de esta red neuronal es la de predecir valores continuos que representen las variaciones diarias de nuestra variable de salida (apertura y variación diaria del IBEX35 en nuestros experimentos), será necesario configurar mediante parámetros esta red como lineal, significando esto que el valor

que pudiera tomar la variable de salida no será mapeado a valores binarios, y que por el contrario, mantendrá la misma naturaleza y rango que los valores de entrada.

Por tanto, tal y como se puede observar, la gran flexibilidad que nos ofrece el paquete *neuralnet* nos permite generar un alto número de configuraciones diferentes, que podrán tener un mayor o menor rendimiento para nuestro análisis dada las característica de nuestros datos y de los modelos buscados. Es por ello que, tal y como se presentará en el epígrafe XX, una de las tareas de análisis que se ha realizado en este proyecto ha sido la de buscar la configuración óptima de la red a partir de la automatización de los análisis para diferentes configuraciones.

Una vez se ha completado el entrenamiento de la red, el objeto devuelto por este módulo será la red neuronal artificial ya entrenada, la cual podrá ser utilizada por otras funciones del paquete *neuralnet* para la realización de predicciones utilizando nuevos datos. Esta funcionalidad de predicción será la que utilizaremos en los módulos posteriores para realizar la validación de nuestro predictor.

6.5 Módulo *modelValidator.*

La quinta fase de la metodología *CRISP-DM* se corresponde con las tareas de evaluación del modelo generado, encargadas de demostrar la validez del predictor que se ha entrenado en la fase previa. Para ello, el módulo *modelValidator* se encargará de computar diferentes herramientas estadísticas que servirán para cuantificar las capacidades predictivas de nuestra red neuronal, siendo estos valores calculados sobre los conjuntos de prueba generados por el módulo *dataPartitioner.*

Tras la ejecución de este módulo obtendremos por consola el cálculo de los diferentes estadísticos para cada uno de los 5 subconjuntos de datos de prueba, y un cálculo medio de todos ellos, tal y como se muestra en la Figura 19.

```
"------------------------------"
"ANN 1  --> RECM = 1.18953 , R2 = 0.00033 , %Correct = 0.55405"
"ANN 2  --> RECM = 1.16703 , R2 = 0.00516 , %Correct = 0.51351"
"ANN 3  --> [DID NOT CONVERGED] "
"ANN 4  --> [DID NOT CONVERGED] "
"ANN 5  --> RECM = 1.05816 , R2 = 0.00477 , %Correct = 0.48649"
"------------------------------"
"Mean perf: RECM = 1.13824 , R2 = 0.00342 , %Correct = 0.51802"
"------------------------------"
"> DONE!"
```

Figura 19: Información de salida del módulo *dataValidator*.

Los estadísticos computados por este módulo son los que se presentan a continuación:

6.5.1 Raíz del Error Cuadrático Medio, RECM.

Mediante el cálculo del RECM obtendremos un estimador que nos medirá las diferencias entre los valores predichos por nuestro estimador y los valores reales esperados, o visto de otra manera, la desviación típica de esta diferencia. Para su cálculo, haremos uso de la siguiente ecuación [33]:

$$\mathbf{RECM} = \sqrt{\frac{\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}{n}}$$

$\hat{y}_i = Valor\ del\ predicho\ para\ la\ muestra\ i.$
$y_i = Valor\ real\ para\ la\ muestra\ i.$
$n = Número\ de\ muestras.$

Para el cálculo del RECM se ha de calcular la suma de las diferencias entre el valor estimado y el valor real, elevado al cuadrado. Esto último se hace con el fin de evitar las posibles compensaciones que se pudieran producir entre errores de signo contrario. El valor de la suma será después dividido entre el número de muestras para así obtener el valor medio de la suma cuadrática de los errores, o tal y como se conoce a dicho estadístico, el Error Cuadrático Medio (ECM). Sobre este valor, calcularemos su raíz cuadrada, con el fin de obtener una magnitud que sea equivalente a la de los valores de la muestra, y nos permita hacer comparaciones entre semejantes.

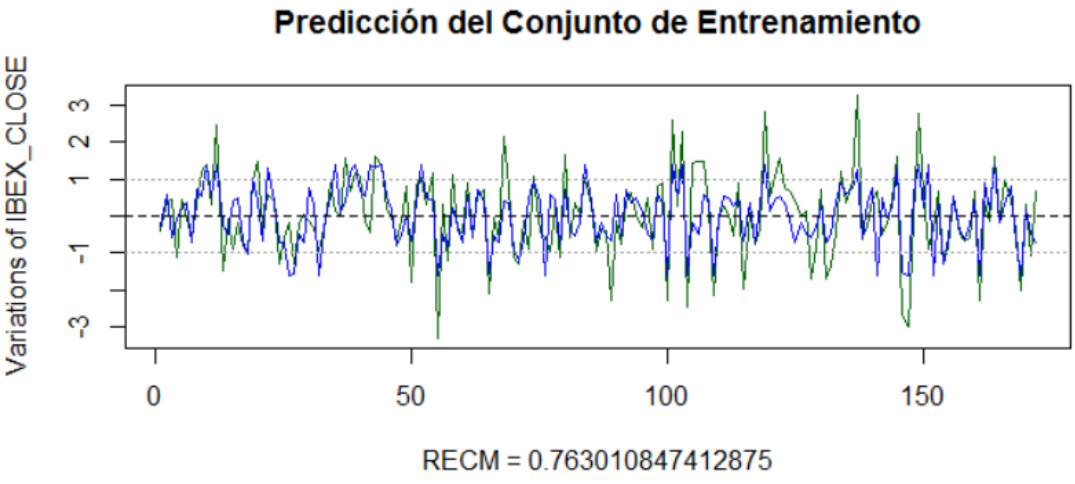

Figura 2: Gráfica de predicción del cierre del IBEX35 y cálculo del RECM.

Este estadístico nos servirá para valorar que tan buen predictor es nuestra red neuronal artificial, significando un valor bajo que el error de predicción de esta es también bajo. Gracias a su uso, simplificamos el proceso de cuantificar el error de

predicción, que en otras circunstancias habría que realizar analizando visualmente la diferencia entre las gráficas de los valores reales y los valores estimados (Ver figura 20), de manera menos rigurosa.

6.5.2 Bondad del ajuste, R^2.

Si representáramos en una gráfica, para cada una de las muestras, el valor real de dicha muestra en el eje X, y el valor estimado por nuestra ANN en el eje Y, obtendremos una gráfica de correlación de la predicción que nos servirá para visualmente reconocer la calidad de nuestro predictor, tal y como se muestra en la figura 21.

En un escenario ideal, dónde el rendimiento de nuestro predictor sea perfecto, los datos reales y estimados serán idénticos, y la representación gráfica de la correlación mostrará que estos se ajustan linealmente en un eje de 45º. En la situación contraria, dónde nuestro predictor sea altamente ineficiente, la representación que obtendremos en la gráfica de correlación será la de una nube aleatoria circular de puntos alrededor del eje central. Por tanto, para estimar la calidad de nuestro predictor, el análisis que deberemos hacer sobre este tipo de gráficas será el de descubrir la existencia de linealidad sobre la nube de puntos.

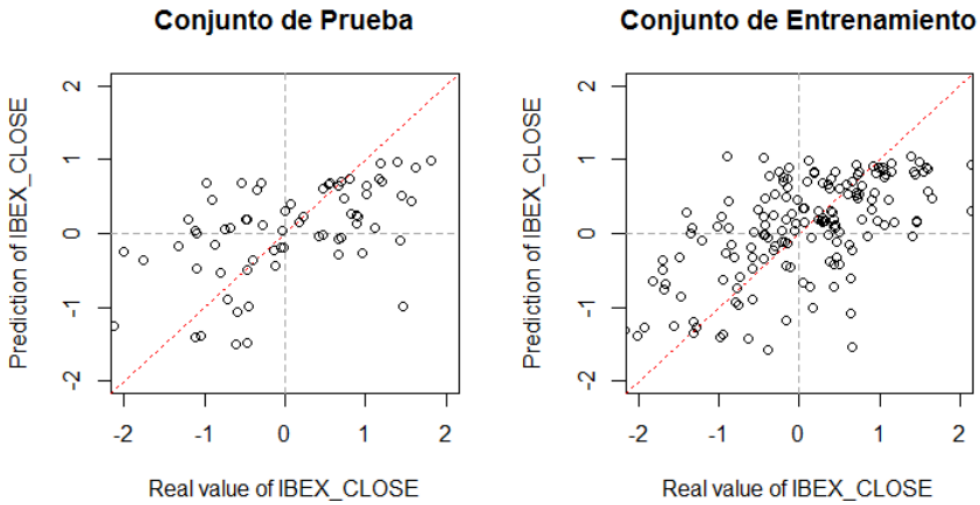

Figura 21: Gráfica de correlación del predictor para ambos conjuntos de datos.

Una forma cuantitativa de detectar la existencia de esta linealidad es mediante el cálculo del estadístico R^2, también conocido como coeficiente de determinación o bondad del ajuste. En términos generales, este estadístico determina la proporción de la variación de los datos que puede explicarse por el modelo de regresión lineal estimado para el conjunto de puntos. En el caso en el que los datos no disten mucho de la línea de regresión estimada, aceptaremos la existencia de una relación lineal

entre los datos, y la bondad del ajuste tomará valores cercanos a 1. Por el contrario, si nos encontráramos ante la situación de aleatoria en la distribución de los datos, el valor del estadístico será cercano a 0.

Para su cálculo, haremos uso de la siguiente expresión [34]:

$$\boldsymbol{R^2 = \frac{\sigma_{XY}^2}{\sigma_X^2 \sigma_Y^2}}$$

$\boldsymbol{\sigma_{XY}^2} = Covarianza\ de\ X\ e\ Y.$
$\boldsymbol{\sigma_X} = Varianza\ de\ X.$
$\boldsymbol{\sigma_Y} = Varianza\ de\ Y.$

Por tanto, con el cálculo de este estadístico podremos comprobar la existencia de correlaciones lineales entre los resultados esperados y lo resultados estimados, sin necesidad de tener que representar gráficamente la correlación del predictor.

6.5.3 Tasa de acierto, %C.

Puesto que los resultados que obtendremos de nuestra ANN están distribuidos en un rango continúo de valores, de cara a calcular la eficacia de la predicción comprobaremos si el valor predicho coincide con el valor esperado, como se ha visto en el cálculo del RECM. Sin embargo, para la creación de reglas de contratación que nos puedan ser útiles en nuestra operativa bursátil, la predicción que pueda realizar el predictor sobre el signo de dicho valor de salida, puede ser suficiente para poder decidir si iniciar operación de compra o venta.

Es por ello, que de forma complementaria a los estadísticos anteriores, también se ha calculado un estadístico que nos indica la cantidad de aciertos en el signo que se ha producido con respecto al número total de muestras. Este indicador es la Tasa de acierto (%C), medida en términos porcentuales, dónde un valor de 0.8 nos indicará un grado de acierto del 80% de los signos en las predicciones realizadas. Esto nos permitirá estimar el rendimiento de nuestra red neuronal actuando exclusivamente como un clasificador entre los incrementos y decrementos de la variable endógena.

- Capítulo 7 –
Análisis y resultados.

Tras la implementación del sistema que servirá como herramienta para la ejecución del proceso de minería de datos, en este capítulo haremos uso de dicho sistema para iniciar un conjunto de análisis que nos permita obtener resultados sobre los dos experimentos planteados y nos acerque a obtener conclusiones relevantes. Para ello, utilizaremos técnicas de análisis que nos permita validar no solo nuestros resultados, sino las capacidades del sistema implementado.

7.1 Análisis de los resultados y estudio de la topología.

Como se ha visto previamente, las redes neuronales artificiales que estamos utilizando como herramientas permiten un alto grado de configuración, dónde entre otros parámetros podemos elegir cuál será su topología (en número de capas ocultas y neuronas), sus funciones de activación y propagación o el algoritmo que se ejecuta para el entrenamiento de la red. Esta flexibilidad en cuanto a la configuración hace difícil la labor de encontrar la configuración óptima para cada uno de los diferentes modelos. El rendimiento de la red y su capacidad de predicción se verá afectado de la elección tomada. En la teoría no existe una metodología precisa que indique las pautas a seguir para elegir dicha configuración, siendo necesario analizar diferentes situaciones y comprobar la calidad de la red para cada una de ellas.

Para hacer esto, haremos uso del sistema implementado, realizando un análisis para cada una de las posibles topologías y tipo de algoritmo, entre los tres que ofrece el paquete *neuralnet*: Algoritmo de Retropropagación (*Backprop),* Retro-propagación (*Backprop),* Retro-propagación con resiliencia y retroceso en los pesos (*Rprop+),* Retro-propagación con resiliencia sin retroceso en los pesos *(Rprop-).* De las topologías analizadas, se han descartado aquellas en las que el número de neuronas de la primera capa oculta era mayor que el de la segunda, por la pérdida de información que esto supone al primero comprimir y después separar los datos. Sobre esta restricción, hemos calculado todas las posibles topologías para un número máximo de 8 neuronas por capa, tal y como se observa en la Tabla1 y Tabla2. Para cada una de las configuraciones, hemos obtenido los diferentes estadísticos computados por el módulo *dataValidator*: la Raíz del Error Cuadrático Medio (RECM), la Bondad del Ajuste (R^2) para la regresión entre los datos reales y estimados, y el coeficiente de acierto del signo, obviando el valor de la predicción (%C). A continuación se muestran los resultados obtenidos.

Exp1. **NK225** {OPEN, HIGH, LOW, CLOSE} → **IBEX** {OPEN}										
TOPOLOGIA		**Backprop**			**Rprop+**			**Rprop-**		
Capa 1	Capa 2	**RECM**	**R^2**	**%C**	**RECM**	**R^2**	**%C**	**RECM**	**R^2**	**%C**
1		0.90569	0.38725	0.72703	0.90572	0.38722	0.72703	0.90576	0.38717	0.72703
2		1.04055	0.213	0.63964	0.96373	0.33438	0.66486	0.98267	0.309	0.68919
3		NC	NC	NC	0.94323	0.35236	0.68108	0.97899	0.31648	0.71622
4		NC	NC	NC	0.99636	0.32275	0.67297	0.98654	0.31061	0.68378
5		NC	NC	NC	1.04253	0.28294	0.70001	1.06278	0.28472	0.69459
6		NC	NC	NC	1.01262	0.28776	0.66486	1.00951	0.29412	0.71081
7		NC	NC	NC	1.10867	0.21959	0.64189	1.16287	0.22026	0.67027
8		NC	NC	NC	1.15649	0.19233	0.67027	1.09823	0.27105	0.60811
1	1	0.86563	0.42261	0.71622	0.90998	0.38232	0.72432	0.90626	0.38602	0.72973
2	1	0.98179	0.18373	0.64865	0.9449	0.35186	0.6973	0.95529	0.33526	0.72297
2	2	NC	NC	NC	0.97949	0.27457	0.68919	0.96099	0.37337	0.7027
3	1	NC	NC	NC	1.00216	0.30973	0.70811	0.99212	0.29348	0.69932
3	2	NC	NC	NC	0.96295	0.34032	0.69459	0.99154	0.30624	0.71892
3	3	NC	NC	NC	1.08407	0.20026	0.69595	1.03872	0.2744	0.67568
4	1	NC	NC	NC	0.95658	0.34624	0.71892	1.00384	0.26549	0.64414
4	2	NC	NC	NC	1.04891	0.24907	0.67117	1.1783	0.20091	0.68468
4	3	NC	NC	NC	1.15432	0.20034	0.70721	0.9934	0.26136	0.71622
4	4	NC	NC	NC	1.0685	0.25373	0.68018	1.0389	0.31378	0.77027
5	1	NC	NC	NC	1.04574	0.27967	0.64865	1.02808	0.24225	0.62613
5	2	NC	NC	NC	1.31458	0.12561	0.64865	1.22181	0.25532	0.73423
5	3	NC	NC	NC	1.78532	0.20109	0.75676	1.11684	0.29505	0.74324
5	4	NC	NC	NC	NC	NC	NC	1.0939	0.26913	0.75
5	5	NC	NC	NC	1.17843	0.17933	0.62162	NC	NC	NC
6	1	NC	NC	NC	1.14939	0.19727	0.65676	1.10261	0.25675	0.68468
6	2	NC	NC	NC	1.04347	0.31449	0.6982	1.23055	0.22299	0.64527
6	3	NC	NC	NC	NC	NC	NC	1.09927	0.25311	0.70946
6	4	NC	NC	NC	NC	NC	NC	1.19535	0.14511	0.63514
6	5	NC	NC	NC	1.16886	0.22632	0.62162	1.34714	0.0763	0.63514
6	6	NC	NC	NC	NC	NC	NC	NC	NC	NC
7	1	NC	NC	NC	1.16653	0.21733	0.65878	1.1325	0.26964	0.63063
7	2	NC	NC	NC	1.25173	0.11801	0.62162	1.24183	0.17827	0.63964
7	3	NC	NC	NC	1.15552	0.16867	0.66667	1.2655	0.16094	0.67568
7	4	NC	NC	NC	NC	NC	NC	NC	NC	NC
7	5	NC	NC	NC	1.35832	0.20278	0.64865	1.28747	0.2157	0.64189
7	6	NC	NC	NC	NC	NC	NC	1.3432	0.20234	0.60811
7	7	NC	NC	NC	NC	NC	NC	NC	NC	NC
8	1	NC	NC	NC	1.18429	0.19908	0.63514	1.20607	0.16615	0.67568
8	2	NC	NC	NC	1.68691	0.15678	0.72973	1.26384	0.14934	0.64189
8	3	NC	NC	NC	1.31126	0.13748	0.66216	2.10338	0.05945	0.65541
8	4	NC	NC	NC	1.47813	0.06823	0.66216	1.39823	0.06471	0.67568
8	5	NC	NC	NC	NC	NC	NC	1.37933	0.13962	0.62162
8	6	NC	NC	NC	NC	NC	NC	NC	NC	NC
8	7	NC	NC	NC	NC	NC	NC	NC	NC	NC
8	8	NC	NC	NC	NC	NC	NC	NC	NC	NC

Tabla 1: Resultados del experimento 1.

Exp2. **NK225** {OPEN, HIGH, LOW, CLOSE} **+ GDAX** {OPEN} **+ IBEX** {OPEN} → **IBEX** {SESVAR}										
TOPOLOGIA		**Backprop**			**Rprop+**			**Rprop-**		
Capa 1	Capa 2	**RECM**	**R^2**	**%C**	**RECM**	**R^2**	**%C**	**RECM**	**R^2**	**%C**
1		NC	NC	NC	1.13407	0.0083	0.47671	1.13378	0.00493	0.47123
2		NC	NC	NC	1.20265	0.00698	0.47123	1.12854	0.02458	0.47671
3		NC	NC	NC	1.25765	0.01874	0.53151	1.51392	0.00238	0.50411
4		NC	NC	NC	1.25698	0.01324	0.49315	1.41411	0.02428	0.45205
5		NC	NC	NC	1.42657	0.01196	0.47397	1.3523	0.00893	0.50137
6		NC	NC	NC	1.51319	0.01561	0.48219	1.39608	0.00575	0.50411
7		NC	NC	NC	1.45162	0.01138	0.51781	1.52169	0.00613	0.49315
8		NC	NC	NC	1.45732	0.00896	0.50959	1.51773	0.01318	0.48493
1	1	NC	NC	NC	1.1397	0.00371	0.48493	1.11789	0.00817	0.53425
2	1	NC	NC	NC	1.24899	0.022	0.46918	1.1822	0.02309	0.4863
2	2	NC	NC	NC	1.22548	0.00093	0.41096	1.1795	0.00418	0.4726
3	1	NC	NC	NC	1.28929	0.00031	0.52329	1.25511	0.00731	0.47945
3	2	NC	NC	NC	NC	NC	NC	1.16731	0.00042	0.56164
3	3	NC	NC	NC	NC	NC	NC	1.41548	0.02093	0.46575
4	1	NC	NC	NC	1.40352	0.00776	0.47945	1.42572	0.02039	0.49041
4	2	NC	NC	NC	1.38061	0.01075	0.54338	4.66911	0.01943	0.55251
4	3	NC	NC	NC	1.55977	0.00158	0.4726	1.45989	0.01382	0.45548
4	4	NC	NC	NC	1.70175	0.0003	0.46575	NC	NC	NC
5	1	NC	NC	NC	1.41547	0.00391	0.5137	1.57134	0.00226	0.49772
5	2	NC	NC	NC	1.64657	0.00857	0.43836	1.52469	0.02674	0.52055
5	3	NC	NC	NC	NC	NC	NC	1.87496	0.00052	0.50685
5	4	NC	NC	NC	1.60771	0.00048	0.54795	1.62497	0.00579	0.42466
5	5	NC	NC	NC	NC	NC	NC	NC	NC	NC
6	1	NC	NC	NC	1.99995	0.02547	0.50411	1.50461	0.00466	0.46918
6	2	NC	NC	NC	1.5777	0.01995	0.49658	2.05877	0.0073	0.5411
6	3	NC	NC	NC	NC	NC	NC	3.24312	0.00092	0.45205
6	4	NC	NC	NC	NC	NC	NC	NC	NC	NC
6	5	NC	NC	NC	NC	NC	NC	NC	NC	NC
6	6	NC	NC	NC	NC	NC	NC	1.71842	0.02046	0.4863
7	1	NC	NC	NC	1.89626	0.01253	0.51142	1.67672	0.00967	0.4863
7	2	NC	NC	NC	NC	NC	NC	1.81366	0.01275	0.42466
7	3	NC	NC	NC	2.30554	0.00264	0.47945	1.69763	0.03498	0.5
7	4	NC	NC	NC	NC	NC	NC	NC	NC	NC
7	5	NC	NC	NC	16.07126	0.05429	0.52055	1.95984	0.01494	0.53881
7	6	NC	NC	NC	2.04799	0.05358	0.50685	NC	NC	NC
7	7	NC	NC	NC	NC	NC	NC	3.20549	0.04501	0.53425
8	1	NC	NC	NC	10.29794	0.03076	0.4726	1.99227	0.01646	0.53425
8	2	NC	NC	NC	1.66804	0.00193	0.43836	1.64242	0.01802	0.53425
8	3	NC	NC	NC	1.81474	0.00515	0.58904	2.24866	0.00273	0.49315
8	4	NC	NC	NC	NC	NC	NC	1.82452	0.01342	0.53425
8	5	NC	NC	NC	NC	NC	NC	3.82575	0.06967	0.56164
8	6	NC	NC	NC	2.50706	0.02179	0.52055	2.15065	0.02797	0.45205
8	7	NC	NC	NC	1.85134	0.00116	0.60274	NC	NC	NC
8	8	NC	NC	NC	5.13617	0.00433	0.46575	NC	NC	NC

Tabla 2: Resultados del experimento 2.

Para los análisis realizados se han utilizado los datos bursátiles de las sesiones diarias comprendidas entre el 01 de Enero de 2014 y el 01 de Enero de 2015, representando esto un total de 243 muestras (al descartarse fines de semana y días festivos). De dicha muestra, como ya se mencionó, el 70% de los datos han sido utilizados para la fase de entrenamiento, y el 30% restante se ha utilizado para el cálculo de los estadísticos que utilizaremos para medir la calidad de los modelos. De cada análisis, con el fin de obtener estadísticos más rigurosos, se está computando 5 análisis que han hecho uso de redes neuronales diferentes, pero con la misma inicialización de los pesos, para así poder hacer comparaciones más fiables.

Tal y como se observa en la Tabla 1, la primera evidencia con la que nos encontramos es con la gran inestabilidad mostrada por el algoritmo de Retropropagación (*backprop)* que solo ha conseguido converger en aquellas topologías de mayor simplicidad, nunca superior a dos neuronas. La no convergencia de una red neuronal implica que el algoritmo encargado de reajustar los pesos de la red con el fin de minimizar la señal de error es incapaz de encontrar dicho mínimo. Esta capacidad de convergencia puede controlarse ajustando el valor de la tasa de aprendizaje del algoritmo, que representa la intensidad de cambio que se produce en los pesos en cada iteración. Para nuestro análisis, se ha utilizado una tasa de aprendizaje de 0.005, la cual se había comprobado previamente ser la que mejor capacidad de convergencia ofrecía.

El rendimiento demostrado por los otros dos algoritmos (*rprop+ y rprop-)* es bastante parecido, obteniendo unos RECM bastante similares en todas las topologías probadas y no convergiendo solo en aquellas de mayor complejidad. Por tanto, ante la indiferencia de seleccionar uno u otro, elegiremos *rprop+* al tratarse de la opción ofertada por el paquete *neuralNet* por defecto.

La tendencia mostrada por los datos respecto a la topología demuestra que según se va aumentando la complejidad de la estructura de la red, tanto en número de capas, como en número de neuronas por capa, el RECM incrementa de igual manera, e inversamente el estadístico R^2 y la tasa de acierto van en decremento. Esto es, la capacidad de predicción de nuestro predictor disminuye según la red se va volviendo más compleja.

De hecho, para el experimento 1, resulta que la topología que, siguiendo la tendencia, muestra mejores RECM, es la formada por capas de una única neurona (topología [1]: RECM = 90572; topología [1-1]: RECM = 0.90998). Una intuición de lo que esto podría significar es que posiblemente el uso de redes neuronales no sea la técnica más adecuada para construir los modelos planteados, puesto que el computo realizado por una red que está compuesta únicamente por una neurona, es el equivalente al de realizar una regresión lineal entre la variable de salida y alguna de las variables de entrada. Por ello, ante esta posibilidad, en el epígrafe 7.2 analizaremos efectivamente

si el rendimiento que podemos obtener del uso de redes neuronales se asemeja al de utilizar otras técnicas más simples, como podría ser los modelos de regresión lineal.

Los resultados obtenidos para el experimento 1 se asemejan a los obtenidos en el análisis del modelo 2, dónde podemos observar que de igual forma, se descarta la utilización del algoritmo backprop, que en este caso no ha logrado converger en ninguna de las pruebas, y que las topologías de redes más simples son las que obtienen un menor RECM.

Finalmente, valorando los resultados obtenidos para ambos experimentos, vemos como la hipótesis planteada para el experimento 1, dónde predecimos la apertura del IBEX35, nos produce un predictor capaz de anticipar con un 70% de acierto si se va a producir un incremento o un decremento de esta variable con respecto a la apertura del día anterior. Este resultado, apoya la existencia de fenómenos causales entre el mercado japonés y el mercado español, que demuestran la interrelación existente entre los mercados bursátiles dentro del marco globalizado.

Por el contrario, tal y como se observa de los resultados del modelo 2, las tasas de acierto de las variaciones obtenidas se sitúan en torno a un valor del 50%. Esto viene a indicar que del total de las predicciones del signo que se han estimado para el conjunto de datos de prueba, solo se ha acertado la mitad de ellas, errando en la otra mitad de casos. Por tanto, a priori, diremos que las capacidades predictivas de nuestro sistema no son destacables, puestos que obtienen el mismo rendimiento que utilizar una función aleatoria que de forma arbitraria genere señales de compra o venta. En cualquier caso, en el epígrafe 7.3 de este capítulo, generaremos diferentes reglas de contratación utilizando filtros que limiten la cantidad de señales de compra-venta que se van a realizar, y estudiaremos el posible rendimiento de las reglas obtenidas. Con esa información, sí podremos realizar una valoración real de las capacidades predictivas de este modelo.

7.2 Validez del uso de la red neuronal.

Como se ha visto en el epígrafe anterior, una primera conclusión que se deriva del estudio de las diferentes topologías es que según se incrementa la complejidad de la red neuronal utilizada, mayor es el error que se produce y menor es la capacidad predictiva de esta. Es llamativo observar, que esta evidencia se cumple hasta el punto en el que las topologías más óptimas son aquellas formadas por una o dos capas de una única neurona. Nos encontramos por tanto ante la arquitectura de una red neuronal denominada *perceptrón* [35].

Como ya se ha visto anteriormente, internamente en las redes neuronales, los procesos ejecutados equivalen a primero calcular una ponderación de las señales de

entrada, para posteriormente sobre la suma de dicha ponderación ejecutar un algoritmo que se encargará de minimizar el error obtenido en la salida, mediante el reajuste de los pesos de las ponderaciones. Normalmente, las redes neuronales realizan este proceso en cada una de sus múltiples neuronas, y finalmente entregan el resultado conjunto a través de su capa de salida. Este proceso, como se trata de un *perceptrón,* con una única neurona, equivale a realizar una regresión lineal simple, en la que una variable endógena (en la red, la variable de salida), es explicada por un conjunto de variables (las variables de entrada), que se relacionan de forma lineal al multiplicarse por un conjunto de coeficientes (los pesos de la neurona de la capa oculta). Este modelo de regresión lineal, se podría estimar mediante una técnica de Mínimos Cuadrados Ordinarios, que se encargará de reajustar los coeficientes con el fin de minimizar el error cuadrático medio del modelo, de forma similar al proceso realizado por las redes neuronales artificiales. Por tanto, es evidente las equivalencias computacionales existentes entre una ANN con una única neurona y estimar por MCO un modelo de regresión lineal simple.

Puesto el objetivo de este capítulo es el de no solo analizar los resultados obtenidos, sino la calidad del proceso realizado, dedicaremos este epígrafe a comprobar la validez del uso de las redes neuronales artificiales como herramienta frente a modelos de regresión simple. Sin embargo, el hecho de demostrar que por medio de modelos de regresión lineal simple se puede obtener resultados iguales, no significa invalidar el uso de las ANNs como herramienta predictiva, sino la demostración de que es posible aplicar herramientas más simple y con menor coste computacional. Como herramienta, las ANNs te permiten estimar formas funcionales complejas, entre las que se incluyen las dependencias lineales que pueden ser estimadas a partir de modelos regresivos lineales. Será tarea del analista de datos, comprobar a partir de los resultados obtenidos, la conveniencia entre utilizar una herramienta u otra.

Para realizar esta comprobación, nosotros trabajaremos sobre el modelo número 1, puesto que es el único que presenta, a priori, capacidades predictivas mayores al 50%. Sobre este modelo, realizaremos un modelo de regresión lineal simple entre cada una de las variables de entradas (OPEN, HIGH, LOW, CLOSE del N225) y la variable endógena (OPEN del IBEX35).

Como se observa en la Figura 22, al realizar las gráficas dónde se representa la regresión lineal entre las variables del NIKKEI y la apertura del IBEX, se puede observar como la línea de regresión (en rojo) presenta una tendencia creciente, sobre la cual los datos, con cierto grado de dispersión, se encuentran ajustados. Esto demuestra la existencia de una cierta presencia de dependencia lineal entre las variaciones producidas por las variables del NIKKEI y del IBEX35, que podrían interpretarse como las relaciones existentes consecuencia de las correlaciones entre mercados globales.

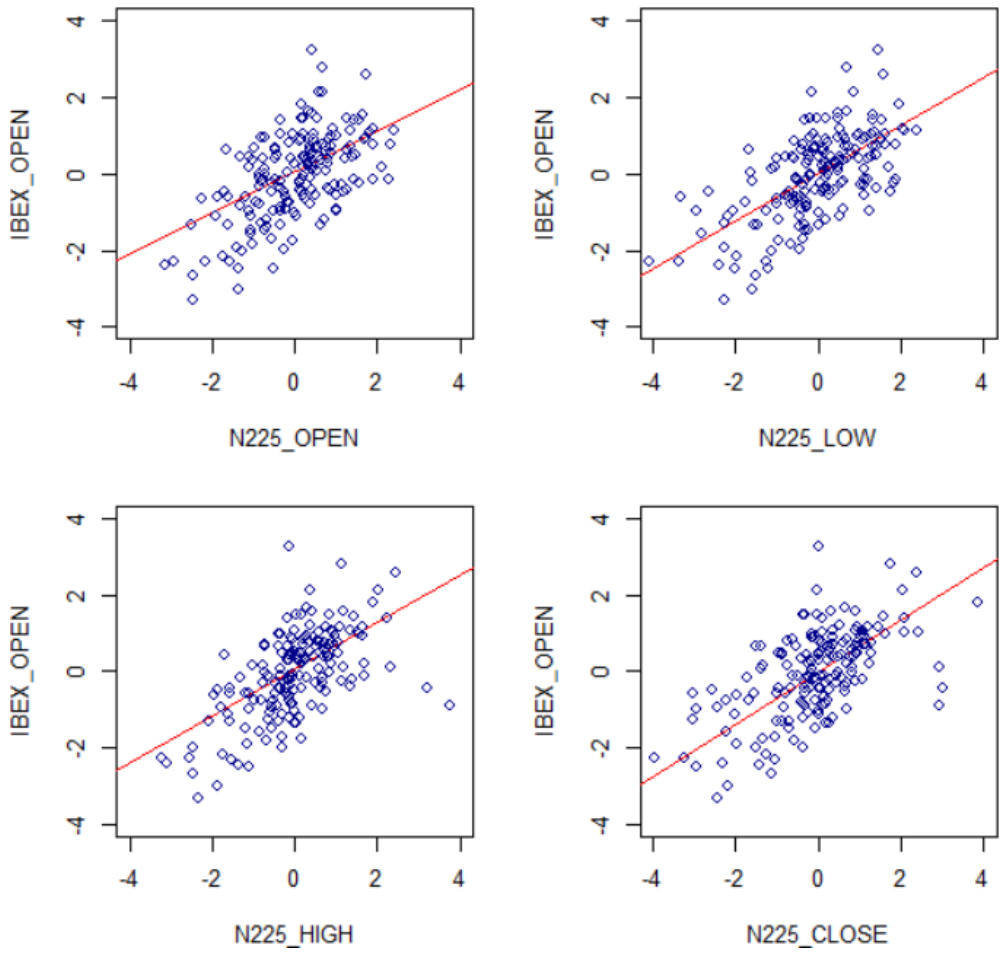

Figura 22: Regresión lineal de la apertura del IBEX35 con las variables del N225.

El modo de comprobar si este modelo de regresión demuestra mejores rendimientos que las redes neuronales de nuestro sistema, será mediante el cálculo de los mismos estadísticos que hemos utilizado previamente. De igual forma, con el fin de que los valores de dichos estadísticos sean bastante fiables, realizaremos un alto número de repeticiones con diferentes configuraciones del conjunto de datos para cada uno de los modelos y calcularemos el valor medio de dichos estadísticos. Los resultados obtenidos son los que se muestran en la Tabla 3.

	RECM	**R^2**	**%C**
$N225_{OPEN} \sim IBEX_{OPEN}$	0.99978	0.21405	0.67391
$N225_{HIGH} \sim IBEX_{OPEN}$	0.90613	0.36475	0.69197
$N225_{LOW} \sim IBEX_{OPEN}$	0.90912	0.34567	0.73194
$N225_{CLOSE} \sim IBEX_{OPEN}$	0.93046	0.32015	0.69405

Tabla 3: Estadísticos de los modelos de regresión lineal.

Analizando los resultados de la tabla, observamos como efectivamente con el cálculo de modelos lineales simples podemos obtener predictores con una tasa de acierto en el signo, cercanas a las obtenidas por las redes neuronales, con una desviación estándar de +/- 3 puntos porcentuales, dependiendo de la variable a utilizar. Esto nos indica que efectivamente, la relación de nuestra variable a predecir con cualquiera de las variables explicativas guarda una relación lineal simple, tal y como se esperaba tras el estudio de la topología.

De nuevo, decimos que estos resultados no invalidan el uso de redes neuronales, ya que con su uso podemos alcanzar un mismo resultado que con la aplicación de regresiones lineales, pero con la posibilidad de estimar formas funcionales de mayor complejidad, si las hubiera. Por tanto, con el uso de las redes neuronales nos aseguramos el poder estimar un mayor número de tipos de funciones, incurriendo eso sí, en un mayor coste de cómputo.

7.3 Análisis del rendimiento económico.

La utilidad directa de la información que estamos extrayendo durante la realización de este proceso de minería de datos es la de su aplicación en la toma de decisiones durante nuestra operativa bursátil. El objetivo perseguido es el de la generación de reglas de contratación que sean coherentes con la información obtenida y que nos permita obtener un cierto beneficio económico, sin incurrir en un factor de riesgo muy elevado.

Como hemos visto, gracias al primer modelo hemos podido obtener un predictor capaz de predecir el signo de las variaciones de la apertura del IBEX35 con una tasa de acierto del 70%, demostrando la existencia de relaciones causales entre los movimientos de la bolsa de Tokio y Madrid.

Si bien, el conocer la existencia de estas relaciones puede sernos de utilidad a la hora de modelar las estructuras existentes detrás de los mercados financieros, esta información, difícilmente podrá ser convertida a una regla de contratación que nos pueda aportar un beneficio económico real. Esto sucede porque en el momento en el que se obtiene la estimación de la apertura del IBEX, todavía no es posible operar dentro del mercado Español, y por tanto, no podremos abrir o cerrar una posición que nos pudiera aportar una ventaja real.

De forma contraria sucede con las características del modelo 2. En este caso, la hipótesis que se está modelando contiene un gran interés económico, puesto que estamos prediciendo las variaciones que se van a producir durante la sesión diaria del IBEX, lo cual nos permitirá posicionar nuestras operaciones de compra o venta al inicio de la sesión, apoyándonos en la información estimada. Una vez finalizada la sesión,

podremos cerrar dicha operación, obteniendo el beneficio asociado en el caso de acierto del predictor.

Sin embargo, como ya se indicó, a priori, las capacidades predictivas de este modelo son del 50%, lo cual no es rendimiento que nos permita la predicción efectiva de nuevos valores, ya que conseguiríamos un rendimiento similar mediante la utilización de un generador aleatorio de señales de compra/venta.

En cualquier caso, es importante señalar que la tasa de acierto obtenida, está calculada sobre la hipótesis de predecir y operar cada una de los registros diarios con los que contamos. Esto es, para cada día registrado en nuestro conjunto de datos, se obtiene un valor estimado, a partir del cual se realiza una operación bursátil. Sin embargo, nuestra red neuronal no siempre va a predecir variaciones del cierre del IBEX muy significativas, lo cual se puede interpretar como la certeza con la que la red predice el signo de las variaciones. Cuanto mayor sea, en términos absolutos el valor predicho, mayor certeza sobre el signo de dicha variación.

Haciendo uso de esto, se podría intentar alcanzar una tasa de acierto haciendo uso de filtros que limitarán el número de operaciones bursátiles a ejecutar, basándose en si la variable estimada supera o no un cierto valor preestablecido. Para ello, la definición de nuestras reglas de contratación debe ser la que se presenta a continuación:

$$Si \ \left(\hat{Y} > +f\right) \rightarrow \ Generar\ señal\ de\ compra.$$
$$Si \ \left(\hat{Y} < -f\right) \rightarrow \ Generar\ señal\ de\ venta.$$

$\hat{Y} = \ Estimación\ de\ la\ variable\ endógena.$
$f \ = \ Valor\ del\ filtro$

Haciendo uso de estas reglas, podremos generar todas las señales de compra y venta que conformarán nuestra operativa bursátil. Hay que recordar que, en el modelo 2, la variable a predecir es la que representa la tasa de variación producida durante la sesión diaria (SESVAR), y que por tanto, deberá ser al inicio de la sesión cuándo abramos las posiciones de compra o venta. El momento en el que deberemos cerrar esta posición y obtener su rendimiento, será al final de la sesión.

En nuestro análisis, realizaremos una comprobación de cuál es la rentabilidad obtenida para diferentes valores del filtro, comparando cada una de las estrategias que generemos a partir de las comparaciones del valor calculado mediante el Ratio de Sharpe. Con ello, podremos comprobar si existe beneficio en la utilización de filtros y, en caso afirmativo, identificaremos la opción que mayor rendimiento genere.

Para ello, tomaremos provecho de la relación existente entre la rentabilidad que obtendremos de iniciar una operación de compra y la rentabilidad obtenida para una operación de venta, cuándo la apertura y cierre de la operación se realizan en el mismo momento para ambos casos. En el caso de la operación de compra, la rentabilidad que obtendremos será la equivalente a la tasa de variación del precio de venta del activo al cierre, con respecto a su precio de compra en la apertura. Por el contrario, para la operación de venta, el valor de la rentabilidad será igual al de la tasa de variación del precio de compra al cierre con respecto al precio de venta en la apertura. Por tanto, como se puede observar, el valor de la rentabilidad para ambos tipos de operaciones será de mismo valor pero con signo inverso.

Por tanto, la forma de calcular la rentabilidad de nuestra estrategia bursátil para un determinado periodo será mediante el uso de las variaciones producidas durante cada una de las sesiones, que como sabemos ya hemos calculado y almacenado en nuestra variable SESVAR. Los valores de esta variable, los multiplicaremos por 1 o -1 en función de si se trata de una operación de compra o venta, respectivamente. Finalmente, a partir de la suma de los valores obtenidos de esta operación, obtendremos la rentabilidad para dicho periodo.

A partir de la rentabilidad calculada para cada uno de los diferentes filtros, calcularemos cada uno de los ratios de Sharpe asociados a ellos. Los resultados, calculados sobre los datos históricos del año 2014, son los que se muestran a continuación:

Valor del filtro	0.0	0.1	0.2	0.3	0.4	0.5
Nº de operaciones	209	196	177	154	117	97
Rent. acumulada	0.1770%	0.1792%	0.2109%	0.1999%	0.1106%	0.0845%
R. Sharpe	1.48669	1.51353	1.82770	1.80073	1.04504	0.83173

Tabla 4: Rendimiento obtenido en función del filtro

Como podemos ver, la aplicación de filtros sí nos aporta una mejora del rendimiento de nuestro predictor, mejorando nuestros niveles de rentabilidad y reduciendo la asunción de riesgo al reducir el número de operaciones a las que nos exponemos. Recordando que la tasa de acierto de nuestro predictor es del 50%, podemos esperar una muy baja rentabilidad, tal y como se puede ver en la tabla. Sin embargo, aun obteniendo una baja rentabilidad, el valor obtenido para el ratio de Sharpe supera la unidad, lo cual implica que nuestra exposición al riesgo debe ser bastante baja.

Para todos los valores de filtro que hemos analizado, encontramos el punto óptimo al aplicar un filtro de 0.2, dónde solo operamos aquellos incrementos y decrementos que nuestro predictor haya estimado, en términos absolutos, por encima de dicho valor. En este caso, el rendimiento obtenido se mejora, y el de igual manera sucede con el valor del ratio de Sharpe.

Por tanto, las conclusiones a las que llegamos es que si bien, las capacidades predictivas de nuestra red neuronal para el segundo modelo no permiten la obtención de notables rentabilidades, podemos hacer uso de un filtro ajustado a 0.2 con el fin de optimizar su rendimiento.

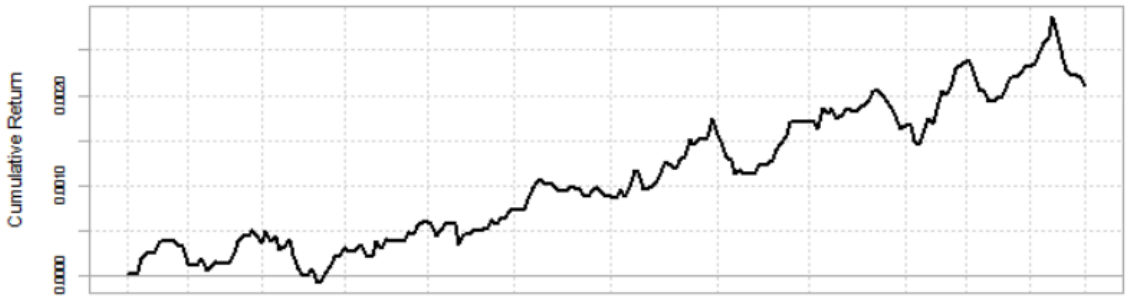

Figure 13: Evolución de la rentabilidad acumulada.

- Capítulo 8 –
Conclusiones y futuro desarrollo.

Este último capítulo servirá para presentar las conclusiones obtenidas a través de la realización de este proyecto, así como plantear posibles vías de trabajos sobre las cual se podría continuar desarrollando las ideas presentadas.

8.1 Conclusiones finales.

Tras haber completado todas las etapas planteadas para este Proyecto de Fin de Titulo, cabe analizar las conclusiones que hemos obtenido tras su realización. Para ello, podemos realizar un análisis en tres niveles: conclusiones sobre los resultados de los experimentos, conclusiones del proceso y conclusiones sobre la aplicación de la minería de datos en el campo de la predicción bursátil.

De los resultados obtenidos en los experimentos, se observa la existencia de una doble vertiente a través de la que un modelo puede ser evaluado: una vertiente estadística y una vertiente económica. Como hemos visto con los resultados del primer experimento, es posible la obtención de un modelo con altas capacidades predictivas pero que luego no sea de interés económico dada su imposibilidad de transformarse en reglas de contratación que puedan materializarse en un beneficio económico real.

De manera contraria sucede con el modelo dos, dónde la hipótesis planteada nos permitía realizar una evaluación económica, e incluso la optimización de sus resultados, sin tratarse de un modelo estadístico globalmente significativo para la realización de predicciones.

Por tanto, se concluye la importancia de, durante la etapa de planificación, realizar una correcta definición de los modelos que se quieran estimar acorde con los objetivos perseguidos con dicho proceso de minería de datos, ya sea bien el análisis de las relaciones causales subyacentes en las estructuras de los mercados financieros, o la obtención de rendimiento económico a partir de la información obtenida de dichos modelos.

De los experimentos realizados, también podemos ver cómo del análisis de los resultados no solo podemos obtener información de interés sobre los modelos planteados, sino que también sobre la eficacia de las herramientas utilizadas. Como ya se ha visto en el capítulo anterior, del análisis de la topología de la red neuronal hemos podido concluir que podría haberse obtenido un resultado semejante mediante la

utilización de otras técnicas más sencillas, como por ejemplo, modelos de regresión lineal.

Esto hace evidente la necesidad de siempre evaluar la eficacia de las herramientas seleccionadas a lo largo de nuestro proceso y de mostrar flexibilidad ante la necesidad de hacer uso de otras herramientas más adecuadas cuándo las características del modelo así lo requieran. Dichas herramientas, pueden ser integradas al proceso sustituyendo a las herramientas actuales, (ej. hacer uso de regresiones lineales en vez de redes neuronales artificiales), o pueden integrarse al procese complementando a las que ya se utilizan, optimizando su rendimiento (ej. la utilización de filtros en la evaluación del rendimiento económico).

A nivel de proceso, concluimos con la demostración de la eficacia de la metodología CRISP-DM, en la ejecución del proceso de minería de datos, y dónde se comprueba la importancia de todas sus etapas para el alcance de los objetivos propuestos. Será en la implementación de todas estas etapas donde entra en valor el papel del analista de datos, cuyas decisiones, basadas en las características de las herramientas, los modelos y, fundamentalmente, los datos, buscarán maximizar la eficacia del proceso, y la calidad de la información obtenida. Ejemplos de este tipo de decisiones han sido planteados en el Capítulo 6 de esta memoria, durante la presentación de la implementación del sistema.

Aun cuando los resultados obtenidos por nuestros modelos no aportan resultados notables, esto no invalida la eficacia del proceso implementado, ya que incluso la invalidación de los modelos planteados es un resultado en sí mismo. Además, la flexibilidad con la que se ha dotado al sistema implementado, permite de forma sencilla (únicamente modificando la *string* de configuración), la evaluación de nuevos modelos que nos sirvan para estudiar las relaciones causales entre mercados y de sus posibles aplicaciones prácticas, obtener rendimientos económicos.

Finalmente, la pregunta que cabría hacerse, y el objetivo fundamental de este proyecto, es valorar en qué medida nos beneficiamos de las técnicas de minería de datos en su aplicación de predicción bursátil. Las conclusiones extraídas de la resolución de este proyecto son que, efectivamente, las ventajas aportadas por este tipo de técnicas son bastante notables. Desde los ejemplos de la literatura presentados en el epígrafe 4.3, hasta la propia implementación de nuestro sistema minería de datos y los resultados obtenidos de su ejecución, son hechos que nos apoyan en afirmar su utilidad al aplicarse al sector financiero. Partiendo de las ventajas de hacer uso de las de las TICs (velocidad de cómputo, ejecución paralela, manejo de grandes volúmenes de datos, integración e interoperabilidad, etc.), hasta las avanzadas técnicas aportadas por cada uno de los grandes campos de estudio que conforman el campo de la minería de datos (inteligencia artificial, estadística, aprendizaje automático, etc.) permiten

afrontar una tarea, a priori, de bastante complejidad, alcanzando resultados bastante satisfactorios.

8.2 Futuro desarrollo.

La magnitud de los campos de estudios que abarca este Proyecto de Fin de Titulo, permite que, a partir del trabajo elaborado, se puedan desarrollar varias vías de desarrollos sobre las cuales poder seguir trabajando.

Una primera vía sería, a partir del sistema diseñado, continuar creando, estimando y validando nuevos modelos que nos permitan seguir estudiando las relaciones causales entre mercados. Recordar que el diseño del sistema permite hacer esto de forma sencilla, haciendo modificaciones en la *string* de configuración. De hecho, para esta vía de desarrollo, se plantea la posibilidad de acoplar un nuevo módulo previo, que implementando algoritmos basados en información mutua [36], sean capaces de fabricar modelos conformados por variables que presenten cierta dependencia entre ellas, y así incrementar las posibilidades de encontrar y estimar las relaciones causales entre mercados.

Explotando el carácter modular del sistema, otra posible vía de desarrollo sería la de implementar y probar la eficacia de otras herramientas de minería de datos diferentes a las utilizadas (arboles de decisión, reglas de asociación, máquinas de soporte vectorial, etc.), para así probar su rendimiento sobre el ámbito del mercado financiero. Para ello, como ya se ha dicho en el epígrafe anterior, será fundamental conocer previamente las características del problema que queramos afrontar, y hacer uso siempre de la herramienta más adecuada, pudiendo incorporarla sustituyendo a las ya existentes o complementando su análisis.

Finalmente, otra interesante vía de trabajo que se propone, podría ser la evaluación de modelos que integren no solo información financiera, sino que también incluyeran conjuntos de datos de los que se pudieran extraer factores de naturaleza no económica (factores políticos, factores sociales, etc.). Es, el análisis sobre la integración de conjuntos de datos de naturaleza muy diferente, una de las claves principales que impulsan el fenómeno del *Big Data* y la minería de datos como una de las tendencias principales en el corto y largo plazo.

Bibliografía y referencias.

[1] “En 2020 más de 30 mil millones de dispositivos…”, http://goo.gl/N1aIuJ

[2] “25 Hottest Skills of 2014 on LinkedIn”, http://goo.gl/cKsLBx

[3] “The real story of how big data analytics helped…”, http://goo.gl/GXwK71

[4] “CRISP-DM, still the top methodology for analytics…”, http://goo.gl/LvP8DC

[5] “What is CRISP-DM methodology”, http://goo.gl/eo7qjI

[6] “La metodología CRISP-DM”, http://goo.gl/5NSjKG

[7] “Four main languages for Analytics, Data Mining…”, http://goo.gl/DWcNZc

[8] CRAN, manual paquete *httr*, http://goo.gl/7ZchV9

[9] CRAN, manual paquete *jsolite,* http://goo.gl/Xbj9cI

[10] CRAN, manual paquete *neuralnet,* http://goo.gl/jNUKNs

[11] CRAN, manual paquete *neuralnet*, http://goo.gl/GiAStk

[12] Página oficial *RStudio,* http://goo.gl/36bL6b

[13] Página oficial *Yahoo Queary Language,* https://goo.gl/qpBVeM

[14] Wikipedia: Minería de datos, http://goo.gl/ip8D2a

[15] Pang-Ning Tan , Michael Steinbach , Vipin Kumar, Introduction to Data Mining, (First Edition), Addison-Wesley Longman Publishing Co., Inc., Boston, MA, 2005, pp.327-328

[16] Hamidah Jantan et al. / (IJCSE) International Journal on Computer Science and Engineering Vol. 02, No. 08, 2010, 2526-2534, http://goo.gl/4IK7wb

[17] Wikipedia: Algorithmic trading, http://goo.gl/7U0Vp7

[18] “Wall Streets Need For Trading Speed…”, http://goo.gl/OUJoUo

[19] Christian González Martel, “*Nuevas perspectivas del análisis técnico de los mercados bursátiles mediante el aprendizaje automático. Aplicaciones al índice general de la bolsa de Madrid*.” Dirigida por Fernando Fernández Rodríguez. Universidad de Las Palmas de Gran Canaria, 2003.

[20] Frederik Hogenboom, Financial Events Recognition in Web News for Algorithmic Trading, 2012, pp 368-377

[21] Gheorghe Ruxanda, Laura Maria Badea, “Configuring Artificial Neural Networks for stock market predictions”, Technological and Economic Development of Economy, 20:1, 116-132

[22] Fernando Fernández-Rodríguez, Christian González-Martel, Simón Sosvilla-Rivero. "On the profitability of technical trading rules based on artificial neural networks: Evidence from the Madrid stock market", 1999.

[23] Q.A. AL-Radaideh Adel Abu Assaf, E.Alnagi. "Predicting Stock Prices Using Data Mining Techniques" The International, 2013.

[24] Chin-Yin Huang, Philip K.P. Lin, "Application of integrated data mining techniques in stock market forecasting", 2014.

[25] Application of Neural Networks in Diagnosing Cancer Disease using Demographic Data. International Journal of Computer Applications 1 Nº26, 2010, pp. 76–85

[26] "Neural Networks in Medicine", http://goo.gl/NL0bvH

[27] Wikipedia: Función sigmoide, http://goo.gl/5gg9iU

[28] Wikipedia: Tangente hiperbólica, http://goo.gl/dWIqt4

[29] http://www.ieee.cz/knihovna/Zhang/Zhang100-ch03.pdf

[30] Wikipedia: Ratio de Sharpe, https://goo.gl/k4Y3lQ

[31] "Ten largest stock exchanges in the world…", http://goo.gl/w1ZxJT

[32] Página oficial *Yahoo Finance,* http://goo.gl/toJRRi

[33] Wikipedia: Root Mean Square Error, https://goo.gl/Hhbjxi

[34] Wikipedia: Coeficiente de determinación, https://goo.gl/s24PpL

[35] Wikipedia: Perceptrón, https://goo.gl/29jVvW

[36] Wikipedia: Información mutua, https://goo.gl/tkkzUB