UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
PROGRAMA DE MAESTRÍA Y DOCTORADO EN INGENIERÍA
INGENIERÍA DE SISTEMAS – OPTIMACIÓN FINANCIERA

MINERÍA DE DATOS PARA LA TOMA DE DECISIONES BURSÁTILES. EL CASO DE AMÉRICA MÓVIL.

TESIS
QUE PARA OPTAR POR EL GRADO DE:
MAESTRO EN INGENIERÍA

PRESENTA:
VÍCTOR MANUEL SALINAS GÓMEZ

TUTOR PRINCIPAL
DR. ELIO AGUSTÍN MARTÍNEZ MIRANDA
PROGRAMA DE MAESTRÍA Y DOCTORADO EN INGENIERÍA

CIUDAD UNIVERSITARIA, CDMX. NOVIEMBRE DE 2020.

JURADO ASIGNADO:

Presidente: DR. REYES ZÁRATE FRANCISCO JAVIER

Secretario: DRA. SOSA CASTRO MAGNOLIA MIRIAM

1er. Vocal: DR. MARTÍNEZ MIRANDA ELIO AGUSTÍN

2do. Vocal: M.I. RODRÍGUEZ RUBIO JORGE

3er. Vocal: M.I. MALFAVÓN RUIZ YONAHANDY

Lugar o lugares donde se realizó la tesis: Ciudad Universitaria, Ciudad de México.

TUTOR DE TESIS:

DR. ELIO AGUSTÍN MARTÍNEZ MIRANDA

FIRMA

Agradecimientos

A mi querida Universidad Nacional Autónoma de México y a la Facultad de Ingeniería, por recibirme nuevamente en sus espacios, por formarme y brindarme conocimiento a manos llenas.

Especialmente al Dr. Elio Martínez por su amistad, sus invaluables consejos y que gracias a su conocimiento y experiencia ayudó de manera importante a mejorar este proyecto.

Con enorme gratitud a la Dra. Miriam Sosa, la M.I. Yonahandy Malfavón, el Dr. Francisco Reyes y el M.I. Jorge Rodríguez por ser mis maestros, amigos y por sus valiosas aportaciones para enriquecer este trabajo.

A mis compañeros de clase, por compartir conmigo esta experiencia y formar lazos entrañables de amistad.

Índice general

Introducción

Actualmente, el desarrollo de tecnologías de análisis de datos masivos e inteligencia artificial ha permitido lograr avances significativos en ramas como la medicina, la computación, las finanzas, videojuegos, por mencionar algunos. Particularmente en el caso de las finanzas, el uso de inteligencia artificial ha desarrollado exponencialmente la aparición de algoritmos que generan cantidades abismales de información y que incluso operan directamente en los mercados bursátiles, explotando la ventaja competitiva de las computadoras sobre el tiempo de procesamiento de los cálculos y ejecutando las órdenes en fracciones ínfimas de tiempo.

Una de las herramientas en la que sientan sus bases la inteligencia artificial y el aprendizaje de máquinas es la minería de datos: algoritmos que buscan relaciones existentes entre los datos existentes de manera que puedan ser utilizables para tareas futuras o que permitan el reforzamiento del aprendizaje de algún modelo.

En cuanto a aplicaciones financieras, la minería de datos se emplea con frecuencia en la generación de reportes, detección de fraudes, control y gestión de pagos, entre otros.

Particularmente, en el área de finanzas bursátiles se emplea como auxiliares en la toma de decisiones de inversión, basándose en información generada en forma de noticias, fluctuaciones en el precio de los activos, anuncios de gobiernos o dependencias, y recientemente también han cobrado relevancia las publicaciones realizadas por dirigentes nacionales en redes sociales. Todo este conglomerado de información proveniente de distintas fuentes, brinda mejores oportunidades de poder encontrar variables que tengan mayor certeza a la hora de obtener una recomendación de parte del modelo. De cualquier manera, intentar pronosticar el comportamiento futuro de un activo financiero es empresa ardua, y se han desarrollado diversos enfoques para llevarla a cabo.

En esta investigación se desarrollan distintos modelos de minería de datos con la finalidad de obtener una recomendación de inversión para un activo que cotiza en el mercado bursátil mexicano, de manera que el inversionista disponga de una herramienta que sea auxiliar en la toma de decisiones. En el caso del mercado mexicano, la información disponible para poder formar un conjunto de datos que tenga todas las variables descritas con anterioridad, sigue siendo limitada. No obstante, es posible construir variables a partir de otras, con la finalidad de que el conjunto de datos sea más robusto y por lo tanto haya mejores oportunidades de encontrar un modelo suficientemente preciso.

A diferencia de conjuntos de datos convencionales, los que buscan modelar el comportamiento de un activo deben tratarse como una serie de tiempo financiera. Por lo tanto, es necesario tener en consideración la correlación que guardan las nuevas observaciones respecto a las previas, y este comportamiento debe estar incorporado en el modelo. En esta investigación se exponen estas consideraciones y se toman en cuenta para la validación de los datos.

Los algoritmos que se analizan y emplean para obtener recomendaciones son regresión logística, clasificador Naive-Bayes, árboles de decisión, k-vecinos más cercanos, análisis discriminante lineal, clasificación mediante perceptrones multicapa y máquinas de vectores de soporte, siendo el modelo de k-vecinos más cercanos el que tuvo una mayor exactitud, mientras que el modelo de bosques aleatorios obtuvo una mejor estabilidad en las recomendaciones a lo largo del proceso.

Todo el proceso se realizó mediante el lenguaje Python, aunque existen múltiples alternativas disponibles para llevarlo a cabo, algunas de ellas son gratuitas y son relativamente simples de usar. La decisión de llevarlo a cabo mediante este lenguaje de programación es debido a la sencillez de su sintaxis, a la amplia disponibilidad de librerías para análisis de datos y que además es posible trasladar la información a algún servicio de cómputo en la nube y realizar el procesamiento en dichas plataformas, acortando significativamente el tiempo necesario para obtener resultados.

La información contenida en esta investigación resultará atractiva para inversionistas, gestores de portafolios, corredores de bolsa, analistas, administradores de riesgo y en general para las personas interesadas en finanzas bursátiles. También resultará útil para personas que deseen introducirse en el análisis de datos financieros, empleando algoritmos y modelos de minería de datos, ya que se abordan temas como la recolección y limpieza de datos,

la preparación del modelo así como algunos ajustes necesarios para obtener mejores resultados.

Aunque existen diversas investigaciones aplicadas a mercados emergentes, muy pocas de ellas están enfocadas en México o en América Latina. Esto representa un área de oportunidad importante para los mercados bursátiles nacionales, teniendo en cuenta que en países desarrollados, un alto porcentaje de las transacciones se realizan de forma automatizada. De tal modo que la presente investigación es, hasta la fecha, una de las pocas que se enfocan en el mercado bursátil mexicano empleando algoritmos de minería de datos.

Con los resultados obtenidos en la investigación, es posible afirmar que los algoritmos de minería de datos constituyen una buena herramienta como predictores de los precios de algún activo, teniendo como base información del precio, volumen y la obtenida con base en el análisis fundamental y técnico.

La estructura en la cual se presenta la investigación es la siguiente: en el primer capítulo se exploran estudios recientes referentes a la aplicación de técnicas de minería de datos y aprendizaje máquina aplicados en los mercados bursátiles como una revisión de la literatura, particularmente en mercados emergentes.

Para el capítulo subsecuente, se detallan conceptos fundamentales necesarios para entender qué es la minería de datos, qué herramientas utiliza, conocer el potencial de sus aplicaciones y tener claro qué se requiere para poder emplearla y obtener resultados satisfactorios. Si bien el conocimiento de todas estas herramientas requiere de una profundización en conceptos matemáticos, cálculo vectorial, álgebra lineal y estadística, lo que se pretende es mostrar un esbozo que permita al lector comprender de mejor manera el proceso desarrollado y obtener los conceptos fundamentales de la minería de datos.

En el tercer capítulo se describe el conjunto de datos que será empleado para construir el modelo y posteriormente poder obtener recomendaciones de él. La combinación de información obtenida de los estados financieros (análisis fundamental), junto con indicadores y osciladores (análisis técnico) brinda una visión global del comportamiento de una emisora, y son estos los enfoques más empleados por los inversionistas.

Posteriormente, en el capítulo IV se realiza la construcción del modelo, tomando como base el conjunto de datos desarrollado en el tercer capítulo. Se exploran distintos modelos, se realizan ajustes tales como selección de variables y elección de hiperparámetros, con la finalidad de obtener las mejores predicciones y posteriormente se elige al que será empleado como resultado.

Con el modelo que provee las predicciones con mayor exactitud, se realiza una simulación con la finalidad de conocer el rendimiento final de una inversión basándose exclusivamente en los resultados del modelo.

En el apartado final se muestran las conclusiones, así mismo se exponen las limitantes que se tuvieron durante el desarrollo de la investigación y se aportan algunas ideas respecto a trabajos futuros que habrán de desarrollarse.

Objetivo

Con base en la información de precio, volumen y la obtenida mediante análisis fundamental y técnico de algún activo en particular, se busca comparar mediante métricas el desempeño de distintos modelos de minería de datos en la predicción del comportamiento futuro del precio del activo.

Elegir entre los mejores, aquel que haya mostrado una mejor estabilidad en cuanto a sus predicciones, de manera que exista un nivel de certeza aceptable y pueda ser utilizado como una herramienta auxiliar para el inversionista en la toma de decisiones.

El activo financiero que se analizará será una acción de alta bursatilidad que cotiza en el mercado mexicano, de la cual se cuenta con información suficiente para la construcción del dataset. El conjunto de datos se usará para la construcción del modelo, y posteriormente se realizará una simulación de inversión empleando las recomendaciones del modelo, contrastando con el índice más representativo del mercado mexicano.

Hipótesis

Los algoritmos de minería de datos presumiblemente tendrían la capacidad de pronosticar eficazmente el comportamiento futuro de series de tiempo financieras.

Capítulo I

Estado del arte

Los mercados bursátiles son un punto clave en toda economía en crecimiento, y cada inversionista en el mercado tiene el objetivo principal de maximizar sus ganancias y minimizar el riesgo asociado a ellas. Como resultado, numerosos estudios han sido dirigidos hacia la obtención de predicciones de los mercados empleando el análisis técnico y el fundamental utilizando como herramienta varias técnicas computacionales y algoritmos.

Con el desarrollo de tecnologías de análisis de datos y las finanzas actuales, la combinación de ellos desempeña un papel importante en los servicios financieros modernos, tales como los sistemas de recomendaciones. Comparado con el esquema tradicional, las finanzas basadas en datos muestran una serie de ventajas tales como mayor transparencia, mayor participación, mejor colaboración y menores costos entre intermediarios (Wang, 2019).

Anticipar el comportamiento futuro del precio de una acción no es tarea sencilla, independientemente del procedimiento que se tome en cuenta, por ello es esencial estudiar y aprender sobre los mercados financieros de forma extensa. Debido a que existen un número significativo de incertidumbres (tales como las condiciones económicas generales, factores sociales y eventos políticos, tanto nacionales como internacionales), es difícil predecir el comportamiento de los mercados financieros (Lin, 2018).

De acuerdo con (Wanjawa y Muchemi, 2014), todas las inversiones que se realizan en los mercados bursátiles están dirigidas por alguna forma de predicción. Confirmando la aseveración anterior, (Dunne, 2015) expone que existen tres enfoques principales para la predicción del mercado de valores: el análisis fundamental, el análisis técnico (*charting*) y el análisis tecnológico (*machine learning*). Aunque autores como (Ahmadi et al., 2018) simplemente

consideran dos categorías: análisis técnico y fundamental. En la presente investigación, se busca combinar los tres tipos de análisis planteados.

Los datos usados para realizar el análisis fundamental generalmente se presentan como datos no estructurados, lo que representa un reto para la captura de la información. Sin embargo, algunos autores como (Li et al., 2014), (Ballings et al., 2015), (Coyne, Madiraju y Coelho, 2017) y (Sorto, Aasheim y Wimmer, 2017) consideran que esta información ha probado ser de utilidad como predictor para el movimiento del precio de las acciones. Es necesario hacer hincapié en que la información existente en el mercado mexicano sigue siendo limitada, y que en la mayoría de las ocasiones es necesario llevar a cabo en primera instancia una limpieza de los datos. En el caso de la información fundamental, cada emisora presenta su reporte en el formato que considera conveniente, y de ahí la necesidad de considerarlos como datos no estructurados

Por otro lado, en el análisis técnico, el analista busca predecir el precio futuro de los instrumentos financieros realizando un estudio de las tendencias ocurridas durante el pasado y el presente para un instrumento en particular. (Ahmadi et al., 2018). Existe una gran cantidad de estudios referente a este tipo de análisis, incluso combinado con otro tipo de herramientas tales como el trabajo desarrollado por (Do Van, Minh Hai y Hieu, 2018) en donde se emplea una combinación de análisis técnico con el análisis de componentes principales; a su vez, (Zhang et al., 2018) combina el análisis técnico con herramientas econométricas como los modelos ARIMA y los GARCH. De acuerdo con los resultados mostrados por estos trabajos, es posible anticipar los movimientos en los precios de las acciones con una exactitud suficiente. De estos modelos econométricos, se retoman las características propias de una serie de tiempo financiera, tales como la dependencia de las observaciones, la tendencia, estacionalidad, temporalidad y ciclicidad, y se deberán tener en cuenta durante la construcción del modelo de minería de datos.

Un alto porcentaje de inversionistas emplea análisis técnico hoy en día, y su fundamento es que el comportamiento de los precios en los activos corresponde a las leyes de la oferta y la demanda, buscando a través de los gráficos la ocurrencia de patrones y movimientos predecibles que puedan ser explotados en el futuro. El factor psicológico también juega un papel importante en las fluctuaciones del precio del activo, y el análisis técnico refleja perfectamente el comportamiento de las masas. El análisis fundamental sigue siendo empleado en gran medida para inversiones a largo plazo, y muestra la salud financiera que goza la empresa, dan indicios acerca de sus expectativas futu-

ras en cuanto a utilidades y brindan un mejor panorama sobre la situación de la misma. De manera que no existe posibilidad alguna de afirmar que un tipo de análisis es mejor o peor que otro; ambos poseen características que los convierten en herramientas eficaces para obtener información que permita tomar decisiones de inversión. En todo caso, dependerá del horizonte de inversión que se busque y del nivel de riesgo que se esté dispuesto a tolerar.

Elaborar pronósticos de los mercados de valores es un área atractiva para investigadores, inversionistas y analistas financieros a pesar de su evidente dificultad (Zhang et al., 2018). Con la masificación de los datos, es frecuente encontrar cada vez más estudios dirigidos a utilizar herramientas de aprendizaje automático e inteligencia artificial para obtener conocimiento de los mercados.

Algunos de los modelos que han sido implementados en el aprendizaje automático con la finalidad de obtener predicciones financieras son las redes neuronales, sistemas adaptativos de inferencias difusas, algoritmos genéticos, analisis de series de tiempo, regresión, máquinas de soporte vectorial y análisis de componentes principales (Do Van et al., 2018; Lin, 2018). En este sentido, las herramientas que se emplean en la presente investigación se encuentran validadas por estos y otros autores, en donde en su mayoría se han logrado resultados prometedores.

La aplicación de algoritmos de minería de datos ha sido empleada ampliamente para pronosticar el comportamiento futuro de las series de tiempo financieras, tanto en mercados desarrollados como emergentes. Por ejemplo, el trabajo desarrollado por (Abuzir y M.Baraka, 2019) muestra avances muy prometedores con el uso de redes neuronales artificiales para pronosticar precios futuros, lo cual contrasta con la presente investigación, donde las redes neuronales no lograron obtener los mejores resultados. En la investigación desarrollada por (Gupta et al., 2019) se muestra el uso de algoritmos de agrupamiento, clasificación y regresión para el pronóstico del precio de acciones, incorporando como un componente adicional el análisis de sentimientos obteniendo información de redes sociales. Si bien, los algoritmos de agrupamiento y regresión son empleados en esta investigación, la información disponible sobre el sentimiento del mercado es prácticamente nula. Con ayuda de una librería especializada en obtención de información de diversas redes sociales, se experimentó recopilar información al respecto sin lograr resultados satisfactorios, y que de haberse integrado al modelo habría generado ruido no deseado.

A pesar del incremento en trabajos relacionados con la predicción de mer-

cados bursátiles, tanto con enfoques en análisis técnico como fundamental, algunos trabajos como (Fama, 1965) y (Malkiel y Fama, 1970) afirman que el mercado tiene un comportamiento estocástico y que por lo tanto no es predecible. Lo anterior condujo a dos famosas hipótesis: la de hipótesis de la caminata aleatoria (*random walk*) y la hipótesis de los mercados eficientes.

Sin embargo, trabajos como el de (Basak et al., 2019) indican que debido a la volatilidad implícita de los mercados de valores, convierten en un reto el realizar predicciones, y se hace hincapié en que al minimizar el error de tales predicciones, se estará gestionando también el riesgo. Concluyen con que el análisis y pronóstico de mercados bursátiles, a menudo tienen características particulares por país, o incluso por región, en cuanto a su comportamiento, cultura y muchos otros factores cruciales. En este sentido, la falta de información relacionada a la aplicación de algoritmos de minería de datos para el mercado mexicano, presenta un área de oportunidad significativo, teniendo en cuenta que debido a la globalización y a la situación comercial de México, el movimiento de los precios presenta un alto grado de correlación con las bolsas de valores norteamericanas y con el tipo de cambio.

Sin duda existe una amplia variedad de criterios y sistemas para operar en los mercados de capitales, y no necesariamente uno es mejor que otro; depende principalmente de los objetivos del inversionista y de la gestión de riesgo. De manera que la investigación desarrollada no pretende cuestionar o debatir sobre las reglas que deberá seguir el inversionista, sino que busca formar parte de la gama de herramientas a su alcance para ayudar en la toma de decisiones informadas.

Capítulo II

Minería de datos

2.1. Introducción

Los modernos sistemas computacionales están acumulando datos a una tasa inimaginable y desde una gran variedad de fuentes: desde equipos de punto de venta, transacciones de tarjetas de crédito, satélites de observación en el espacio y también un crecimiento exponencial del volumen de información disponible en internet.

Algunos de los ejemplos de grandes volúmenes de información se enlistan a continuación:

- Los satélites con los que cuenta la NASA actualmente, generan aproximadamente un terabyte de datos cada día. Esto es más que el total de información que los satélites previos habían transmitido.
- Muchas compañías mantienen grandes *data warehouses* que contienen transacciones de clientes. Un data warehouse pequeño podría contener más de cien millones de transacciones u operaciones de una corporación.
- Existen vastas cantidades de datos almacenados cada día por dispositivos que lo hacen de manera automática, tales como transacciones realizadas con tarjetas de crédito, *logs* de webs y servidores, incluso datos no estructurados tales como video grabado por circuitos cerrados de televisión (CCTV).
- A enero de 2020, existen más de 2500 millones de usuarios activos de Facebook, con un estimado de 8,100 millones de publicaciones cada día.

- A mediados de 2019, se estima que existen alrededor de 330 millones de usuarios activos de Twitter, enviando 500 millones de *tweets* cada día.

En paralelo a los avances en dispositivos de almacenamiento de información —lo que ha hecho posible que se pueda almacenar tal cantidad de datos a un costo relativamente bajo— se ha ido descubriendo poco a poco que estos datos contienen conocimientos ocultos que pueden ser críticos para el crecimiento o declive de una empresa, por ejemplo, o que podrían conducir a descubrimientos importantes de alguna ciencia, poder pronosticar con mayor precisión el clima y los desastres naturales, inclusive posibles curas de enfermedades mortales... conocimiento que literalmente podría significar la diferencia entre la vida y la muerte.

En el entorno financiero, los datos constituyen la materia prima que permite a inversionistas, administradores de riesgo, operadores bursátiles, por mencionar algunos, tomar decisiones con mayor certeza. Por lo tanto, la minería de datos constituye una herramienta con un enorme potencial de aplicación en este sector.

2.2. Conceptos fundamentales

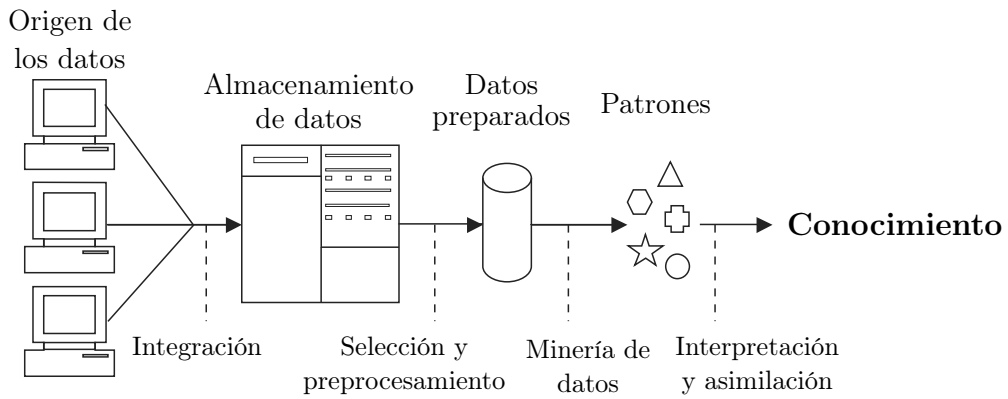
La minería de datos (*data mining*) forma parte de un proceso llamado *knowledge discovery in databases* (KDD) el cual se define como la extracción no trivial de información previamente desconocida y potencialmente útil desde una base de datos (tales como reglas, constantes, regularidades) (Frawley, Piatetsky-Shapiro y Matheus, 1992).

Una definición más simple es que KDD se refiere al proceso de identificar patrones válidos, novedosos, potencialmente útiles y principalmente entendibles.

La minería de datos es el proceso interdisciplinario de descubrimiento de patrones en grandes conjuntos de datos (*datasets*) empleando metodologías que combinan el uso de herramientas como aprendizaje de máquina (*machine learning*), estadística y sistemas de base de datos.

El término minería de datos realmente no representa los resultados obtenidos de su aplicación, pues lo que se busca es obtener conocimiento (y no los datos, *per se*); el término más apropiado debería ser "minería de conocimiento a partir de datos".

Figura 1: Diagrama esquemático del proceso KDD.



Fuente: (Bramer, 2016).

Existe una creencia popularizada de que minería de datos es sinónimo de KDD mientras que otros consideran a la minería de datos como un paso esencial dentro del descubrimiento del conocimiento. En la Figura 1, se presenta de manera esquemática el proceso KDD (Bramer, 2016).

En términos simples, la minería de datos es una herramienta que a través del descubrimiento de patrones y procesos secuenciados funciona como auxiliar en la toma de decisiones, otorgando un valor agregado a todo proyecto aquel en el que se integre.

La minería de datos emplea un sinfín de algoritmos y modelos para descubrimiento de patrones. Es posible clasificarlos de acuerdo con el tipo de variable descriptiva que presenten los datos (categóricas o contínuas), el tipo de variable objetivo (categórica o contínua) y el tipo de aprendizaje que se desea obtener. De manera resumida se muestran en la Tabla 1.

La materia prima de la minería de datos es la información, la cual puede estar disponible de forma pública o restringida (incluso en ambientes empresariales). A toda la información en su conjunto se le conoce como *dataset*, conformado por una determinada cantidad de atributos (también conocido como variables o *features* en inglés) y una gran cantidad de instancias (también llamadas entradas o registros).

Existen dos tipos de datos, los cuales deben ser tratados y procesados de manera distinta: los datos etiquetados y los no etiquetados. Para los primeros, existe un atributo específico (llamado clase) cuyo valor se busca predecir por nuevas instancias que no han sido observadas con anterioridad; cuando se

Tabla 1: Clasificación de modelos de minería de datos.

	Aprendizaje supervisado		Aprendizaje no supervisado
	Objetivo contínuo	Objetivo categórico	Objetivo exploratorio
Predictores continuos	Regresión lineal Redes neuronales k-vecinos cercanos Ensamblados	Regresión logística Redes neuronales Análisis discriminante k-vecinos cercanos Ensamblados	Componentes principales Agrupamiento (*Clustering*) Filtrado colaborativo
Predictores categóricos	Regresión lineal Redes neuronales Árboles de regresión Ensamblados	Redes neuronales Árboles de clasificación Regresión logística Naive Bayes Ensamblados	Reglas de asociación Filtrado colaborativo

Fuente: (Bramer, 2016)

tiene un *dataset* con estas características el enfoque que se plantea es el de aprendizaje supervisado. Para los datos no etiquetados, no existe un atributo para pronosticar, sino que se pretende extraer la mayor cantidad posible de información de los datos crudos, sin saber con exactitud qué resultados esperar; a este enfoque se le conoce como aprendizaje no supervisado.

Específicamente hablando de aprendizaje supervisado, cuando el atributo a predecir pertenece a un conjunto de categorías (por ejemplo, que puede tomar valores como "bueno" o "malo", o "bicicleta", "auto", "tren", "avión", etcétera) se trata de una tarea de clasificación. Cuando el atributo puede tomar un valor numérico (por ejemplo, el precio por metro cuadrado de construcción, o la cantidad de personas que vieron una película), se trata de una tarea de regresión.

Dicho de otro modo, los principales objetivos de la minería de datos son:

- *Objetivos predictivos.* Son llevados a cabo usando parte de las variables para predecir una o más de las otras variables, por ejemplo, tareas de clasificación, regresión, detección de anomalías (*outliers*).

- *Objetivos descriptivos.* Son logrados mediante la identificación de patrones que describen los datos y que son fácilmente entendibles por el usuario, por ejemplo, tareas de agrupamiento, descubrimiento de reglas de asociación y descubrimiento secuencial de patrones (Gorunescu, 2013).

2.3. Preprocesamiento de datos

Usualmente, la información contenida en los *datasets* incluye datos que no son relevantes, información erróneamente capturada o incompleta, valores faltantes, entre muchos errores potenciales que afectan directamente la precisión y desempeño de la tarea de minería de datos.

El primer paso a realizar una vez que se ha recabado la información que será empleada es preparar el *dataset* para que pueda ser procesado de forma óptima por el modelo.

2.3.1. Limpieza del *dataset*

Los *datasets* pueden contener valores erróneos por una gran variedad de razones, incluyendo errores de medición, subjetividad, funcionamiento deficiente de instrumentos de automatización, errores humanos, por mencionar algunos. Los valores erróneos pueden ser clasificados en aquellos que son posibles valores del atributo y los que no: por ejemplo, el número 84.39 podría ser capturado como 8.439 o una variable categórica como "color" podría ser capturada accidentalmente como azul en lugar de café. Estos valores se consideran como no válidos en lugar de ruido. De manera que los valores no válidos pueden ser fácilmente detectados y corregidos u omitidos del *dataset*.

Algunos otros elementos que deben ser tomados en cuenta son:

- Un atributo numérico podría tomar solo 6 valores enteros uniformemente distribuidos. La mejor forma de tratar esta variable sería como una variable categórica en lugar de una continua.
- Todos los valores de un atributo son idénticos. La variable debería ser omitida del *dataset*.
- Todos los valores de un atributo son idénticos con excepción de uno. Se debe analizar para determinar si se trata de un error o si es un valor poco frecuente pero que puede ser tomado por la variable.
- Algunos valores podrían estar fuera del rango normal del atributo, por ejemplo debido a que accidentalmente fueron capturados de forma equivocada, o si son registrados de forma automática, se presentó una avería en el instrumento de medición, etcétera.

El último punto es especialmente relevante, pues puede tratarse verdaderamente de un error o bien podría tratarse de *outliers*, es decir, valores genuinos que fueron registrados junto con todos los demás. Antes de descartar una instancia que presente este comportamiento, es conveniente analizar e investigar el origen de los datos.

2.3.2. Valores faltantes

En muchos de los *datasets* del mundo real, existen instancias que no cuentan con un valor en todos los atributos, ya sea por que alguno de ellos no fue registrado, o no aplica para la instancia correspondiente (por ejemplo en una serie de tiempo financiera, podría tenerse una instancia de una fecha en la que el mercado no operó).

Aunque existen muchas formas de tratar este tipo de instancias, algunas de las más comunes son las que se describen:

- Descartar todas aquellas instancias en las que al menos un atributo no contiene valor, y conservar únicamente las restantes.
- Reemplazar los valores faltantes por el dato más frecuente, o por el valor promedio.
- Realizar una interpolación entre los valores existentes para completar el valor faltante.

El responsable de llevar a cabo el proceso de minería será el encargado, con base en su experiencia, de determinar cuál es el método más conveniente para tratar los valores faltantes en el *dataset*.

2.3.3. Normalización y estandarización

Un problema común al que hay que enfrentarse cuando se trabaja con grandes volúmenes de datos numéricos es la diferencia existente en las dimensiones de los datos. Mientras que algunos atributos podrían tener valores en un rango de 0 a 100, algún otro podría rebasar fácilmente el millón. Por ejemplo, si en un *dataset* cualquiera una de las variables es la edad de una persona y otro atributo consiste en el ingreso anual.

Algunos algoritmos emplean medidas tales como la distancia entre dos puntos (o datos) para determinar alguna regla de asociación o de agrupamiento, de manera que la diferencia tan importante entre las dimensiones

puede ocasionar alteraciones en el funcionamiento del modelo. Para solucionar este problema, existen principalmente dos alternativas para que todos los atributos coexistan en rango que solvente este inconveniente.

El primero de ellos es la normalización de los datos, proceso que consiste en reescalar los valores de las instancias de manera que el rango de ésta vaya de 0 a 1. Sean x_n el valor de una instancia n, $x_{\text{mín}}$ y $x_{\text{máx}}$ los valores mínimo y máximo, respectivamente, de todas las instancias para el atributo a reescalar, el valor normalizado de la instancia x_n^* es determinado como:

$$x_n^* = \frac{x_n - x_{\text{mín}}}{x_{\text{máx}} - x_{\text{mín}}}\,. \tag{2.1}$$

Con la misma nomenclatura, la estandarización de los atributos emplea una transformación utilizando una función Z, de manera que los valores reescalados de la instancia tengan media igual a 0 y varianza igual a 1. Para poder determinar los valores estandarizados de las instancias z_n^*, es necesario aplicar la siguiente transformación:

$$z_n^* = \frac{x_n - \overline{x}}{s}\,, \tag{2.2}$$

donde:

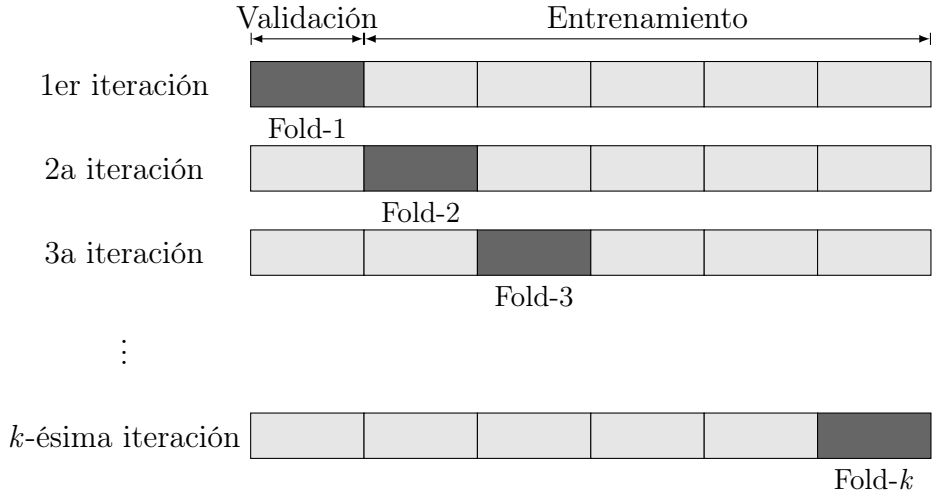
$\overline{x}$ = valor promedio del atributo,
s = desviación estándar del atributo.

2.3.4. Conjuntos de entrenamiento y prueba

Los distintos algoritmos de minería de datos emplean conjuntos de datos para aprender y descubrir relaciones existentes entre las variables explicativas. Sin embargo, generalmente la información disponible es limitada, por lo que se requiere optimizar aquella con la que se cuenta, de manera que sea posible usarla como parte de la etapa de aprendizaje. Los distintos algoritmos de minería de datos establecen relaciones y reglas con base en la información que le es suministrada en la etapa de aprendizaje (o entrenamiento), y que posteriormente sean empleados con información nueva y que otorgue un pronóstico.

En los procesos de aprendizaje supervisado, lo usual es dividir el *dataset* en dos subconjuntos: entrenamiento (*training*) y prueba (*test*), los cuales serán aplicados en la etapa de aprendizaje del modelo y para evaluar su desempeño, respectivamente.

Figura 2: Validación cruzada típica.



Fuente: Elaboración propia con fines ilustrativos.

Cuando se sigue este camino, puede suceder que se sobreajuste o se subajuste el modelo. Lo que se busca es evitar que cualquiera de esas dos condiciones ocurran ya que afectan la capacidad de predicción del modelo.

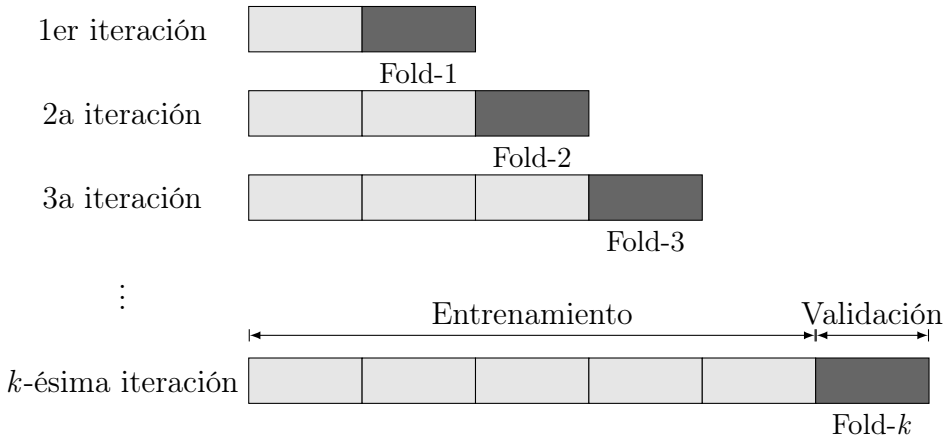
Uno de los métodos más usuales es el de dividir el *dataset* en una proporción determinada, por ejemplo, 80 % de los datos para entrenamiento y el 20 % restante para prueba y validación.

El subconjunto de entrenamiento, a su vez se divide en dos partes durante la selección de variables y elección del modelo; se realiza de esta manera para tener un esbozo sobre cómo será el desempeño del modelo cuando se presenten nuevos datos nunca antes vistos.

El enfoque más popular para realizar este tipo de procedimientos es conocido como validación cruzada (*cross-validation*, CV). De acuerdo con (Browne, 2000), la validación cruzada se empleó por primera vez en 1951 cuando se buscó evaluar el uso de una ecuación de regresión lineal para predecir una variable de criterio mediante una función lineal de variables predictivas. Consiste en dividir el subconjunto de entrenamiento en k grupos de igual tamaño, *k-1* grupos se emplean para el entrenamiento y el grupo restante se usa para validar los resultados del modelo. De manera gráfica es posible apreciarlo en la Figura 2.

Sin embargo, para el aprendizaje de un modelo que corresponda a una

Figura 3: Validación cruzada para una serie de tiempo.



Fuente: Elaboración propia con fines ilustrativos.

serie de tiempo, no sería correcto emplear tal enfoque, sino que deberá adaptarse a las características de una serie financiera. Los modelos de aprendizaje supervisado asumen que los datos son independientes idénticamente distribuidos (i.i.d.), sin embargo esto no necesariamente sucede con una serie de tiempo financiera. Para atender esta necesidad, es posible dividir el *dataset* de entrenamiento en k grupos distintos de manera que cada uno de ellos sea incremental del anterior, de esta manera es posible evaluar el desempeño del modelo ante el surgimiento de nuevos datos dependientes del tiempo. Para observar de mejor forma el comportamiento de este tipo de validación cruzada, véase la Figura 3.

2.3.5. Selección de variables

Con frecuencia, se trabaja con *datasets* con unas dimensiones considerablemente grandes. Algunos de sus atributos, sin embargo, son irrelevantes o insignificantes para la variable objetivo. La contribución de este tipo de atributos generalmente es poca a comparación de los atributos críticos, tanto que podría ser nula. Además, conservar este tipo de atributos podría causar una serie de problemas tales como: ocupación de espacio innecesario de almacenamiento, generación de ruido provocando que el desempeño del modelo sea inferior al esperado, y por otro lado, al tener una mayor cantidad de atributos

toma más tiempo al modelo realizar su entrenamiento. Para solucionar estos inconvenientes, es necesario realizar el proceso de selección de variables.

La selección de variables es el proceso de obtener un subconjunto del *dataset* original de tal forma que cumple con un criterio de selección de atributos, el cual elige aquellos que son más relevantes del *dataset* (Cai et al., 2018).

Este proceso es uno de los más importantes al realizar cualquier tarea de reconocimiento de patrones. Sin embargo, (Guyon y Elisseeff, 2003) señala que podría ser omitido cuando se tenga un conocimiento previo de aquellos atributos que tengan un impacto en el modelo, o que por requerimientos explícitos deban considerarse. Tampoco es necesario realizar este proceso cuando no se tiene una variable objetivo determinada, y se está realizando el proceso de minería con el objetivo de un aprendizaje no supervisado como se describió en la sección 2.2.

Para la implementación de la selección de variables, los atributos numéricos o categóricos deben ser tratados de forma diferente, y también debe ser tomado en cuenta si la variable a predecir es numérica o categórica.

Los métodos de selección de variables pueden basarse en estadística, teoría de la información, entre otros, y pueden ser clasificados de acuerdo con varios estándares. Una clasificación es la propuesta por (Cai et al., 2018).

(a) Con base en el *dataset* de entrenamiento (etiquetado, no etiquetado o parcialmente etiquetado), los métodos de selección pueden dividirse en supervisados, no supervisados y semisupervisados.

(b) Con base en su relación con los modelos de aprendizaje, pueden dividirse en filtros, envolventes y embebidos.

(c) Con base en el criterio de evaluación, pueden dividirse en correlación, distancia euclidiana, consistencia, dependencia y medida de información

(d) Con base en las estrategias de búsqueda de atributos, pueden clasificarse en incrementales hacia adelante, eliminación hacia atrás y modelos híbridos.

(e) Con base en el tipo de salida, pueden ser organizados como aquellos que asignan una calificación a cada atributo, y los que crean un subconjunto de variables.

Recapitulando, la importancia de la selección de variables es:

- Permite al modelo realizar el proceso de entrenamiento en menor tiempo.
- Reduce la complejidad del modelo, por lo que es más fácil de interpretar.
- Puede mejorar la precisión del modelo si el subconjunto de variables es el adecuado.
- Reduce el riesgo de sobreentrenamiento [1].

En la práctica es usual emplear métodos de selección hacia adelante o de eliminación hacia atrás.

La selección hacia adelante implica iniciar sin variables en el modelo, y posteriormente se van agregando de manera secuencial. La primera variable que se considerará introducir en la ecuación será la que tenga mayor correlación, positiva o negativa, con la variable dependiente. Dicha variable se introducirá en el modelo sólo si cumple el criterio de entrada. Si se introduce la primera variable, a continuación se considerará la variable independiente cuya correlación parcial sea la mayor y que no esté en la ecuación. El procedimiento termina cuando ya no quedan variables que cumplan el criterio de entrada.

En la eliminación hacia atrás, por el contrario, se inicia integrando todas las variables al modelo y después se van excluyendo una tras otra. Aquella

[1] Se conoce como sobreajuste, sobreentrenamiento u *overfitting* en inglés, al efecto negativo que presenta un algoritmo cuando es entrenado con unos ciertos datos para los que se conoce el resultado deseado, pero que no son suficientemente representativos del fenómeno que se desea analizar, de manera que se corre el riesgo de que se presente una falla de generalización de las predicciones.

El algoritmo de aprendizaje debe alcanzar un estado en el que será capaz de predecir el resultado en otros casos a partir de lo aprendido con los datos de entrenamiento, generalizando para poder resolver situaciones distintas a las acontecidas durante el entrenamiento. Sin embargo, cuando un sistema se entrena demasiado (se sobreentrena) o se entrena con datos extraños, el algoritmo de aprendizaje puede quedar ajustado a unas características muy específicas de los datos de entrenamiento que no tienen relación causal con la función objetivo.

Durante la fase de sobreajuste el éxito al responder las muestras de entrenamiento sigue incrementándose mientras que su actuación con muestras nuevas va empeorando. El sobreentrenamiento es el uso de modelos o procedimientos que violan la parsimonia, esto es, que incluye más elementos de los necesarios o que usa enfoques más complicados que el necesario (Hawkins, 2004)

variable que tenga la menor correlación parcial con la variable dependiente será la primera en ser considerada para su eliminación. Si satisface el criterio de eliminación, se eliminará. Tras haber excluido la primera variable, se pondrá a prueba aquella variable, de las que queden en la ecuación, que presente una correlación parcial más pequeña. El procedimiento termina cuando ya no quedan en la ecuación variables que satisfagan el criterio de eliminación.

2.4. Algoritmos de clasificación

Una de las tareas del aprendizaje supervisado es la de realizar la clasificación de una clase basada en nuevas observaciones. La clasificación es una actividad que realizamos cotidianamente. Esencialmente consiste en agrupar objetos de manera que queden asociados a una categoría mutuamente excluyente que en minería de datos se denomina como "clase".

El ejemplo más simple podría ser clasificar figuras geométricas basado en el número de lados que tienen, en triángulo, rectángulo, pentágono, etcétera. A continuación se describen brevemente algunos de ellos.

2.4.1. Naïve Bayes

El algoritmo de clasificación de Naïve Bayes emplea la rama de las matemáticas conocida como *teoría de la probabilidad* para encontrar cuál es la más probable clasificación de una instancia.

Los estadísticos utilizan la palabra experimento para describir cualquier proceso que genere un conjunto de datos. Un ejemplo simple de experimento estadístico es el lanzamiento de una moneda al aire. En tal experimento sólo hay dos resultados posibles: águila o sol. (Walpole, Myers y Myers, 1999).

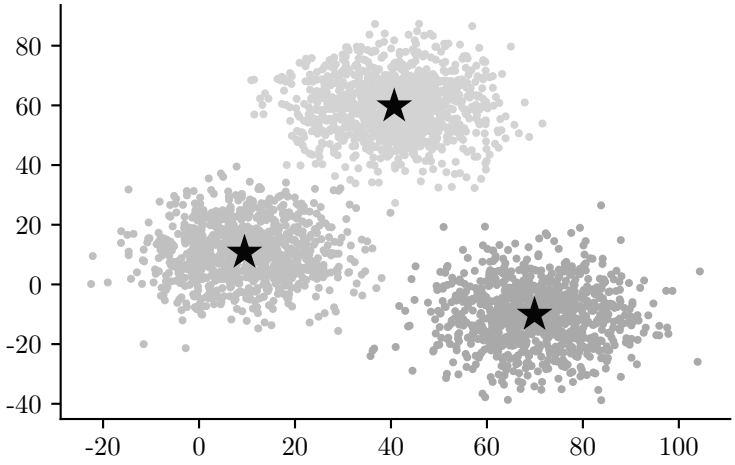
Por lo tanto, la probabilidad de ocurrencia de un evento A es la suma de los pesos de todos los puntos muestrales en A. Por lo tanto,

$$0 \leq P(A) \leq 1\,.$$

Si un experimento puede dar como resultado cualquiera de N diferentes resultados que tienen las mismas probabilidades de ocurrir, y si exactamente n de estos resultados corresponden al evento A, entonces la probabilidad del evento A es:

$$P(A) = \frac{n}{N}\,.$$

Figura 4: Agrupamiento (*clusters*).



Fuente: Elaboración propia con fines ilustrativos.

A esta probabilidad se le conoce como probabilidad *a priori*. Por otro lado, la probabilidad de que ocurra un evento B cuando se sabe que ya ocurrió algún evento A se llama probabilidad condicional y se denota con $P(B|A)$.

La probabilidad condicional de B, dado A se define como:

$$P(B|A) = \frac{P(A \cap B)}{P(A)} \quad \text{siempre que } P(A) > 0\,.$$

Este algoritmo, básicamente calcula las probabilidades de asociación basado en los datos previos y las emplea en para el pronóstico de nuevas clases.

2.4.2. Agrupamiento (*clustering*)

Existen varios casos en los que se requiera agrupar un conjunto de datos dadas sus características, sin embargo podrían ser tantos los atributos para evaluar que hacerlo de forma manual resultaría poco conveniente. La idea de los algoritmos de agrupamiento es la de estimar la clasificación de una instancia nueva basado en las categorías cuyas instancias están más cerca de la nueva observación. De manera gráfica es sencillo distinguir la tarea desempeñada por el algoritmo como se muestra en la Figura 4.

Para determinar qué categoría es la más cercana, el algoritmo calcula la distancia empleando alguna medida determinada. Algunas de las más usuales son la distancia euclidiana y la distancia Manhattan.

Algunos puntos a considerar para calcular las distancias entre dos puntos son las siguientes:

1. La distancia de un punto A a sí mismo es cero.

2. La distancia entre un punto A y B es la misma que la distancia del punto B al A.

De manera que para determinar la distancia euclidiana entre un conjunto de datos en dos dimensiones, utilizando la notación (a_1, a_2) para una instancia de entrenamiento y (b_1, b_2) para una instancia nueva, por el teorema de Pitágoras es:

$$\sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2}\,.$$

De forma generalizada, la distancia entre dos puntos en un espacio n-dimensional es:

$$\sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2 + \cdots + (a_n - b_n)^2}\,.$$

El algoritmo calcula las distancias entre los puntos más cercanos y determina cuál es la mínima para agregarlo a tal categoría. Sin embargo, en ocasiones los rangos de los atributos presentan una dimensionalidad totalmente distinta entre ellos, por lo que es necesario normalizar o estandarizar los datos para que se obtengan medidas coherentes a través del modelo, como se mencionó en la sección 2.3.3.

Otro tipo de aproximación para determinar las distancias es mediante la geometría propuesta por Hermann Minkowski donde la distancia entre dos puntos es la suma de las diferencias absolutas de sus coordenadas. A esta métrica también se le conoce como distancia rectilínea, distancia de ciudad o distancia Manhattan.

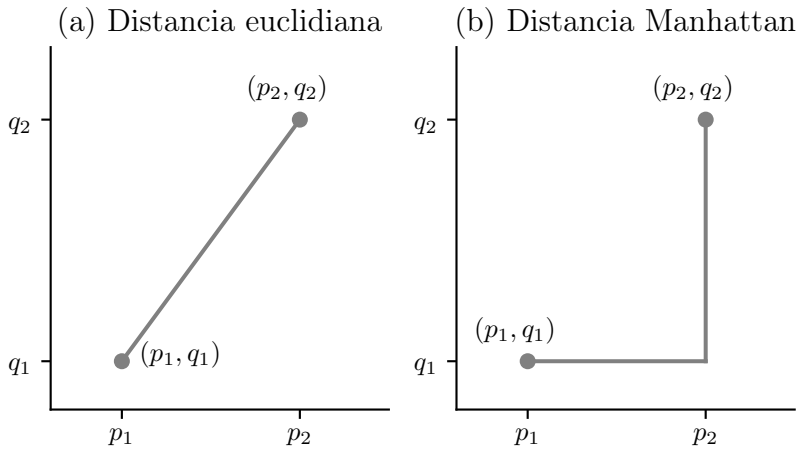
Formalmente, la distancia d_1 entre dos puntos $\mathbf{p}$ y $\mathbf{q}$ está data por:

$$d_1(\mathbf{p}, \mathbf{q}) = \|\mathbf{p} - \mathbf{q}\|_1 = \sum_{i=1}^{n} |p_i - q_i|\,, \tag{2.3}$$

donde $\mathbf{p} = (p_1, p_2, \cdots, p_n)$ y $\mathbf{q} = (q_1, q_2, \cdots, q_n)$ son vectores.

De manera gráfica es posible apreciar mejor la diferencia entre ambas métricas descritas, por lo que pueden compararse en la Figura 5.

Figura 5: Comparativo entre métricas de distancia



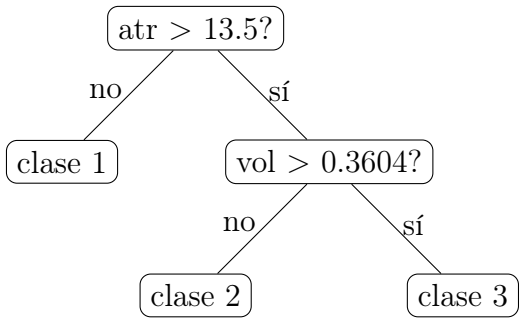
Fuente: Elaboración propia con fines ilustrativos.

2.4.3. Árboles de decisión

Un árbol de decisión es una herramienta que se utiliza para respaldar decisiones que incluyen varias alternativas, incluidos los resultados de eventos, el costo de los recursos utilizados y su uso. Las aplicaciones de los árboles de decisión se usan típicamente en la investigación y el análisis de decisiones. (Batra, 2018) De manera gráfica, un árbol de decisión puede representarse como se muestra en la Figura 6.

Existen dos tipos de árboles de decisión:

Figura 6: Árbol de decisión para una tarea de clasificación



Fuente: Elaboración propia con fines ilustrativos.

- *Árbol de clasificación.* Es usado para predecir el valor categórico al cual pertenece la instancia.
- *Árbol de regresión.* Son usados generalmente cuando la clase es un valor numérico real.

Los árboles de decisión de clasificación son conocidos como uno de los métodos más aceptados para el descubrimiento de patrones. Examinan metódicamente la información que contiene hechos importantes para revelar reglas y relaciones valiosas, y en general, se utilizan con el propósito de predicción. Si lo comparamos con otras metodologías de minería de datos, los árboles de decisión para clasificación se aplican ampliamente en varias áreas (Breiman, Friedman, Stone y Olshen, 1984).

Una técnica importante que mejora significativamente la exactitud del modelo es la de crear múltiples árboles de decisión mediante alguna de las siguientes técnicas: *bagging*, bosques aleatorios, árboles reforzados y bosques de rotación (Batra, 2018).

Un término clave en los árboles de decisión es la ganancia de información. Éste parámetro mide la reducción esperada en entropía. Dicho de otro modo, decide qué atributo va a un nodo de decisión. Para minimizar la profundidad del árbol de decisión, el atributo con la mayor reducción de entropía es la mejor opción.

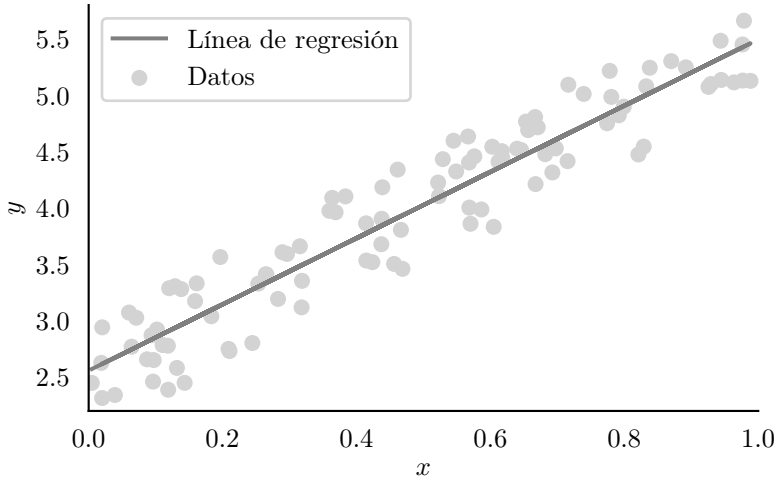
2.5. Algoritmos de regresión

En el mundo real, muchos atributos son naturalmente continuos, por ejemplo la estatura, el peso, longitud, temperatura y velocidad. Por lo que es esencial que para tener un mejor sistema de minería de datos éste pueda manejar tales atributos. En algunos casos, los algoritmos pueden ser adaptados para emplear variables continuas. En otros más, es muy difícil o imposible de realizarlo (Bramer, 2016).

2.5.1. Regresión lineal

La regresión lineal sirve para modelar la relación lineal existente entre dos variables. Una de ellas es la variable dependiente y mientras que la otra es la variable independiente x. De manera gráfica, en la Figura 7 se observan dichos componentes.

Figura 7: Representación gráfica de la regresión lineal simple



Fuente: Elaboración propia con fines ilustrativos.

El modelo de regresión lineal simple usualmente se escribe de la siguiente manera:

$$y = \beta_0 + \beta_1 x + \varepsilon\,, \tag{2.4}$$

donde y es la variable dependiente, β_0 es el intercepto de y, β_1 es el gradiente o pendiente de la línea de regresión, x es la variable independiente y ε es el error aleatorio. Es usualmente asumido que el error ε sigue una distribución normal con media $\mathrm{E}(\varepsilon) = 0$ y una varianza constante $\mathrm{Var}(\varepsilon) = \sigma^2$ (Yan, 2009).

Dados los valores de y y x es posible estimar los parámetros desconocidos de la ecuación encontrando los valores para los cuales la distancia al cuadrado de las distancias entre los valores observados y los predichos sean mínimos. A este método se le conoce como mínimos cuadrados (Field, Miles y Field, 2012).

Por otro lado, el modelo de regresión múltiple existen varios predictores, de manera que y es determinada con base en una combinación de cada variable independiente multiplicada por su respectivo coeficiente de regresión, de acuerdo con (Field et al., 2012), de la siguiente forma:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n + \varepsilon\,. \tag{2.5}$$

2.5.2. Regresión logística

En pocas palabras, la regresión logística es una regresión múltiple pero con una variable objetivo que es una clase categórica y variables independientes que son continuas o categóricas. En su forma más simple, esto significa que podemos predecir a cuál de las categorías es probable que pertenezca una observación nueva dada cierta otra información. Un ejemplo trivial es observar qué variables predicen si una persona es hombre o mujer. Podríamos medir la estatura, el índice de masa corporal, el consumo de alcohol, entre otros. Usando la regresión logística, podríamos encontrar que todas estas variables predicen el género de la persona, pero la ventaja de esta técnica es que también permite predecir si una persona, no contenida en nuestro conjunto de datos original, es probable que sea hombre o mujer (Field et al., 2012).

La regresión logística conserva muchas similitudes con los modelos descritos en la sección 2.5.1. En su forma más simple, cuando sólo existe una variable independiente x, la ecuación que determina la probabilidad de la variable objetivo y está dada por:

$$P(y) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1)}} \,. \tag{2.6}$$

De la misma forma, es posible extender la expresión anterior cuando se tienen múltiples variables independientes:

$$P(y) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n)}} \,. \tag{2.7}$$

A pesar de las similitudes entre la regresión lineal y la regresión logística, hay una muy buena razón por la cual no es posible aplicar la regresión lineal directamente a un problema donde la variable objetivo es categórica. La razón es que uno de los principales supuestos de la regresión lineal es precisamente que la relación entre las variables es lineal (Field et al., 2012).

Por lo tanto, para que la regresión lineal sea un modelo válido, los datos observados deben contener una relación lineal. Cuando la variable objetivo es categórica, este supuesto es violado (Berry, 1993).

Como se trata de un problema de clasificación, lo que se desea determinar es la probabilidad de que una observación sea agrupada en una u otra clase. De la ecuación 2.6, ordenando los términos obtenemos la expresión:

$$\frac{P(y)}{1 - P(y)} = \exp\left(\beta_0 + \beta_1 x_1\right), \tag{2.8}$$

y al obtener el logaritmo en ambos términos:

$$\log\left[\frac{P(y)}{1-P(y)}\right] = \beta_0 + \beta_1 x_1\,. \tag{2.9}$$

La expresión anterior se conoce como *logit*. Como puede observarse, la relación es lineal para x. Si los coeficientes son positivos, un incremento en la variable independiente x resultará en una mayor probabilidad de ocurrencia.

2.6. Evaluación del desempeño de los modelos

Según (Ting, 2010), se llama matriz de confusión a una tabla que contiene de manera resumida el desempeño de un algoritmo de clasificación. Es una matriz bidimensional: en una dimensión muestra la clase verdadera y en la segunda dimensión la clase que el algoritmo asigna. Para poder entender la matriz de confusión, es necesario identificar los siguientes valores:

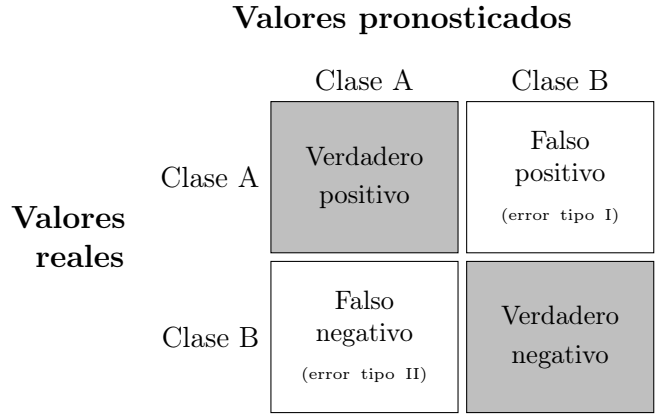
Verdadero positivo, (*True positive*, TP). El algoritmo predijo que los datos pertenecen a la clase A y es cierto.

Falso positivo, (*False positive*, FP). El algoritmo predijo que los datos pertenecen a la clase A y realmente pertenecen a la clase B. En estadística se conoce como *error tipo I*.

Falso negativo, (*False negative*, FN). El algoritmo predijo que los datos pertenecen a la clase B y realmente pertenecen a la clase A. En estadística se conoce como *error tipo II*.

Verdadero negativo, (*True negative*, TN). El algoritmo predijo que los datos pertenecen a la clase B y es cierto.

Por ser los nombres más usuales, se usarán a partir de ahora en idioma inglés. La siguiente, es una matriz de confusión de ejemplo para un algoritmo de clasificación con dos clases, A y B:



La matriz de confusión anterior es la que se genera cuando se presentan variables objetivo dicotómicas, es decir, que solo pueden tener dos posibles valores. A partir de la matriz de confusión podremos calcular los valores de *recall* (exhaustividad), *precision* (precisión) y *accuracy* (exactitud) (Powers, 2011). Dado que los términos son ampliamente utilizados en inglés, serán estos los que prevalezcan a continuación.

Recall De todas las clases positivas, cuántas fueron pronosticadas correctamente, y debe ser tan alto como sea posible.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}\,. \tag{2.10}$$

Precision De todas las clases positivas, cuántas son realmente positivas.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}\,. \tag{2.11}$$

Accuracy Respecto al total, cuántas clases fueron pronosticadas correctamente.

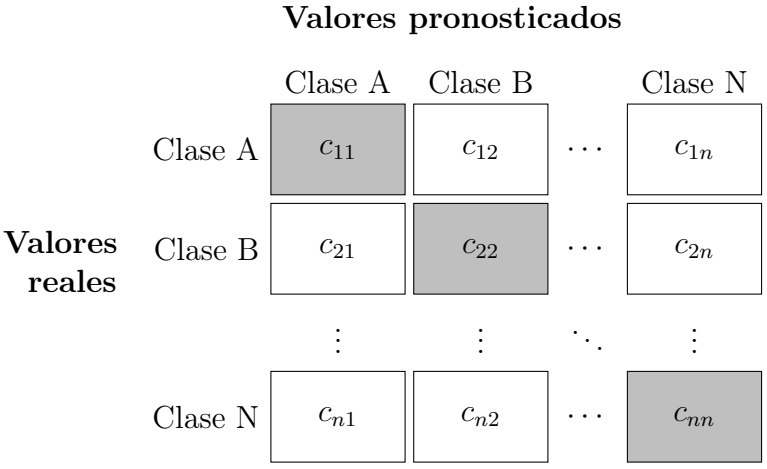
$$\text{Accuracy} = \frac{\text{TP}+\text{TN}}{\text{Total}}\,. \tag{2.12}$$

Es muy complicado comparar dos modelos cuyo valor de *precision* es bajo y el valor de *recall* es alto, y viceversa. Por lo que, para hacerlos comparables, se usa una métrica llamada *F-score*. El F-score permite medir los valores de

precision y recall al mismo tiempo, empleando medias armónicas en lugar de medias aritméticas.

$$\text{F-score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}\,. \tag{2.13}$$

Sin embargo, con frecuencia será necesario evaluar un modelo que tenga más de dos clases por clasificar, por lo que será necesario calcular una extensión de la matriz de confusión. Si el modelo puede pronosticar N cantidad de clases, la matriz de confusión multiclase tendrá dimensiones N × N. A continuación se muestra una matriz de confusión con tales características:



A diferencia de la matriz de confusión para una variable objetivo dicotómica, cuando existen más de dos posibles resultados del modelo de clasificación, es necesario determinar para cada clase i los valores de verdadero y falso positivos y verdadero y falso negativos, como se indica a continuación:

$$\text{TP}_i = c_{ii}\,, \tag{2.14}$$

$$\text{FP}_i = \sum_{j=1}^{n} c_{ji} - \text{TP}_i\,, \tag{2.15}$$

$$\text{FN}_i = \sum_{k=1}^{n} c_{ik} - \text{TP}_i\,, \tag{2.16}$$

$$\mathrm{TN}_i = \sum_{j=1}^{n}\sum_{k=1}^{n} c_{jk} - \mathrm{TP}_i - \mathrm{FP}_i - \mathrm{FN}_i\,. \tag{2.17}$$

Las expresiones anteriores son las formas generalizadas para un modelo con N clases. En la Figura 8 es fácilmente visible la ubicación de cada uno de estos elementos dentro de la matriz de confusión.

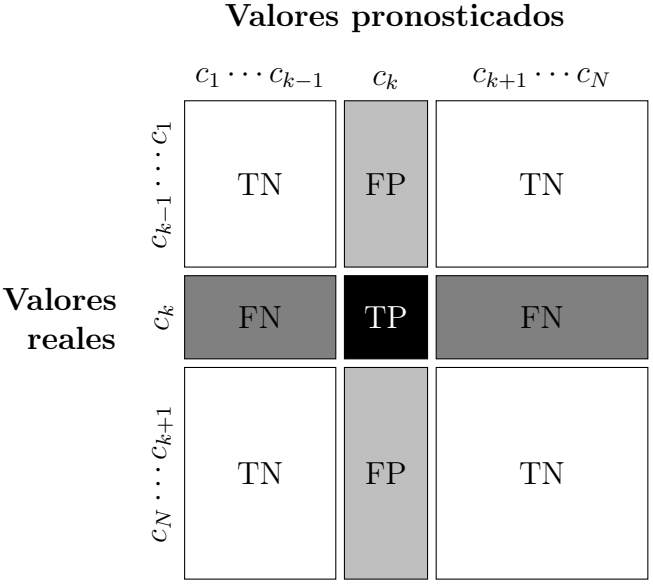
En este caso, los valores *precision*, *recall*, *accuracy* y F-score también deben ser calculados para cada una de las clases i que posea el modelo.

Para recordar fácilmente, *positive* o *negative* se refieren a la predicción realizada por el modelo; *true* o *false* se refiere a si la predicción es correcta o no.

Cuando se trata de modelos de regresión, las métricas generalmente empleadas son la raíz del error cuadrático medio (*root-mean-square error, RMSE)* y el error medio absoluto (*mean absolute error, MAE).*

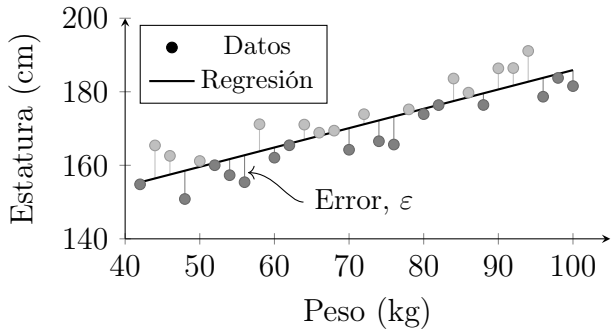
Al respecto, (Chai y Draxler, 2014) señala que aunque ambas métricas han sido empleadas para determinar el desempeño de modelos de regresión a lo largo de los años, no existe un consenso sobre cuál es el más apropiado para

Figura 8: Matriz de confusión para una clasificación multi-clase.



Fuente: (Krüger, 2016).

Figura 9: Visualización de los errores para un modelo de regresión lineal.



Fuente: Elaboración propia con fines ilustrativos.

medir el error en los modelos. Los errores, ε, son las diferencias existentes entre el valor predicho y el valor real, y pueden ser tanto positivos como negativos; una forma sencilla de visualizarlos es como se muestra en la Figura 9. En la literatura, es usual llamar a este tipo de errores por su nombre en inglés, por los que serán los que se empleen de ahora en adelante.

Para simplificar, se asume que hemos tomado una muestra de n errores de un modelo ε calculados como $(\varepsilon_i, i = 1, 2, \cdots, n)$. También se considera que los errores no presentan sesgo. Estos supuestos conllevan a que, para el caso del RMSE, se considere que los errores presentan una distribución normal; por lo tanto, al usar el RMSE, permite tener una imagen completa de la distribución de los errores. El RMSE y el MAE son determinados como:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\varepsilon_i^2}\,, \tag{2.18}$$

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n} \mid \varepsilon_i \mid \ . \tag{2.19}$$

En lo que respecta a ciencia de datos, durante el proceso de aprendizaje, lo usual es que el modelo sea evaluado con el error medio cuadrado (MSE), $\text{MSE} = \text{RMSE}^2$, mientras que durante la realización de pronósticos se use el RMSE. Se esperaría que un modelo de regresión funcione mejor cuando el RMSE sea lo más cercano a cero.

Adicionalmente, otro tipo de herramientas que se emplean con frecuencia para medir el desempeño de un modelo son las llamadas curvas ROC

(acrónimo de *Receiver Operating Characteristic*, o Característica Operativa del Receptor). Este tipo de representaciones gráficas son una representación de la sensibilidad contra la especificidad para un sistema de clasificación según se varíe el umbral de discriminación. Otra manera de verlo es como la representación de la razón de verdaderos positivos (TPR) frente a la razón de falsos negativos (FNR).

Las curvas ROC proporcionan herramientas para seleccionar los modelos posiblemente óptimos y descartar modelos subóptimos. Además, también son independientes de la distribución de las clases en la población.

Capítulo III

Procesamiento del *dataset*

3.1. Introducción

Para construir el *dataset* que se someterá al proceso de minería se requirió consultar una fuente confiable y que provea de información suficiente y precisa para realizar el análisis. Existen diversas plataformas actualmente que ofrecen información financiera actualizada, algunas de ellas requieren de una suscripción tal como el caso de Bloomberg, Reuters o Economática; mientras que otras más proveen información gratuita, aunque limitada la mayoría de las ocasiones y como ejemplo podríamos mencionar a Yahoo! Finanzas o a Investing.

Existen en internet distintas aplicaciones que permiten, a través de una API, conectarse a sus servidores para descargar información sobre distintas emisoras. Sin embargo, la mayoría de ellos se enfocan en información del mercado estadounidense, mientras que para otros mercados, la información es escasa o nula.

El *dataset* que se analizará consta de cuatro partes:

1. La información de la cotización diaria en el precio de apertura, máximo, mínimo y cierre, así como el volumen.
2. Información fundamental de la empresa
3. Datos de análisis técnico
4. Variable objetivo (*target)*

Para las primeras dos partes del *dataset*, la información se obtuvo a través de Economática, mientras que la tercera parte se calculó utilizando Python con algunas librerías auxiliares.

3.2. Información de precios y volumen

Las bolsas de valores diariamente reciben instrucciones de parte de las casas de bolsa, donde los inversionistas ofertan y demandan instrumentos financieros (mercado secundario). Básicamente, la instrucción contiene el ID de la casa de bolsa que la emite, el *ticker symbol* del instrumento que se desea operar, el tipo de operación y el precio. Este tipo de instrucciones son las que conforman la dinámica del mercado, pues mientras algunas instrucciones son de compra, muchas otras son de venta.

La manera más común de visualizar la dinámica de los precios de cualquier instrumento financiero es a través de las gráficas de velas japonesas (como la mostrada en la Figura 10), sin embargo, no son las únicas formas de visualización. Es importante tenerlo en cuenta, pues en este tipo de visualizaciones es donde el análisis técnico presenta una ventaja.

Para cada intervalo de tiempo, se presentan cuatro precios que son de interés, a saber: el precio de apertura, el precio máximo, el precio mínimo y el precio de cierre. Tales precios son representados en la gráfica de velas mediante el cuerpo de la vela (para el precio de apertura y cierre) y una sombra (para el precio máximo y mínimo). Es usual que el color de la vela esté relacionado con la dinámica del mercado para ese intervalo específico. Es decir, si el precio cerró por encima del precio de apertura, el cuerpo de la vela se presenta en color verde o blanco; si por el contrario, el precio de cierre quedó por debajo del precio de apertura, la vela se muestra con una tonalidad roja o negra. Los datos que se requieren para la construcción del *dataset* son precisamente esos cuatro precios para un intervalo de un día, por lo que emplearán los atributos descritos en la Tabla 2.

Tabla 2: Información relacionada al precio y al volumen.

ID	Atributo	Descripción
1	fecha	Indica el día correspondiente a las cotizaciones del activo o instrumento financiero.

Continuación de la Tabla 2

ID	Atributo	Descripción
2	num_operaciones	Número de instrucciones recibidas en la bolsa de valores para el activo o instrumento financiero elegido.
3	num_titulos	Número de títulos operados en el día. Matemáticamente es la suma del número de títulos de todas y cada una de las instrucciones recibidas en la bolsa de valores. Cabe recalcar que se contabiliza tanto en posiciones de compra como en posiciones de venta: por ejemplo, si 500 acciones fueron compradas, luego vendidas, después recompradas y finalmente revendidas, el número de títulos sería 2000, incluso si se tratase de las mismas 500 acciones.
4	volumen	El volumen representa la cantidad, en términos monetarios, que fue operada en un periodo específico. $$\text{Volumen} = \sum_{i=1}^{n} \text{Núm. de títulos}_i \times \text{Precio}_i \,,$$ donde: Núm. de títulos$_i$ se refiere a la cantidad de títulos negociados en la órden i. Precio$_i$ es el precio al cual se concretó la operación i.
5	cierre	Es el precio al cual cotizó el activo al final del periodo.
6	apertura	Es el precio al cual cotizó el activo al inicio del periodo.
7	minimo	Es el precio mínimo al cual cotizó el activo durante el periodo.

Continuación de la Tabla 2

ID	Atributo	Descripción
8	maximo	Es el precio máximo al cual cotizó el activo durante el periodo.
9	promedio	Es el precio promedio al que fue operado el instrumento durante el periodo. Se obtiene dividiendo el volumen sobre el número de títulos. $$\text{Promedio} = \frac{\text{Volumen}}{\text{Núm. de títulos}}.$$
10	rendimiento	En términos simples es la apreciación o depreciación del precio de un activo durante un intervalo de tiempo. $$\text{Rendimiento} = \frac{p_i - p_{i-1}}{p_i}.$$

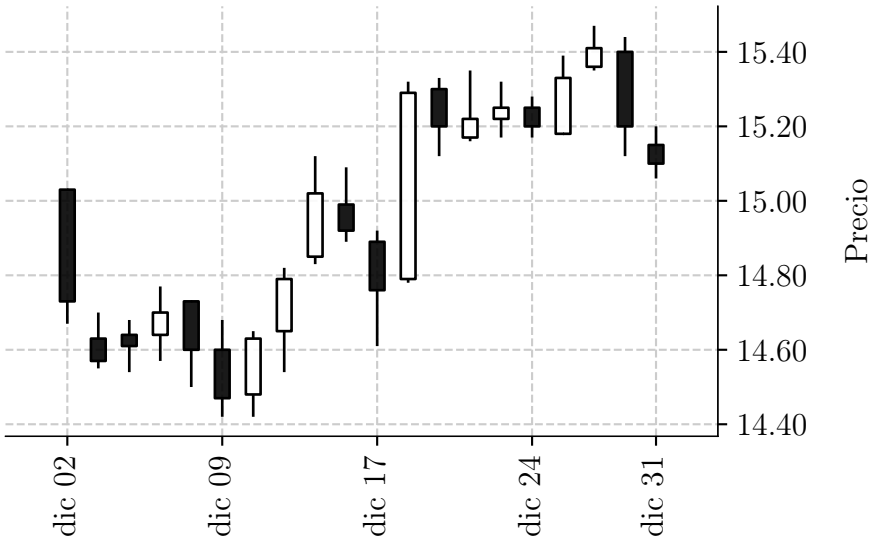
Fuente: Elaboración propia.

De los diez atributos anteriormente descritos, los nueve primeros son proporcionados como datos con información obtenida de Economática.

Uno de los procesos que demandan mayor tiempo de dedicación en cualquier proyecto de *machine learning* o de minería de datos, es la limpieza e ingeniería de atributos. La limpieza busca eliminar posibles inconsistencias, valores incorrectos o datos faltantes en el *dataset* que serían potencialmente riesgosos para la eficiencia del modelo; para ello es indispensable deshacerse de aquellos atributos cuya cantidad de elementos faltantes sea significativa, o que presenten datos inconsistentes tales como caracteres en variables numéricas, errores tipográficos, etcétera. La ingeniería de atributos consiste en construir nuevas variables a partir de los datos crudos que pudieran ser relevantes para el modelo en sí. Un ejemplo de la ingeniería de atributos podría ser la creación de variables *dummy* o variables ficticias; éstas son empleadas cuando uno de los atributos es categórico y como tal no es posible introducirlo al modelo elegido. Cada una de las categorías se convierte en un nuevo atributo el cual se señala con un cero o un uno dependiendo el valor que haya tenido la instancia.

Empleando ingeniería de atributos, se añadió el atributo número diez, rendimiento. Es relevante conocer la variación que ha presentado el precio

Figura 10: Gráfico de velas para AMX. Diciembre de 2019.



Fuente: Elaboración propia con datos de Economática.

del instrumento en un intervalo de un día. Existen dos metodologías que son aceptadas generalmente para el cálculo del rendimiento. La primera de ellas es a través del logaritmo natural del cociente entre el precio en el tiempo $t+1$ y el precio en el tiempo t. La segunda consiste en calcular el cociente de la diferencia entre el precio en el tiempo $t+1$ y el tiempo t respecto al precio en el tiempo t. La opción elegida para el *dataset* es la segunda por facilidad de cálculo y por ser el criterio más utilizado en entornos corporativos.

3.3. Información fundamental

El análisis fundamental es una metodología de análisis bursátil que busca establecer el valor teórico de una acción (precio objetivo) de una compañía y de anticipar cuál será su comportamiento futuro en el mercado bursátil, con base en el estudio detallado de toda la información económico-financiera disponible de la empresa (balance general, estado de resultados, razones financieras, etcétera), así como de la información del sector, de la coyuntura económica, entre otros. El análisis fundamental estudia cualquier informa-

ción que pueda servir para tratar de predecir el comportamiento futuro de la empresa.

Si el valor teórico de la acción de la compañía es mayor que su precio de mercado (cotización de la acción), la empresa se considera infravalorada y se recomendaría la compra, dado que se esperaría que ambos valores se acercaran. Por el contrario, si se encontrara un valor inferior al de mercado, se considera que está sobrevalorada y se recomendaría la venta.

Algunos de los indicadores más empleados son los llamados múltiplos o razones financieras, los cuales consisten en emplear valores numéricos tomados de los estados financieros de las compañías y relacionarlos para conocer información valiosa sobre la misma. Los datos obtenidos a partir de los estados financieros, estado de resultados y flujos de efectivo son empleados para realizar un análisis cuantitativo que permite determinar (entre otros) la liquidez de la compañía, el apalancamiento, el crecimiento, los márgenes de ganancia, rentabilidad, rendimiento y valuación de la empresa.

Para complementar el *dataset* que se empleará en el modelo, se añadieron los atributos descritos en la Tabla 3.

Tabla 3: Información fundamental de la emisora.

ID	Atributo	Descripción
11	pr_utilidad	El múltiplo precio a utilidades es la primera de las medidas de valor de mercado, la razón (o múltiplo) precio a utilidades o P/U se define: $$\text{P/U} = \frac{\text{Precio por acción}}{\text{Utilidad por acción}}.$$
12	pr_valor_libros	Múltiplo de valor de mercado (VM) a valor en libros (VL) es segunda medida que se menciona con frecuencia es la razón de valor de mercado a valor en libros: $$\text{VM/VL} = \frac{\text{Valor de mercado por acción}}{\text{Valor en libros por acción}}.$$

Continuación de la Tabla 3

ID	Atributo	Descripción
13	pr_ventas	El Múltiplo de precio a ventas resume la relación que guarda el valor de mercado de la acción, a los ingresos ventas que se obtuvieron: $$\text{Precio/Ventas} = \frac{\text{Precio por acción}}{\text{Ingresos netos por acción}}\,.$$
14	pr_ebitda	Múltiplo precio a EBITDA. Es la relación existente entre el precio y los ingresos antes de intereses, impuestos, depreciación y amortización. $$\text{Precio/EBITDA} = \frac{\text{Precio por acción}}{\text{EBITDA}}\,.$$
15	pr_flj_cj_libre	Múltiplo precio a flujo de caja libre (FCj). Es una razón que busca corregir algunos defectos del múltiplo precio a utilidades, empleando el efectivo generado por la compañía. El flujo de caja libre es la diferencia entre los flujos de efectivo neto y los gastos de capital. El múltiplo se determina de la siguiente forma: $$\text{Precio/FCj} = \frac{\text{Precio por acción}}{\text{Flujo de caja libre}}\,.$$
16	capit_bursatil	Capitalización bursátil. Se refiere al valor total de mercado, en términos monetarios, de las acciones de una compañía; también se le conoce como capitalización de mercado y permite conocer el tamaño de una compañía: $$\text{Capit. burs.} = \text{Núm. de acciones} \times \text{Precio}\,.$$

Continuación de la Tabla 3

ID	Atributo	Descripción
17	enterprise_value	Es una medida del valor total de una compañía, incluyendo la capitalización bursátil, las deudas a corto y largo plazo menos el efectivo y/o sus equivalentes. Se puede definir idealmente como el monto que un inversionista debería pagar si quisiera adquirir el cien por ciento de una compañía. Se resta el efectivo y sus equivalentes, porque si un inversionista adquiriera la compañía, el efectivo disponible pasaría a ser propiedad del nuevo dueño. $\text{EV} = \text{Capit. burs.} + \text{Deuda} - \text{Efectivo}\,.$
18	ev_ebitda	Valor de la empresa a EBITDA. Es la relación existente entre el valor de la empresa y el las ganancias antes de intereses, impuestos, depreciación y amortización que reporta la compañía. $\dfrac{\text{Enterprise value}}{\text{EBITDA}}\,.$
19	ev_ventas	Valor de la empresa a ventas. Se relaciona el valor de la empresa respecto a las ventas por acción. $\dfrac{\text{Valor de la empresa}}{\text{Ventas por acción}}\,.$

Fuente: Elaboración propia a partir de información de (Ross et al., 2015).

3.3.1. Información técnica

Dentro del análisis bursátil, la contraparte del análisis fundamental es el análisis técnico, el cuál se encarga del estudio de los movimientos del mercado, empleando como herramienta principal las gráficas de cotizaciones con el propósito de poder pronosticar futuras tendencias en el precio. La

expresión "movimientos del mercado" incluye las tres fuentes principales de información disponibles para el técnico: precio, volumen e interés abierto (el cual se emplea sólo cuando el instrumento analizado son futuros u opciones) (Murphy, 2016).

La filosofía o el fundamento lógico detrás del análisis técnico sugiere tres premisas elementales:

1. Los movimientos del mercado lo descuentan todo
2. Los precios se mueven por tendencias
3. La historia se repite

El análisis técnico considera que el precio de un instrumento financiero refleja todos los posibles elementos ambientales que podrían afectarlo, tales como situaciones políticas, psicológicas, medio ambientales u otras, teniendo por conclusión que lo único que haría falta sería realizar un análisis del movimiento de los precios. Lo que el análisis técnico está diciendo en realidad es que los movimientos del precio deberían reflejar los cambios de la oferta y la demanda (Murphy, 2016). Si la demanda supera a la oferta (o si la oferta disminuye), los precios deberían subir. Si la oferta supera a la demanda (o si la demanda disminuye), los precios deberían bajar.

Los gráficos en sí mismos no hacen que los mercados suban o bajen, sino que simplemente reflejan la psicología alcista o bajista del mercado.

El único propósito de tener una representación gráfica de los movimientos de los precios es identificar tendencias que están en las primeras etapas de su desarrollo, con el fin de que las transacciones vayan en la dirección de dichas tendencias.

Una gran parte del análisis técnico y del estudio de los movimientos del mercado tiene que ver con el estudio de la psicología humana. Los gráficos son imágenes que revelan la psicología alcista o bajista del mercado, y dado que estos patrones han funcionado bien en el pasado, se asume que con cierta probabilidad seguirán funcionando bien en el futuro.

Con la finalidad de incorporar información generada a través del análisis fundamental, se calcularon los atributos descritos en la Tabla 4 para el *dataset*.

Tabla 4: Información técnica de la emisora.

ID	Atributo	Descripción
20 21 22 23	sma10 sma20 sma50 sma200	Las medias móviles simples (*simple moving average*, SMA) son promedios aritméticos calculados para una ventana de k días, calculados a través de la suma de los precios de cierre de los últimos k días divididos entre k. Las medias móviles simples para un periodo pequeño son más sensibles a los cambios en el precio del subyacente, mientras que las de periodos más largos son más lentas para reaccionar. Matemáticamente se determinan de la siguiente forma: $$\mathrm{SMA}(A,n) = \frac{A_1 + A_2 + \ldots + A_n}{n} \ ,$$ donde: A_n = Precio de un activo A en el tiempo n. n = Número total de periodos.
24 25 26	bb_upperband bb_middleband bb_lowerband	Las Bandas de Bollinger (*Bollinger Bands*, BB) son una herramienta de análisis técnico definidas como un conjunto de dos líneas con valor de dos desviaciones estándar tanto positivas como negativas alejadas de una media móvil simple. Como la desviación estándar es una medida de la volatilidad, cuando ésta se incrementa el canal se hace más ancho, mientras que cuando disminuye las bandas se contraen. La forma de calcularlas es a través de las siguientes expresiones: $$\mathrm{bolU} = \mathrm{SMA}(\mathrm{TP},n) + m \cdot \sigma(\mathrm{TP},n) \ ,$$ $$\mathrm{bolD} = \mathrm{SMA}(\mathrm{TP},n) - m \cdot \sigma(\mathrm{TP},n) \ ,$$

Continuación de la Tabla 4

ID	Atributo	Descripción
		donde:
		$\begin{aligned} \text{bolU} &= \text{Banda de Bollinger superior.} \\ \text{bolD} &= \text{Banda de Bollinger inferior.} \\ \text{TP} &= \tfrac{1}{3}(\text{Máximo} + \text{Mínimo} + \text{Cierre})\,. \\ n &= \text{Días de suavizamiento (usado 20).} \\ m &= \text{Núm. desv. estándar (usado 2).} \\ \sigma(\text{TP}, n) &= \text{Desv. est. del TP en } n \text{ periodos.} \end{aligned}$
27	atr	El rango verdadero medio (*average true range*, ATR) es un indicador de análisis técnico que mide la volatilidad del mercado. El rango de una acción es la diferencia entre el precio alto y bajo en cualquier día dado. Revela información acerca de lo volátil que es un activo. Los rangos grandes indican alta volatilidad y los rangos pequeños indican baja volatilidad. El ATR está defindo por: $$\text{TR}_i = \text{máx} \begin{cases} H_i - L_i, \\ \text{Abs}(H_i - C_p), \\ \text{Abs}(L_i - C_p) \end{cases}$$ $$\text{ATR} = \tfrac{1}{n} \sum_{i=1}^{n} \text{TR}_i$$ donde: $\begin{aligned} H_i &= \text{Precio máximo en el periodo } i. \\ L_i &= \text{Precio mínimo en el periodo } i. \\ C_p &= \text{Precio de cierre del periodo previo.} \\ \text{TR}_i &= \text{Rango verdadero en el periodo } i \\ n &= \text{Ventana de tiempo empleada.} \end{aligned}$ En la práctica es usual emplear una ventana de 14 días, mismos que se usaron para este atributo en el dataset.

Continuación de la Tabla 4

ID	Atributo	Descripción
28 29 30	stoch_slow_k stoch_slow_d smi	El índice de momento estocástico (*stochastic momentum index*, SMI) es una versión más refinada del oscilador estocástico, para lo cual emplea un rango más ámplio de valores y tiene una mayor sensibilidad a los precios de cierre. Calcula la distancia del precio de cierre actual respecto a la media entre el máximo y mínimo del día. El SMI tiene un rango de valores de entre +100 y -100. Cuando el precio de cierre actual es mayor que el promedio entre el máximo y el mínimo, el valor resultante es positivo. Cuando el precio de cierre actual es menor que la media entre el máximo y mínimo, el valor del SMI es negativo. El SMI es determinado como se indica: $$\begin{aligned} \mathrm{hl}_1 &= \mathrm{máx}\,(H_n) - \mathrm{mín}\,(H_n) \ , \\ \mathrm{cm}_1 &= C_i - \tfrac{1}{2}\mathrm{hl}_1 \,, \\ \mathrm{cm}_2 &= \mathrm{EMA}(\mathrm{EMA}(\mathrm{cm}_1, 25), 2) \ , \\ \mathrm{hl}_2 &= \mathrm{EMA}(\mathrm{EMA}(\mathrm{hl}_1, 25), 2) \ , \\ \mathrm{SMI} &= 100 \times \tfrac{1}{2}\tfrac{\mathrm{cm}_2}{\mathrm{hl}_2} \ , \end{aligned}$$ donde: $$\begin{aligned} C_i &= \text{Precio de cierre en el periodo } i. \\ \mathrm{máx}\,(H_n) &= \text{Precio máximo en } n \text{ periodos.} \\ \mathrm{mín}\,(H_n) &= \text{Precio mínimo en } n \text{ periodos.} \end{aligned}$$ Los valores más comunes son emplear una ventana de 13 días y una media móvil exponencial de 9 días para la construcción de una línea de señal.

Continuación de la Tabla 4

ID	Atributo	Descripción
31 32	aroondown aroonup	El indicador Aroon es una herramienta de análisis técnico que es usada para identificar cambios de tendencia en el precio de un activo, así como la fuerza de esa tendencia. Básicamente, el indicador mide el tiempo transcurrido entre los máximos y el tiempo transcurrido entre los mínimos en un lapso determinado. Tendencias alcistas fuertes encontrarán nuevos máximos en poco tiempo, mientras que tendencias bajistas fuertes encontrarán nuevos mínimos rápidamente. El indicador consiste en dos líneas, una que mide la fuerza de las tendencias alcistas y otra que mide la calidad de las tendencias bajistas. Ambas líneas son determinadas mediante las expresiones: $$\text{Aroon Up} = \frac{n - \text{máx}(n)}{n} \times 100 \ ,$$ $$\text{Aroon Down} = \frac{n - \text{mín}(n)}{n} \times 100 \ ,$$ donde: $\text{máx}(n)$ = Número de periodos desde que ocurrió el máximo en n periodos. $\text{mín}(n)$ = Número de periodos desde que ocurrió el mínimo en n periodos. En la práctica es usual utilizar diferentes periodos en función del horizonte de inversión. Para la creación de este atributo, se eligió una ventana de 14 días.

Continuación de la Tabla 4

ID	Atributo	Descripción
33	clv	La ubicación del valor de cierre (*close location value*, CLV) es una métrica empleada en análisis técnico que permite evaluar el precio de cierre actual respecto al máximo y mínimo presentados durante el intervalo. Los valores del CLV varían entre +1.00 y -1.00, donde un valor más alto significa que el precio de cierre está más cerca del máximo del periodo, y un valor más cercano al límite inferior significa que el precio de cierre está próximo al mínimo. $$\text{CLV}_i = \frac{(C_i - L_i) - (H_i - C_i)}{H_i - L_i}$$ donde: C_i = Precio de cierre en el periodo i. H_i = Precio máximo en el periodo i. L_i = Precio mínimo en el periodo i.
34	chaikin_vol	La volatilidad de Chaikin es nombrada así en torno a su creador Marc Chaikin. Este análisis mide la acumulación (distribución) del oscilador de promedios móviles de convergencia-divergencia (*moving average convergence divergence*, MACD). Para calcular el oscilador de Chaikin, se sustrae una media móvil exponencial de 10 días de la línea de acumulación-distribución a una media móvil exponencial de 3 días de la línea de acumulación-distribución. El resultado es el momentum predecido por los osciladores alrededor de la línea de acumulación-distribución. $$M_i = \text{CLV}_i \times \text{volumen del periodo } i \ ,$$ $$\text{ADL}_i = M_{i-1} + M_i \ ,$$ $$\text{CO} = \text{EMA}(\text{ADL}, 3) - \text{EMA}(\text{ADL}, 10) \ ,$$

Continuación de la Tabla 4

ID	Atributo	Descripción
		donde: M_i = Monto medio operado en el periodo i. ADL = Línea de acumulación-distribución.
35	cmo	El oscilador de momentum de Chande fue nombrado así en honor a su creador, Tushar Chande. Este indicador mide el momentum día a día sin realizar suavizamiento alguno, resultando en una mayor cantidad de señales de entrada (salida). El rango de este oscilador es de entre +100 y -100. La fórmula calcula la diferencia entre la suma de las ganancias recientes y la suma de las recientes pérdidas y luego divide el resultado entre la suma de todo el movimiento de precios en el mismo periodo. $\text{CMO} = \frac{\text{s máx} - \text{s mín}}{\text{s máx} + \text{s mín}} \times 100$, donde: s máx = Suma de los precios máx. en n periodos. s mín = Suma de los precios mín. en n periodos. Para la construcción del dataset, se empleó una ventana de 14 días, valor típicamente empleado en la práctica.
36	ema10	Las medias móviles exponenciales (*exponential moving average*, EMA) son una variación de las medias móviles simples, con la diferencia de que otorgan una mayor ponderación a las observaciones más recientes. Una media móvil exponencial reacciona más rápido que una media móvil simple la cual aplica el mismo peso a todos los datos. $\text{Df} = \frac{\lambda}{1+n}$

Continuación de la Tabla 4

ID	Atributo	Descripción
		$\mathrm{EMA}_i(A,n) = A_i \cdot \mathrm{Df} + \mathrm{EMA}_{i-1}(A,n) \cdot (1 - \mathrm{Df})$ donde: λ = Factor de suavizamiento. n = Periodos a calcular. Df = *Decay factor*. Como notación, las indicaremos con las siglas EMA y entre paréntesis, el elemento que se calculará seguido del número de periodos considerado.
37	emv14	El Indicador de facilidad de movimiento (*ease of movement value*, EMV) es un estudio técnico que busca cuantificar una mezcla entre el momentum y la información de volumen en un solo valor. Pretende emplear este valor para distinguir si los precios tienen potencial de subir o bajar con poca resistencia al movimiento direccional. Teóricamente, si los precios se mueven con facilidad, continuarán haciéndolo por un periodo de tiempo de manera que el activo puede ser negociado con beneficios para el inversionista. $\text{Dist. movida}_i = \frac{1}{2}\left[(H_i + L_i) - (H_{i-1} - L_{i-1})\right]$, $\text{Box ratio}_i = \frac{\text{Volumen}_i}{\text{Escala}} \times \frac{1}{H_i - L_i}$, $\text{EMV}(1) = \frac{\text{Distancia movida}}{\text{Box ratio}}$, donde: H_i = Precio máximo en el periodo i. L_i = Precio mínimo en el periodo i. H_{i-1} = Precio máximo en el periodo $i-1$. L_{i-1} = Precio mínimo en el periodo $i-1$. Volumen_i = Volumen operado en el periodo i.

Continuación de la Tabla 4

ID	Atributo	Descripción
		La escala puede variar en un rango que va desde 1,000 y hasta 1,000,000,000 en función del volumen promedio diario de la acción. Mientras ésta sea negociada con volúmenes mayores, convendrá emplear una escala más amplia para que el indicador EMV se mantenga con valores por debajo de dos dígitos. En la práctica es usual emplear una ventana de 14 unidades de tiempo, mientras que para el dataset se usó una escala de 100,000,000.
38	volat	En términos simples, la volatilidad es el rango de la variación en en el precio que experimenta un activo en un determinado tiempo; si el precio se mantiene relativamente estable, entonces el instrumento tiene baja volatilidad y viceversa. Un activo cuyo precio presenta una alta volatilidad alcanza nuevos máximos y nuevos mínimos rápidamente, presenta un comportamiento errático y sube o baja de precio de manera acelerada. Existen diversos modelos de volatilidad, cada uno de ellos con ventajas y desventajas. Entre los modelos más avanzados están aquellos que modelan de forma dinámica la volatilidad, tales como los GARCH o los MGARCH. Algunos otros modelos involucran volatilidad estocástica. (Engle y Patton, 2000) Para el alcance planteado en capítulos anteriores, se empleará el modelo de volatilidad propuesto por (Garman y Klass, 1980), determinado de la siguiente forma: $\sigma = \sqrt{\frac{N}{n}\sum\left[\frac{1}{2}\left(\log\frac{H_i}{L_i}\right)^2 - (2\log 2 - 1)\left(\log\frac{C_i}{O_i}\right)^2\right]}$,

Continuación de la Tabla 4

ID	Atributo	Descripción
		donde: N = Número de días operativos al año. n = Número de periodos analizados. H_i = Precio máximo en el periodo i. L_i = Precio mínimo en el periodo i. C_i = Precio de cierre en el periodo i. O_i = Precio de apertura en el periodo i. Para la construcción del dataset, se consideró que existen en promedio 250 días al año que son hábiles bursátilmente hablando, mientras que la ventana de tiempo se mantuvo en 10 días.
39	macd	El promedio móvil de convergencia divergencia (*moving average convergence divergence*, MACD) es un indicador de momentum de continuidad de tendencias, que muestra la relación entre dos medias móviles del precio de un instrumento. El MACD es calculado sustrayendo la media móvil exponencial a 28 periodos (lenta o de largo plazo) de la media móvil exponencial a 14 periodos (rápida o de corto plazo) del precio de cierre. El MACD otorga avisos de entrar en posiciones largas cuando éste cruza por encima de la línea de señal; por el contrario alerta sobre posibles entradas cortas cuando cae por debajo de la línea de señal. Es posible determinar el MACD para cualquier otro periodo de tiempo, pero se sugiere que la relación de tiempo entre la señal rápida y lenta sea de 1:2. Para el dataset, fue calculado mediante la expresión: $\text{MACD}_i = \text{EMA}\left(C_i, 14\right) - \text{EMA}\left(C_i, 28\right)$,

Continuación de la Tabla 4

ID	Atributo	Descripción
		donde: C_i = Precio de cierre en el periodo i.
40	macd_signal	La señal MACD es un complemento consistente en una media móvil de 9 periodos de la línea MACD. Su función principal es la de mostrar señales de entrada o salida. $\text{MACD signal} = \text{EMA}\,(\text{MACD}, 9)$.
41	macd_histogram	El histograma MACD es la representación gráfica de la diferencia existente entre la línea MACD y la línea de señal. $\text{MACD histogram} = \text{MACD} - \text{MACD signal}$.
42	mfi	El índice de flujo de dinero (*money flow index*, MFI) es un oscilador que usa precio y volumen para identificar zonas de sobrecompra y zonas de sobreventa en un activo o instrumento financiero. También es utilizado frecuentemente para identifica divergencias, las cuales nos advierten de posibles cambios en el precio. El oscilador tiene un rango que va de 0 a 100. A diferencia de otros osciladores, el MFI incorpora el volumen a la ecuación, por esta razón, algunos analistas le conocen también como un RSI ponderado por volumen. Aunque es posible determinarlo para cualquier otro periodo, en la práctica es usual que sea a 14 unidades de tiempo. En primera instancia, es necesario calcular el flujo de efectivo como $\text{Flujo de efectivo}_i = \text{TP}_i \times \text{Volumen}_i$,

Continuación de la Tabla 4

ID	Atributo	Descripción
		donde: $\text{TP}_i = \frac{1}{3}(\text{Máximo} + \text{Mínimo} + \text{Cierre})$ $\text{Volumen}_i = \text{Volumen operado en el periodo } i.$ Posteriormente, por separado se determina cuáles precios típicos (TP) fueron mayores al del periodo anterior, así mismo cuáles fueron inferiores. Se realiza la suma de los mayores (positivos) y los inferiores (negativos) para una ventana de 14 periodos. $\text{MFR} = \frac{\text{Flujo de efectivo positivo a 14 periodos}}{\text{Flujo de efectivo negativo a 14 periodos}}$, y posteriormente se determina $\text{Money flow index} = 100 - \frac{100}{1-\text{MFR}}$.
43	rsi	El índice relativo de soporte (*relative support index*, RSI) es un indicador de momentum que mide la magnitud de los cambios recientes en el precio para evaluar zonas de sobrecompra o sobreventa de un activo o instrumento financiero. El RSI se muestra como un oscilador y puede presentar valores en un rango que va de 0 a 100. Generalmente, cuando el precio se encuentra en una tendencia lateral se usa el límite de 70 para considerar que hay sobrecompra y 30 si es que existe sobreventa. Cuando existe una tendencia alcista, los parámetros usualmente considerados son 80 y 40 respectivamente. Para una tendencia bajista, dichos parámetros se consideran en 60 y 20 respectivamente. El RSI es calculado en dos partes, la primera de ellas se determina como: $\gamma = \frac{\text{Ganancia promedio}}{\text{Pérdida promedio}}$,

Continuación de la Tabla 4

ID	Atributo	Descripción
		$\text{RSI}_{\text{Paso uno}} = 100 - \frac{100}{1+\gamma}$.
		Una vez que se han completado 14 periodos de datos, es posible calcular la segunda parte de la fórmula, que está dada por:
		$\text{PLR} = \frac{\text{Ganancia promedio previa}\times 13+\text{Ganancia actual}}{\text{Pérdida promedio previa}\times 13+\text{Pérdida actual}}$,
		$\text{RSI}_{\text{Paso dos}} = 100 - \left[\frac{100}{1+\text{PLR}}\right]$.
		El valor de RSI usualmente empleado es el de 14 días (indicado RSI14), mismo que se incorpora en el dataset.
44	sar	El sistema de paro y reversa parabólico (*stop and reversal system*, SAR) es un indicador técnico empleado para determinar la dirección del precio de un activo, así como advertir de posibles cambios en la dirección del precio.
		Generalmente se representa con líneas punteadas sobre el gráfico de precios: cuando el SAR se encuentra por arriba de las barras de precio indica que los vendedores están al mando, mientras que si el SAR está por debajo de los precios, indica una señal bajista. Es determinado de la forma siguiente:
		$\text{SAR}_{i+1} = \text{SAR}_i + \alpha(\text{EP} - \text{SAR})$,
		donde:
		SAR_i = SAR del periodo i.
		α = Factor de aceleración, típicamente 0.02.
		EP representa el precio más alto en una tendencia alcista o el precio más bajo en una tendencia bajista.

Continuación de la Tabla 4

ID	Atributo	Descripción
45	sd	La desviación estándar (*standard deviation*, σ) es la medida estadística de la volatilidad, mide qué tan dispersos son los precios respecto a su promedio. Si el precio cotiza en un rango pequeño, la desviación estándar resultará en un valor pequeño, mientras que si los precios varían considerablemente, la desviación estándar presentará un valor significativo. $$\sigma\left(C,n\right)=\sqrt{\sum_{i=1}^{n}\frac{\left(C_i-\overline{C}\right)^2}{n-1}}\ ,$$ donde: C_i = Precio de cierre en el periodo i. $\overline{C}$ = Precio de cierre promedio durante n periodos. Para conformar este atributo en el dataset se considera un periodo de 10 días.

Fuente: Elaboración propia a partir de información de diversos autores listados en el apartado de Referencias.

3.4. Variable objetivo (target)

Podemos ver las cotizaciones del activo como una serie de tiempo financiera, de manera que podemos identificar cuatro componentes principales en dicha serie:

1. Tendencia
2. Estacionalidad
3. Ciclicidad
4. Ruido

En particular, es relevante la identificación de tendencias para la aplicación del modelo, ya que de ello dependerán los resultados que se obtengan.

Figura 11: Identificación de tendencias para AMX/L. Jun 19-Ene 20

Fuente: Elaboración propia con datos de Economática.

En una serie de tiempo financiera es simple identificar tres tipos de movimientos del precio: movimientos bajistas, movimientos alcistas o movimientos laterales (vea Figura 11). Los movimientos bajistas se dan cuando los nuevos máximos de un activo o instrumento financiero, son cada vez menores respecto a los de periodos anteriores. Los movimientos alcistas suceden cuando los nuevos mínimos de un activo o instrumento financiero, superan a los de periodos anteriores. Los movimientos laterales se dan cuando no existen nuevos máximos y tampoco nuevos mínimos en los nuevos periodos.

Como inversionista, siempre se busca tener un rendimiento positivo $r\,\%$ al realizar una inversión, asumiendo un cierto nivel de riesgo, que sea al menos superior a los costos de transacción e impuestos. Para obtener un rendimiento positivo, no es necesario que los precios suban, sino que depende de la posición que tome el inversionista:

Posición larga. El inversionista compra un activo o instrumento financiero a un determinado precio, esperando que en el futuro la cotización de dicho activo sea superior y que al venderlo pueda obtener un beneficio.

Posición corta. El inversionista realiza la venta de un activo o instrumento financiero a un precio determinado, esperando que en el futuro la co-

tización de dicho activo sea inferior y que al comprarlo pueda obtener un beneficio.

En este contexto, se desea que el modelo pueda predecir si tales rendimientos potencialmente podrían ser alcanzados durante los siguientes n días, sin desestimar la posibilidad de que por el contrario los rendimientos no sean alcanzados.

Sería muy complicado intentar estimar el precio en el tiempo $t+n$, además de que tal precio podría significar una variación menor que $r\,\%$ por lo que no sería relevante para el inversionista.

Lo que realmente se pretende es tener nociones del comportamiento de los precios en forma dinámica, siguiendo su tendencia natural, lo cual solo es posible si analizamos las variaciones que han tenido a lo largo de varios días. Por ejemplo, el precio de cierre en el tiempo $t + n$ podría ser significativamente inferior a $r\,\%$ pero podría haber sido precedido por una racha alcista en donde se alcanzó (pudiendo incluso rebasar) durante varios días el rendimiento mínimo esperado por el inversionista.

La construcción de la variable objetivo (en inglés, target) consiste en calcular un valor que nos de indicios sobre la tendencia para los siguientes n días. El valor de esta variable debe estar relacionado con el valor de $r\,\%$ y también en función de una ventana de tiempo para calcularlo. La idea es que si la variable objetivo arroja resultados positivos, el inversionista pueda tomar la decisión de entrar en una posición larga, mientras que valores negativos significarían tomar una posición corta y valores cercanos a cero se interpretarían como una señal de mantener (o no hacer nada).

Para determinar este indicador, se toma la idea original de la volatilidad, midiendo cuando exceda un cierto límite $r_t\,\%$. En primer lugar, basaremos los cálculos en el precio típico, que es un promedio entre el precio de cierre, el precio máximo y el mínimo. Posteriormente, también se calculará el rendimiento entre el precio de cierre y los siguientes n días, de manera que al realizar la suma de los valores absolutos obtengamos el rendimiento acumulado.

Es importante señalar el hecho de que calcularemos rendimientos sobre precios futuros para que el modelo aprenda sobre la relación existente entre los atributos dependientes y la variable objetivo, y que en la fase de evaluación y producción pueda obtener resultados confiables.

Una vez que se tienen los rendimientos esperados para los siguientes n días, es necesario codificar la variable objetivo teniendo un parámetro de re-

ferencia, que indicará las distintas señales que otorga el mercado. La variable objetivo (target) se generará mediante:

$$\text{target}_i = \begin{cases} \text{“comprar”}, & \text{para } r_i \geq r_t \,. \\ \text{“mantener”}, & \text{para } -r_t \leq r_i < r_t \,. \\ \text{“vender”}, & \text{para } -r_t < r_i \,. \end{cases}$$

La instrucción “comprar” (*buy*) indicará que es momento de tomar una posición larga ante una probable tendencia al alza, o bien, que es momento de comprar el activo para cerrar una posición corta. La instrucción “mantener” (*hold*) anticipa un movimiento lateral de precio, en donde el inversionista dependiendo si está o no dentro de una posición, deberá determinar si la abandona o la mantiene. Finalmente, la señal de “vender” (*sell*) proporcionará señales de que es probable que se esté formando una tendencia a la baja, por lo que sería una opción realizar una operación en corto o bien cerrar una posición larga.

El supuesto principal para la construcción de este indicador y poder ser considerado como la variable objetivo es que es posible pronosticar comportamientos futuros en los mercados financieros observando y analizando información histórica (uno de los fundamentos del análisis técnico). Dicho de otro modo, si en el pasado existió un comportamiento particular, seguido por una consecuencia de forma repetitiva, se esperaría que en el futuro cuando se presente nuevamente el comportamiento, podamos anticipar las consecuencias.

Capítulo IV

Construcción del modelo y predicciones

Con la información descrita en la sección 3.4, debemos plantear la construcción de un modelo de clasificación utilizando aprendizaje supervisado. Es de clasificación porque existen tres posibles resultados (*outputs*) posibles, y de aprendizaje supervisado porque se conoce cuál es el comportamiento de la variable objetivo en función de las variables dependientes.

En el mercado existe un sinnúmero de *software* para la construcción de modelos de minería de datos; algunos de ellos son gratuitos aunque con ciertas limitaciones en cuanto al soporte técnico, por lo que para ambientes productivos podrían ser poco recomendables. El software comercial, brinda a su vez una plataforma robusta para la construcción y puesta en marcha del modelo, pero presenta la principal desventaja de ser una caja negra.

Un modelo de caja negra, o más específicamente un modelo financiero de caja negra, es un término general utilizado para describir un programa informático diseñado para transformar diversos datos en estrategias de inversión útiles (Kenton, 2019).

En ciencia, informática e ingeniería, una caja negra es un dispositivo, sistema u objeto que se puede ver en términos de sus entradas y salidas, sin ningún conocimiento de su funcionamiento interno. Su implementación es opaca o "negra".

Casi cualquier cosa podría denominarse una caja negra: un transistor, un algoritmo o incluso el cerebro humano. Lo opuesto a una caja negra es un sistema donde los componentes internos o la lógica están disponibles para inspección, lo que se conoce comúnmente como una caja blanca.

Un ejemplo de software que emplea cajas negras podría ser *RapidMiner* (https://rapidminer.com) o *SAS* (https://www.sas.com). Aplicaciones con modelos de cajas grises son por ejemplo *Orange* (https://orange.biolab.si). Y finalmente, el software (o lenguajes de programación, en este caso) cuyos modelos son cajas blancas, por excelencia son *Python* (https://www.python.org) y *R* (https://www.r-project.org).

La ventaja de trabajar con cajas blancas es que en todo momento puede revisarse el algoritmo, para tener certeza sobre lo que hace y cómo funciona. Las cajas negras, por su parte, tienen una serie de ventajas como que están optimizadas para realizar determinadas tareas en menos tiempo, pero que no son accesibles para el usuario. Las cajas grises son una combinación de las otras dos.

Para la construcción y evaluación del modelo, se usará Python, siendo éste un lenguaje que ha tenido un desarrollo importante en el área de ciencia de datos, y que cuenta con la ventaja adicional de poder monitorear en todo momento el algoritmo.

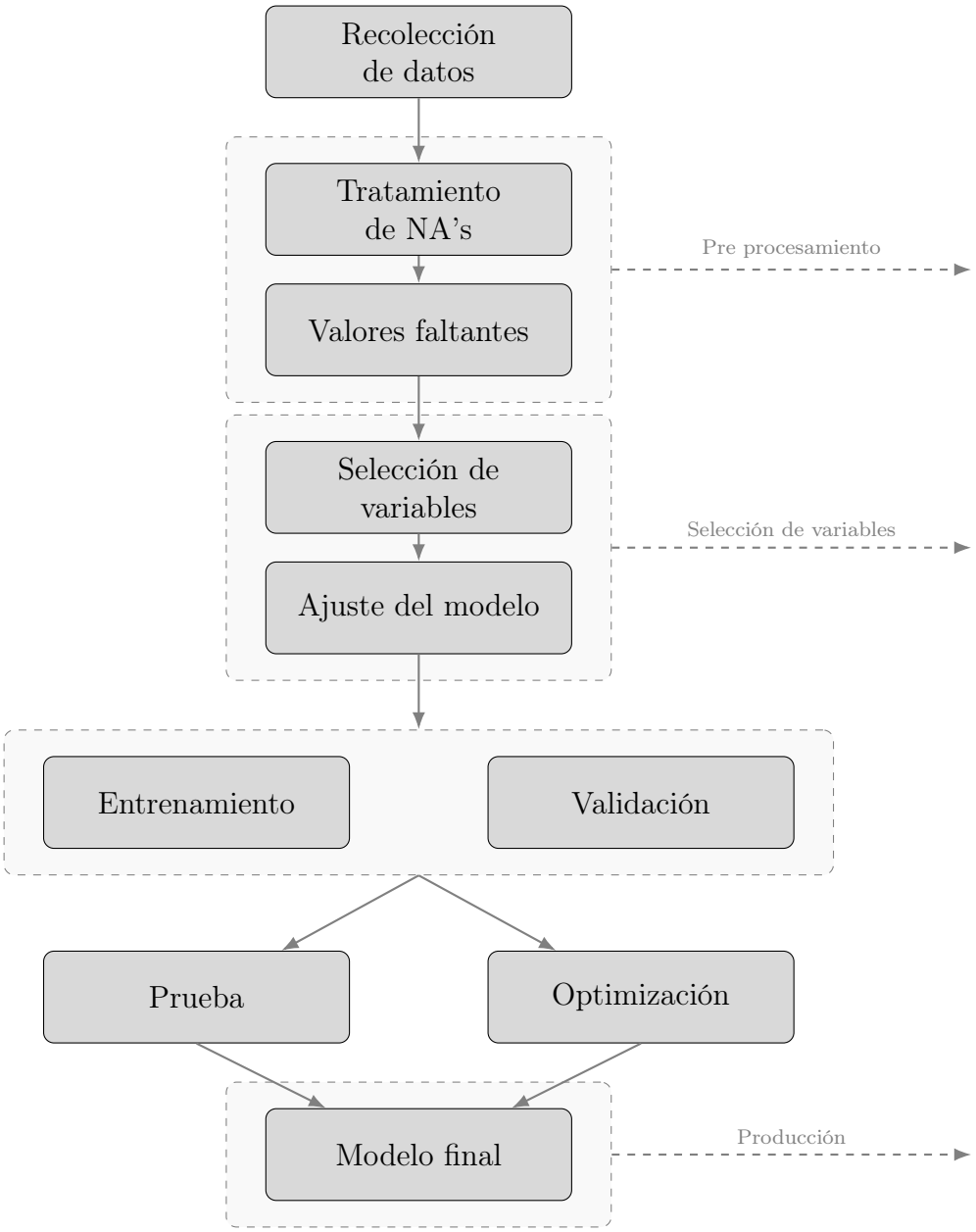
A manera esquemática, en la Figura 12 se muestra en un diagrama de flujo el proceso que se seguirá para construir y desarrollar el modelo.

4.1. Caso de estudio: América Móvil, S.A.B. de C.V.

Planteando un escenario local, se buscó realizar el modelo para una empresa nacional que cotice en alguna bolsa de valores en México. La elección de esta compañía no fue arbitraria, se tomó en cuenta la solidez de la empresa, la información disponible en el mercado, y principalmente el nivel de liquidez que tiene en el mercado, siendo una de las de mayor bursatilidad actualmente. La empresa elegida fue América Móvil, cuya clave de pizarra (o *ticker*) en ambas bolsas es AMX; en particular, el análisis se realizará para la serie L.

América Móvil es la empresa líder en servicios integrados de telecomunicaciones en Latinoamérica. Excluyendo China y la India, es la más grande a nivel mundial en términos de suscriptores móviles. Con el despliegue de su plataforma de comunicaciones de clase mundial le permite ofrecer a sus clientes un portafolio de servicios de valor agregado y soluciones de comunicación mejoradas en 25 países de América Latina, los Estados Unidos y

Figura 12: Diagrama de flujo para la construcción y desarrollo del modelo.



Fuente: Elaboración propia.

Europa Central y del Este (América Móvil, 2014).

De acuerdo con información de primer reporte trimestral del 2020, América Móvil cerró con 77.2 millones de suscriptores en México, después de agregar 215 mil suscriptores móviles de postpago y 80 mil clientes prepagos en el primer trimestre del año. Su principal empresa de telefonía móvil, Telcel, posee en México una cuota de mercado de más del 62.6 %, de acuerdo con cifras de (The Ciu, 2019), y mantiene una tasa anual de crecimiento constante del 2.2 %.

A pesar de contar con poco más del 60 % de participación en el mercado, (The Ciu, 2019) estima que a nivel de ingresos, Telcel obtiene 70.9 % de todos los operadores móviles existentes en México.

Con las cifras reveladas, es evidente que se trata de una empresa plenamente consolidada, con la bursatilidad suficiente para poder operar en el mercado de manera fluida, sin problemas de deslizamiento. [1]

4.2. Exploración y visualización

Uno de los primeros pasos a realizar en toda tarea de ciencia de datos es la de visualizar la información con la que se cuenta, de manera que se pueda tener un esbozo del comportamiento de los datos. Habrá algunas relaciones que no sean evidentes a simple vista, pero eso es tarea para el modelo que se entrenará posteriormente. Por ejemplo, en la Figura 11, se visualizó el comportamiento de la acción de América móvil para el periodo de junio de 2019 a enero de 2020, aunque por simplicidad, se omitirán los indicadores y osciladores descritos en la Tabla 4.

Para la variable objetivo, los datos están mayormente concentrados en la señal de "mantener", lo cual tiene mucho sentido, asumiendo que los rendimientos tienen una distribución normal con media igual a cero. Como se describió en la sección 3.4, es necesario establecer un límite para identificar señales de compra o venta: para el caso de América móvil se estableció en 2 %. Si el límite se establece demasiado alto, se obtendrán pocas señales de

[1] El deslizamiento se refiere a la diferencia entre el precio esperado de un trade y el precio al cual es ejecutado. El deslizamiento puede ocurrir en cualquier momento, pero es más común en periodos de alta volatilidad. También puede ocurrir cuando debido a la baja bursatilidad, o al ingreso de una orden suficientemente grande, no existe el volumen suficiente para poder operar al precio elegido para mantener el *spread* entre el precio *bid* y *ask*.

compra/venta, mientras que un límite demasiado bajo resultará en la obtención de señales que podrían no ser confiables, o cuyo rendimiento esperado no exceda siquiera a los costos de transacción.

Para el *dataset* con que se realizará el entrenamiento del modelo, se cuenta con el 35.06 % de instancias clasificadas como *hold*, 30.48 % clasificadas como *buy* y el restante 34.46 % como *sell*.

Los mapas de calor (*heathmaps*) son sumamente útiles durante las primeras etapas de visualización de datos. En ellas es posible apreciar de forma gráfica la correlación existente entre las variables independientes; en la Figura 13 se puede observar el mapa de calor para las variables del *dataset*. Es evidente que en aquellos indicadores que se calculan directamente con el precio de cierre, existe una alta correlación. Ya desde esta gráfica sería posible intuir sobre las variables que deberían ser seleccionadas para el entrenamiento del modelo, pero más adelante se establecerán distintos mecanismos para hacerlo.

Existen otras herramientas visuales tales como las gráficas tipo *pairplot*, en las cuales es posible observar la distribución de los datos de cada una de las variables con respecto a otras y al mismo tiempo agruparlo por cada una de las clases que dispone el *dataset*. A nivel de información, es una gráfica que provee al analista de una gran ventaja, pues puede tomar decisiones sobre ciertos modelos dependiendo de la distribución que siguen (algunos modelos asumen normalidad en los datos para poder funcionar de manera eficiente).

Más adelante en la sección 4.4 se muestra una gráfica de este tipo con las variables que fueron seleccionadas para realizar el entrenamiento del modelo.

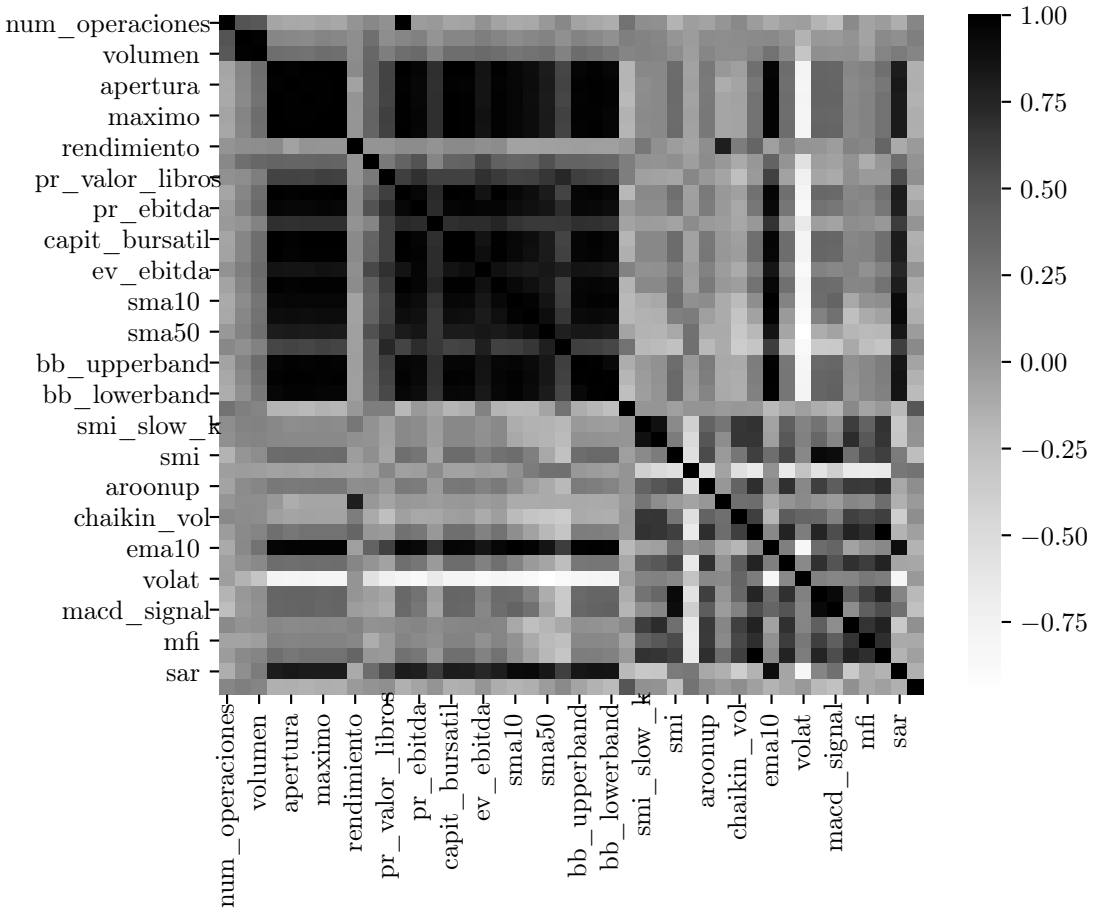
4.3. Tratamiento y limpieza

Una vez que el *dataset* quedó construido como se describió en el Capítulo III, es necesario realizar el procesamiento inicial del mismo. Como los datos fueron obtenidos desde una fuente externa, es necesario verificar que no existan datos irregulares (valores `NA`, que pueden ser caracteres importados de manera erronea debido a la codificación del archivo), duplicados o incorrectamente capturados.

En Python es fácilmente verificable con la instrucción `pd.describe()`, que devuelve el número total de instancias, la media, desviación estándar, valores mínimo y máximo, así como los tres cuartiles principales.

Con la finalidad de verificar los valores faltantes, es posible visualizarlos

Figura 13: Mapa de calor con la correlación entre todas las variables del *dataset*.



Fuente: Elaboración propia.

a través de la instrucción `pd.isna()`. Debido a que al calcular varios de los indicadores descritos en la Tabla 4 se emplea una ventana móvil, muchos de los atributos poseen datos faltantes para las primeras instancias. Existen dos formas para poder solventar esta situación: la primera de ellas es eliminar aquellos valores `NA` mediante la instrucción `pd.dropna(inplace=True)`, sin embargo al aplicar esta instrucción, se pierden todas las instancias que contienen al menos un valor faltante; la segunda opción, por otro lado, es realizar los cálculos con suficientes periodos de información anteriores a la fecha de

inicio que se desea calcular, de esta manera los valores que se pierden debido a los cálculos no quedan dentro del periodo a analizar. Para la construcción del modelo, se elegirá la segunda opción para mantener la integridad de las fechas establecidas previamente.

El *dataset* presenta una escala que no es congruente entre los atributos. Basta con calcular los rangos para los atributos `apertura`, `cierre`, `maximo` y `minimo`, y compararlo con el del `volumen`. Para resolver esta situación, tal como se describe en la sección 2.3.3, es necesario recurrir a la transformación de los datos.

Para que cada uno de los atributos posea una media igual a cero, y varianza igual a uno, se aplicó el procedimiento de estandarización. La elección sobre si se deben estandarizar o normalizar los datos corresponde al analista, y se debe tener en cuenta el tipo de modelos y métricas de evaluación que se emplearán, ya que algunas de ellas no aceptan valores negativos (como cuando se usa chi-2 para evaluar el desempeño de un modelo).

Como se describió en la sección 3.4, la variable objetivo cuenta con tres posibles resultados, los cuales corresponden a una cadena de texto. Sin embargo, los distintos modelos no tienen la capacidad de interpretar el significado de cadenas de texto, y realizar el entrenamiento, sin llevar a cabo una transformación previa, otorgará un desempeño poco satisfactorio de los mismos.

Por lo tanto, es necesario codificar los posibles resultados de la variable objetivo: los distintos lenguajes de programación o software que se emplean en el manejo de datos poseen funciones que permiten realizar esta tarea. La codificación no es una simple traducción de los valores categóricos a datos numéricos, sino que debe establecerse como tal que se trata de clases y no de un número.

Es decir, si las clases "comprar", "mantener" y "vender" se transforman simplemente en `1`, `2` y `3`, se corre el riesgo de que el modelo asuma que 3 es mejor que 2, y que a su vez 2 es mejor que 1, aún cuando no exista tal relación entre los datos.

Python incluye estas herramientas dentro de su librería Scikit-Learn, las cuales pueden ser ejecutadas mediante el comando `LabelEncoder()` (Buitinck et al., 2013).

4.4. Selección de atributos

Un modelo de minería de datos se alimenta de los datos que el analista le provee, realiza los cálculos y asociaciones necesarias y construye un conjunto de reglas (o algoritmo) para poder realizar un pronóstico. Ponerlo de esta manera parece simple, sin embargo, es necesario hacer notar que a medida que el número de variables de un *dataset* se incrementa, también lo hace la complejidad del modelo (en forma exponencial, de hecho), los tiempos de cómputo y principalmente la facilidad para que una persona pueda interpretarlo.

Los dos principales problemas de un modelo de minería de datos son el sobreaprendizaje y la dimensionalidad.

Para resolver este inconveniente, sería recomendable que el modelo se alimentara únicamente de las variables suficientes que permitan obtener los mejores resultados. Tal como se desarrolló en la sección 2.3.5, existen distintas metodologías que permiten al analista de datos realizar la selección de las variables más representativas.

4.4.1. Selección basada en filtros

Un enfoque simple para realizar la selección de variables es utilizar herramientas como el mapa de calor mostrado en la Figura 13 y remover aquellas variables cuya correlación es mayor a un límite determinado. Es de interés que las variables no estén altamente correlacionadas en forma alguna (positiva o negativamente), ya que incorporarlas al modelo no aportará beneficios, puesto que tales variables presentan un comportamiento conjunto, y por su parte aportarán complejidad y dificultad para interpretarse.

Estableciendo un límite (en valor absoluto) de 0.90, las variables que deberían ser eliminadas (sin un órden en particular) del *dataset* son: `ev_ventas`, `macd`, `bb_middleband`, `bb_lowerband`, `sma10`, `bb_upperband`, `sma20`, `sma50`, `ev_ebitda`, `minimo`, `pr_ventas`, `ema10`, `pr_ebitda`, `apertura`, `maximo`, `capit_bursatil`, `sar`, `rsi`, `volat`, `volumen`, `macd_signal`, `enterprise_value` y `promedio`.

4.4.2. Selección basada en métodos envolventes

La idea principal detrás de los métodos envolventes es la de buscar un subconjunto de atributos, los cuales provean el mejor desempeño para un clasificador específico.

Como se describió en la sección 2.3.5, los métodos de selección de variables hacia adelante y eliminación de variables hacia atrás, son ejemplos de métodos envolventes. Éstos métodos emplean métricas de desempeño tales como el p-valor, el criterio de información de Akaike, error medio cuadrado, F-score, etcétera.

Los dos principales métodos con estas características son la eliminación hacia atrás y la selección hacia adelante.

La eliminación hacia atrás consiste en integrar todas las variables a un modelo y después ir excluyendo una tras otra. Aquella variable que tenga la menor correlación parcial (es decir, la que sea menos influyente) con la variable dependiente será la primera en ser considerada para su eliminación. Si satisface el criterio de eliminación, se retira del *dataset*. Se vuelve a evaluar con las variables restantes y se elimina aquella con la correlación parcial más pequeña. El procedimiento termina cuando ya no quedan en la ecuación variables que satisfagan el criterio de eliminación.

La selección de variables hacia adelante es un procedimiento en el que las variables se introducen secuencialmente en el modelo. La primera variable que se considerará incorporar será la que tenga mayor correlación, positiva o negativa, con la variable dependiente (la de mayor influencia). Dicha variable se introducirá en el modelo solo si cumple el criterio de entrada. Si se introduce la primera variable, a continuación se considerará la variable independiente cuya correlación parcial sea la mayor y que no esté previamente incluida en el modelo. El procedimiento termina cuando ya no quedan variables que cumplan el criterio de entrada.

La ventaja de este tipo de algoritmos de selección de variables es que permiten al analista evaluar cuál sería el desempeño del modelo en función de las variables que se seleccionen.

Para el *dataset* con que se cuenta, se entrenarán los modelos enlistados a continuación. Algunos de ellos presentan opciones para utilizarse como modelos de regresión o de clasificación, pero dada la tarea que se persigue, se emplearán los segundos. El nombre se muestra en español e inglés.

(a) Regresión logística (*Logistic regression*).

(b) Naive-Bayes (*Gaussian NB*).

(c) Árboles de decisión (*Decision tree*).

(d) k-vecinos más cercanos (k-*neighbors*).

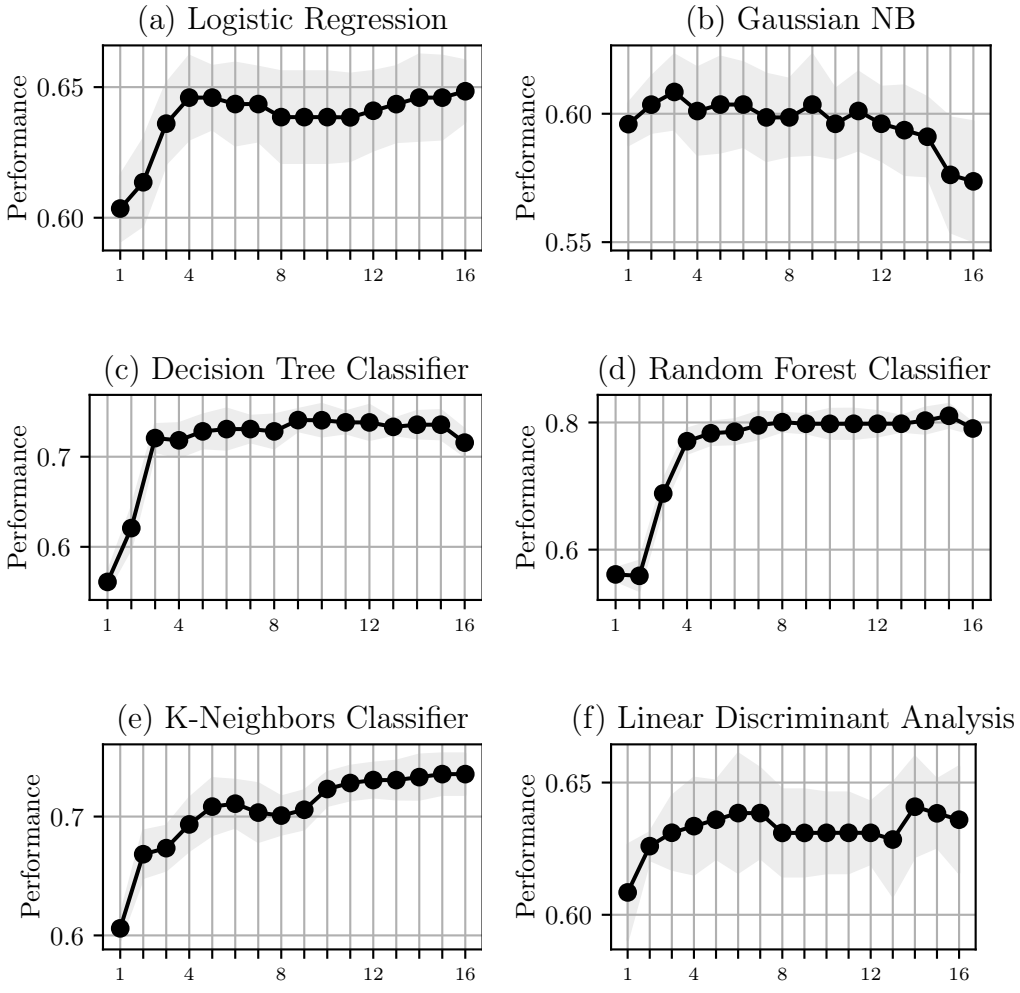
(e) Análisis discriminante lineal (*Linear discriminant analysis*).

(f) Perceptrón multicapa (*Multi-layer perceptron*).

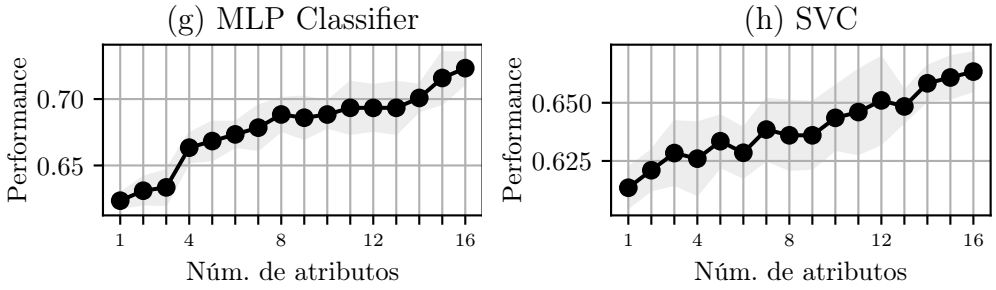
(g) Máquinas de vectores de soporte (*Support vector machines*).

En las figuras incluidas en la Tabla 5, se muestran los resultados de emplear el algoritmo de selección hacia adelante, empleando cada uno de los modelos descritos.

Tabla 5: Selección de variables con selección hacia adelante.



Continuación de la Tabla 5



Fuente: Elaboración propia.

Como se puede observar, en general todos los modelos tienen un mejor desempeño a medida que se incrementa el número de variables. Sólo el modelo Gaussian NB presenta un decaimiento significativo en el desempeño del modelo a medida que se agregan variables.

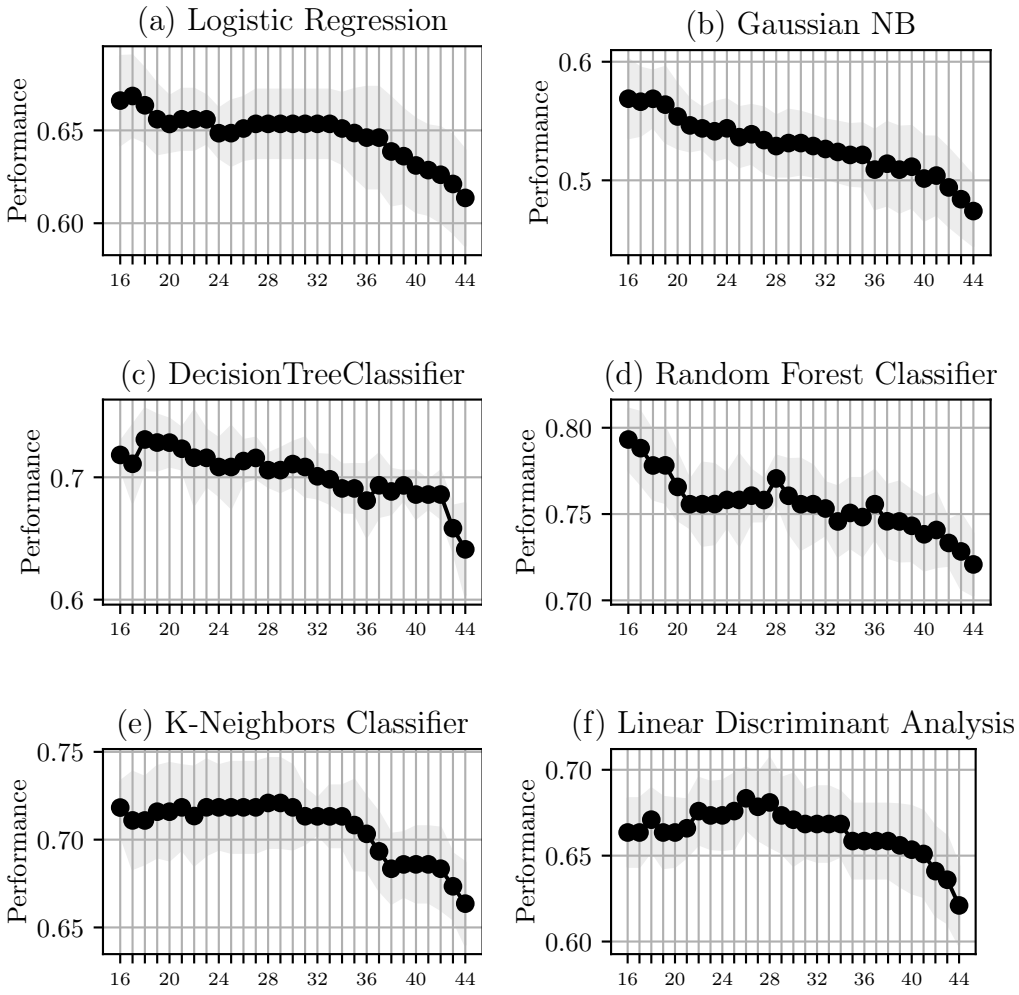
La métrica empleada para definir el desempeño del modelo en las gráficas de la Tabla 5, fue el *accuracy*. Evidentemente, cualquier otra métrica de clasificación puede ser utilizada. Así mismo, se empleó una validación cruzada de 5 *k-folds* para ajustar el modelo y obtener el score de cada variable ingresada.

A pesar de que el *dataset* contiene 44 variables explicativas, se puede tomar como criterio para medir el comportamiento del modelo la utilización de un número de variables igual a la raíz cuadrada del número de instancias, por lo que para el *dataset* analizado se midió el desempeño con las primeras 16 variables seleccionadas. Más adelante en la sección 4.5 se evaluará el desempeño del modelo con los hiperparámetros óptimos empleando las variables que previamente fueron seleccionadas como las mejores para cada uno de los modelos.

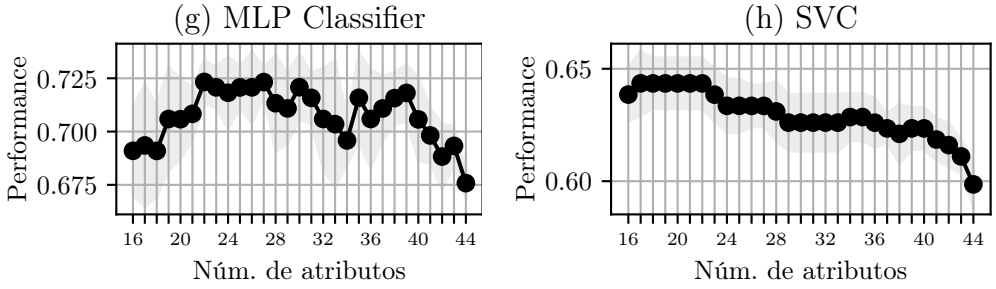
Hasta este punto, con la combinación de variables elegidas, aparentemente el modelo que presenta un mejor comportamiento es el clasificador con bosques aleatorios. Sin embargo, habrá que explorar más la información y tener certeza de que no se esté presentando un caso de sobreaprendizaje con el modelo.

Manteniendo todos los parámetros constantes, se procede a realizar la eliminación de variables hacia atrás. Los resultados de la selección de variables son los mostrados en la Tabla 6.

Tabla 6: Selección de variables con eliminación hacia atrás.



Continuación de la Tabla 6



Fuente: Elaboración propia.

Como este método elimina una a una las variables, se grafica iniciando desde las variables seleccionadas (16) hasta el total de variables disponibles (44). Es notable que en la medida que el número de variables crece, también el desempeño del modelo presenta un decaimiento.

Es notable que el modelo de bosques aleatorios también presenta el mejor desempeño utilizando la metodología de eliminación de variables hacia atrás. El desempeño del modelo aparentemente converge en las 16 variables a niveles cercanos a 0.80 de *accuracy*. Con excepción de *linear discriminant analysis* y MLP *classifier*, los demás modelos presentan el comportamiento esperado, un mejor desempeño empleando únicamente la cantidad de variables suficientes (cercana a 16 variables).

Llama la atención el comportamiento del modelo MLP *classifier*, ya que tiene un desempeño errático a medida que se modifica el número de variables. Sin embargo, si se compara con su contraparte de selección de variables por selección hacia adelante, su comportamiento es más estable. Aparentemente se alcanza un mejor desempeño con el método de eliminación hacia atrás cuando se tiene del orden de 20 a 24 variables en el modelo.

Tabla 7: Atributos elegidos mediante selección hacia adelante.

Modelo	**Score**	**Atributos seleccionados**
Logistic regression	0.6484	`apertura, maximo, pr_ventas, enterprise_-value, sma20, sma50, sma200, bb_upperband, bb_middleband, bb_lowerband, smi, clv, ema10, emv14, sar, sd`

Continuación de la Tabla 7

Modelo	Score	Atributos seleccionados
Gaussian NB	0.6085	promedio, smi_slow_d, clv
Decision Tree	0.7407	cierre, sma20, sma200, bb_upperband, smi, aroondown, ema10, volat, sd
Random Forest	0.8105	pr_valor_libros, pr_fl_cj_libre, capit_bursatil, ev_ebitda, sma20, sma200, atr, aroondown, aroonup, chaikin_vol, ema10, volat, macd_signal, macd_histogram, sar
k-neighbors	0.7357	apertura, minimo, pr_utilidad, pr_valor_libros, pr_ventas, pr_flj_cj_libre, sma10, sma20, sma50, sma200, b_upperband, bb_middleband, ema10, volat, macd, sd
LDA	0.6409	num_titulos, volumen, apertura, pr_utilidad, pr_flj_cj_libre, ev_ventas, sma200, smi, cmo, volat, macd, rsi, sar, sd
MLP	0.7233	promedio, pr_utilidad, pr_valor_libros, capit_bursatil, enterprise_value, sma10, sma50, bb_upperband, atr, smi, aroondown, ema10, volat, macd_signal, mfi, sd
SVC	0.6634	pr_utilidad, pr_valor_libros, pr_ventas, ev_ebitda, sma50, sma200, bb_upperband, bb_middleband, atr, smi, aroondown, aroonup, chaikin_vol, macd_histogram, sar, sd

Fuente: Elaboración propia.

Tabla 8: Selección de variables con eliminación hacia atrás.

Modelo	Score	Atributos seleccionados
Logistic regression	0.6685	num_titulos, cierre, apertura, minimo, maximo, rendimiento, pr_valor_libros, pr_flj_cj_libre, enterprise_value, sma200, atr, aroondown, aroonup, volat, macd, mfi, sd

Continuación de la Tabla 8

Modelo	Score	Atributos seleccionados
Gaussian NB	0.5689	num_operaciones, cierre, apertura, maximo, promedio, pr_valor_libros, enterprise_value, ev_ventas, bb_upperband, smi_slow_d, chaikin_vol, ema10, emv14, macd, macd_signal, macd_histogram, mfi, sd
Decision Tree	0.7309	apertura, minimo, promedio, pr_utilidad, pr_valor_libros, pr_ventas, pr_flj_cj_libre, ev_ebitda, bb_upperband, bb_middleband, smi_slow_d, smi, volat, macd_signal, macd_histogram, mfi, sar, sd
Random Forest	0.7932	pr_valor_libros, pr_flj_cj_libre, capit_bursatil, ev_ebitda, ev_ventas, sma10, sma50, sma200, bb_middleband, aroondown, chaikin_vol, ema10, emv14, macd, macd_signal, sar
k-neighbors	0.7209	num_operaciones, cierre, apertura, minimo, maximo, promedio, pr_valor_libros, pr_ventas, pr_ebitda, pr_flj_cj_libre, capit_bursatil, ev_ebitda, ev_ventas, sma10, sma20, sma200, bb_upperband, bb_middleband, bb_lowerband, atr, smi, aroondown, aroonup, chaikin_vol, cmo, macd, macd_signal, mfi, sd
LDA	0.6834	num_operaciones, volumen, apertura, minimo, rendimiento, pr_utilidad, pr_valor_libros, pr_ventas, pr_ebitda, pr_flj_cj_libre, capit_bursatil, enterprise_value, ev_ebitda, sma20, bb_upperband, atr, smi_slow_k, smi_slow_d, aroondown, cmo, ema10, emv14, volat, macd_signal, sar, sd
MLP	0.7232	cierre, minimo, pr_utilidad, pr_valor_libros, pr_ventas, pr_ebitda, capit_bursatil, ev_ebitda, sma10, sma20, sma200, bb_lowerband, atr, smi, aroondown, chaikin_vol, emv14, macd, macd_signal, macd_histogram, rsi, sar

Continuación de la Tabla 8

Modelo	Score	Atributos seleccionados
SVC	0.6435	`cierre, minimo, maximo, pr_utilidad, pr_-valor_libros, pr_ebitda, enterprise_value, ev_ebitda, sma20, smi_slow_k, smi, aroondown, aroonup, chaikin_vol, emv14, sar, sd`

Fuente: Elaboración propia.

De forma tabular se muestran los *scores* de cada modelo, con sus correspondientes atributos seleccionados en las Tablas 7 y 8. Es notable que al aplicar esta metodología, empiezan a aparecer en prácticamente todos los modelos, los indicadores fundamentales que se calcularon en la Tabla 3; por su parte, en los métodos de selección basados en filtros, los atributos que tienen una mayor participación son los de información técnica, quedando la información fundamental ligeramente relegada.

Realizar la selección de variables con esta metodología tiene una deficiencia: los modelos no han sido "calibrados". Dado que el ajuste del modelo se realizó con los parámetros por *default*, es posible que los resultados no sean los óptimos, sin embargo sirve como punto de partida para realizar la calibración con los atributos seleccionados y posteriormente, si así se desea, de forma iterativa encontrar la mejor combinación.

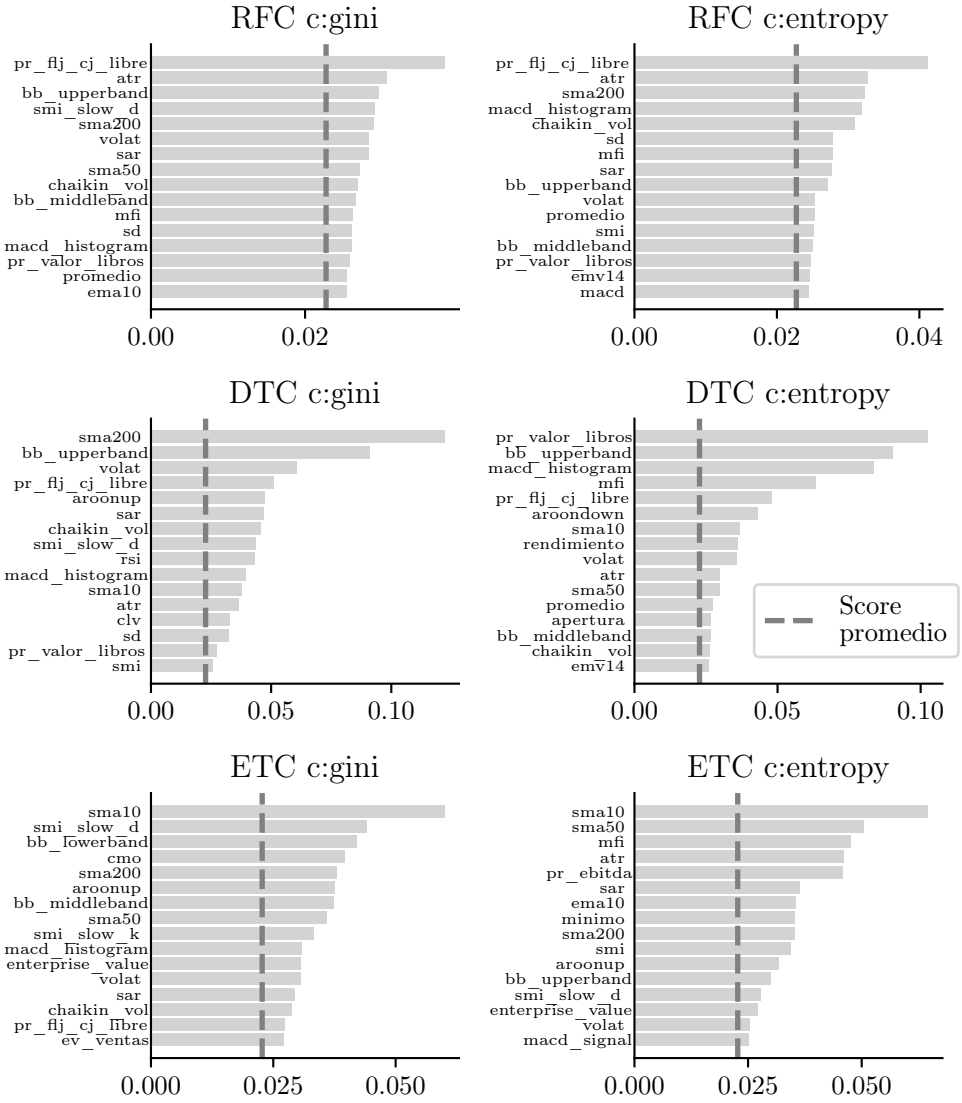
4.4.3. Selección basada en la importancia del atributo

Una alternativa más para seleccionar las variables que deberían pertenecer al *dataset* es la selección mediante la "importancia" del atributo empleando modelos basados en árboles de decisión.

Como se explicó en la sección 2.4.3, para que un árbol de decisión cree un nuevo nodo, la variable de la que se desprenderá debe cumplir con un criterio preestablecido.

La importancia de las variables son esencialmente la media de la mejora de los árboles individuales en el criterio de división producido por cada variable. En otras palabras, es cuánto mejoró la puntuación (conocida como "impureza" en la notación de árboles de decisión) al dividir el árbol utilizando la variable específica. Estas importancias pueden ser usadas para otorgar un puntaje a cada variable y seleccionar un subconjunto de ellas.

Figura 14: Selección de atributos con importancia de las variables.



Fuente: Elaboración propia.

En la figura 14 se muestran los resultados obtenidos aplicando esta metodología para los siguientes modelos, considerando para cada uno de ellos dos criterios distintos, el de Gini y el de entropía (o ganancia de información):

(a) Random Forest Classifier

(b) Decision Tree Classifier

(c) Extra Tree Classifier

En el eje de las ordenadas se muestran las primeras 16 variables que obtuvieron una mayor importancia (ordenadas de mayor a menor) de acuerdo con el modelo y el criterio aplicado, mientras que con una línea punteada se muestra el score promedio para las 44 variables.

El score promedio para cada modelo puede servir como criterio de selección de las variables; con los resultados obtenidos se observa que el criterio previamente explicado sobre que el número de atributos es aproximadamente igual a la raíz cuadrada del número de observaciones tiene una alta confiabilidad.

4.5. Optimización de hiperparámetros

Los hiperparámetros son parámetros ajustables que se eligen para entrenar un modelo y que rigen el propio proceso de entrenamiento. Por ejemplo, para entrenar una red neuronal profunda, debe decidir el número de capas ocultas en la red y la cantidad de nodos de cada capa antes de entrenar al modelo. Estos valores suelen permanecer constantes durante el proceso de entrenamiento. (Microsoft Corporation, 2020).

Durante el proceso de aprendizaje se evalúan y mejoran numerosos parámetros. Por el contrario, un hiperparámetro es una variable cuyo valor se establece antes del entrenamiento, por lo que ni se evalúa ni se corrige.

Aunque existen diversos métodos para realizar la optimización de hiperparámetros, nos centraremos en dos: muestreo de cuadrícula (*Grid search*) y muestreo aleatorio (*Random search*).

El muestreo de cuadrícula se puede usar si el espacio de hiperparámetros se puede definir como una opción entre valores discretos y si tiene el poder computacional suficiente para buscar de forma exhaustiva todos los valores en el espacio de búsqueda definido. En el muestreo aleatorio, los valores de

los hiperparámetros se seleccionan aleatoriamente del espacio de búsqueda definido.

En cuanto al tiempo de cálculo, el método de muestreo de cuadricula toma más tiempo que el de muestreo aleatorio, ya que mientras en el primero es el analista quien define específicamente las combinaciones posibles que deberán probarse, en el método aleatorio elige hiperparámetros al azar dentro del espacio de búsqueda.

Cuando ya se han determinado los hiperparámetros óptimos, el analista de datos puede elegir ajustar nuevamente los modelos (usando los resultados de la optimización) para elegir nuevamente el conjunto de variables que deberán ser integradas al modelo.

Por ejemplo, la siguiente combinación de hiperparámetros fue propuesta para realizar una búsqueda exhaustiva para el modelo de clasificación con bosques aleatorios, empleando el *dataset* desarrollado en el capítulo III,

```
grid_params = [{'n_estimators':[100,250,500,750,1000],
'criterion':['gini', 'entropy'],'max_features':[None],
'max_leaf_nodes':[None]}]
```

y, una vez realizada la búsqueda, se obtuvo la siguiente combinación óptima con un *accuracy* de 0.683168:

```
{'criterion': 'entropy', 'max_features': None, 'max_leaf_nodes':
None, 'n_estimators': 500}.
```

Para conocer más acerca del funcionamiento de cada uno de los hiperparámetros, se sugiere revisar la documentación de Scikit-Learn.

4.6. Entrenamiento del modelo y obtención de predicciones

Poniendo todos los elementos anteriormente descritos en un sólo flujo de trabajo que permita determinar cuál es el mejor subconjunto de variables, los hiperparámetros óptimos de cada modelo con la combinación de variables seleccionadas, y determinando qué modelo obtiene mayores puntajes de *accuracy* en conjuntos de entrenamiento y prueba, se obtiene el mejor modelo posible.

El flujo de trabajo se divide por etapas, realizando los siguientes procedimientos en cada una de ellas:

1. Entrenamiento y evaluación de los modelos con los datos sin procesa-

miento.

2. Entrenamiento y evaluación de los modelos con los datos después de estandarizarlos.

3. Se realiza la selección de variables, empleando un modelo embebido sin optimización (ver Figura 15(a)), posteriormente se entrenan y evalúan los modelos con los datos sin procesamiento.

4. Con los resultados de la selección de variables, se entrenan y evalúan los modelos con los datos después de estandarizarlos.

5. Mediante la optimización de hiperparámetros, se realiza una nueva selección de variables (ver Figura 15(b)), posteriormente se entrenan y evalúan los modelos con los datos sin procesamiento.

6. Con los resultados de la segunda selección de variables, se entrenan y evalúan los modelos con los datos después de estandarizarlos.

7. Se realiza la optimización de hiperparámetros de cada modelo, y con los resultados obtenidos se entrenan y evalúan utilizando los datos después de estandarizarlos.

En la Figura 15 se muestran los resultados de la selección de variables antes y después de la optimización de los hiperparámetros. Después de realizar la selección con los mejores hiperparámetros, se reafirman las variables que deben ser seleccionadas para el modelo. Con una línea punteada se observa la importancia promedio de todas las variables, y es notable que aproximadamente las mejores 16 se encuentran por encima de dicho promedio.

De manera resumida, se concentra en la tabla 9 los resultados de *accuracy* obtenidos en cada etapa para el conjunto de validación *validation set*. Es posible apreciar que desde las primeros cálculos, aún sin realizar optimización de hiperparámetros y selección de variables, el modelo que presentó un mejor desempeño es el de bosques aleatorios.

Es de hacer notar que el desempeño de los modelos, a medida que se fue avanzando en las distintas etapas de procesamiento fue mejorando, salvo en algunas excepciones.

Tres de los ocho modelos probados obtuvieron su máximo *accuracy* en las etapas 6 y 7; y aunque se presentan puntajes más altos en la etapa 4, aún

Tabla 9: Resultados por modelo. (*Accuracy*)

Modelo	Etapa 1	Etapa 2	Etapa 3	Etapa 4	Etapa 5	Etapa 6	Etapa 7	Mejor etapa
LR	0.356436	0.306931	0.336634	0.524752	0.336634	0.445545	0.475248	4
NB	0.267327	0.306931	0.346535	0.346535	0.346535	0.346535	0.346535	7
SVM	0.267327	0.306931	0.39604	0.594059	0.39604	0.485149	0.514851	4
KNN	0.554455	0.306931	0.316832	0.60396	0.316832	0.613861	0.693069	7
DT	0.623762	0.247525	0.564356	0.623762	0.564356	0.574257	0.613861	4
RF	0.663366	0.306931	0.683168	0.663366	0.683168	0.683168	0.673267	6
LDA	0.455446	0.247525	0.485149	0.455446	0.445545	0.445545	0.435644	3
MLPc	0.445545	0.445545	0.445545	0.524752	0.445545	0.514851	0.326733	4
Mayor score	**RF**	**LR**	**RF**	**RF**	**RF**	**RF**	**KNN**	

Fuente: Elaboración propia.

se considera que los modelos no cuentan con la estabilidad suficiente como para seleccionar un buen modelo.

Durante la última etapa de optimización, el mejor modelo resulta ser k-vecinos más cercanos con el mayor accuracy (0.693069). Sin embargo, dado que a lo largo de todas las etapas existe una mayor estabilidad en el modelo de bosques aleatorios por lo que sería este el que se elegiría como el ganador ($accuracy = 0.673267$).

Para medir el desempeño del modelo con nuevos datos, es posible ahora medir cómo es su comportamiento utilizando el conjunto de prueba (*test set*) a través de la matriz de confusión. Para obtenerla, es necesario realizar las predicciones `y_pred` y compararlas con los valores reales `y_test`.

A continuación se presentan las matrices de confusión para los tres modelos que obtuvieron el mayor desempeño en cuanto a su *accuracy* para el conjunto de validación.

Para k-vecinos más cercanos:

	buy	hold	sell
buy	20	1	4
hold	5	29	11
sell	2	8	21

Para árboles de decisión:

	buy	hold	sell
buy	15	4	6
hold	2	31	12
sell	6	9	16

Para bosques aleatorios:

	buy	hold	sell
buy	20	3	2
hold	6	28	11
sell	5	6	20

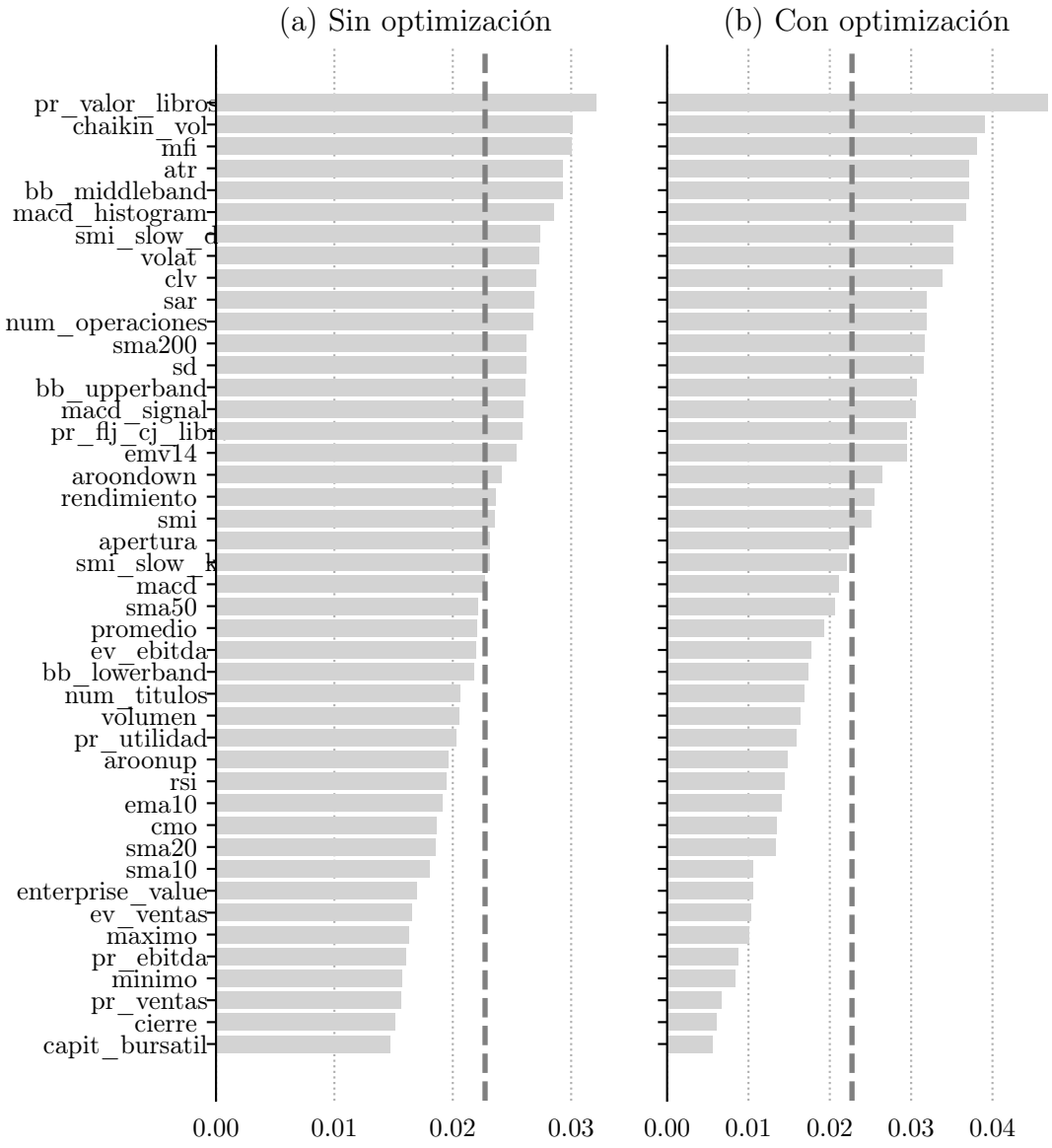
Una vez que se cuenta con toda la información anterior, es necesario entrenar nuevamente el modelo, pero ahora usando la totalidad del *dataset*, con la finalidad de poder salvar el modelo y utilizarlo en producción.

4.7. Simulación de una estrategia simple

Con base en los resultados del pronóstico obtenidos, se buscará simular una estrategia de inversión, medir el desempeño de la misma. La estrategia aquí descrita destaca por su simplicidad, y busca incorporar la utilización del modelo como herramienta para la toma de decisiones.

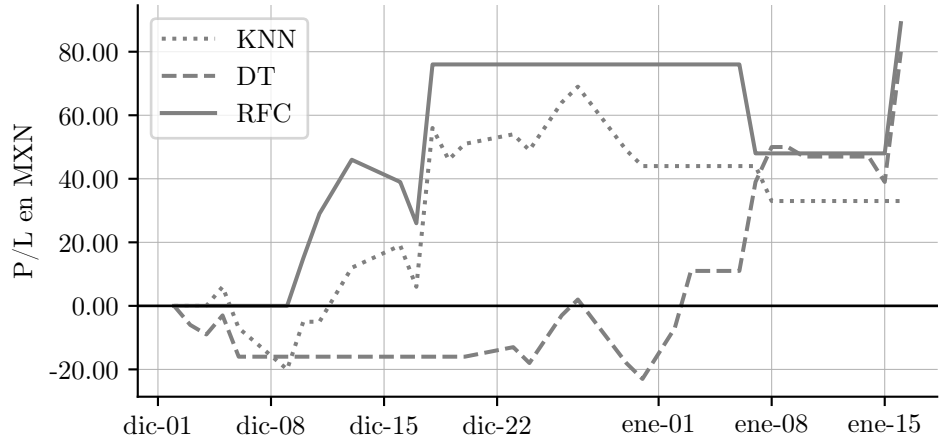
La estrategia planteada consiste en que con los datos hasta el día *k-1*, realizar un pronóstico sobre lo que sucederá en el día *k*. Si el resultado es "comprar", se envía al *broker* la orden de comprar 100 acciones al precio de al apertura en la jornada del día *k*. Si el resultado del pronóstico es *mantener*, no se realizan compras ni ventas de parte del inversionista. Si el pronóstico

Figura 15: Variables seleccionadas después de la optimización de hiperparámetros.



Fuente: Elaboración propia.

Figura 16: Beneficios/Pérdidas de la aplicación de los tres mejores modelos.



Fuente: Elaboración propia.

indica que se debería "vender", se realiza una operación en corto por 100 acciones al precio de apertura de la jornada en el día k. Todas las operaciones se cierran el mismo día en que fueron enviadas al *broker* al precio de cierre. No existe una estrategia de administración del riesgo, por lo que no se consideran órdenes *stop-loss* ni *take-profit* a lo largo de la jornada.

La estrategia se planteará con información del 2 de diciembre de 2019 al 16 de enero de 2020, es decir 31 jornadas bursátiles. Se comparará el desempeño de los 3 modelos que obtuvieron el mayor *accuracy*, tomando como *benchmark* el rendimiento del Índice de Precios y Cotizaciones (IPC).

Para el caso del modelo k-vecinos más cercanos, generó 16 recomendaciones de compra y 1 de venta, las restantes 14 señales fueron de no hacer nada. Siguiendo las recomendaciones hechas por este modelo, el inversionista obtiene un rendimiento de 2.22 % en las 31 jornadas de la simulación. En términos anuales, el rendimiento es de 17.89 %.

Por otro lado, el modelo de árboles de decisión, generó 14 señales de compra y 4 de venta, mientras que fueron 13 recomendaciones de no realizar operaciones. El inversionista, siguiendo las recomendaciones obtenidas del modelo, obtiene un rendimiento sobre su inversión del 5.38 %, que en términos anuales equivale a 84.01 %.

Finalmente, el modelo de bosques aleatorios generó 8 recomendaciones de compra, 23 de no operar y ninguna de realizar ventas. Si se siguen las reco-

mendaciones del modelo, el inversionista obtendría un rendimiento de 5.98 % sobre su inversión, equivalente a un rendimiento anualizado de 106.81 %.

Para el mismo periodo, si el inversionista hubiera comprado algún ETF que replique el comportamiento del IPC, habría obtenido un rendimiento de 5.69 %. De manera que el único modelo que supera al *benchmark* es el de bosques aleatorios. Aunque el rendimiento por encima del índice de referencia parezca marginal, hay que tener en cuenta que en la simulación no se está considerando ninguna acción para administrar la exposición al riesgo.

De manera gráfica, en la Figura 16 se representan los beneficios acumulados obtenidos empleando los tres modelos descritos con anterioridad.

Conclusiones, limitaciones, recomendaciones y futuras investigaciones

Conclusiones

El comportamiento de los mercados bursátiles es errático, cambia de un momento a otro y en general es difícilmente predecible, como pudo revisarse en la literatura. Sin embargo, eso no significa que no sea posible obtener resultados aceptables en cuando a una metodología o modelo que permita al inversionista tomar decisiones informadas.

Como pudo apreciarse, los modelos basados en árboles obtuvieron un desempeño sobresaliente sobre los modelos basados en probabilidad, incluso superando a las redes neuronales, lo que reafirma lo revisado en el estado del arte. Desde etapas tempranas, el modelo de bosques aleatorios obtuvo un desempeño que destacó, logrando obtener el mayor rendimiento durante la simulación.

El hecho de realizar una validación de los datos empleando la validación cruzada para una serie de tiempo, implica que el modelo se elija bajo las condiciones propias de una serie de tiempo financiera, logrando de esta manera un mejor desempeño y por ende un mayor poder predictivo. Aunque este paso muchas veces es omitido, otorga una mayor certidumbre sobre la calidad del aprendizaje que están obteniendo los modelos.

La optimización de los modelos mediante la selección de los mejores hiperparámetros, no solo trae consigo mejores valores de exactitud en los modelos, sino que además reduce significativamente los tiempos de cálculo, tanto para realizar el entrenamiento como para determinar las métricas de desempeño para los subconjuntos de validación y pruebas en la fase de construcción del

modelo. Además de que realizar este tipo de ajustes, evita en la medida de lo posible que se presente el problema del sobreaprendizaje. Con los resultados obtenidos mediante las matrices de confusión y una vez realizados los pronósticos, existe evidencia suficiente para determinar que no existió problema de sobreaprendizaje.

Aunque la exactitud de los modelos, con valores que rondan el setenta por ciento, parece relativamente baja, se debe tener en cuenta que por las características que se mencionaron a lo largo de los capítulos anteriores, la dificultad de poder pronosticar con certeza implica un enorme reto. Sin embargo, al observar los rendimientos obtenidos, se obtiene evidencia de la efectividad, que aunque se trata de una simulación sencilla, en combinación con una adecuada gestión del riesgo podría traducirse en beneficios aceptables para un inversionista dado el nivel de riesgo.

Así mismo, los rendimientos obtenidos, salvo en el modelo de k-vecinos más cercanos, guarda una estrecha relación con el rendimiento obtenido mediante la inversión en el índice que se tomó como *benchmark*, el IPC. En este caso particular, dado que América Móvil, S.A.B. de C.V. tiene un peso significativo dentro del índice, se podría decir que existe una gran correlación entre ambos instrumentos financieros.

Aunque los resultados son aceptables para el caso de esta emisora, no es posible generalizarlos para empresas con menor grado de liquidez; aunque se puede inferir que por ser menos bursátiles, también presentan menor volatilidad por lo tanto la cantidad de señales de compra, venta o mantener se reducirían drásticamente, provocando que el modelo entrenado presentara una condición de sobreaprendizaje, limitando su desempeño y al mismo tiempo su capacidad predictiva. Se podría decir bajo el razonamiento anterior que la utilización de estos modelos requieren de una emisora con la volatilidad adecuada para poder obtener las señales. Adicionalmente, la falta de liquidez dificulta las transacciones diarias, por lo que para posiciones muy grandes sería complicado entrar y salir el mismo día.

La incorporación de información fundamental (aunque limitada) otorga mayor conocimiento sobre la operación de la empresa, de manera que se obtiene un modelo ajustado a las características particulares de la emisora. Para el caso del mercado mexicano, la obtención de la información fundamental se vuelve complicada pues no se cuenta con datos en forma estructurada que faciliten su recolección.

Dejando a un lado momentáneamente la hipótesis de los mercados eficientes, el modelo sigue siendo vulnerable ante situaciones extraordinarias

adversas tales como la aparición de noticias o cambios abruptos en los tipos de cambio, debido a que no se contempla este tipo de variables dentro del modelo, por lo que es necesario que se emplee en combinación con medidas de mitigación del riesgo.

Por lo tanto, con los resultados obtenidos en esta investigación, existe evidencia experimental de que no es posible rechazar la hipótesis planteada. Los algoritmos de minería de datos brindan buenas oportunidades como predictores de los movimientos en los precios, partiendo desde información de los precios, información fundamental e información técnica.

Limitaciones

El principal reto al afrontar este tipo de modelos es que deben ser entrenados y optimizados para cada emisora que se desee analizar, por lo que es un proceso largo y costoso en cuanto a tiempos de cómputo, por lo que no es una alternativa que esté al alcance de todos.

Por tratarse de características particulares de la emisora, el sector, e incluso de la fase económica que prevalezca, es muy probable que para otra empresa (incluso del mismo ramo), el modelo que mejor se ajuste sea uno distinto, por lo que no es posible generalizar su uso.

A su vez, dadas las características de los mercados, las condiciones en las que se desarrollan las negociaciones, la incertidumbre económica y política, aspectos como el riesgo país, barreras de entrada, solo por mencionar algunas, provocan que el modelo tenga una fecha de caducidad implícita, y continuamente se debe estar ajustando y calibrando para que refleje las condiciones que prevalecen.

Por otro lado, la construcción del *dataset* se enfoca exclusivamente en datos internos de la emisora y no se incluyen variables que reflejen las condiciones existentes en el mercado tales como las noticias, anuncios corporativos, análisis de sentimientos, índices de mercado y por sector, tipos de cambio, etcétera.

Dadas las características propias del mercado mexicano, la cantidad de información disponible sigue siendo limitada, sobre todo si se compara con mercados de dimensiones significativamente mayores como los de Nueva York y Londres. En consecuencia, se encuentran excluidas del análisis muchas variables fundamentales que son de relevancia para obtener una mejor recomendación.

Por último, dado que es requerida la información fundamental, la implementación de un modelo con estas características se encuentra limitado a ser utilizado solamente en acciones, quedando excluidos algunos otros instrumentos como *commodities*, divisas, criptoactivos, etcétera.

Recomendaciones

Los datos son el ingrediente principal de todo modelo de minería de datos o de *machine learning*, por lo que siempre se deberá recurrir a fuentes confiables para la obtención de las cotizaciones. Existen múltiples recursos gratuitos que proveen esta información, pero se debe ser cauteloso de que haya congruencia para lograr resultados satisfactorios.

La etapa exploratoria de los datos es una de las más importantes, ya que permite de manera visual identificar la distribución de las variables, y puede advertir sobre posibles correlaciones, mismas que deberían evitarse. Más aún, puede proporcionar indicios sobre si existe sesgo en los datos, lo cual podría ocasionar sobreaprendizaje en el modelo.

Es importante recalcar que en un problema de minería de datos aplicado a series de tiempo, la validación cruzada no debe realizarse en la forma clásica, ya que no se puede garantizar que los datos sean independientes idénticamente distribuidos. Por lo tanto, se debe contemplar la utilización de validación cruzada incremental o por bloques para que se tome en cuenta el efecto de la dependencia en el tiempo de las instancias.

Al realizar la optimización de los modelos, se debe tener en cuenta las variables que hayan sido seleccionadas con algún algoritmo previo, ya que realizarlo con todas las variables del *dataset* resultará en un modelo optimizado para tales variables. Se esperaría que un modelo optimizado y con las variables seleccionadas tenga un mejor desempeño que un modelo genérico.

La manera óptima de realizar la selección de variables y el ajuste del modelo es mediante un algoritmo recurrente, es decir que se itere sobre cada modelo, ajustando los parámetros y al mismo tiempo ajustando las variables seleccionadas por algún método. Los enfoques que aquí se abordaron pueden ser aplicados tanto al caso de variables objetivo continuas como categóricas.

A medida que los mercados evolucionan, es necesario crear nuevos modelos cuando exista evidencia de su baja precisión: para ello se recomienda que cada cuatro meses se analice nuevamente el desempeño del modelo, agregando la información fundamental y técnica que se haya generado, o cuando

exista un cambio de tendencia significativo en el mercado.

Futuras investigaciones

Con la experiencia obtenida en el desarrollo del modelo expuesto, y con la finalidad de solventar algunas de las limitaciones previstas, se buscaría dar un enfoque de *reinforcement learning* para que el modelo vaya aprendiendo cada día con los datos que se vayan generando en el mercado.

La manera de abordar esta investigación podría ser mediante el uso de redes neuronales recurrentes, como una extensión del modelo *multi-layer perceptron*, empleando una robusta cantidad de capas.

Dado que la temporalidad no es un problema para abordar el análisis técnico financiero, se contempla el análisis y desarrollo de modelos empleando datos intradía, pensando en una futura implementación para un algoritmo de *trading* de alta frecuencia (HFT), asumiendo el riesgo que conlleva usar una temporalidad menor en lo que a la vigencia del modelo se refiere.

Se espera que en desarrollos futuros se podrían aplicar los criterios y las variables que se han descrito con anterioridad para integrar el modelo como parte de un algoritmo que administre un portafolio, haciendo rebalanceos tomando en cuenta los resultados del algoritmo.

Tal como se ha realizado en otros mercados emergentes, la utilización de información no estructurada para el análisis de sentimientos, proveniente de redes sociales o sitios web de noticias financieras, podrá fungir como un componente importante para realizar predicciones más precisas.

Por último, se puede abordar también el incluir información proveniente de los mercados de futuros para el contrato inmediato siguiente, ya que por el principio de convergencia, pueden dar indicios sobre hacia dónde se moverá el precio.

Referencias

Abuzir, D. E. Y., y M.Baraka, M. A. (2019). Financial Stock Market Forecast Using Data Mining in Palestine. *Palestinian Journal of Technology and Applied Sciences (PJTA)*, *2*(1), 38-48.

Ahmadi, E., Jasemi, M., Monplaisir, L., Nabavi, M. A., Mahmoodi, A. y Amini Jam, P. (2018). New efficient hybrid candlestick technical analysis model for stock market timing on the basis of the support vector machine and heuristic algorithms of imperialist competition and genetic. *Expert Systems with Applications*, *94*, 21 - 31.

América Móvil. (2014). *América móvil - acerca de nosotros.* Descargado 2020-03-13, de `https://www.americamovil.com/Spanish/acerca-de-nosotros/nuestra-empresa/default.aspx`

Appel, G. (2005). *Technical analysis: Power tools for active investors.* Financial Times/Prentice Hall.

Ballings, M., Van den Poel, D., Hespeels, N. y Gryp, R. (2015). Evaluating multiple classifiers for stock price direction prediction. *Expert Systems with Applications*, *42*(20), 7046 - 7056.

Basak, S., Kar, S., Saha, S., Khaidem, L. y Dey, S. R. (2019). Predicting the direction of stock market prices using tree-based classifiers. *The North American Journal of Economics and Finance*, *47*, 552 - 567.

Batra, R., Mridulaand Agrawal. (2018). Comparative analysis of decision tree algorithms. *Nature Inspired Computing*, 31–36.

Berry, W. D. (1993). *Understanding regression assumptions.* SAGE Publications.

Blau, W. (1973). Stochastic momentum. *Technical analysis of stocks and commodities*, *1*.

Bollinger, J. (2001). *Bollinger on bollinger bands* (first ed.).

Bramer, M. (2016). *Principles of data mining* (third ed.). Springer-Verlag London.

Breiman, L., Friedman, J., Stone, C. J. y Olshen, R. (1984). *Classification and regression trees*. Chapman and Hall/CRC.

Browne, M. W. (2000). Cross-validation methods. *Journal of Mathematical Psychology*, *44*(1), 108 - 132.

Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., ... Varoquaux, G. (2013). API design for machine learning software: experiences from the scikit-learn project. En *Ecml pkdd workshop: Languages for data mining and machine learning* (pp. 108–122).

Cai, J., Luo, J., Wang, S. y Yang, S. (2018). Feature selection in machine learning: A new perspective. *Neurocomputing*, *300*, 70 - 79.

Chai, T., y Draxler, R. R. (2014). Root mean square error (RMSE) or mean absolute error (MAE)? *Geoscientific Model Development Discussions*, *7*(1), 1525-1534.

Chande, T., y Kroll, S. (1994). *The new technical trader: Boost your profit by plugging into the latestindicators*. Wiley.

Coyne, S., Madiraju, P. y Coelho, J. (2017). Forecasting stock prices using social media analysis. En *2017 ieee 15th intl conf on dependable, autonomic and secure computing, 15th intl conf on pervasive intelligence and computing, 3rd intl conf on big data intelligence and computing and cyber science and technology congress(dasc/picom/datacom/cyberscitech)* (p. 1031-1038).

Do Van, T., Minh Hai, N. y Hieu, D. (2018, 01). Building unconditional forecast model of stock market indexes using combined leading indicators and principal components: application to vietnamese stock market. *Indian Journal of Science and Technology*, *11*, 1-13.

Dunne, M. (2015). *Stock market prediction.*

Engle, R. F., y Patton, A. J. (2000). What good is a volatility model? *Quantitative finance*, *1*, 237-245.

Fama, E. F. (1965). The behavior of stock-market prices. *The journal of Business*, *38*(1), 34–105.

Field, A., Miles, J. y Field, Z. (2012). *Discovering statistics using r*. SAGE Publications.

Frawley, W. J., Piatetsky-Shapiro, G. y Matheus, C. J. (1992). Knowledge discovery in databases: An overview. *AI Magazine*, *13*(3), 57.

Garman, M. B., y Klass, M. J. (1980). On the estimation of security price volatility from historical data. *Journal of business*, *1*, 67-78.

Gorunescu, F. (2013). *Data mining: Concepts, models and techniques*. Springer Berlin Heidelberg.

Gupta, A., Bhatia, P., Dave, K. y Jain, P. (2019). Stock market prediction using data mining techniques. *Available at SSRN 3370789*.

Guyon, I., y Elisseeff, A. (2003). An introduction to variable and feature selection. *The Journal of Machine Learning Research*, *3*, 1157–1182.

Hawkins, D. M. (2004). The problem of overfitting. *Journal of Chemical Information and Computer Sciences*, *44*(1), 1-12.

Kenton, W. (2019). *Black box model.* Descargado 2020-05-04, de `https://www.investopedia.com/terms/b/blackbox.asp`

Krüger, F. (2016). *Activity, context, and plan recognition with computational causal behaviour models* (Tesis Doctoral, University of Rostock). Descargado de `https://www.uni-rostock.de`

Li, Q., Wang, T., Gong, Q., Chen, Y., Lin, Z. y kwang Song, S. (2014). Media-aware quantitative trading based on public web information. *Decision Support Systems*, *61*, 93 - 105.

Lin, Z. (2018). Modelling and forecasting the stock market volatility of sse composite index using garch models. *Future Generation Computer Systems*, *79*, 960 - 972.

López de Prado, M. (2018). *Advances in financial machine learning* (first ed.). Wiley.

Malkiel, B. G., y Fama, E. F. (1970). Efficient capital markets: A review of theory and empirical work. *The journal of Finance*, *25*(2), 383–417.

Marek, P., y Čadková, V. (2020). Optimization and testing of money flow index. *APLIMAT 2020 - 19th Conference on Applied Mathematics*.

Microsoft Corporation. (2020). *Ajuste de los hiperparámetros de un modelo mediante azure machine learning.* Descargado 2020-05-15, de `https://docs.microsoft.com/es-es/azure/machine-learning/how-to-tune-hyperparameters`

Murphy, J. J. (2016). *Análisis técnico de los mercados financieros* (first ed.). Planeta.

Powers, D. (2011). Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. *Journal of Machine Learning Technologies*, *2*(1), 37-63.

Ross, P., Jaffe, J., Westerfield, R. y Bradford D. Jordan, P. (2015). *Corporate finance.* McGraw-Hill Education.

Sorto, M., Aasheim, C. y Wimmer, H. (2017). Feeling the stock market: A study in the prediction of financial markets based on news sentiment. *SAIS 2017 Proceedings.*, *30*.

The Ciu. (2019). *Telecomunicaciones móviles al primer trimestre 2019,*

méxico: Líneas y estructura del mercado. Descargado 2020-05-04, de https://www.theciu.com/publicaciones-2/2019/5/20/telecomunicaciones-mviles-al-primer-trimestre-2019-mxico-lneas-y-estructura-del-mercado

Thomsett, M. (2010). *Cmf-chaikin money flow: Changes anticipating price reversal*. Pearson Education.

Ting, K. M. (2010). Confusion matrix. En C. Sammut y G. I. Webb (Eds.), *Encyclopedia of machine learning* (pp. 209–209). Boston, MA: Springer US.

Walpole, R., Myers, R. y Myers, S. (1999). *Probabilidad y estadística para ingenieros*. Pearson Educación.

Wang, Y. (2019). Analysis of financial business model towards big data and its applications. *Journal of Visual Communication and Image Representation*, 102729.

Wanjawa, B. W., y Muchemi, L. (2014). *Ann model to predict stock prices at stock exchange markets.*

Wilder, J. W. (1978). *New concepts in technical trading systems*. Trend Research.

Yan, X. (2009). *Linear regression analysis: Theory and computing*. World Scientific Publishing Company Pte Limited.

Zhang, Q., Zhou, X., Pan, Z., Hu, G., Tang, S. y Zhao, C. (2018). Stock market prediction on high-frequency data using generative adversarial nets. *Mathematical Problems in Engineering*, *2018*, 4907423.