Universidad Nacional Autónoma de México

Facultad de Ciencias

Construcción de indicador de la dirección del precio de acciones: modelo de aprendizaje automático basado en minería de opinión

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

Actuaria

QUE PRESENTA:

Carol Sánchez Garibay

TUTOR

Fís. Jimmy Hernández Morales

Ciudad Universitaria, Ciudad de México, 2021

Agradecimientos

Son muchas las personas que han contribuido en el proceso y conclusión de este trabajo pero me gustaría hacer una mención especial a algunas de ellas. En primer lugar, quiero agradecer a la Universidad Nacional Autónoma de México por haberme dado las herramientas necesarias para mi desarrollo académico y profesional así como el pensamiento crítico que caracteriza a los que estudiamos en esta honorable institución.

Agradezco a Jimmy Hernández Morales, asesor y director de esta tesis y mi profesor de la materia de Introducción a la ciencia de datos y *Machine Learning* donde nació la inspiración para realizar esta investigración. La experiencia, conocimiento y compromiso de Jimmy desempeñaron un papel fundamental en la culminación de este trabajo y estoy segura de que fue la mejor elección para dirigir este proyecto.

El desarrollo de esta tesis no hubiera sido posible sin el apoyo de mi familia que es mi principal motor para enfrentar los retos que se me presentan en la vida. A mi mamá por ser mi mayor ejemplo de fortaleza, valentía, resiliencia y amor incondicional y por enseñarme que a pesar de tener miedo, debo seguir adelante. A mi papá por siempre recordarme que no valemos por lo que tenemos sino por lo que somos y por enseñarme a luchar como guerrera por mis sueños. A mi esposo Javier por no soltar mi mano, por cada palabra de aliento, por cada consejo, por su paciencia y por ser mi gran compañero de vida. A mis hermanos Leo, Chris, July y Bicho a quienes sirvo como referencia para trabajar duro hasta alcanzar sus metas y que quiero ver cumpliendo sus sueños. A mis grandes amigas y amigos con los que encontré momentos de distracción cuando el objetivo parecía inalcanzable.

Para mí, terminar esta tesis representa el final de un proyecto de muchos años de esfuerzo y por eso extiendo el agradecimiento a todas las personas que me acompañaron en mi formación profesional desde que era pequeña. A mi tía Paty a quien siempre he admirado por ser una mujer independiente y empoderada. A mi tía Norma por darme la tranquilidad de saber que la familia está unida gracias a su esfuerzo. A mi Nina porque no olvido todas las veces que me apoyó para seguir estudiando y, finalmente, a mis primos (tíos) con los que crecí y que significan mucho para mi.

Esta tesis fue todo menos fácil pero pude demostrarme a mí misma que con autodisciplina, objetivos claros y perseverancia se pueden lograr cosas increíbles. Estoy satisfecha con el resultado y espero que sea de utilidad para aquellas personas que se tomen el tiempo de leerla.

Índice general

Resumen

En esta tesis se expone el problema de predecir la dirección de los precios de las acciones del sector financiero que cotizan en la bolsa mexicana de valores. Se propone un modelo de clasificación donde la variable objetivo se define como función de la variación porcentual diaria del precio del conjunto de acciones que integran el sector y donde las opiniones de los usuarios de redes sociales fungen como variables predictoras de dicha variable objetivo.

Al inicio del trabajo, se expresa la importancia que ha tomado el procesamiento del lenguaje natural en los últimos años, así como el avance y modernización de las herramientas y tecnologías que permiten explotar la información. Se habla de la motivación técnica y personal de la realización del trabajo y también se explica la estructura de los textos en idioma español y como se puede procesar para tener *insights* (información clave) sobre el comportamiento de los mercados financieros. Para terminar con la primera parte, se describe el proceso de obtención de los datos necesarios para la construcción del modelo.

El fundamento matemático y estadístico de los modelos de aprendizaje automático supervisado se aborda en la segunda parte; también se habla de la terminología y las cuatro fases necesarias para el desarrollo de modelos de aprendizaje de máquina. El tema central de esta sección son las Máquinas de Soporte Vectorial que es el método de clasificación lineal que se usa para el modelo. Las definiciones y uso de métricas de evaluación de desempeño y selección de modelos también se explican en esta parte del proyecto.

El procesamiento y transformación del texto se aborda en el capítulo 3. Se explica detalladamente la construcción de una de las variables predictoras que es el sentimiento de la opinión que usa un modelo intermedio en el que primero se vectoriza el texto usando un método llamado TF-IDF y luego asigna el sentimiento: Positivo, Negativo o Neutral. También se analizan el resto de las variables predictoras cuantitativas que provenienen de las opiniones de los usuarios y del contexto de la red social Twitter.

Dado que la aplicación de la teoría matemática está orientada a un tema financiero, se dedicó un capítulo al análisis del comportamiento de dicho sector en el mercado accionario mexicano. En esta parte, también se define la *variable objetivo* que indica la oportunidad de compra de acciones con base en la dirección de su precio. Luego, se unen las piezas para aplicarlas en el caso de estudio y dar los resultados del mismo. Se presenta gráficamente el cálculo de cada métrica y se explica el motivo de la selección del modelo final así como la importancia de cada variable a través de los valores SHAP.

Para culminar, se propone como trabajo futuro, la utilización de información en tiempo real como el *streaming* (flujo continuo de datos), el reentrenamiento del modelo con más variables y fuentes de información. También se propone la definición de otra variable objetivo que prediga oportunidades de venta de acciones del sector. Además, se discute la utilización de los modelos de aprendizaje de máquina en producción ya que existe desconfianza por parte de las personas que no conocen el fundamento matemático de este tipo de herramientas.

- o

Capítulo 1

Introducción

La estadística y las ciencias de la computación han permitido usar los datos contenidos en nuevas fuentes de información como las redes sociales para transformarlos en conocimiento. Por su parte, la inteligencia artificial y, particularmente, el aprendizaje de máquina (aprendizaje supervisado, aprendizaje automático o Machine Learning) han hecho posible explicar fenómenos de manera muy precisa y aplicarlos en una gran variedad de ramas del conocimiento que van desde la medicina hasta la mercadotecnia.

El aumento de la capacidad computacional y el acceso a tecnologías de la información hacen posible producir, almacenar y enviar más datos que nunca. Estos datos alimentan los modelos de Aprendizaje Automático y son el impulso principal del auge que esta ciencia ha experimentado en los últimos años. El Procesamiento del Lenguaje Natural (NLP por sus siglas en inglés) es uno de los campos que ha despertado más interés debido a la amplia gama de aplicaciones útiles tanto en la academia como en la industria privada. Por su parte, el crecimiento de la web ha facilitado emitir opiniones sobre acontecimientos o noticias y publicarlos en redes sociales, blogs, foros o cualquier otro lugar en internet, haciendo imposible para los humanos analizar toda esta información sin la utilización de tecnologías para el tratamiento de grandes cantidades de datos, por este motivo, se han desarrollado herramientas de *web scraping* que son utilizadas para extraer datos de sitios web simulando la navegación de un humano.

Dado lo anterior, en este trabajo se extrae información de una red social para construir un modelo que indica cuando hay una oportunidad de compra de las acciones que integran el sector financiero del Índice de Precios y Cotizaciones de la Bolsa Mexicana de Valores (S&P BMV IPC). Las principales fuentes de información para construir el modelo fueron publicaciones de Twitter y datos diarios de los precios de las acciones del sector financiero del IPC. Python fue el lenguaje de programación seleccionado para la extracción y procesamiento de los datos así como para el desarrollo del modelo.

1.1. Motivación

La motivación del presente trabajo proviene de los acontecimientos políticos y económicos que sucedieron en México durante el último trimestre del 2018, como la elección presidencial para el periodo 2019-2024 y la cancelación de un nuevo aeropuerto en Texcoco. Los precios de las acciones que cotizan en el mercado mexicano sufrían grandes pérdidas mientras que los inversionistas compartían sus opiniones en redes sociales. Así surge la hipótesis de que existe una relación entre las publicaciones en redes sociales y los precios de las acciones, por

lo tanto, es posible modelar ese comportamiento.

Es claro que no es posible procesar el gran volumen de información que los usuarios generan en las redes sociales así que es necesario construir algoritmos que permitan extraer, transformar y modelar la información para entender la relación que existe con el mercado accionario.

Por último, la gran mayoría de los trabajos que implican el procesamiento de datos en forma de texto están basados en el idioma inglés, cuyas estructuras sintáctica y gramatical son completamente distintas al español. Por esta razón, constituye un gran reto construir un modelo de Procesamiento de Lenguaje Natural (NLP) basado en español por la dificultad de adaptarlo a las particularidades del idioma como la jerga popular, abreviaturas, analogías, sarcasmo, etc.

1.2. Objetivos

La presente tesis tiene la finalidad de procesar opiniones sobre un tema particular para reconocer la dirección semántica de estas opiniones y predecir oportunidades de compra de acciones en el mercado de valores mexicano. Por otra parte, se pretende dar a conocer el fundamento matemático detrás de los modelos de aprendizaje de máquina y su capacidad para llevar a conclusiones importantes que pueden ayudar a la toma de decisiones de inversión. Finalmente, se desea discutir el uso de este tipo de herramientas de manera productiva ya que existe escepticismo entre los analistas financieros acerca del funcionamiento de este tipo de instrumentos.

1.3. El texto

Antes que nada, se debe conocer la estructura del texto para poder entender cómo puede ser abordado el tema. Esta estructura tiene dos componentes básicos y funcionales; el primero es la semántica que es la ciencia lingüística que estudia la denotación y connotación de las palabras. El segundo es la sintaxis que se ocupa de dar coherencia a las oraciones de tal manera que se entienda la semántica, dando orden y relación a las palabras que componen el texto.

La semántica lingüística da significado a las palabras a través de semas y puede ser denotativa o connotativa. Es denotativa cuando el mensaje se expresa objetivamente, mientras que es connotativa si se le añade alguna valoración personal mediante gestos o entonaciones.

1.3.1. La opinión

Una opinión es una idea, juicio o concepto que una persona tiene o se forma acerca de algo y que expresa de manera connotativa, es decir, que dependen de la experiencia del emisor para darle dirección a esta idea. Por ejemplo, la frase “este coche tiene cuatro ruedas” no es una opinión ya que se puede comprobar su veracidad, mientras que la frase “Este es el mejor coche de la actualidad” sí representa una opinión porque depende de lo que el emisor espere de un coche o su experiencia con el.

En el Procesamiento del Lenguaje Natural, una opinión está compuesta de dos factores: la dirección y la magnitud. La dirección hace referencia al partido que toma el emisor sobre el hecho, es decir, positivo cuando está de acuerdo y negativo cuando no lo está. La magnitud indica qué tan de acuerdo se siente el emisor con lo que dice. Por ejemplo, la frase “hoy es un buen día para el IPC” que no tiene la misma intensidad que la frase “hoy es el mejor día del año para el IPC” a pesar de que ambas frases hacen referencia al mismo hecho.

Una de las grandes dificultades que se presentan al hacer minería de opinión son las frases pragmáticas, es decir, aquellas que se dicen con ironía, sarcasmo o intencionalidad, ya que en este tipo de opiniones es difícil interpretar el sentimiento del emisor cuando se procesan de manera automática. Por ejemplo, supongamos que la noticia es que el mercado accionario mexicano perdió 10 % en el mes y una persona publica “¡Qué buen día

para el IPC!", está siendo sarcástica al respecto entonces los algoritmos clásicos de sentimientos detectarían erróneamente la dirección de esta opinión. Existen modelos que usan factores como el tema del que se habla, la audiencia y el autor para detectar este tipo de frases. Por simplicidad y enfoque, en este caso de estudio no se llegará a esa profundidad en el análisis del texto así que se asume que habrá opiniones con error en la predicción de la dirección de la opinión.

1.4. Obtención de opiniones

Con el crecimiento de la web, se desarrollaron las interfaces de programación de aplicaciones (APIs) que permiten la comunicación entre aplicaciones mediante un conjunto de reglas y especificaciones proveidas por un usuario. Las redes sociales como Twitter han desarrollado sus propias interfaces para que los programadores puedan hacer consultas masivas a sus bases de datos a través de una conexión autenticada.

Por su parte, el *web scraping* es un término utilizado para indicar el uso de un programa o algoritmo para extraer datos de archivos HTML que se encuentran en internet. Estos programas interactúan con las páginas web a través de APIs y usan parámetros que permiten hacer consultas de datos. Actualmente, este tipo de herramientas son muy usadas en las grandes empresas para rastrear los precios o actividades de sus competidores para hacer ofertas o lanzamientos de productos aunque también se han identificado usos maliciosos que han llevado a su prohibición en algunos paises.

Para obtener las publicaciones de la red social se usó un algoritmo de *web scraping* incluido en un paquete de Python que hace consultas parametrizadas a la API de Twitter y almacena los resultados en archivos de texto en formato *json* (notación de objetos JavaScript). Cada publicación resultante de la consulta se almacena como en la figura 1.1 en la que se cuenta con toda la información necesaria para identificarla.


```
{
 "usernameTweet": "lopezobrador_",
 "ID": "1047210080637149184",
 "text": "El 68 es sinónimo de autoritarismo.",
 "url": "/lopezobrador_/status/1047210080637149184",
 "nbr_retweet": 3964,
 "nbr_favorite": 13591,
 "nbr_reply": 957,
 "datetime": "2018-10-02 14:41:54",
 "is_reply": false,
 "is_retweet": false,
 "user_id": "82119937"
}
```


Figura 1.1: Ejemplo del formato de los archivos de texto obtenidos al ejecutar el algoritmo de web scraping sobre las publicaciones de Twitter

La información requerida para este proyecto se acota en la extracción de opiniones que puedan ser relevantes para el sector financiero; por lo tanto, se parametrizó la consulta de tal manera que se obtuvieran datos relacionados con los acontecimientos politicos, económicos y financierons más relevantes de los años 2018 y 2019. Esta información se obtuvo mediante la búsqueda parametrizada de publicaciones que contuvieran los *cashtags* de los símbolos (*tickers*) de las acciones (**$GFNORT**, **$BBAJIO**, **$GENTERA**, **$GFINBU** y **$BSMX**) o *hashtags* sobre las noticias más relevantes de ese momento tales como **#ComisionesBancarias**, **#NuevoAeropuerto**, **#Elecciones2018**, etc. También se incluyeron las publicaciones de personajes importantes de la política, la economía y el periodismo del país como el presidente, directores de instituciones y periodistas

reconocidos. De igual manera, sólo se tomaron en cuenta aquellas publicaciones escritas en español para que los criterios de análisis que usan la estructura del idioma y de normalización de texto fueran válidos.

1.5. Modelo de Sentimientos

Una de las variables predictoras que se pretende usar en el presente trabajo es la dirección semántica (positiva o negativa) de la opinión de los usuarios de la red social, sin embargo, la dificultad de usar esta variable es que las opiniones no están clasificadas y se requiere esta información para desarrollar un modelo de sentimientos. El presente trabajo propone una solución a este tema: clasificar una porción de los datos e integrar la información proveniente de la Sociedad Española de Procesamiento del Lenguaje Natural [13] para tener suficiente información en el entrenamiento del modelo de sentimientos y poder usarlo como *input* (entrada) para el modelo final.

En el diagrama de la figura 1.2, se observa el flujo de la construcción del modelo en términos de las fuentes (cilíndros), procesamiento (rombos) y transformación en información cuantitativa (rectángulos) que se usa en los modelos (círculos): asignación de sentimientos y de dirección de precios.

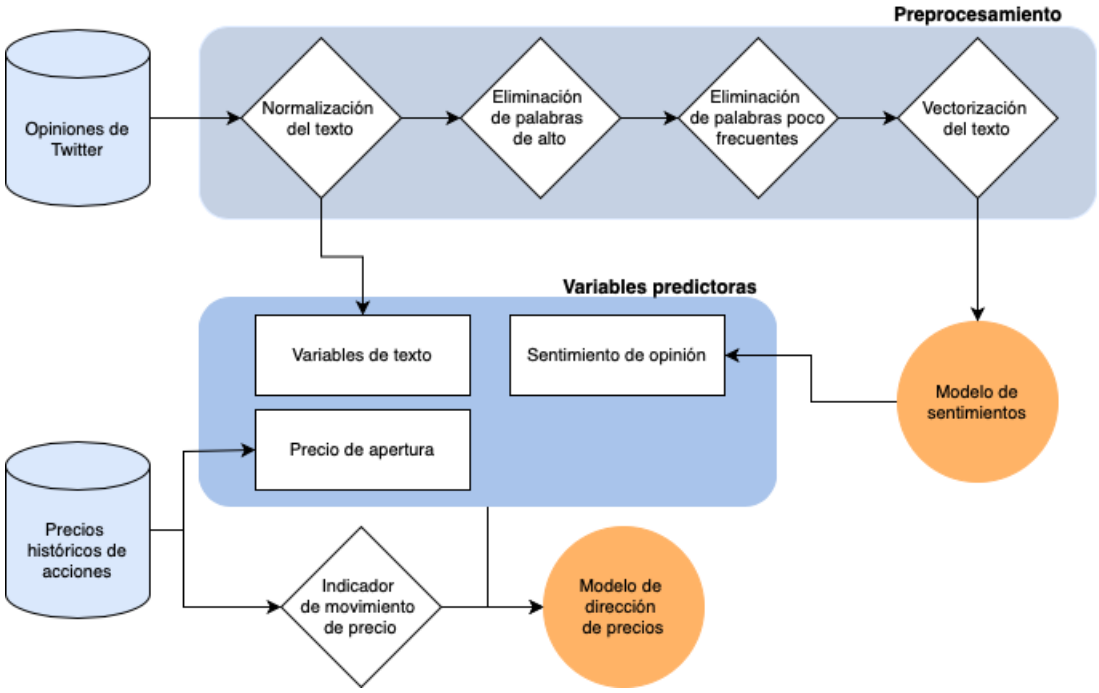


Figura 1.2: Diagrama del sistema: extración, transformación e integración de datos para la construcción del modelo predictivo.

Capítulo 2

Modelo de Aprendizaje Automático

Este capítulo se basa en el libro "The elements of statistical learning: data mining, inference and prediction" [2] ya que provee la intuición necesaria para el entendimiento de los modelos de aprendizaje automático.

Los algoritmos de Aprendizaje Automático tienen aplicaciones muy variadas y han constituido un gran avance en el modelado de fenómenos o acontecimientos mediante el uso de datos. Estos algoritmos van desde la clasificación de correos electrónicos no deseados hasta la detección de enfermedades, y han llegando a convertirse en un recurso necesario para la toma de decisiones en todas las industrias. Entre estos modelos se encuentran los **supervisados** y los **no supervisados** que varían según la presencia de una variable de respuesta en el conjunto de entrenamiento.

Los modelos **supervisados** usan variables predictivas, covariables o *inputs* $X = (X_1^T, ..., X_p^T)$ que tienen cierta relación con una variable de respuesta u *output* $Y = (y_1, ..., y_m)$ de la cual se desea modelar el comportamiento. Este tipo de modelos son conocidos como de "aprendizaje con profesor" ya que el "estudiante" da una respuesta para cada x_i en el conjunto de entrenamiento y el supervisor o "profesor"provee el error asociado a dicha respuesta mediante una función de pérdida definida. Si suponemos que (X, Y) son variables aleatorias representadas por una densidad de probabilidad conjunta $f(x, y)$, entonces el aprendizaje supervisado puede ser visto como un problema de estimación donde se tratan de determinar las propiedades de la densidad condicional $f(y|x)$ conociendo su comportamiento en un conjunto de entrenamiento. Por otro lado, los inputs x_i se introducen en un sistema artificial conocido como Algoritmo de Aprendizaje, que se actualiza segun la diferencia entre el resultado real y el del sistema artificial, esperando que sean lo suficientemente cercanos para ser predictivos en conjuntos de datos no observados. Por su parte, en los modelos **no supervisados** no se cuenta con la variable de respuesta sino que solo se tiene un conjunto de observaciones $(x_i, ..., x_N)$ con función de densidad conjunta $f(x_1, \ldots, x_N)$ a la cual se le desean determinar las propiedades sin un supervisor o 'profesor".

Como en este trabajo usaremos el aprendizaje supervisado para identificar oportunidades de compra de acciones financieras del IPC basándonos en la actividad de los usuarios de redes sociales, se profundizará unicamente en la explicación de este tipo de modelos.

Los modelos de aprendizaje supervisado se nombran dependiendo del tipo de medida del *output*. Si el output es numérico, entonces se trata de regresión; si el output está dentro de un conjunto de clases o categorías, entonces es una clasificación. A pesar de esta diferencia, la misión de ambos modelos es aproximar una función $\hat{f}$ que describa el comportamiento del *output*. En el diagrama de la figura 2.1 se muestran, de manera superficial,

las cuatro fases necesarias para hacer un modelo de aprendizaje supervisado. En la primera etapa del proceso, conocida como *preprocesamiento*, se recolecta, procesa y divide la información en dos partes: la primera (datos de entrenamiento) se usará directamente en el modelo de aprendizaje mientras que la segunda (datos de prueba) quedará sin usarse hasta la tercera etapa. En la fase de *aprendizaje* se usan los datos de entrenamiento para construir un modelo matemático capaz encontrar patrones de comportamiento con la información proveída. En la *evaluación*, se retoman los datos de prueba y se predice su comportamiento con el modelo de aprendizaje construido en la fase anterior para saber si el método funciona en datos no observados previamente. En caso de que el rendimiento del modelo no sea lo suficientemente bueno, se vuelve a entrenar, entrando en un proceso iterativo entre la fase de aprendizaje y la de evaluación. Cuando la métrica de evaluación en los datos de prueba es lo suficientemente buena, entonces se procede a la fase de *predicción*, en la cual ya puede usarse el algorimo para predecir el comportamiento de observaciones nunca antes vistas.

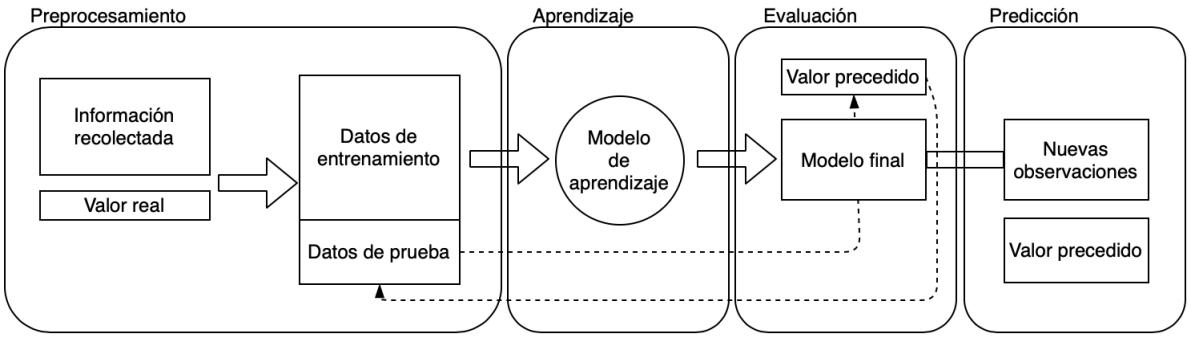


Figura 2.1: Diagrama de las cuatro fases que integran los modelos de aprendizaje automático supervisado

2.1. El problema de la estimación

Para construir modelos de aprendizaje automático, es necesario construir un marco de referencia (framework) con los conceptos y criterios necesarios para resolver el problema de la estimación de la variable de respuesta. Considerando el caso de una variable objetivo cuantitativa, sea $X \in \mathbb{R}^p$ un vector aleatorio de entrada y $Y \in \mathbb{R}$ la variable aleatoria de salida, con distribución conjunta $\mathbb{P}(X, Y)$. Se busca una función $f(X)$ para predecir Y dados los valores de la entrada X. Para esto, se requiere una función de pérdida[1] $L(Y, f(X))$ para penalizar el error de predicción como:

- El error cuadrático $(Y - f(X))^2$
- El error absoluto $|Y - f(X)|$
- La norma L_p $|Y - f(X)|^p$.

Si tomamos el error cuadrático (utilizado con mayor frecuencia por ser derivable en cualquier punto) como función de pérdida, llegamos a un criterio para seleccionar f,

$$EPE(f) = \mathbb{E}(Y - f(X))^2 = \int [y - f(x)]^2 \mathbb{P}(dx, dy), \tag{2.1}$$

el error de predicción esperado (cuadrático). Condicionando[2] sobre X, se puede escribir el EPE como

$$EPE(f) = \mathbb{E}_X \mathbb{E}_{Y|X}([Y - f(X)]^2 | X) \tag{2.2}$$

[1] que relaciona un evento con su respectivo costo asociado

[2] lo cual equivale a factorizar la densidad $\mathbb{P}(X, Y) = \mathbb{P}(Y|X)\mathbb{P}(X)$ donde $\mathbb{P}(Y|X) = \frac{\mathbb{P}(X,Y)}{\mathbb{P}(X)}$ y separar la integral doble.

y vemos que es suficiente con minimizar el EPE puntualmente:

$$f(x) = \arg\min_c \mathbb{E}_{Y|X}([Y-c]^2|X=x). \tag{2.3}$$

La solución es

$$f(x) = \mathbb{E}(Y|X=x), \tag{2.4}$$

la esperanza condicional. Por lo tanto, cuando la función de pérdida es el error cuadrático, la mejor predicción de Y en un punto $X=x$ es la media condicional y es, a su vez, representado por la suma de la varianza y el sesgo en el conjunto de entrenamiento τ:

$$\mathbb{E}(Y-f(X))^2 = var_\tau(f) + sesgo^2_\tau(f) \tag{2.5}$$

Por su parte, cuando la variable objetivo G es categórica (clasificación), se requiere una función de pérdida distinta para penalizar los errores de predicción. Esta función puede ser representada por una matriz $\mathbf{L}$ de dimensión $K \times K$, donde $K = |\mathcal{G}|$ y $\mathcal{G}$ es el conjunto de valores que puede tomar el estimador $\hat{G}$ de G. Cada entrada $L_{i,j}$ de la matriz es el precio de clasificar incorrectamente una observación de la clase $\mathcal{G}_i$ en la clase $\mathcal{G}_j$, por lo tanto, $L_{i,i} = 0$ y $L_{i,j} \geq 0 \ \forall \ i \neq j$. La función de pérdida más común en este tipo de modelos es la *Cero-Uno* donde cada entrada $L_{i,j}$ de la matriz es una función indicadora del error:

$$L_{i,j} = \begin{cases} 1 \text{ si } \mathcal{G}_i \neq \mathcal{G}_j, \\ 0 \text{ si } \mathcal{G}_i = \mathcal{G}_j. \end{cases} \tag{2.6}$$

Entonces se toma la esperanza del error con respecto a la distribución conjunta $\mathbb{P}(G,X)$ y se condiciona sobre X:

$$EPE = \mathbb{E}(L(G,\hat{G}(X))) = \mathbb{E}_X \sum_{k=1}^{K} L(\mathcal{G}_k, \hat{G}(X))\mathbb{P}(\mathcal{G}_k \mid X), \tag{2.7}$$

y se minimiza en cada punto para tener la función de estimación

$$\hat{G}(x) = \arg\min_{g\in\mathcal{G}} \sum_{k=1}^{K} L(\mathcal{G}_k, g)\mathbb{P}(\mathcal{G}_k \mid X=x) \tag{2.8}$$

que se simplifica con la función de pérdida 0-1, quedando como

$$\hat{G}(x) = \arg\min_{g\in\mathcal{G}}[1 - \mathbb{P}(g \mid X=x)] \tag{2.9}$$

o, simplemente,

$$\hat{G}(x) = \mathcal{G}_k \text{ si } \mathbb{P}(\mathcal{G}_k \mid X=x) = \max_{g\in\mathcal{G}} \mathbb{P}(g \mid X=x). \tag{2.10}$$

Esta solución es conocida como *clasificador Bayesiano* y nos dice que se clasificarán las observaciones en la clase más probable usando la distribución condicional $\mathbb{P}(G \mid X)$.

2.2. Métodos lineales de clasificación

Debido a que el predictor $\hat{G}$ de una observación toma valores en un conjunto discreto $\mathcal{G}$, es posible dividir el espacio p-dimensional de *inputs* en regiones donde cada una de ellas corresponde a una clase y las fronteras (o

límites de decisión) de esas regiones pueden ser funciones lineales (o suavizadas) como se muestra en la figura 2.2.

Hay varios métodos para encontrar las fronteras lineales en una clasificación; uno de ellos es ajustar una regresión lineal y clasificar según el signo de la evaluación de cada observación en la función. Ahora, suponiendo que hay K clases y el modelo lineal ajustado para la k-ésima clase siendo $\hat{f}_k(x) = \hat{\beta}_{k_0} + \hat{\beta}_k^T x$ con $\hat{\beta}_k$ el coeficiente de regresión. El límite de decisión entre la clase k y la clase l es el conjunto de puntos que cumplen $\hat{f}_k(x) = \hat{f}_l(x)$, esto es $\{x \mid (\hat{\beta}_{k0} - \hat{\beta}_{l0}) + (\hat{\beta}_k - \hat{\beta}_l)^T x = 0)\}$ un conjunto afín entre las dos clases.

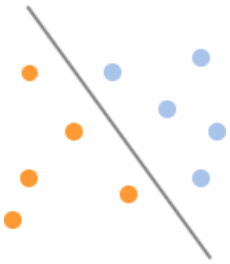


Figura 2.2: Clasificador lineal de dos clases (naranja y azul).

Los métodos lineales más populares son el análisis de discriminante lineal y la regresión logística lineal cuya diferencia principal es en la forma en que ajustan la función lineal en los datos de entrenamiento. Sin embargo, debido a que el presente trabajo solo utiliza el método de optimización del hiperplano separador (máquinas de soporte vectorial), unicamente se profundizará en la explicación del mismo.

2.2.1. Máquinas de Soporte Vectorial Lineal

Las máquinas de soporte vectorial (SVM por sus siglas en inglés) son un tipo de modelo de datos desarrollados por Vladimir Vapnik en 1996. En la clasificación binaria, el objetivo es encontrar el hiperplano que distinga mejor una clase de la otra y maximice su distancia al punto más cercano de cada una. Este método provee una única solución al problema de clasificación, encontrando un hiperplano separador y mejorando su desempeño mediante la maximización del margen entre las clases. La figura 2.4 muestra el funcionamiento del SVM en un problema de clasificación binaria.

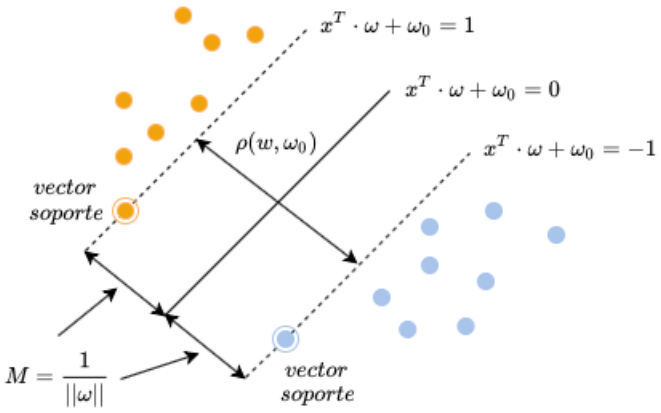


Figura 2.3: Clasificador de Máquina de Soporte Vectorial en el caso separable. El hiperplano separador es la línea sólida y las líneas punteadas representan los límites del margen $\rho(\omega, \omega_0)$ de ancho $\frac{2}{||w||}$

Sea el conjunto de datos de entrenamiento τ conformado por N pares $(x_1, y_1), \ldots, (x_N, y_N)$ con $x_i \in \mathbb{R}^p$ y $y_i \in \{-1, 1\}$. Se define un hiperplano como $\{x | f(x) = x^T\omega + \omega_0\}$ donde ω es un vector unitario ($||\omega|| = 1$). La regla de clasificación inducida por $f(x)$ es

$$G(x) = signo(x^T\omega + \omega_0) \tag{2.11}$$

Como las clases son separables, es posible encontrar una función $f(x) = x^T\omega + \omega_0$ con $y_i f(x_i) > 0 \forall i$. Por lo tanto, se puede encontrar el hiperplano que genere el margen más amplio entre los datos de la clase 1 y -1:

$$max\ M \tag{2.12}$$

sujeto a

$$y_i(x_i^T\omega + \omega_0) \geq M, i = 1, \ldots, N \tag{2.13}$$

donde M representa la distancia (margen) entre el hiperplano y cada clase. Equivalentemente, (como $M = \frac{1}{||w||}$) este problema puede resolverse como:

$$min\ ||\omega|| \tag{2.14}$$

sujeto a

$$y_i(x_i^T\omega + \omega_0) \geq 1, i = 1, \ldots, N, \tag{2.15}$$

donde se elimina la restricción de la norma de ω. Tenemos un problema de optimización convexo que, computacionalmente, es más conveniente expresar como

$$min\ \frac{||\omega||^2}{2} \tag{2.16}$$

sujeto a

$$y_i(x_i^T\omega + \omega_0) \geq 1. \tag{2.17}$$

Luego, la distancia ρ entre los hiperplanos definidos por $y_i(x_i^T\omega + \omega_0) = 1$ y $y_i(x_i^T\omega + \omega_0) = -1$ es vista como:

$$\begin{aligned}
\rho(\omega, \omega_0) &= \min_{\{x_i : y_i = 1\}} \frac{|x_i^T \cdot \omega + \omega_0|}{||\omega||} + \min_{\{x_j : y_j = -1\}} \frac{|x_j^T \cdot \omega + \omega_0|}{||\omega||} \\
&= \frac{1}{||\omega||}\left(\min_{\{x_i : y_i = 1\}} |x_i^T \cdot \omega + \omega_0| + \min_{\{x_j : y_j = -1\}} |x_j^T \cdot \omega + \omega_0|\right) \\
&= \frac{2}{||\omega||}.
\end{aligned} \tag{2.18}$$

Los vectores soporte son aquellos que cumplen con $\min_{\{x_i : y_i = 1\}} \frac{|x_i^T \cdot \omega + \omega_0|}{||\omega||}$ o $\min_{\{x_i : y_i = 1\}} \frac{|x_i^T \cdot \omega + \omega_0|}{||\omega||}$.

La restricción del problema es una función lineal y diferenciable. Se desea seleccionar ω y ω_0 de tal manera que se maximice el margen M. La resolución del problema es equivalente a minimizar la función primitiva de Lagrange (véase Apéndice A) con respecto a ω y ω_0, que es:

$$L_P = \frac{||\omega||^2}{2} - \sum_{i=1}^{N} \alpha_i [y_i(x_i^T \cdot \omega + \omega_0) - 1], \tag{2.19}$$

y que se minimiza con respecto a ω y ω_0. Luego, igualando la derivada a cero, se obtiene:

$$\omega = \sum_{i=1}^{N} \alpha_i y_i x_i, \tag{2.20}$$

$$0 = \sum_{i=1}^{N} \alpha_i y_i, \tag{2.21}$$

así como la restricción de $\alpha_i \geq 0 \ \forall i$. Sustituyendo en L_P, se obtiene la función objetivo dual Lagrangiana (Wolfe) que nos da un límite inferior de la función

$$L_D = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{k=1}^{N} \alpha_i \alpha_k y_i y_k x_i^T x_k \text{ sujeto a } \alpha_i \geq 0 \text{ y } \sum_{i=1}^{N} \alpha_i y_i = 0. \tag{2.22}$$

La solución es encontrada al maximizar L_D en el espacio N-dimensional positivo. Además, la solución satisface las condiciones de Karush-Kuhn-Tucker que son los requerimientos necesarios y suficientes para que la solución de un problema de programación matemática sea óptima.

De lo anterior, se tiene que:

- si $\alpha_i > 0$ entonces $y_i(x_i^T \omega + \omega_0) = 1$ o, en otras palabras, x_i está en el límite del margen
- si $y_i(x_i^T \omega + \omega_0) > 1$, x_i no está en el límite del margen, y $\alpha_i = 0$.

Finalmente, el vector solución ω está definido en términos de una combinación lineal de los *vectores soporte* x_i (en los que $\alpha_i > 0$) como:

$$\hat{\omega} = \sum_{i=1}^{N} \hat{\alpha}_i y_i x_i \tag{2.23}$$

Por su parte, $\hat{\omega}_0$ se obtiene resolviendo la ecuación anterior para cualquiera de los vectores soporte. Dadas las soluciones ω y ω_0, la función de decisión del clasificador se puede escribir como:

$$\hat{G}(x) = signo(x^T \hat{\omega} + \hat{\omega}_0). \tag{2.24}$$

Métodos kernel en las Máquinas de Soporte Vectorial

En los modelos de clasificación por el metodo de SVM, no siempre es posible separar linealmente las observaciones en el espacio original; por este motivo, existe un proceso llamado *kernelización* que consiste en mapear las observaciones del espacio original a un espacio de mayor dimensión en el que sí se pueden separar. Una vez que las funciones kernel $h_m(x), m = 1, \ldots, M$ son seleccionadas, el procedimiento es igual al descrito previamente solo que se usan las variables $h(x_i) = (h_1(x_i), \ldots, h_M(x_i))$, la función no lineal $\hat{f}(x) = h(x)^T \hat{\omega} + \hat{\omega}_0$ y el clasificador $\hat{G}(x) = signo(\hat{f}(x)$. En la figura 2.4, se muestra (a grandes razgos) la kernelización en la clasificación binaria.

De hecho, no es necesario especificar la transformación $h(x)$ sino que solo se requiere la función kernel:

$$K(x, x') = \langle h(x), h(x') \rangle \tag{2.25}$$

que calcula productos interiores en el espacio transformado. Algunos ejemplos de kernels son:

- Polinomial: $K(x, x') = (1 + \langle x, x' \rangle)^d$
- Red Neuronal: $K(x, x') = tanh(\kappa_1 \langle x, x' \rangle + \kappa_2)$.

Cabe destacar que el uso de funciones kernel no siempre dará mejores resultados que el clasificador lineal sobre el espacio original pero es necesario conocerla para comparar los procedimientos y seleccionar el que tenga mejor rendimiento.

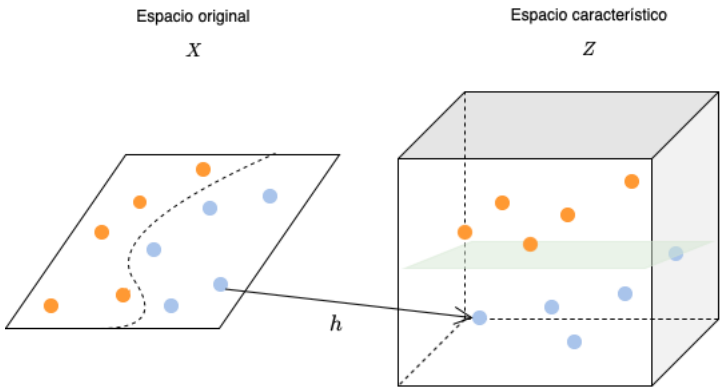


Figura 2.4: Aplicación de una función kernel h a las observaciones del espacio original al espacio característico donde son linealmente separables.

2.3. Evaluación y selección del modelo

La generalización de un método de aprendizaje se relaciona con su capacidad de predicción en datos de prueba. La evaluación de este desempeño es importante porque nos da una medida cuantitativa de la calidad del modelo elegido. Hay dos objetivos que se deben tomar en cuenta para la elección de modelos:

- **Selección**: evaluar el rendimiento de diferentes modelos para elegir al mejor.
- **Validación**: una vez elegido el mejor modelo, estimar el error de predicción en datos de prueba.

Si se tienen suficientes observaciones para entrenar el modelo, se divide la información en tres partes: el conjunto de entrenamiento, el de validación y el de prueba. El modelo se construye con el conjunto de entrenamiento; después se estima el error de predicción en el conjunto de validación (selección del modelo) y, finalmente, se evalua la capacidad de generalización en el conjunto de prueba.

A medida que se añaden más variables a un modelo, disminuye el sesgo de las predicciones y la varianza incrementa. Muy pocas variables producen un alto sesgo lo cual indica que hay subajuste (**underfitting**) y demasiadas variables conducen al incremento de la varianza que, a su vez, produce sobreajuste (*overfitting*) y mayor carga computacional. Los mejores estimadores logran encontrar un buen equilibrio entre sesgo y varianza, es decir, un balance entre la complejidad y el ajuste.

Existen varios métodos para estimar el riesgo de predicción R, uno de ellos es el error de entrenamiento que, dada una variable objetivo categórica G en $\mathcal{G} = 1, 2, \ldots, K$, un vector de *inputs* X y un modelo de probabilidades $p_k(X) = \mathbb{P}(G = k|X)$ que da una estimación $\hat{G}(X) = \arg\max_k \hat{p}_k(X)$ del conjunto de entrenamiento τ, permite calcular el riesgo como

$$\hat{R}_\tau(G, \hat{G}) = \sum_{i=1}^{N} (L(G, \hat{G})). \tag{2.26}$$

La desventaja de usar esta métrica es que representa una estimación sesgada hacia abajo del riesgo de predicción, de hecho,

$$Sesgo(\hat{R}_\tau) = \mathbb{E}(\hat{R}_\tau) - R = -2 \sum_{i=1} Cov(\hat{G}_i, G_i), \tag{2.27}$$

lo cual implica que cuando se ajusta un modelo, habrá una relación inversa entre el sesgo y la varianza. Por este motivo, existen otros métodos para estimar el riesgo de predicción en los que se busca encontrar el balance

entre el ajuste y la complejidad.

En los modelos de aprendizaje de máquina, la validación cruzada (*cross-validation*) es el método más usado para estimar el riesgo de predicción. Este método calcula el riesgo esperado en el conjunto de prueba $R = \mathbb{E}[L(G, \hat{G})]$ tomando diferentes conjunto de entrenamiento, lo cual garantiza que el modelo funcione similarmente en conjuntos de datos distintos. Esta validación cruzada puede hacerse de diferentes formas, siendo la más popular el *K-fold* que divide las observaciones en K partes iguales. En la k-ésima validación, se entrena el modelo con $k-1$ partes y se calcula el error de predicción en el conjunto k restante. Esto se repite para cada $k = 1, \ldots, K$ y luego se promedian los K errores para dar la métrica final

$$CV(\hat{f}) = \frac{1}{N} \sum_{i=1}^{N} L(y_i, \hat{f}^{-\mathcal{K}(i)}(x_i)), \tag{2.28}$$

donde $\hat{f}^{-\mathcal{K}(i)}$ es la función estimada sin la k-ésima parte de la información.

2.3.1. Métricas de clasificación binaria

Las métricas específicas para evaluar la calidad de los modelos de clasificación binarios se basan en la confusión del modelo al asignar una clase incorrecta a cada observación. Estas métricas se calculan a partir de una matríz C_{2x2} fig. 2.5 tal que $C_{i,j}$ es igual al número de observaciones de la clase i que fueron predecidas en la clase j. Esta matriz se conoce como **matriz de confusión** y de la cual se obtienen varios datos:

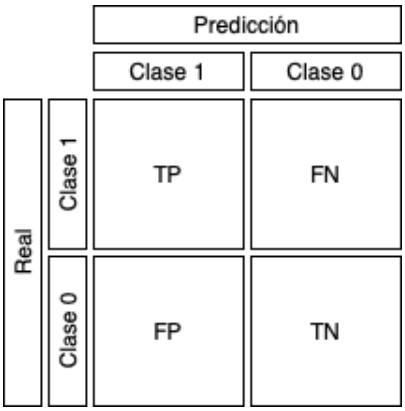


Figura 2.5: Matriz de confusión de un modelo de clasificación binario
TP: verdaderos positivos, FN: falsos negativos, FP: falsos positivos, TN: verdaderos negativos

- Exactitud: capacidad del modelo para indentificar correctamente ambas clases

$$A = \frac{TP + TN}{TP + FP + TN + FN}. \tag{2.29}$$

- Precisión: proporción de observaciones clasificados en la clase 1 frente a los que se predijeron en la clase 1

$$P = \frac{TP}{TP + FP}. \tag{2.30}$$

- Sensibilidad: proporción de observaciones de la clase 1 que fueron identificados correctamente

$$R = \frac{TP}{TP + FN}. \tag{2.31}$$

- F1: media armónica entre la precisión y la sensibilidad que combina ambas métricas en un solo indicador

$$2\frac{P * R}{P + R}. \tag{2.32}$$

A menudo, existe una relación inversa entre la precisión y la sensibilidad, donde es posible aumentar una a costa de la reducción de la otra; por eso, la métrica seleccionada para la evaluación de modelos dependerá de la finalidad del mismo.

Por otro lado, la proporción de cada clase en los datos de entrenamiento también es un factor para definir la metrica se usará para la validación del modelo. Por ejemplo, en un conjunto de datos en que las proporciones son 90 % vs 10 %, no es conveniente usar la exactitud ya que se podría estar identificando bien a todas las observaciones de la clase mayoritaria pero ninguna de la clase minoritaria.

La curva de detección de señales o **curva ROC** (Receiver Operating Characteristic), es un gráfico de un cuadrado unitario que diagnostica la habilidad de un clasificador binario para discriminar correctamente cada clase con diferentes límites de decis ión (o *thresholds*) basados en la probabilidad de pertenecer a una u otra clase. Por su parte, el área debajo de la curva o **AUC** (Area Under Curve) por sus siglas en inglés, mide el área por debajo de la curva ROC y es la forma de ver la probabilidad de que el modelo clasifique una observación positiva más alto que una negativa.

En la gráfica A de la figura 2.6, se muestran las funciones de densidad f_0 y f_1 de las clases 0 y 1 en un problema de clasificación binario. Si estas distribuciones se sobreponen en algún intervalo, entonces cualquier límite de decisión tendrá un error de clasificación FN y/o FP. La gráfica B, muestra la curva ROC donde cada punto (x, y) representa la tasa de falsos positivos ($\frac{FP}{FP+TN}$) contra la tasa de verdaderos positivos ($\frac{TP}{TP+FN}$) para cada límite de decisión de la gráfica A .

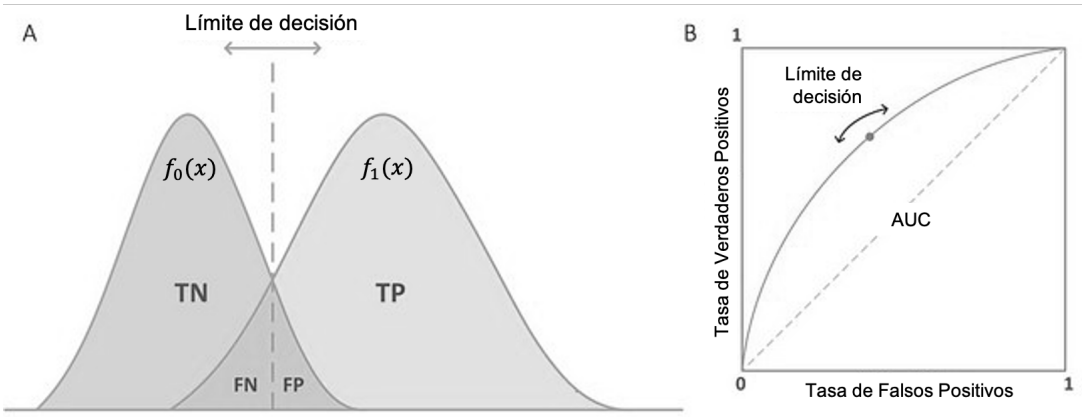


Figura 2.6: Para un problema de clasificación binario, el gráfico de la izquierda (A) muestra las distribuciones de probabilidad f de las clases 0 y 1. La figura de la derecha muestra la curva ROC formada a partir de diferentes límites de decisión en el gráfico A.

El mejor límite de decisión se situaría en la esquina superior izquierda del cuadrado unitario donde se dibuja la curva ROC, indicando que no hay falsos negativos ni falsos positivos, es decir, la clasificación perfecta. Por el contrario, una clasificación totalmente aleatoria daría un punto sobre la función *identidad* ($f(x) = x$), conocida también como *línea de no-discriminación*. De esta manera, los puntos por encima de la diagonal representan los buenos resultados mientras que los puntos debajo en ella indican que el modelo no sirven para diferenciar las clases.

Conocer el modelo a ajustar, las medidas de pérdida o riesgo y las métricas de selección y valiación, hacen posible generar modelos con alta predictibilidad.

Capítulo 3

Minería de texto

Al igual que el video, el audio y las imágenes, la información en forma de texto pertenece al conjunto de datos no estructurados, los cuales no tienen una organización identificable que pueda ser presentada en forma de renglones y columnas (tablas) como en los datos estructurados. Por este motivo, este tipo de datos sólo tienen valor hasta que se almacenan de manera organizada para poder analizarlos.

En la gran mayoría de los modelos de aprendizaje de máquina, no es posible usar datos no estructurados como *input* sino que requieren de variables numéricas para realizar los cálculos del algoritmo. Por consiguiente, se han desarrollado métodos para transformar este tipo de datos en vectores y así poder usar *Machine Learning* para modelarlos.

En Procesamiento de Lenguaje Natural, se le conoce como *documento* a cada uno de los textos que conforman la colección que se va a analizar y pueden ser opiniones, párrafos, capítulos, etc., mientras que el *corpus* es todo este conjunto de documentos. Por otro lado, dada la naturaleza del texto, se pueden encontrar signos de puntuación, caracteres especiales, números, mayúsculas, etc. que alteran la identificación de palabras con el mismo significado. Es necesario conocer la información con la que se cuenta: el tipo de dato, el horizonte de tiempo del análisis, las frecuencias y otras medidas estadísticas para interpretar correctamente los resultados.

3.1. Análisis descriptivo de los datos

Como primer paso en esta investigación que pretende explicar un fenómeno a través de datos, se debe profundizar en la descripción del comportamiento y la distribución de estos datos. Se tomaron 475,635 tuits publicados de 1º de octubre del 2018 al 30 de septiembre del 2019 que contuvieran información sobre el sector financiero y cuyos autores estuvieran relacionados con la política, economía y periodismo del país. La selección de las cuentas (autores) se basó en su popularidad y en la insignia azul[1] de Twitter. Algunas cuentas consultadas son **@lopezobrador_**, **@ricardomonreala**, **@ElFinanciero_Mx**, **@eleconomista**, **@Reforma**, **@lopezdoriga**, **@Adela_Micha**, **@aristeguicnn**, **@CarlosLoret**, **@Banxico**, **@BMVMercados**, entre otras.

La figura 3.1 muestra el volumen de publicaciones realizadas en Twitter durante el periodo de estudio. Podemos notar un decremento en el volumen de publicaciones durante el primer mes del 2019 y otro durante los meses de abril y mayo, lo cual es un indicio de variación estacional anual que podría impactar en la predictibilidad

[1] verificación de autenticidad de las cuentas de interés público

del modelo final por el bajo volumen de información en dichos periodos, por lo que se requiere tomar en cuenta dicho comportamiento.

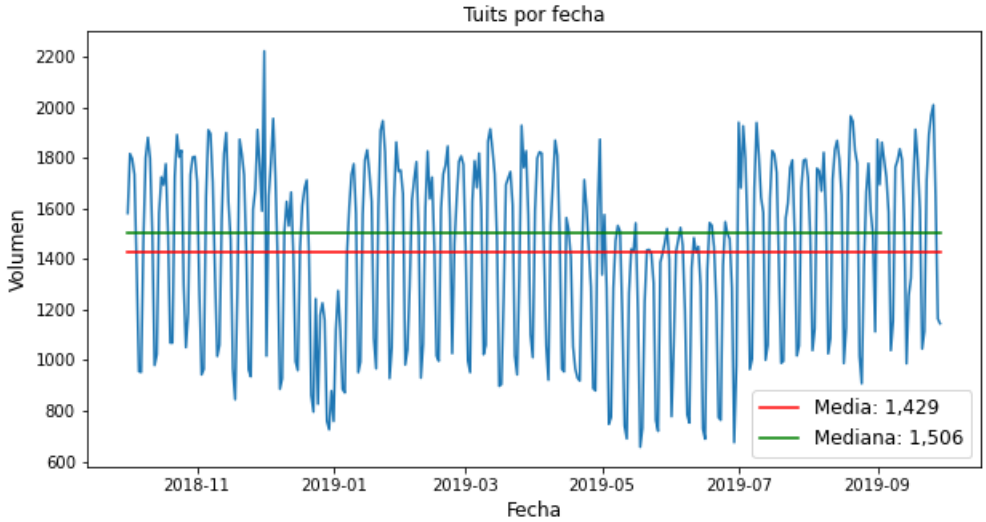


Figura 3.1: Publicaciones en Twitter del 01/10/2018 al 30/09/2019 de usuarios relacionados con el sector financiero

Otro punto a notar en la figura 3.1 es que el número de publicaciones tiene estacionalidad semanal, es decir, una variacion periódica que se repite en un intervalo de una semana. En la figura 3.2 podemos notar que el mayor volumen se encuentra en los días laborales que son los mismos que los días en que está abierto el mercado de valores.

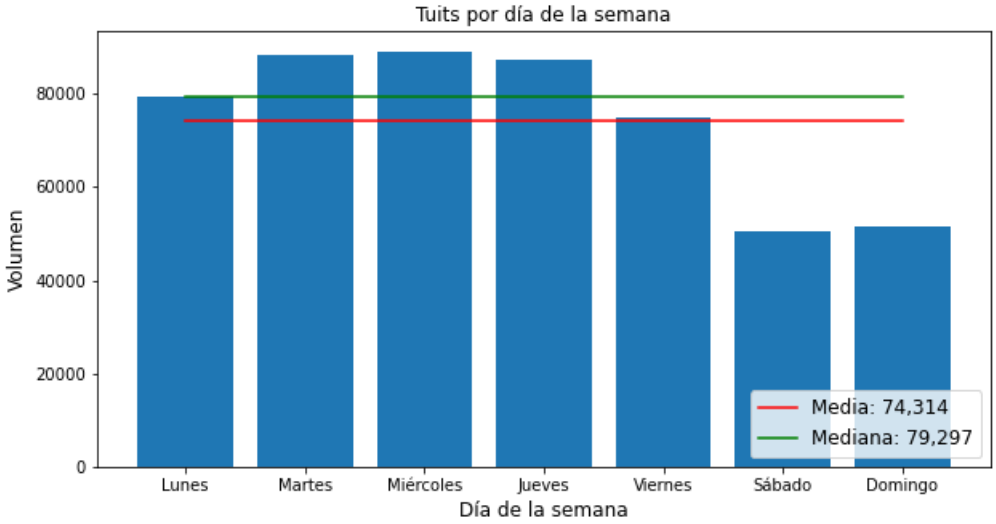


Figura 3.2: Número de publicaciones en Twitter por día de la semana

Así como es importante saber los días con mayor actividad en la red social, también se requiere conocer los

horarios de mayor concurrencia para poder identificar la relación que estos tienen con el del mercado de valores. En la figura 3.3, podemos observar un incremento significativo en la frecuencia de las publicaciones en el horario de las 7:00 a las 23:00 horas, alcanzando su punto máximo cerca de las 12:00 horas. Con esto concluimos que hay suficientes opiniones durante las horas en que opera la bolsa de valores.

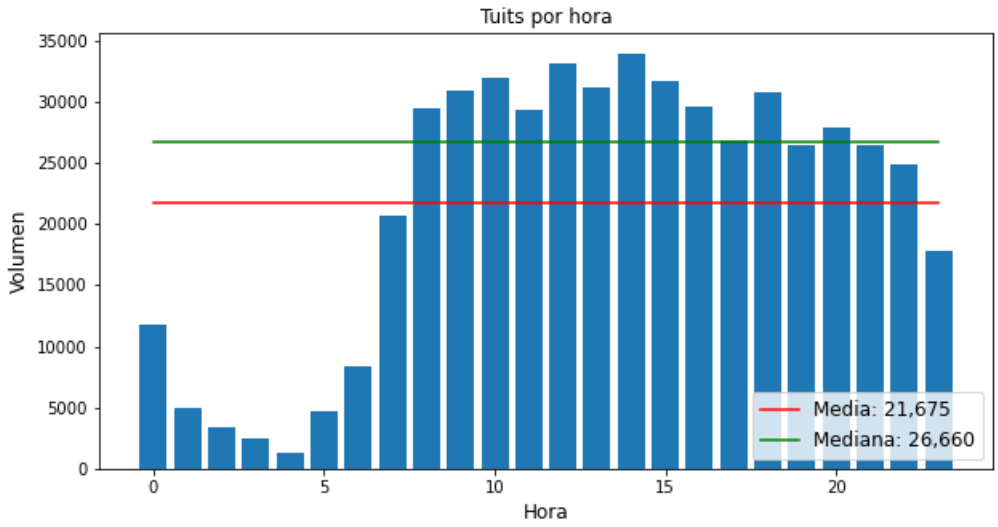


Figura 3.3: Número de publicaciones en Twitter por horario

Una vez conociendo el volumen de información y su comportamiento general durante el periodo de estudio, se procesa el texto para obtener las covariables que modelarán el comportamiento del mercado. En el diagrama 3.4 se muestra el flujo (*pipeline*) de normalización del texto: el cilindro representa el texto sin procesar, los triángulos representan fases de eliminación de ciertas palabras mientras que los rectángulos indican transformaciones al texto; los últimos dos pasos (segmentación y vectorización) solo se llevan a cabo cuando se desea transformar el texto en vectores para usarse directamente en modelos de Aprendizaje Automático mientras que los pasos anteriores se usan más para la extracción de variables de interés como conteos de palabras, frases etc. En las siguientes secciones se detalla cada paso del flujo aplicado a las opiniones de Twitter extraidas para este trabajo.

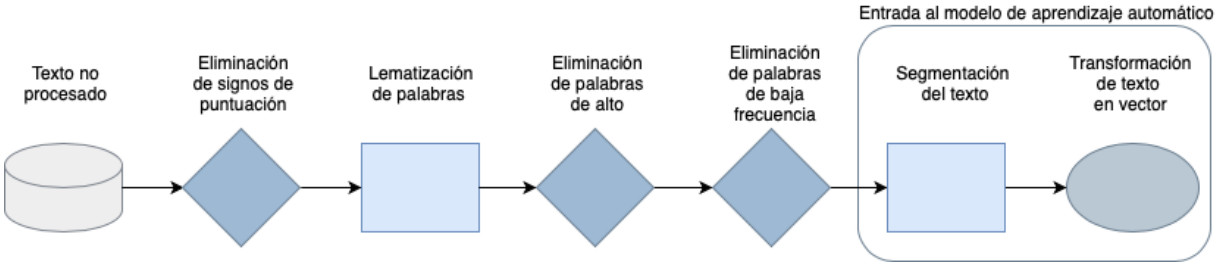


Figura 3.4: Flujo de transformación de texto en modelos de Aprendizaje Automático.

3.2. Normalización

La normalización o estandarización del texto consiste en unificarlo en un mismo formato para que se eviten errores al transformarlo en vectores. Este proceso puede incluir reglas como la eliminación de signos de puntuación, el reemplazo de palabras que se escriben igual pero tienen diferente significado, la eliminación de tildes, etc. Por ejemplo, las palabras "CREDITO", "crédito" y "Credito" tienen el mismo significado; la diferencia entre ellas es que la primera está escrita en mayúsculas y no tiene tilde; la segunda está escrita en minúsculas y tiene tilde y la tercera tiene la primera letra en mayúsculas, el resto en minúsculas y no tiene tilde. La normalización hará que estas tres palabras estén escritas de la misma manera para que el modelo les asigne el mismo valor.

3.2.1. Expresiones regulares

Una expresión regular o *regex* es una secuencia de caracteres que conforma un patrón de búsqueda en texto y está constituida por metacaracteres (caracteres con un significado especial) y caracteres regulares (que tienen un significado literal) que dan la posibilidad de buscar o reconocer cadenas de texto de manera flexible.

Los metacaracteres incluyen operadores lógicos como la unión ("|"), la agrupación ("()"), el número de ocurrencias del patrón ("?", "*", "+") y los comodines ("."), que facilitan la búsqueda de coincidencias en el texto. Un ejemplo de expresión regular es la secuencia "s.*" que coincidirá con el conjunto de palabras que comiencen con la letra "s" seguida por cero o más caracteres: "s", "si", "sal" y "señor".

Para la realización del presente trabajo y con ayuda de expresiones regulares, se erigieron las siguientes reglas para la normalización de los textos:

1. Todas las palabras están escritas en minúsculas
2. Todas las palabras están escritas sin tilde
3. Eliminar los números
4. Eliminar los signos de puntuación tales como "?", "!", ".", "%", etc.
5. Eliminar las referencias a sitios web que inicien con "www.", "http" o "https"
6. Eliminar las referencias a imágenes de Twitter que inician con "pic.twitter"
7. Eliminar los espacios seguidos

Como se ha mencionado antes, la normalización del texto depende del caso de estudio en cuestión ya que podrían existir caracteres especiales que impacten en el resultado del análisis. Para este caso de estudio, los caracteres especiales "#", "$" y "@" no serán eliminados ya que tienen un significado particular en el contexto de Twitter.

3.2.2. Lematización

La *lematización* es el proceso mediante el cual las palabras de un texto que pertenecen a un mismo paradigma flexivo o derivativo son llevadas a una forma normal que representa a toda la clase. Esta forma normal, llamada lema, es típicamente la palabra utilizada como entrada en los diccionarios de la lengua: el infinitivo para las conjugaciones verbales, el masculino singular para adjetivos, etc.

En Procesamiento de Lenguaje Natural, la lematización es muy importante ya que permite agrupar palabras en un solo vocablo manteniendo su significado y reduciendo el número de variables en los modelos de aprendizaje de máquina en los que se vectoriza el texto. Por ejemplo, las palabras "financiera", "financiero", "finaniceras", "financieros" serán reemplazadas por "financiero"; o las palabras "económica", "económicas", "económico" y "económicos" serán reemplazadas por "económico", con lo cual se consigue conservar el significado y reducir el vocabulario.

3.3. Palabras de alto

En computación, se les conoce como palabras de alto o *stopwords* al conjunto de palabras que sirven para modificar o acompañar a otras pero cuyo significado no es relevante para la frase. Generalmente, este conjunto de palabras contiene artículos, pronombres, preposiciones, adverbios e incluso algunos verbos que se eliminan del análisis ya que son demasiado frecuentes en el *corpus* y, como veremos más adelante, podrían adquirir una falsa importancia en el proceso de vectorización. En la minería de texto, nos interesa saber cuales son las palabras que generan más impacto y una de las formas de identificarlas es mediante su frecuencia en el texto, sin embargo, es claro que una stopword tendrá una frecuencia mayor por lo que se debe tener cuidado con el tratamiento de ellas para que no alteren el resultado de los modelos.

A pesar de su alta frecuencia, hay palabras que no se eliminan debido a que sirven para modificar la dirección semántica del texto. Por ejemplo, la palabra "no" en "No va a subir el precio" altera completamente la oración y si la palabra "no" no estuviera, cambiaría la dirección de la opinión.

Por ejemplo, en el texto "Aumentamos pronósticos de UT, PO a $51 de BBajio porque esperamos que el precio de la acción aumente y creemos que los préstamos a empresas son el nicho con mayor potencial de crecimiento en México donde BBajio es el jugador más especializado del segmento". La palabra que más se repite es "de" con una frecuencia de 4 veces, sin embargo, esta palabra no es capaz de brindarnos información sobre el tema del que se habla. En cambio, la palabra "BBajio" aparece dos veces y nos ayuda a saber que el tema principal de la opinión es sobre la empresa Banco del Bajío. Por tal motivo, la palabra "de" es catalogada como stopword.

La figura 3.5, muestra un comparativo entre las palabras más frecuentes en el texto original y después de eliminar las palabras de alto. En el texto original, de las diez parabras más frecuentes, ocho son palabras de alto; mientras que en el texto procesado, notamos que las diez palabras tienen relación con el ámbito financiero por lo que serán mas valiosas para explicar el fenómeno de cambio en el mercado de valores.

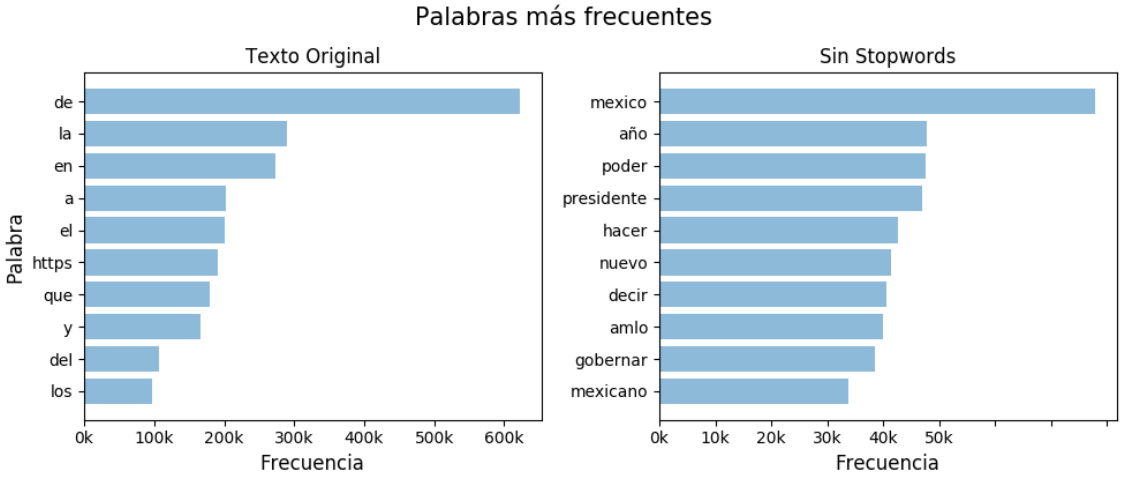


Figura 3.5: Comparativo de las diez palabras más frecuentes en el texto original y en el texto sin stopwords.

3.4. Palabras de baja frecuencia

Así como existen *stopwords* cuya particularidad es su alta frecuencia en el texto, también existen palabras de baja frecuencia; estas palabras se suelen eliminar cuando se hace minería de texto debido a que la gran mayoría de ellas son errores ortográficos (*typos*) o se refieren a un tema fuera del campo semántico que se está

abordando en el caso de estudio.

En este *corpus*, se encontraron 27,522 palabras que se repiten solo una vez. Entre estas palabras encontramos el listado 3.1.

Abuelo	Caliente	Econonmía	Buzo
Deportivo	Futbol	Gemelos	Obsequio
Tela	Hadoop	Odiado	Untable
Pandas	Xochiaca	Músicos	Rangel
Yolanda	Nopales	Servilletas	Zumba

Tabla 3.1: Extracto de palabras menos frecuentes en el texto

Las palabras de la tabla 3.1 no tienen relación con el campo semántico de las finanzas, representan faltas de ortografía o son nombres propios que no tiene sentido agregar al análisis; por ejemplo, la palabra "econonmía" notoriamente quiso decir "economía" pero que no fue añadida en la frecuencia de la palabra en cuestión por esa falta de ortografía.

Una vez hecho el análisis, las palabras en cuestión se eliminan para focalizar el análisis en las palabras que tienen más sentido de acuerdo al caso de estudio y también para eliminar el "ruido" al hacer los cálculos y así optimizar los recursos computacionales.

3.5. Obtención de variables

Existe información cuantitativa básica que puede obtenerse de cada tuit sin tener que recurrir a la vectorización del texto como son:

- Número de palabras
- Número de caracteres
- Número de palabras de alto
- Largo promedio de las palabras

El cálculo de estas variables es fácil y útil en el modelado de la dirección del precio. Por ejemplo, intuitivamente podemos decir que una mala opinión suele tener más palabras que una buena ya que es necesario justificar el punto de vista.

Por su parte, existen otras variables que dependen del análisis y la fuente de información para tener un impacto en el análisis. Por ejemplo, la red social Twitter usa caracteres especiales que tienen un significado particular en las publicaciones. El *hashtag* ("#"), se usa para destacar una palabra o frase sin espacios y tiene la finalidad de encontrar fácilmente opiniones que hablen sobre el tema que dicha palabra o frase. El *cashtag* ("$"), una etiqueta muy usada por la comunidad financiera de Twitter, permite generar y buscar información bursátil de las compañías usando las siglas con las que cotizan en la bolsa de valores. Finalmente, el símbolo "@" se usa para mecionar o "etiquetar" usuarios en los tuits.

De esta manera, se agregaron cinco variables más para predecir la dirección del precio, que solo tienen valor si se usa la fuente de información de Twitter y que son:

- Número de *hashtags*
- Número de *cashtags*

- Número de menciones
- Número de *retweets*
- Número de respuestas

Finalmente, se han extraído nueve variables numéricas sin tener que transformar el texto, sin embargo, se requiere del análisis de estas variables para saber si la distribución de los datos es adecuada para integrarlas a un modelo.

3.6. Análisis univariado de los datos

La medición y descripción de las variables construidas a partir del texto son parte del análisis exploratorio de los datos que ayuda a conocer sus principales características y saber si pueden dar predictibilidad al modelo. En las gráficas de distribución se reflejan los comportamientos de las variables que pretenden explicar el cambio en el precio y obtener las primeras hipótesis sobre el fenómeno que se desea explicar.

Definición 3.1 *Sea X una variable aleatoria con N observaciones, media $\bar{x}$ y desviación estándar σ, el coeficiente de asimetría S^3 mide el grado de asimetría de la distribución de X respecto a $\bar{x}$ y se calcula como*

$$S^3 = \frac{\sum_{i=1}^{N}(x_i - \bar{x})^3}{N\sigma^3}. \tag{3.1}$$

Además, se dice que la distribución de X es simétrica si $S^3 = 0$, asimétrica por la derecha si $S^3 > 0$ y asimétrica por la izquierda si $S^3 < 0$.

La figura 3.6 muestra la distribución del **número de palabras** en cada publicación. Podemos observar que la gran mayoría de los tuits tienen entre 5 y 50 palabras y que la media ronda las 25 palabras. Como el coeficiente de asimetría es 0.8872, esta variable tiene una "cola" más larga del lado derecho de la media. También se observan valores atípicos con hasta 200 palabras por tuit que deben eliminarse debido a que no representan el comportamiento típico de la variable.

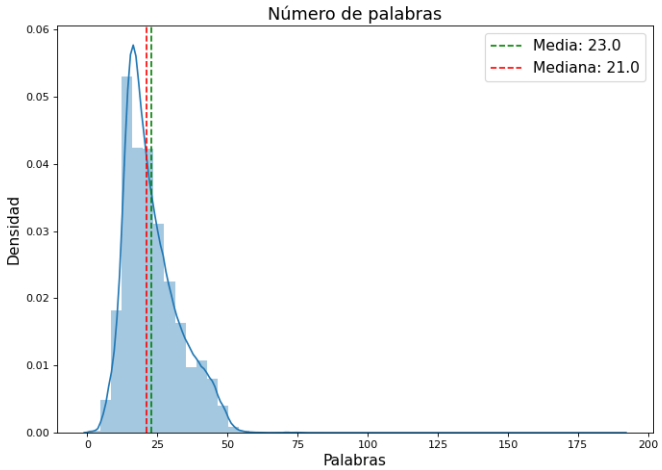


Figura 3.6: Distribución del número de palabras contenidas en las publicaciones con sus respectivas medidas de tendencia central.

En la figura 3.7 podemos observar la distribución del **número de caracteres** contenidos en cada publicación. La red social Twitter tiene un máximo 280 caracteres por tuit, sin embargo, cuando la publicación contiene una referencia a un sitio web (*link*), se puede exceder esa cantidad. El gran volumen de tuits se concentra entre los 30 y los 300 caracteres, no obstante, se observan valores atípicos que alcanzan los 700 caracteres pues incluyen *links* largos. Además, a pesar de que la media y la mediana están cerca, la distribución no es simetrica puesto que su coeficiente de asimetría es de 0.6556; la frecuencia aumenta aceleradamene hasta alcanzar los 120 y después cae más suavemente provocando que la distribución no luzca acampanada.

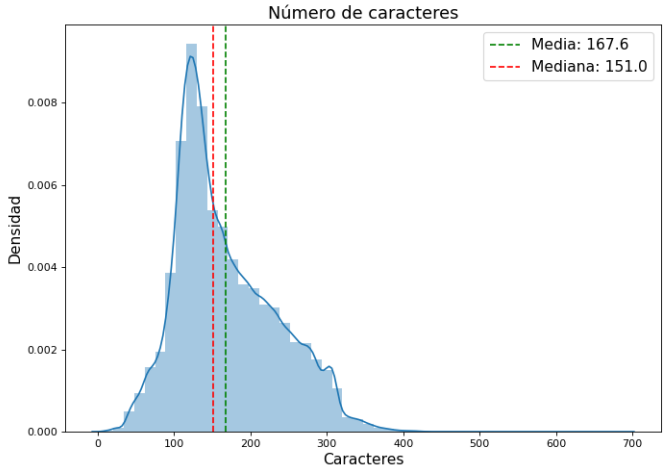


Figura 3.7: Distribución del número de caracteres en las publicaciones con sus respectivas medidas de tendencia central y una aproximación a su función de densidad.

El **número de *stopwords*** es relevante ya que pueden existir publicaciones que hagan un uso excesivo de estas palabras y, por lo tanto, pierden peso al tratar de explicar un fenómeno. En la figura 3.8 puede observarse que el número de este tipo de palabras está entre 1 y 30 en las publicaciones siendo 4 la frecuencia más alta de estas palabras. La distribución presenta un coeficiente de asimetría de 1.0046 lo que implica un sesgo a la derecha de la media.

Una hipótesis de este análisis es que en los días malos para el mercado de valores se escriban palabras más largas que en los días buenos. Por esto, la variable del **largo promedio de palabras** es interesante de calcular. La figura 3.9 muestra que la densidad de esta variable tiene un ligero sesgo a la derecha puesto que su coeficiente de asimetría es de 1.6133 y su valor máximo es de 20 caracteres promedio por palabra.

Como algunas de las variables construidas sobre el contexto de Twitter tienen menor variabilidad y no todas las publicaciones contienen estas variables se analizan sus características con gráficos de caja-brazos (*boxplots*). En la figura 3.10 podemos visualizar las características principales de las cinco variables construidas a partir de los caracteres especiales de Twitter (“#”, “$” y “@”) y la interacción de los usuarios en la red social. En las tres gráficas de izquieda a derecha se observa que, de los tuits que tenían alguno de estos caracteres, la gran mayoría solo tenían uno o dos. Sin embargo, podemos notar que tienen un sezgo largo, lo que significa que hay publicaciones con más de diez de estos caracteres. También se nota del diagrama que el **número de cashtags** tiene un sesgo más largo que las otras dos variables, lo cual provoca que el **número promedio de *cashtags*** sea más alto que la mediana.

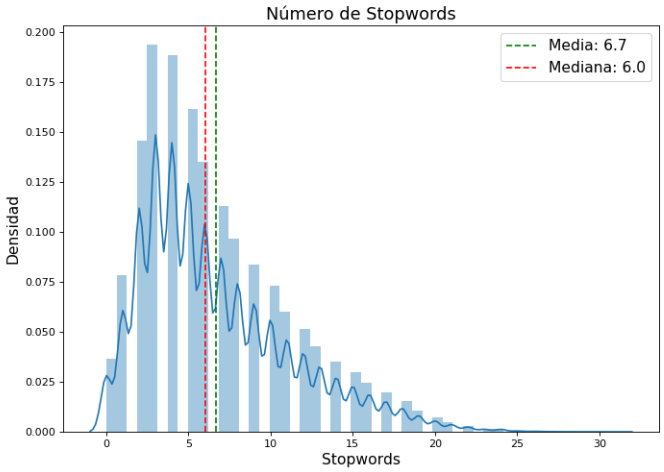


Figura 3.8: Distribución del **número de caracteres** en las publicaciones con sus respectivas medidas de tendencia central y una aproximación a su función de densidad.

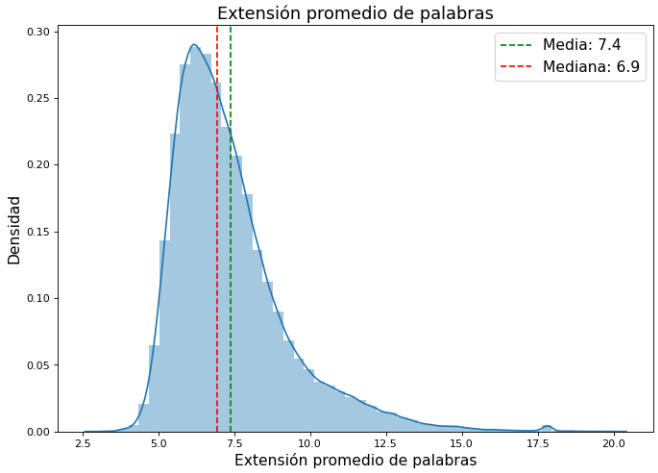


Figura 3.9: Distribución del **largo promedio de las palabras** de las publicaciones. Se incluyen las principales medidas de tendencia central.

En cuanto a las dos variables relacionadas a la interacción de los usuarios (**número de *retweets*** y **número de respuestas**), podemos notar que el 75 % de los tuits tienen un volumen de publicaciones inferior a 10; sin embargo, los valores extremos son notablemente altos lo que significa que hay usuarios muy influyentes que generan interés en otros usuarios.

Habiendo explotado el texto cualitativamente y conociendo sus principales características se procede a transformarlo en vectores para que se pueda obtener la dirección semántica de cada publicación mediante el análisis de sentimientos.

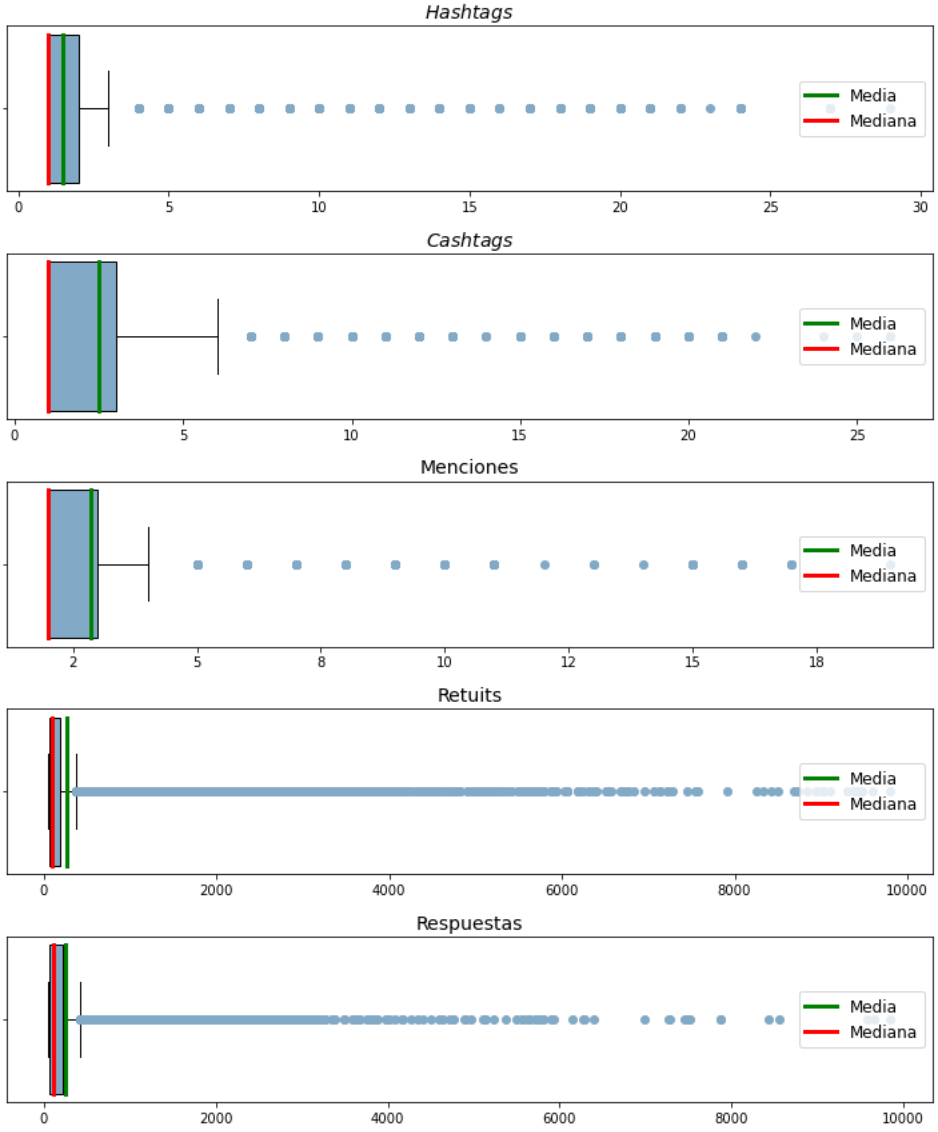


Figura 3.10: Diagramas de caja-brazos del **número de *hashtags***, **número de *cashtags***, **número de menciones**, **número de retuits** y **número de respuestas** respectivamente de todas las publicaciones.

3.7. Sentimiento de la opinión

Además de las variables cuantitativas que se obtienen en la minería de texto, también es posible y útil asignarle un sentimiento a la opinión. Este sentimiento representa la posición del autor respecto al tema del que está hablando y contribuye a indentificar la posición de los inversionistas frente a la situación del mercado.

El sentimiento de una opinión puede ser positivo, negativo o neutral. Es positiva cuando el autor está de acuerdo con lo que escribe; negativa cuando está en desacuerdo y neutral cuando no toma una postura respecto al tema. Para este caso de estudio se hace la suposición de que la postura de una persona al publicar en Twitter sobre algún tema financiero repercute directamente en la manera en que invierte su dinero, por lo tanto, influye en el movimiento del precio de las acciones.

3.7.1. Conversión del texto en vector

Como se ha mencionado antes, la información no estructurada no puede ser procesada de manera directa. Así que se hace la conversión del texto en vectores (*vectorization*) para poder usarlo en modelos matemáticos.

La separación del texto en entidades llamadas *tokens* que pueden ser caracteres o palabras que se encuentran entre un espacio o signo de puntuación se conoce como *tokenization*. Estas entidades pueden tener uno o varios elementos y a partir de ellos se asigna el nombre. Por ejemplo, los unigramas son *tokens* formados por una palabra o carácter, mientras que los bigramas son *tokens* formados por dos palabras o caracteres y así sucesivamente.

Un ejemplo de un trigrama es el conjunto de palabras "Ciudad de México" que, si bien cada palabra tiene un significado, las tres juntas representan un lugar dentro de México y sería más preciso que un modelo tomara en cuenta la frecuencia de las tres palabras juntas que una por una dentro del proceso de vectorización.

Dependiendo del caso de estudio se definen las particularidades de la transformación del texto y se utilizan técnicas como *Bag of Words*, *TF-IDF* y *Word2Vec* para darle una interpretación numérica. En este trabajo se usó el método Frecuencia de Término – Frecuencia Inversa de Documento ($TF-IDF$ por sus siglas en inglés).

El método $TF - IDF$ consiste en asignar la importancia de las palabras en un *corpus* dependiendo de su frecuencia en el mismo. Está conformado por dos conceptos: la Frecuencia del Término (TF) y la Frecuencia Inversa del Documento (IDF). Como su nombre lo indica, $TF_{d,p}$ indica la frecuencia de una palabra p en un documento d. Siendo n_d el número de total de palabras en el documento d y $F_{d,p}$ la frecuencia de p en el mismo, entonces

$$TF_{p,d} = \frac{F_{d,p}}{n_d}. \tag{3.2}$$

El segundo término es la frecuencia inversa del documento IDF_p que sirve para penalizar las palabras que aparecen muy frecuentemente y se calcula como

$$IDF_p = \log(\frac{D}{N_p}) \tag{3.3}$$

donde D es el número de documentos en el *corpus* y N_p es el número de documentos en los que aparece el término p.

Definición 3.2 *Sea $\boldsymbol{D}_1$ una matriz de tamaño $n \times m$ y sea $\eta(\boldsymbol{D}_1)$ el número de entradas distintas de cero de $\boldsymbol{D}_1$. Se dice que $\boldsymbol{D}_1$ es dispersa si*

$$\frac{\eta(\boldsymbol{D}_1)}{n \times m} < 0.5 \tag{3.4}$$

Definición 3.3 *Sea $\boldsymbol{D}_2$ una matriz de tamaño $n \times m$ y sea $\eta(\boldsymbol{D}_2)$ el número de entradas distintas cero de $\boldsymbol{D}_2$. Se dice que $\boldsymbol{D}_2$ es densa si*

$$\frac{\eta(\boldsymbol{D}_2)}{n \times m} \geq 0.5 \tag{3.5}$$

El valor $TF\text{-}IDF_{p,d}$ de p en d queda como

$$TFIDF_{p,d} = TF_{p,d} \times IDF_p \tag{3.6}$$

y, siendo P el número total de palabras distintas en el *corpus*, se tiene como resultado una matriz dispersa M_{DXP} tal que

$$M_{DXP} = \begin{pmatrix} TFIDF_{d_1,p_1} & \cdots & TFIDF_{d_1,p_P} \\ \vdots & \ddots & \vdots \\ TFIDF_{d_D,p_1} & \cdots & TFIDF_{d_D,p_P} \end{pmatrix}. \tag{3.7}$$

Finalmente, esta matriz que contiene la información del texto transformada en números se usa como *input* para el modelo de sentimientos. En la figura 3.11 se observa una muestra de la matriz dispersa obtenida de transformar los textos en vectores donde cada renglón representa un tuit y cada columna representa un *token*; la mayoría de las entradas de la matriz son 0 debido a que el vocabulario de los tuits es muy vasto, sin embargo, sí podemos notar que hay palabras cuya frecuencia es mucho mayor que otras y cuya relevancia es importante para el caso de estudio del presente trabajo.

Tweet Normalizado	economico	editorial	educacion	eeuu
«creo educacion libertar elegir centro»	0	0	0.546059	0
gobernar suspense marianista ratista entornar economico rato lehman brothers	0.600301	0	0	0
marianorajoy destruir educacion no interesar tener poblar manipulable no ahora entender	0	0	0.359271	0
leer mucho gente defensor legitimo educacion religioso decir ahora acabar escuela	0	0	0.477396	0
reportar senado republica aprobar ley general educacion detalle beltrandelrio	0	0	0.397315	0

Figura 3.11: Muestra de la matriz dispersa de dimension $(n \times m)$ que se obtiene de la transformación del texto a vectores a través del método $TF - IDF$.

Para optimizar el recurso computacional se debe hacer una selección de los *tokens* más importantes, es decir, aquellos que contengan la mayor cantidad de información predictiva sin tener que incluirlos todos en el modelo. Los dos criterios para la selección de *tokens* se basan en la relación que tengan con el campo semántico abordado en este trabajo: finanzas, economía y política así como la utilización en contextos positivos y negativos. Estos criterios contribuyen al aprendizaje del modelo para la correcta clasificación de las observaciones.

3.7.2. Clasificador de sentimientos

A pesar de tener un margen de error, asignar el sentimiento a una opinión de manera automática ahorra tiempo al clasificar los datos lo cual representa una gran ventaja cuando se tienen grandes cantidades de datos, como en este caso que se tienen más de 500 mil observaciones.

Para saber la dirección connotativa del *corpus* se requiere información previamente clasificada con la que pueda entrenarse un modelo de clasificación supervisado, sin embargo, las opiniones obtenidas no cuentan con dicha clasificación. Dada esta situación, se plantean dos opciones para tener un conjunto de entrenamiento para el modelo de sentimientos:

1. Clasificar manualmente una proporción de los datos
2. Tomar información previamente clasificada de otra fuente

Para resolver el dilema se hizo una combinación de las dos opciones, es decir, se clasificaron aleatoria y manualmente 10,000 opiniones y se integraron a una base de datos llamada TASS publicada por la Sociedad Española para el Procesamiento del Lenguaje Natural (SEPLN) que contiene tuits en español clasificados. Con dichas acciones se llegó a tener 50,000 observaciones para el entrenamiento del modelo de sentimientos.

En la figura 3.12 se observa que el conjunto de datos de entrenamiento, incluyendo los tuits clasificados manualmente, cuenta con 57 % de observaciones positivas y 43 % negativas. Tener el conjunto de entrenamiento balanceado es una gran ventaja al ajustar un modelo ya que las métricas de desempeño representarán efectivamente la capacidad del modelo para identificar cualquiera de las dos clases. Por el contrario, si se tuviera una clasificación binaria desbalanceada, por ejemplo 90 % vs 10 %, la **exactitud** no sería útil ya que un valor muy alto de esta métrica podría significar una buena clasificación de la clase mayoritaria y un muy mal desempeño

en la clase minoritaria.

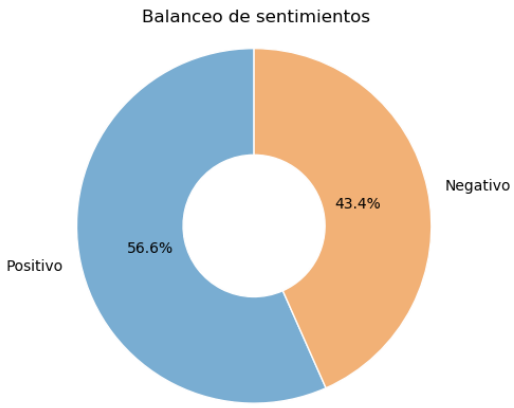


Figura 3.12: Proporción de registros en el conjunto de entrenamiento donde se observa que el 56.6 % corresponde a la categoría "positivo", mientras que el 43.4 % pertenecen a la categoría "negativo"

Por su parte, una buena diferenciación descriptiva entre las clases nos indica que el modelo va a lograr encontrar patrones para diferenciarlas. En la figura 3.13 se puede observar que los 1-gramas con el valor $TF\text{-}IDF$ más alto en ambas clases ("positiva" y "negativa"). Notamos que dichos 1-gramas son diferentes en la clase "positiva" y "negativa". Por ejemplo, la palabra "no" es el 1-grama con el valor más alto en la clase "negativa" lo cual es completamente intuitivo; esto nos da un buen indicio de que el modelo encontrará la información suficiente para clasificar correctamente.

Sentimiento Positivo

n-grama	Σ tf*idf
bueno	941.81
gracia	828.56
dia	692.61
no	533.54
ir	524.67
hoy	521.44
nuevo	506.25
hacer	436.18
gran	415.16
mejor	397.91

Sentimiento Negativo

n-grama	Σ tf*idf
no	1196.44
decir	480.56
politico	439.64
presidente	414.68
hacer	395.96
partir	388.59
gobernar	374.18
si	370.35
ir	325.10
poder	323.53

Figura 3.13: Comparativo de los diez 1-gramas (palabras) con el valor TF-IDF más grande tanto en la clase "positiva" como en la clase "negativa"

Una vez sabiendo que el conjunto de datos está balanceado y que los n-gramas más altos son distintos entre las clases, se dividieron las 50,000 observaciones en tres conjuntos: 80 % para el entrenamiento/validación y 20 % para el *test* en datos no antes vistos, garantizando que la proporción de la variable de respuesta se mantuviera igual en ambos conjuntos.

Como se vió en el capítulo 2, la validación cruzada garantiza que el modelo funcione similarmente en conjuntos de datos distintos, por esta razón se usó este procedimiento para validar el modelo con tres conjuntos evaluados en distintos periodos, obteniendo las métricas de la figura 3.14. Podemos notar que no hay mucha diferencia entre los resultados obtenidos en el conjunto de entrenamiento y en el de validación, además, la desviación es pequeña, por lo tanto, el modelo está generalizando correctamente. Por otro lado, las métricas son buenas tanto en el conjunto de entrenamiento como en el de validación lo cual indica que el clasificador funciona correctamente en datos no vistos.

```
------------------------------------------------------------------------------
SVC(C=1.0, break_ties=False, cache_size=200, class_weight=None, coef0=0.0,
    decision_function_shape='ovr', degree=3, gamma='scale', kernel='linear',
    max_iter=-1, probability=True, random_state=None, shrinking=True, tol=0.001,
    verbose=False)
------------------------------------------------------------------------------
test_accuracy  : 0.858 (+/- 0.007)
train_accuracy  : 0.882 (+/- 0.003)
test_precision  : 0.873 (+/- 0.008)
train_precision  : 0.896 (+/- 0.002)
test_recall  : 0.878 (+/- 0.005)
train_recall  : 0.896 (+/- 0.004)
test_f1  : 0.875 (+/- 0.006)
train_f1  : 0.896 (+/- 0.003)
test_roc_auc  : 0.932 (+/- 0.004)
train_roc_auc  : 0.948 (+/- 0.001)
test_neg_log_loss  : -0.332 (+/- 0.011)
train_neg_log_loss  : -0.292 (+/- 0.004)
```

Figura 3.14: Principales métricas obtenidas de la validación cruzada en el entrenamiento del modelo de sentimientos sobre el 53.3 % de los datos

De igual manera, se calcula la curva ROC para cada validacion; la gráfica 3.15 muestra estas curvas sobrepuestas así como la media de las mismas (línea azul). Se nota que los resultados son estables, es decir, el modelo es insesgado ante cambios en el conjunto de entrenamiento, además, el AUC es el mismo en cada caso lo cual nos da otro indicio de que el modelo funciona correctamente.

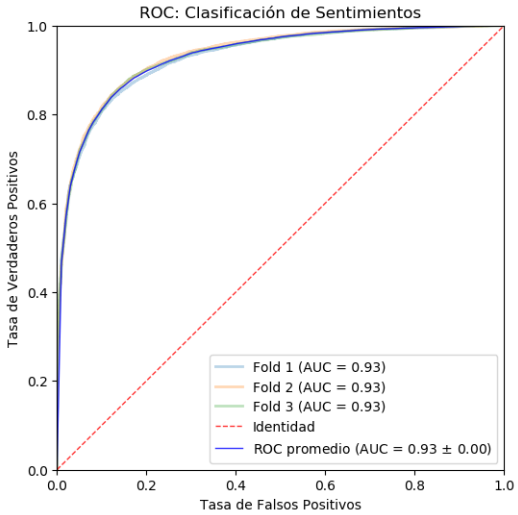


Figura 3.15: Curvas ROC de los conjuntos de prueba de las tres validaciones cruzadas con su respectivo AUC. La función identidad (línea roja) se muestra para compararse con las curvas ROC

Como el desempeño del modelo tiene buenas metricas de clasificación podemos decir que tiene la capacidad de identificar el sentimiento de un tuit; por lo tanto, podemos utilizarlo para clasificar el 100 % de las opiniones y usar el resultado como *input* para el modelo final.

3.8. Relación de las variables predictivas

Una vez teniendo una etiqueta con la dirección connotativa de cada publicacion es interesante conocer la relación que existe entre todas las variables explicativas que se construyeron mediante la minería del texto. Este proceso permite seleccionar las variables que van a contribuir más a la calidad del resultado final y eliminar aquellas que son irrelevantes y pueden disminuir el rendimiento del modelo.

La figura 3.16 también conocida como *pairplot* muestra la relación entre cada par de variables construidas a partir del texto (sin tomar las que se construyeron sobre el contexto de Twitter). Se tiene una matriz de gráficas donde la diagonal representa la distribución de los datos según cada clase (sentimiento) para cada variable y el resto de las gráficas representa una variable en función de otra.

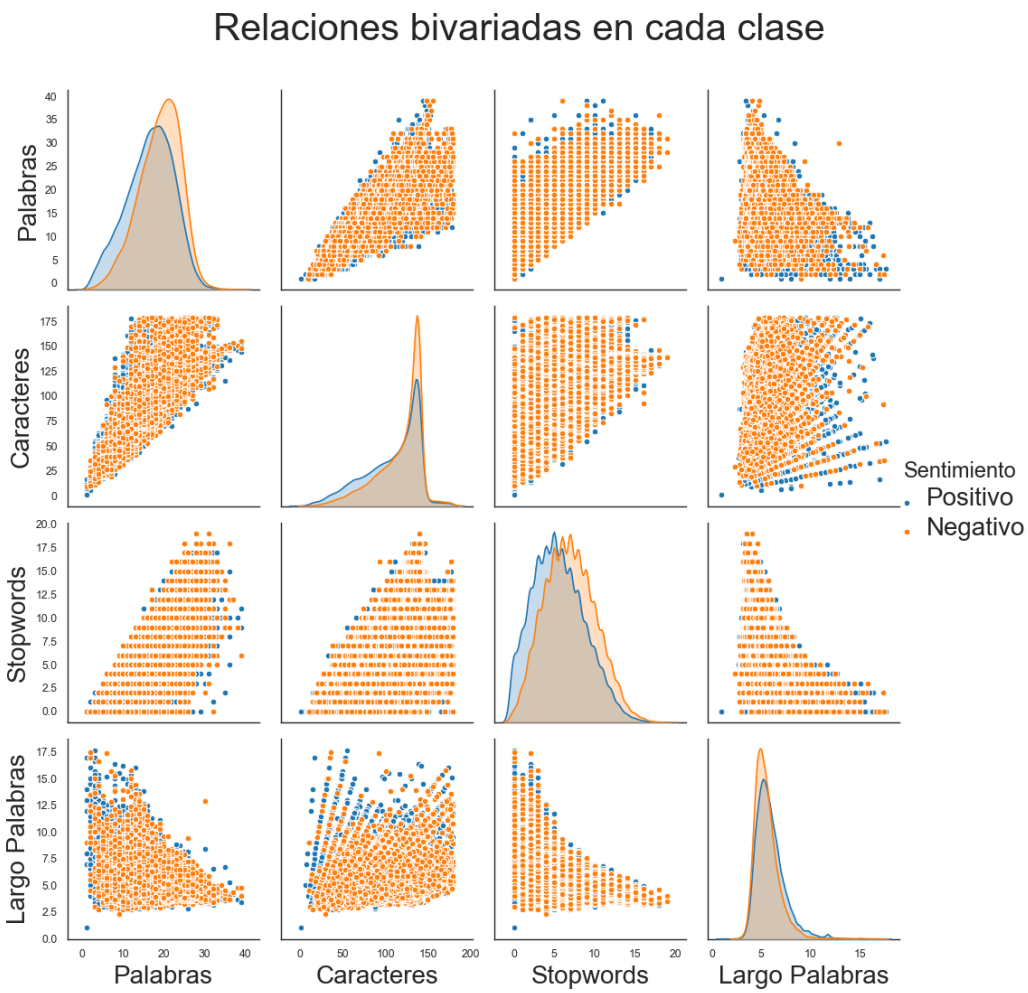


Figura 3.16: Gráfico de comportamiento bivariado de los datos diferenciado por sentimiento de la opinión

Lo primero que puede notarse en la figura 3.16 es que en ninguna de las relaciones bivariadas es posible encontrar diferencias significativas entre las clase (positiva y negativa). Solo en las variables *Palabras* y *Stopwords* podemos notar que la distribución del sentimiento positivo está un poco más cargado a la izquierda que la distribución del sentimiento negativo. Como era de esperarse, las variables *Caracteres* y *Palabras* están estrechamente relacionadas aunque no de manera lineal. Por el contrario, las variables *Largo Palabras* y *Stopwords* tienen una relación inversa, lo cual es intuitivo ya que las Stopwords regularmente son palabras cortas.

Como parte del análisis de relación entre variables es importante saber la fuerza de la relación lineal entre cada par de variables, así podremos detectar si es necesario eliminar alguna en la construcción del modelo final de detección de oportunidades de compra. En la figura 3.17 podemos observar la matriz de correlación de las variables construidas en este capítulo; una correlación fuertemente positiva está representada en color azul marino mientras que una correlación fuertemente negativa se ve en color rojo.

Como ya lo habíamos notado en el *pairplot*, se observa en la matriz 3.17 que hay una correlación fuerte y positiva entre las variables de *Palabras*, *Caracteres* y *Stopwords* por lo que valdría la pena eliminar alguna de ellas en la construcción del modelo final para reducir la complejidad del modelo. Por otro lado, la variable *Largo Palabras* tiene correlación negativa con las variables *Palabras* y *Stopwords*, sin embargo, esta correlación no es tan alta como para eliminar alguna de las variables. Las variables relacionadas con el contexto de Twitter (*Hashtags*, *Menciones* y *Tickers*) son independientes entre sí y con el resto de las variables, lo cual indica que pueden usarse en el modelo de aprendizaje automático y serán capaces de aportar información que no se encuentra en las demás.

Figura 3.17: Matriz de correlación de todas las variables construidas a partir del texto

Una vez obtenida la mayor información cuantitativa del texto podemos dar paso al análisis del sector financiero y construir la variable objetivo con lo que se tendrán las piezas necesarias para la construcción del modelo final, esto lo abordaremos en el siguiente capítulo.

Capítulo 4

El mercado accionario

Como el objetivo de este trabajo es poder demostrar la hipótesis de que las opiniones de la sociedad respecto a temas financieros repercuten en el comportamiento de los activos que cotizan en la Bolsa Mexicana de Valores, es necesario tener la información de dichos activos al momento del análisis para identificar su comportamiento y relación con las opiniones en la red social Twitter.

4.1. Indice de precios y cotizaciones

Los Índices de la Bolsa Mexicana de Valores y S&P Dow Jones, dependiendo de su enfoque y especialidad, son indicadores que buscan reflejar el comportamiento del mercado accionario mexicano en su conjunto o bien de diferentes grupos de empresas con alguna característica en común.

El Índice de Precios y Cotizaciones (S&P/BMV IPC) es el principal indicador del comportamiento del mercado accionario mexicano. Su muestra está integrada por las 35 series accionarias más relevantes en términos de su actividad operativa y tamaño disponible para los inversionistas, asegurando que la muestra sea representativa del segmento de mercado que abarca el índice mediante su rebalanceo cada seis meses: marzo y septiembre.

El peso de cada emisora en la muestra está determinado por el valor de capitalización de sus acciones flotantes, es decir, el número de acciones disponibles al mercado multiplicado por el precio de ellas. El peso máximo que puede tener una emisora dentro de la muestra es 25 % y las cinco más grandes no pueden acumular más del 60 % de representación en la muestra.

En el IPC cotizan acciones de emisoras de diferentes sectores como energía, materiales, industrial, servicios y bienes de consumo no básico, productos de consumo frecuente, salud, servicios financieros, tecnología de la información, servicios de telecomunicaciones y servicios públicos, entre otros. En este trabajo nos centraremos en el análisis del sector VII de Servicios Financieros integrado por entidades financieras, bienes inmobiliarios y sociedades de inversión.

Durante el periodo del 14 de Septiembre de 2018 al 8 de marzo de 2019, el sector VII del IPC (sector financiero) estuvo conformado por las emisoras de la tabla 4.1 donde se muestra el nombre de la emisora, su símbolo (*ticker*) y la identificación con la que se referenciarán en el presente treabajo. En algunas emisoras se presenta una letra después del nombre, esta letra indica la serie de acciones que se estan emitiendo; por ejemplo, en Grupo Financiero Banorte se considera la serie "O" para la información de cotización.

Emisora	Símbolo	Identificación
Banco del Bajío S.A	BBAJIO O	BBAJIO
Banco Santander México B	BSMX B	BSMX
Gentera S.A.B. de C.V.	GENTERA	GENTERA
Grupo Financiero Banorte O	GFNORTE O	GFNORTE
Grupo Financiero Inbursa O	GFINBUR O	GFINBUR
Regional S.A. de C.V.	R A	R

Tabla 4.1: Emisoras que integraron el sector VII del S&P BMV IPC: Servicios Financieros durante el periodo del 14 de Septiembre de 2018 al 8 de marzo de 2019

4.1.1. Obtención de precios

Para estudiar las acciones e identificar su relación con las opiniones de la red social es necesario tener información de comportamiento histórico de dichas acciones, para eso se usó la información de una plataforma digital de mercados financieros llamada *Investing* que proporciona datos de 250 mercados del mundo incluyendo el mercado accionario mexicano. Las variables obtenidas tienen periodicidad diaria y son las siguientes:

- Precio de apertura
- Precio de cierre
- Precio máximo
- Precio mínimo
- Volumen transaccionado
- Variación porcentual

Además de las proporcionadas por *Investing*, pueden calcularse otras variables que indiquen la tendencia de los precios en diferentes plazos anteriores a la fecha de predicción. Como el objetivo de este trabajo es pronosticar el movimiento del precio diario, se construyeron variables adicionales que representen cambios en plazos cortos como la media móvil y la desviación estándar de 3, 5 y 10 días.

4.2. Análisis Exploratorio

Como se mencionó anteriormente, estamos interesados en analizar el comportamiento de las acciones en el periodo de octubre 2018 a septiembre 2019, para lo cual es de suma importancia saber el movimiento de su precio durante este periodo.

La figura 4.1 muestra el precio diario de cada una de las acciones incluidas en el sector financiero del IPC de octubre 2018 a septiembre 2019. A pesar de que el precio de cada una es muy distinto, tienen una tendencia similar a través del tiempo. Por ejemplo, de octubre a diciembre de 2018, todo el sector tuvo una tendencia bajista en su precio debido a que se canceló la construcción de un aeropuerto en el Estado de México, donde todas estas compañías tenían inversiones.

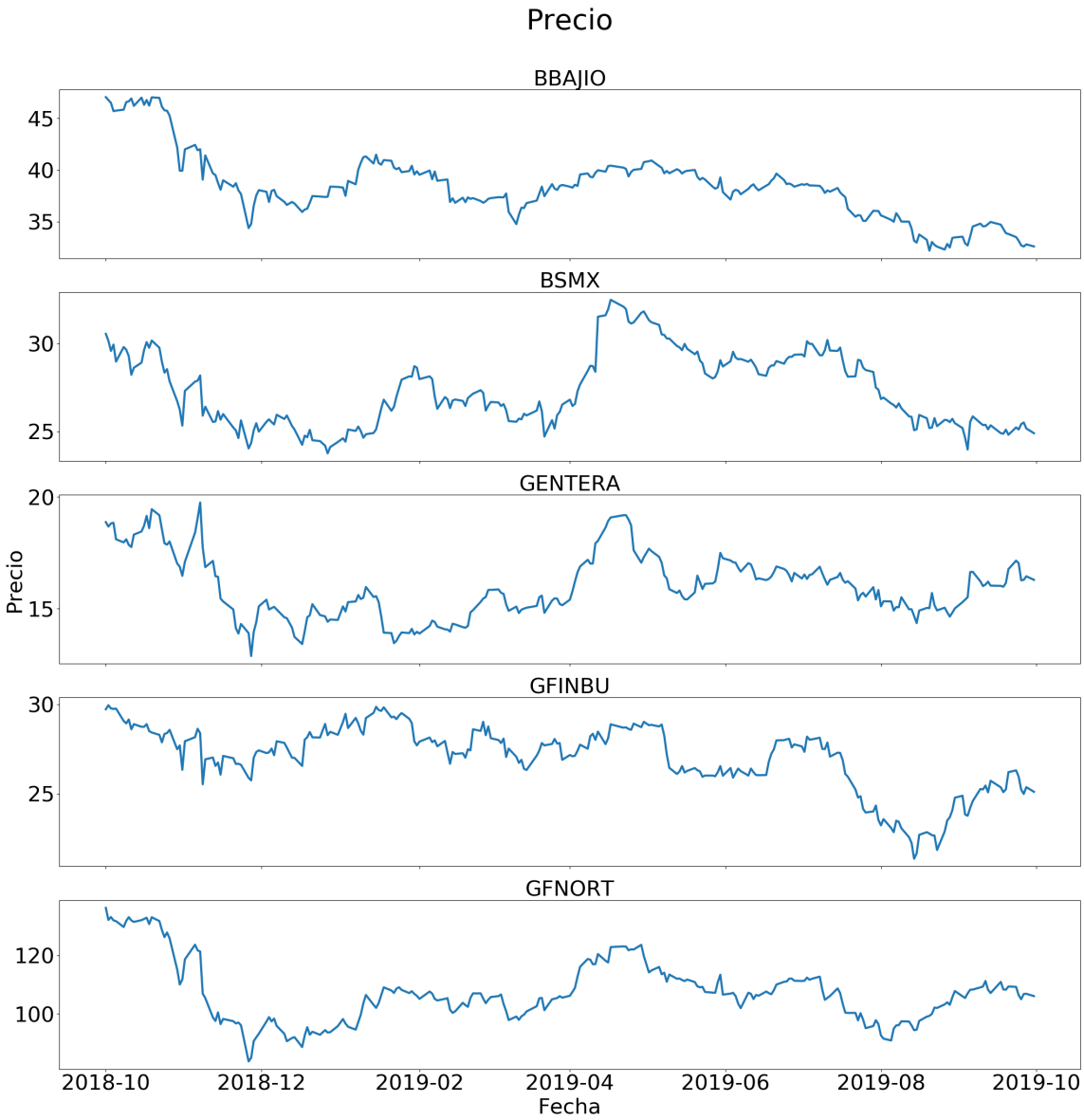


Figura 4.1: Precio de cierre de las acciones que constituyen el sector financiero del IPC

En mayo de 2019 se ve una recuperación sustancial en los precios de las acciones debido a que incrementó la inversión extranjera en la Bolsa Mexicana de Valores en 23 % durante ese mes, lo cual no está relacionado con la opinión de los usuarios en la red social sino con la atración económica del país.

La figura 4.2 muestra la variación porcentual entre el precio de apertura y el precio de cierre de cada una de las acciones que integran el sector y se puede ver que el comportamiento es similar entre todas las acciones. En noviembre de 2018 todas las acciones tienen variación muy alta debido a que en ese mes se lanzó una iniciativa de ley en el Senado de la República para disminuir las comisiones que las entidades bancarias cobran a sus clientes y eso repercutió directamente en todas las acciones del sector.

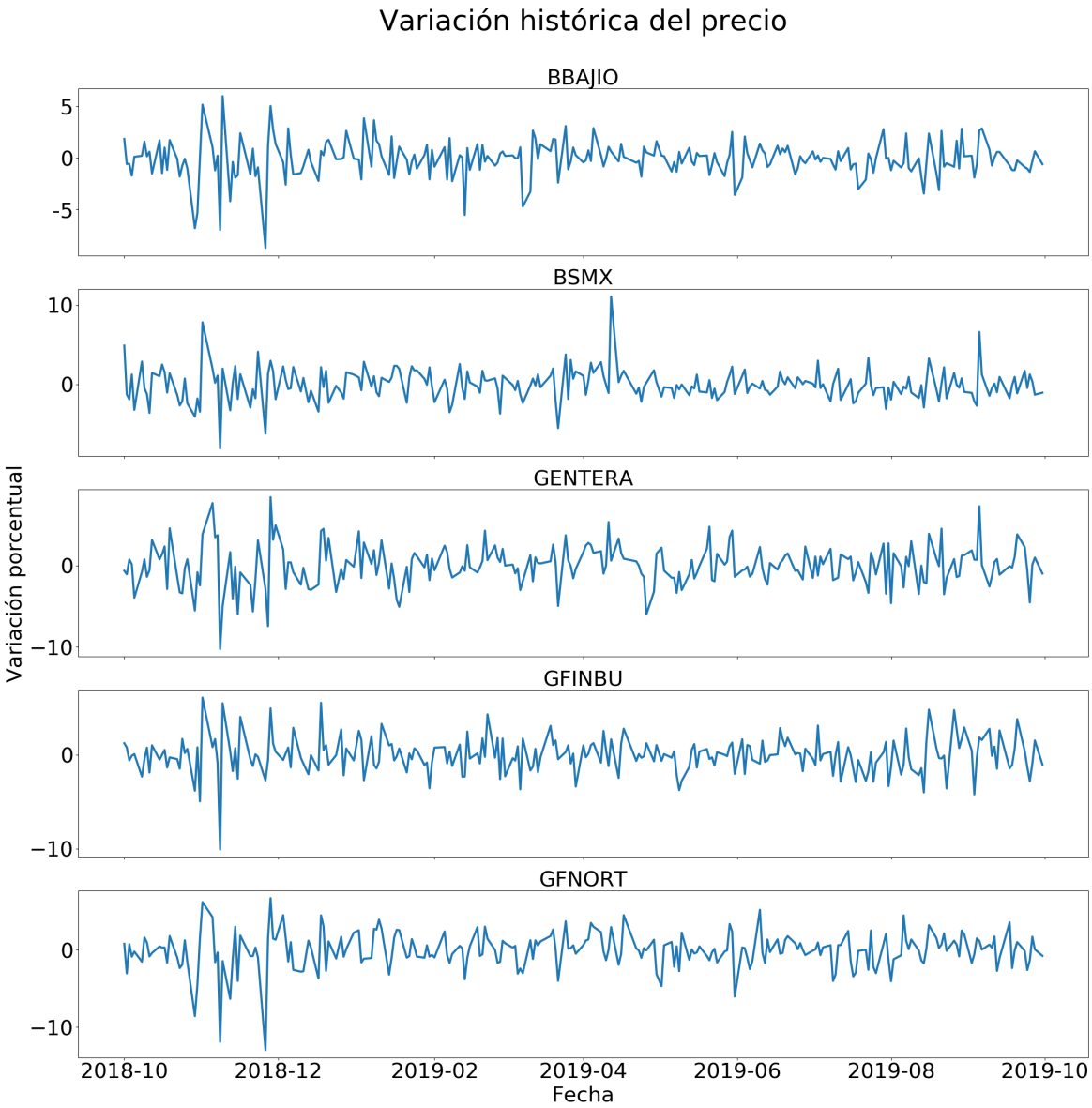


Figura 4.2: Variación diaria porcentual entre el precio de apertura y de cierre de las acciones que constituyen el sector financiero del IPC

Con el comportamiento histórico de los precios y la variación de las acciones se puede concluir que el sector financiero tiene un movimiento sistemático, es decir, que las acciones tienen la misma dirección bajo las mismas circunstancias en el mismo periodo. Por esta razón, es preciso entender la intensidad y dirección con las que se relacionan estas acciones.

Definición 4.1 *Dadas n observaciones de dos variables aleatoras X y Y, el coeficiente de correlación de Pearson r se calcula como:*

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y - \bar{y})^2}}. \tag{4.1}$$

El coeficiente de Pearson toma valores en el intervalo $[-1, 1]$. Un coeficiente negativo entre dos variables aleatorias X y Y indica una correlación lineal inversa, es decir, a medida que X aumenta, Y disminuye. Un coeficiente

positivo indica que la dirección de las variables es la misma. La figura 4.3 muestra el coeficiente de correlación de Pearson para cada pareja de acciones donde podemos notar que todas las acciones están positivamente correlacionadas; la pareja con el coeficiente más alto es GENTERA-GFNORTE mientras que GENTERA-GFINBU tienen el coeficiente más bajo.

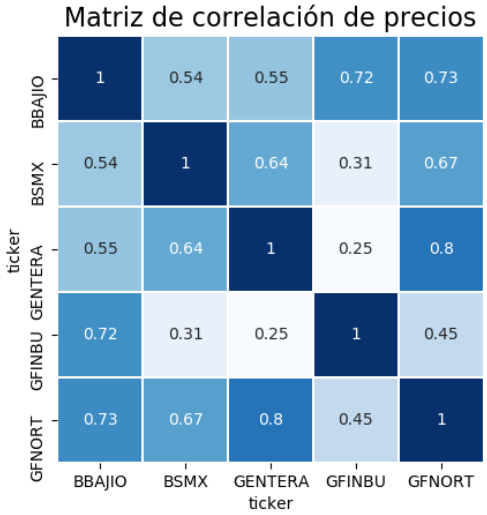


Figura 4.3: Matriz de coeficientes de correlación de Pearson de los precios de las acciones que integran el sector financiero del IPC

Del análisis anterior podemos decir que las acciones que integran el sector financiero son afectadas de la misma forma cuando hay acontecimientos, noticias o especulaciones importantes relacionados con la banca.

4.3. Variable objetivo

En los modelos supervisados de aprendizaje de máquina es necesario definir la variable objetivo, es decir, aquella variable de la cual se quiere predecir su valor o clasificación a partir del resto. Existen diversas formas de indicar que una acción tiene un mejor comportamiento respecto a su pasado; por ejemplo, si su precio de *apertura* del día d es mayor que el del día $d-1$ o si su precio de *cierre* del día d es mayor que el del día $d-1$, entre otros.

Sea y el indicador de la dirección del sector financiero en el día d, $p_{d,i}$ el precio de apertura del día d para la acción i donde $i \in \{$"BBAJIO", "BSMX", "GENTERA", "GFINBUR", "GFNORTE"$\}$ entonces

$$y_d = \begin{cases} 1 & \text{si } 1.003 * \sum_i p_{d-1,i} \leq \sum_i p_{d,i} \\ 0 & e.o.c. \end{cases}. \tag{4.2}$$

El punto de corte entre un día bueno para invertir y el resto se estableció con base en dos factores que son el balanceo de los datos y la ganancia que se puede obtener de la estrategia de inversión. Si se tomaba un valor muy alto, el conjunto de datos quedaba muy desbalanceado repercutiendo en la habilidad del modelo para predecir días positivos y si se tomaba un valor muy cercano a 0 %, el modelo simplemente predeciría la dirección del precio mas no la oportunidad de tener ganancias al invertir en ese día.La tabla 4.2 muestra el balanceo de los datos con distintos puntos de corte.

Punto de corte	Observaciones negativas/positivas	Porcentaje de clase positiva
0.1 %	133/117	46.8 %
0.3 %	154/96	38.2 %
0.5 %	173/77	30.8 %
1.0 %	194/56	22.4 %

Tabla 4.2: Balanceo de clases en el conjunto de datos con diferentes puntos de corte (variación en el precio).

Con este criterio, se definió que un día es positivo si la suma de los precios de apertura es 0.3 % mayor que la suma de los precios de apertura del día anterior, así se logra identificar si vale la pena invertir en ese día dado que ya abrió el mercado accionario y que la gente ya tomó una postura respecto a lo que está sucediendo en el sector. La figura 4.4 muestra la variación porcentual entre el precio de apertura del día anterior y el del día corriente. La línea punteada verde indica la variación de 0.3 %, los días cuyo valor y se encuentre a la derecha de esa línea serán considerados como buenos días para comprar acciones mientras que el resto serán considerados como malos.

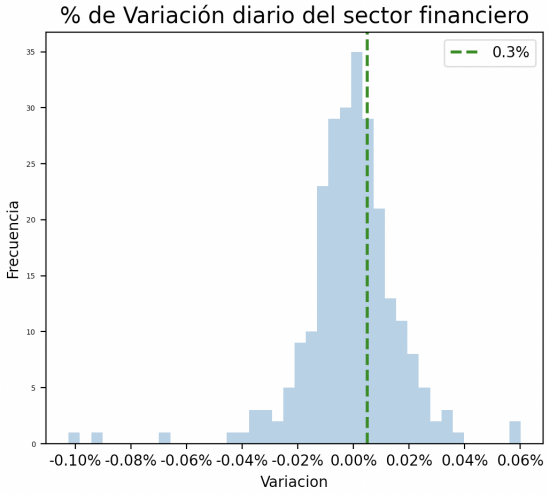


Figura 4.4: Variación porcentual entre el precio de un día y el precio del día inmediato anterior de las acciones del sector financiero del IPC.

Capítulo 5

Resultados

Para obtener el conjunto de datos usados en el modelo, es necesario agrupar la información de las opiniones para poder reunirla con los datos financieros. Esto se debe a que cada registro de la primera tabla corresponde a un tuit mientras que en la segunda corresponde a un día de actividad en el mercado. La tabla 5.1 muestra las funciones de agrupación las variables extraidas de los tuits. Dado que la variable *Sentimiento* es categórica, se realizó una codificación numérica en la que un sentimiento positivo suma un punto y uno negativo resta un punto de tal manera que un número mayor que cero indicará que el día tuvo mayor cantidad de publicaciones positivas y viceversa. Después se une la información mediante la fecha considerando las opiniones del día anterior al que se está prediciendo, es decir, se toma un retraso (*lag*) de un día entre las variables provenientes de los tuits y las financieras (incluyendo la variable objetivo). La razón de hacerlo así es que el horario de operación del mercado accionario es de 8:30 a 15:00 horas en tanto que la red social funciona todo el día, entonces las noticias y opiniones que ocurren por la tarde y noche de un día afectan al precio del día siguiente.

Variable	Funciones de agrupación
Tuits	Conteo total y por hora del día
Sentimiento	Suma
Palabras	Conteo y Promedio
Caracteres	Conteo y Promedio
Stopwords	Conteo y Promedio
Hashtags	Conteo y Promedio
Retweets	Conteo y Promedio
Respuestas	Conteo y Promedio
Menciones	Conteo y Promedio
Cashtags	Conteo y Promedio

Tabla 5.1: Funciones de agrupación aplicadas a las variables extraida de las opiniones de Twitter.

Existen otras funciones de agrupación como la desviación estándar, máximo, mínimo, etc. que se pueden calcular a partir de las variables provenientes del texto, sin embargo, solo se dejaron el conteo y el promedio porque cuando hay más variables que observaciones en un conjunto de datos se debe abordar como un problema de alta dimensionalidad y resolverse con otras metodologías como el Análisis de Componentes Principales (PCA por sus siglas en inglés).

El conjunto de entrenamiento consta de 250 observaciones que corresponden a los 250 días de actividad de los mercados bursátiles en el periodo del 1 de Octubre de 2018 al 30 de Septiembre del 2019. Por su parte, se tienen 49 variables predictoras: 42 provenientes de las opiniones de Twitter y 7 relacionadas a la tendencia del precio de las acciones, mientras que el balanceo del indicador es de 61.8 % para la clase negativa y 38.2 % para la clase positiva.

Existen estrategias de remuestreo para balancear las clases de la variable objetivo, sin embargo, no es conveniente aplicarlas en este caso debido a que las observaciones del conjunto de entrenamiento son históricas. Lo que puede hacerse es recopilar más información histórica para ver si las proporciones se mantienen en el tiempo o si es una particularidad del periodo de estudio (aunque la tendencia general de los precios suele ser alcista), otra opción es tomar lapsos más cortos (intradiarios) donde las fluctuaciones sean mas balanceadas. Ambas opciones se plantean como trabajo futuro.

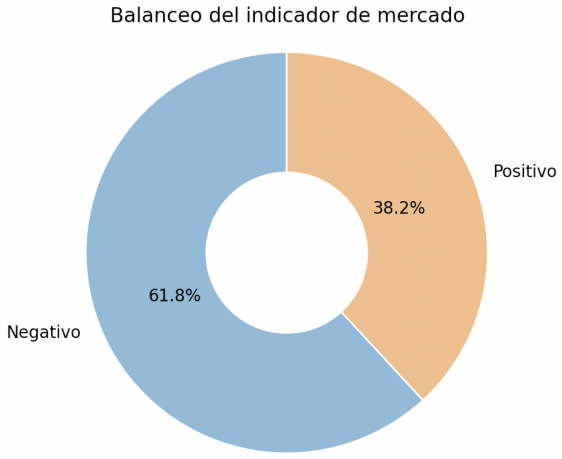


Figura 5.1: Proporción del indicador de mercado construido en el capítulo tres donde la clase *cero* representa los días en que el sector financiero tuvo variación negativa y *uno* los días en que tuvo variación positiva

La magnitud de las variables numéricas es un factor importante que debe considerarse antes de entrenar el modelo; por ejemplo, la variable de *número de caracteres* se mide en miles mientras que el número de *hashtags* se mide en cientos. Por ese motivo, es importante tener las variables en la misma escala para que sean comparables. El método de normalización más común es la estandarización que a cada valor x de la variable X le resta la media μ_X y divide entre la desviación estandar σ_X, es decir, el valor normalizado z de x es

$$z = \frac{x - \mu_X}{\sigma_X}. \tag{5.1}$$

Aunque la normalización de datos sea una práctica muy común, se debe tener precaución en este proceso porque puede distorcionar la distribución de las variables. Si una variable tiene una distribución asimétrica derecha, la estandarización recorrería los datos hacia la izquierda ya que el cálculo de la media aritmética está afectado por los valores atípicos.

En las figuras 5.2, 5.3 y 5.4 se muestra la comparación entre las variables originales y las normalizadas mediante el proceso de estandarización. La intención de mostrar el comparativo es que se identifiquen claramente las variables que tienen mucha dispersión en comparación con el resto. Por ejemplo, la variable original del promedio de *stopwords* de la figura 5.3 parece estar aplanada y dispersa pero cuando se coloca en la misma escala que las demás variables se nota mucha concentración en el intervalo [-2.5, 2.5] mientras que otras como el promedio de *hashtags* oscilan en un rango más grande.

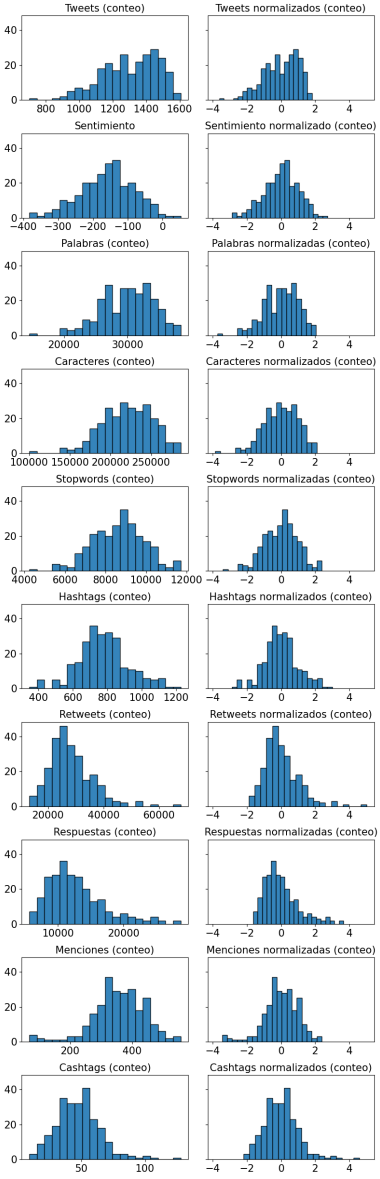


Figura 5.2: Normalización de variables de conteo.

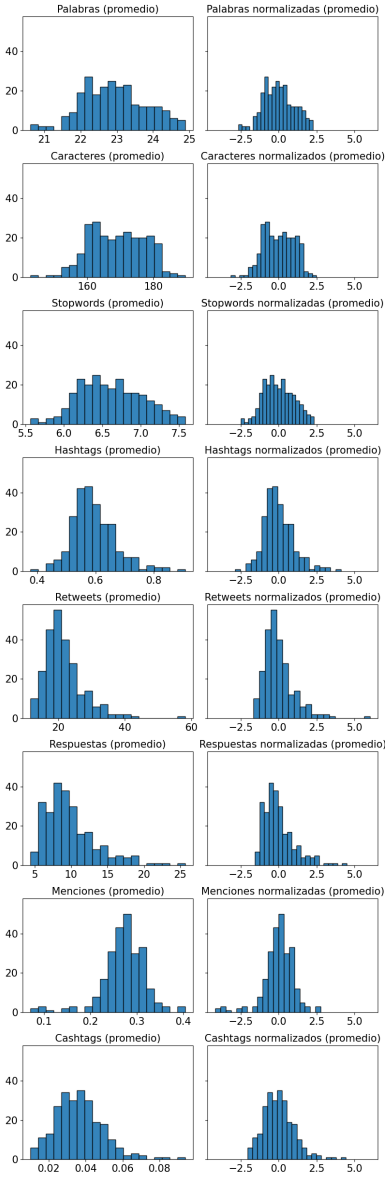


Figura 5.3: Normalización de variables promediadas.

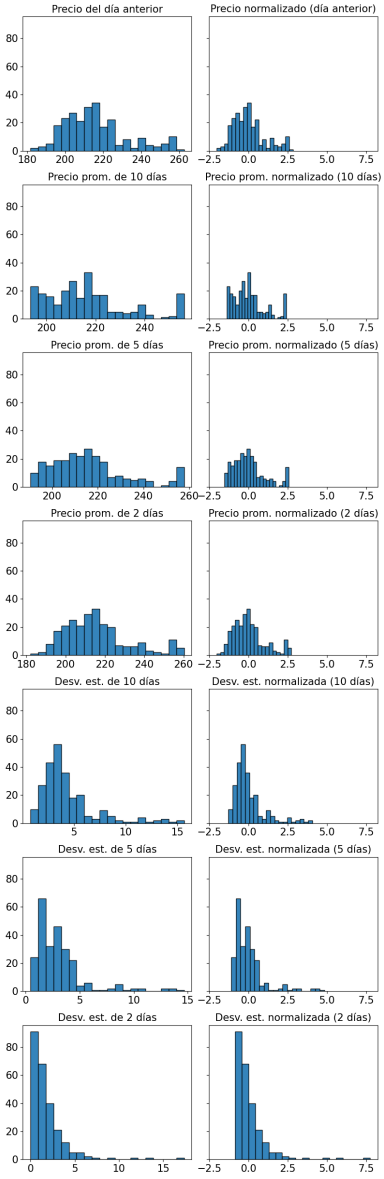


Figura 5.4: Normalización de variables del precio.

5.1. Validación cruzada

Los algoritmos de aprendizaje de máquina requieren de un conjunto de entrenamiento y otro de prueba. En la mayoría de los casos, esta división se hace aleatoriamente tomando el 20 % de los datos para la prueba, no obstante, cuando la información que se usa para predecir el fenómeno es histórica, lo mejor es hacer la división basándose en el tiempo. La validación en series de tiempo (*Time Series Cross-Validation*) es una forma de construir conjuntos de datos de tal manera que el subconjunto de entrenamiento siempre este conformado por observaciones anteriores al subconjunto de prueba.

Sea n_r es el número de registros, n_s el número de validaciones cruzadas, i la validación actual ($i = 1, .., n_s$), tr_i el número de observaciones del conjunto de entrenamiento en la i-ésima validación y $n_r|(n_s - 1)$ la operación módulo entre n_r y $n_s - 1$, entonces el número de observaciones tr_i del subconjunto de entrenamiento es

$$tr_i = \left\lfloor \frac{i \times n_r}{n_s + 1} \right\rfloor + n_r|(n_s + 1), \tag{5.2}$$

mientras que el tamaño del subconjunto de prueba tt_i está dado por

$$tt_i = \left\lfloor \frac{n_r}{n_s + 1} \right\rfloor. \tag{5.3}$$

Dado que el conjunto de datos de este caso de estudio es pequeño ($n_r = 250$), se estableció $n_s = 3$, entonces el número de observaciones del subconjunto de entrenamiento de la primera iteración tr_1 es

$$tr_1 = \left\lfloor \frac{1 \times 250}{3 + 1} \right\rfloor + 250|(3 + 1) = 62 + 2 = 64. \tag{5.4}$$

Por su parte, el número de observaciones del subconjunto de prueba en la primera iteración es

$$tt_1 = \left\lfloor \frac{250}{3 + 1} \right\rfloor = 62. \tag{5.5}$$

La figura 5.5 muestra el comportamiento de la validación cruzada temporal usada en este trabajo. De arriba hacia abajo, las primeras tres barras representan en color azul los datos de entrenamiento y en rojo los de prueba, la última barra muestra en color azul las fechas en las que el indicador fue *negativo* y en color blanco cuando el indicador fue *positivo* segun la definición del capítulo 4 (ecuación 4.2).

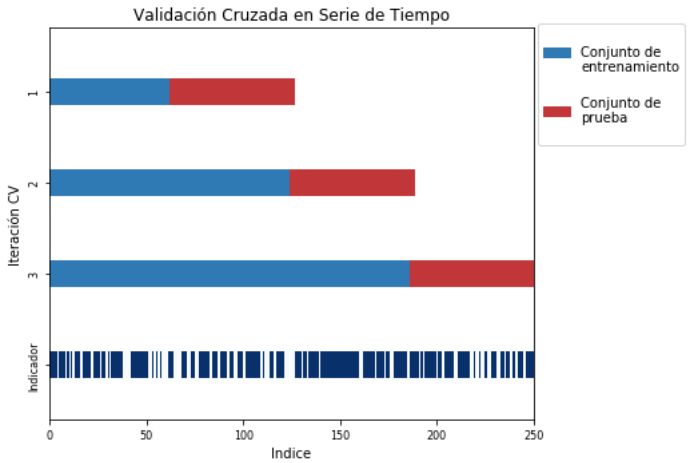


Figura 5.5: Conjuntos temporales de entrenamiento de las iteraciones de la validación cruzada e indicador de la variable objetivo en el tiempo.

La tabla 5.2 muestra los periodos que comprende cada iteración de la validación cruzada así como el balanceo de la variable objetivo según la definición del capítulo 3. Vemos que el balanceo de los datos no es estable ni en el tiempo ni entre los conjuntos de entrenamiento/prueba y esto se debe a que los precios de las acciones de la bolsa no son constantes en el tiempo. Se puede intuir que el conjunto con el peor desempeño será el de prueba de la segunda iteración puesto que es el más desbalanceado de todos, mientras que el conjunto de prueba de la primera iteración podría tener el mejor desempeño debido a que es el mejor balanceado.

Iteración	Periodo de entrenamiento	Periodo de prueba	% de positivos (entrenamiento)	% de positivos (prueba)
1	1/Oct/2018 al 3/Ene/2019	4/Ene/2019 al 3/Abr/2019	39.1 %	46.8 %
2	1/Oct/2018 al 4/Abr/2019	5/Abr/2019 al 4/Jul/2019	43.3 %	27.4 %
3	1/Oct/2018 al 4/Jul/2019	5/Jul/2019 al 30/Sep/2019	38.0 %	39.3 %

Tabla 5.2: Periodos que comprenden los conjuntos de entrenamiento y prueba en cada iteración de la validación cruzada en serie de tiempo.

5.2. Optimización de parámetros

Todo modelo de Aprendizaje de Máquina tiene asociados unos *hiperparámetros* que son parámetros externos que rigen el proceso de entrenamiento. El proceso de optimización de parámetros consiste en encontrar la combinación de hiperparámetros con el mejor desempeño del modelo según las métricas definidas. Los módulos de Python tienen valores predeterminados para los hiperparámetros pero se pueden modificar si se desea mejorar el rendimiento del modelo.

En las máquinas de soporte vectorial se optimizan cuatro hiperparámetros: la función de penalización del error, el número de iteraciones, el peso de las clases (en clasificación binaria) y la regularización. La función de penalización es simplemente la medida de discrepancia entre el estimador y el valor real, el número de iteraciónes es el número de veces que se resolverá el problema de optimización (minimización), el balance del peso de las clases replica la clase minoritaria hasta que se tiene el mismo número de registros que la clase mayoritaria y la regularización relaja las restricciones cuando las clases no son estrictamente separables por un hiperplano.

5.2.1. Curvas de validación

Las curvas de validación son gráficas que muestran una métrica del modelo en función de algúno de los hiperparámetros. Estas curvas permiten encontrar el valor del hiperparámetro donde se obtiene la mejor métrica sin que el modelo se vuelva inestable o demasiado complejo.

En la figura 5.6 se muestra el rendimiento del modelo evaluado en las tres principales métricas: exactitud, precisión y sensibilidad. La exactitud es buena (arriba de 0.5) sin importar el valor de regularización aunque hay un incremento importante alrededor de un valor de 10 y después se estabiliza. La precisión mejora consistentemente hasta que la regularización llega a 10 y después comienza a disminuir mientras que la sensibilidad tiende a incrementar siempre que aumenta la regularización. Por lo anterior y para conservar la estabilidad de las métricas sin sacrificar rendimiento, se seleccionó el valor de 10 para este hiperparámetro.

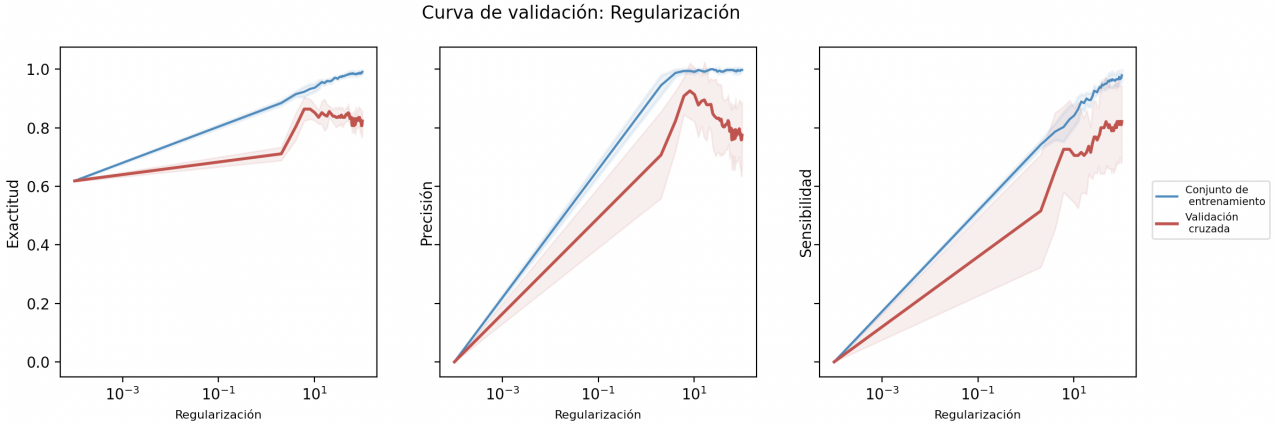


Figura 5.6: Curva de validación del parámetro de regularización evaluado en tres métricas: exactitud, precisión y sensibilidad.

En el proceso de optimización se probaron 40 combinaciónes de hiperparámetros: 2 opciones para la función de penalización, 2 opciones para el número de iteraciones, 2 opciones para el balanceo de clases y 5 opciones para la regularización. Una vez terminada la búsqueda de los hiperparámetros, es decir, después de ajustar los 40 modelos con 3 validaciones cruzadas cada uno (120 modelos en total) se obtuvo la mejor combinación que arrojó una precisión de 0.7312.

Hiperparámetro	Nombre	Valor
Función de penalización	penalty	L2
Iteraciones hasta convergencia	max_iter	100,000
Balanceo de clases	class_weight	No
Regularización	C	10

Tabla 5.3: Combinación de hiperparámetros que arrojan el mejor score en el conjunto de prueba

La tabla 5.3 muestra que la medición mediante el error cuadrático (norma L2) con un grán número de iteraciones en el conjunto original (desbalanceado) y una regularización moderada ($C = 10$). Estos son los parámetros que lograron el mejor ajuste a los datos, por lo tanto, los resultados siguientes mostrarán el desempeño del modelo con estos parámetros.

5.3. Rendimiento del modelo

Teniendo la combinación de hiperparámetros que arrojan la mejor métrica, se requiere revisar que el modelo sea capaz de generalizar lo aprendido para predecir nuevas observaciones, que la clasificación funcione bien para ambas clases y al mismo tiempo identifique el punto de corte entre las clases más adecuado tomando en cuenta que el objetivo del modelo es identificar los días en que es mejor invertir en el sector financiero del mercado accionario mexicano.

5.3.1. Curva de aprendizaje

Una curva de aprendizaje es una gráfica del rendimiento del aprendizaje de un modelo sobre la experiencia o el tiempo. Estas gráficas permiten diagnosticar problemas con el aprendizaje del modelo como el sobreajuste (*overfitting*) o el subajuste (*underfitting*) para saber si los datos son representativos del fenómeno que se quiere predecir. Una buena adaptación de los datos es representada por una curva de aprendizaje cuya brecha entre la

métrica del conjunto de entrenamiento y la del conjunto de prueba se hace más pequeña a medida que se tiene más experiencia (más observaciones).

El aprendizaje de los modelos puede ser medido con métricas como el error de clasificación o la función de pérdida y dependiendo de qué métrica se use, se busca maximizar o minimizar. En los modelos de clasificación se suelen usar la exactitud, la precisión y la sensibilidad que, como ya se detalló en el capítulo 2, indican la proporción de observaciones asignadas correctamente a cada clase.

La figura 5.7 muestra la curva de aprendizaje para tres tamaños de conjuntos de entrenamiento: 63, 126 y 183 observaciones. La línea azul representa el promedio de la exactitud de una validación cruzada de tres iteraciones en el conjunto de entrenamiento mientras que la línea roja representa el promedio en el conjunto de prueba. La sombra de cada línea representa la desviación estándar de la exactitud en cada conjunto. Se espera que el aprendizaje vaya mejorando conforme incrementa la experiencia, es decir, a mayor cantidad de datos se mejorará la puntuación. En la figura 5.7 notamos que la exactitud incrementa en el conjunto de prueba a medida que incrementan el número de observaciones en el conjunto de entrenamiento que indica que el modelo está generalizando correctamente. También podemos notar que las curvas de aprendizaje correspondientes al conjunto de entrenamiento van disminuyendo a medida que se incluyen más datos, lo cual indica que el modelo está aprendiendo bien a medida que adquiere experiencia. La sensibilidad disminuye tanto en el conjunto de entrenamiento como en el de prueba debido a que existe una relación inversa con la precisión de tal manera que no es posible maximizar ambas al mismo tiempo.

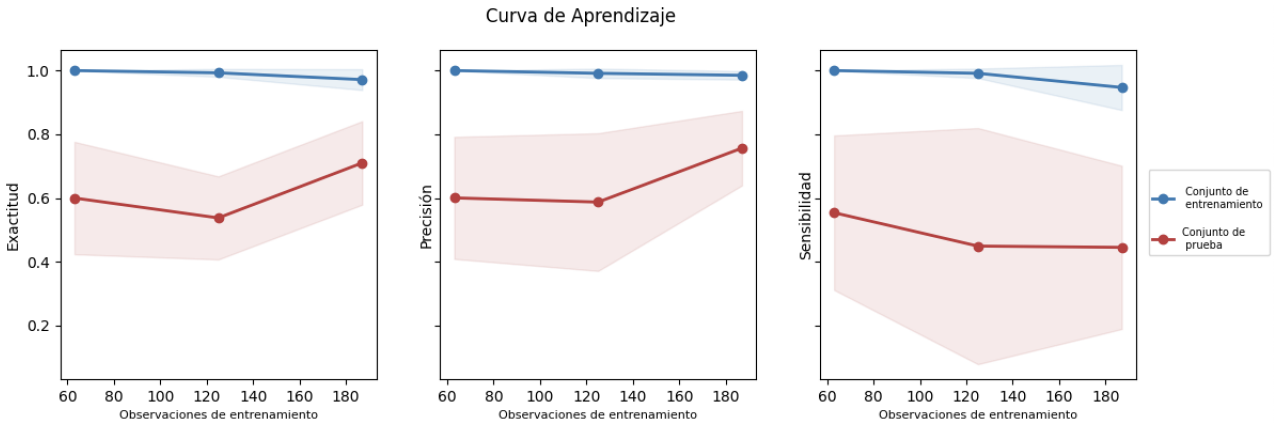


Figura 5.7: Curva de aprendizaje para tres conjuntos de entrenamiento de diferentes tamaños. En azul se observa la exactitud del modelo en el conjunto de entrenamiento y en rojo en el de prueba.

5.3.2. Curva ROC

La curva ROC ayuda a observar la relación entre la tasa de falsos positivos frente a los verdaderos positivos para encontrar el balance entre ambas tasas. En la figura 5.8 podemos observar la curva ROC para las tres validaciones realizadas (azul, verde y anaranjado). Las iteraciones 2 y 3 tienen un buen resultado si se comparan con la función identidad o "suerte", sin embargo, la iteración 1 tiene un rendimiento mucho más bajo debido a que el conjunto de prueba en el que se evaluó esta iteración corresponde a las observaciones de mayo a julio de 2019 donde la variable objetivo tuvo un comportamiento atípico (ver figura 5.5). Pese a esto, el AUC promedio fue de 0.75 lo cual indica que, en general, la clasificación del modelo es mejor que la selección aleatoria.

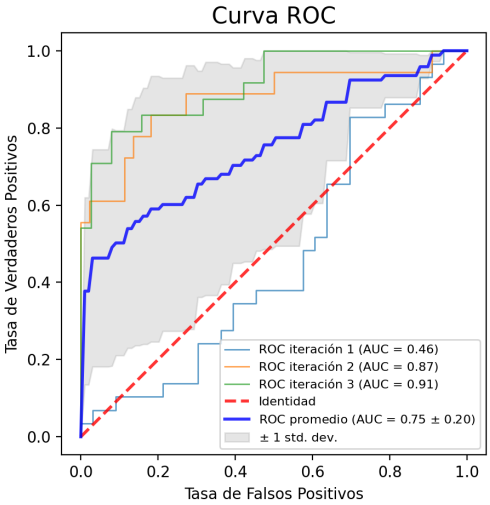


Figura 5.8: Curva ROC de las tres iteraciones en las que se evaluó el modelo entrenado con los mejores hiperparámetros

5.3.3. Curva PR

Las métricas de precision y sensibilidad están inversamente relacionadas ya que al aumentar el umbral de clasificación (*threshold*) aumenta la precisión y disminuye la sensibilidad. Por lo tanto, se debe priorizar alguna de dichas métricas según el objetivo de la clasificación en el modelado de datos. La curva PR resulta de dibujar la precisión en función de la sensibilidad y permite identificar el nivel de sensibilidad sin que haya degradación de la precisión y viceversa. La curva ideal es aquella que se acerque a la esquina superior derecha (alta precisión y alta sensibilidad). En la figura 5.9 observamos que el área bajo la curva promedio es de 0.59 lo cual indica que existe un umbral de clasificación en el que ambas métricas resultan mayores que 0.5. Al igual que se veía en la curva ROC, notamos que la primera iteración tiene un desempeño sustancialmente más bajo que el resto lo cual se debió a un comportamiento anómalo en el periodo que comprende dicha validación.

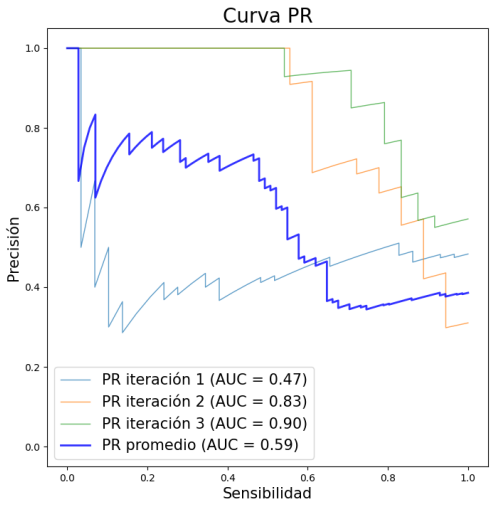


Figura 5.9: Curva PR de las validaciones cruzadas del modelo entrenado con los mejores hiperparámetros

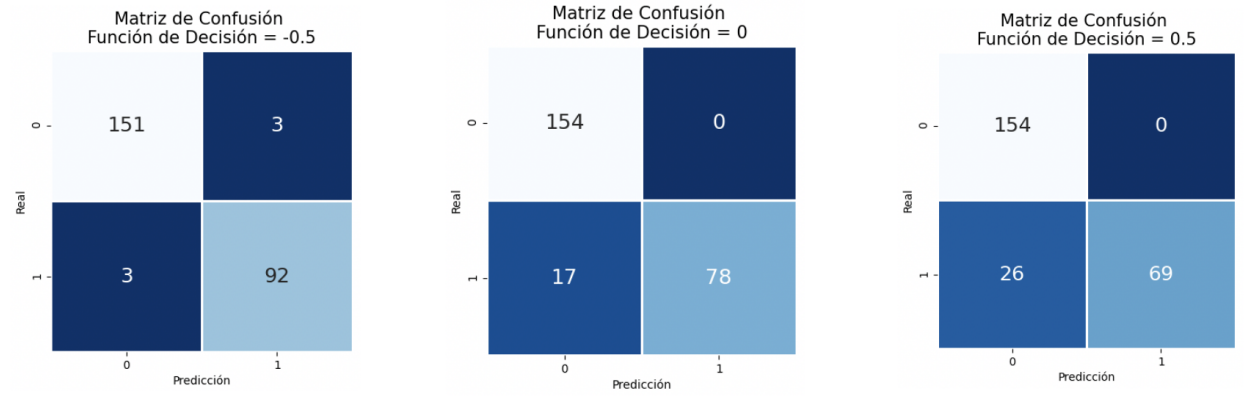
5.3.4. Cambio de límite de clases

Para mejorar la clasificación de las observaciones se selecciona aquel umbral de clasificación (*threshold*) con el que se obtenga la tasa de verdaderos positivos más alta y la menor tasa de falsos positivos. Por lo general, se establece este umbral a partir de la probabilidad de pertenecer a una clase, sin embargo, en las máquinas de soporte vectorial se define a partir de su distancia al hiperplano separador: si la evaluación en la función de desición es positiva, se asigna a la clase positiva y viceversa.

En la figura 5.10 se observan las matrices de confusión correspondientes a tres diferentes umbrales de clasificación. En la matriz de la izquierda se tomó un punto de corte en -0.5, es decir, las observaciones por debajo de -0.5 son clasificadas como negativas y las mayores como positivas, se observa que todas las métricas están por encima de 0.95, lo cual indica que el modelo acierta en la gran mayoría de las clasificaciones. En la matriz del centro se usó el umbral de 0 obteniendo una exactitud de 0.93 que también resulta ser bastante bueno; finalmente, en la matriz de la derecha se usó un umbral de 0.5 que arrojó una exactitud de 0.8955 que es el peor rendimiento de los tres.

Además de la exactitud, contemplamos otras métricas como la precisión, la sensibilidad y la F1 para seleccionar el mejor umbral entre clases, ya que estas métricas validan el rendimiento del modelo cuando una de las clases es más importante y aquí nos interesa más que el modelo logre identificar correctamente los días que representan un buen día para el mercado (clase positiva) pues esto detonaría la inversión en la bolsa de valores.

Como existe una relación inversa entre la presición y la sensibilidad se debe priorizar alguna de ellas para seleccionar el umbral que la maximice. Para esta aplicación es más importante maximizar la presición puesto que así se garantiza que los días que el modelo clasifica como Positivos realmente lo sean. Por lo anterior, se selecciona el umbral en el que la presición es buena pero que el resto de las métricas se mantengan en niveles aceptables que corresponde a la matriz del lado izquierdo.



$$Exactitud = \frac{151+92}{151+3+3+92} = 0.9759 \qquad Exactitud = \frac{154+78}{154+0+17+78} = 0.9317 \qquad Exactitud = \frac{154+69}{154+0+26+69} = 0.8955$$

$$Precisión = \frac{92}{92+3} = 0.9684 \qquad Precisión = \frac{78}{78+0} = 1 \qquad Precisión = \frac{69}{69+0} = 1$$

$$Sensibilidad = \frac{92}{92+3} = 0.9684 \qquad Sensibilidad = \frac{78}{78+17} = 0.8210 \qquad Sensibilidad = \frac{69}{69+26} = 0.7263$$

$$F1 = 2 * \frac{0.9684 * 0.9684}{0.9684 + 0.9684} = 0.9684 \qquad F1 = 2 * \frac{1 * 0.8210}{1 + 0.8210} = 0.9017 \qquad F1 = 2 * \frac{1 * 0.7263}{1 + 0.7263} = 0.8415$$

Figura 5.10: Matrices de confusión y métricas de clasificación para tres limites de decisión: $-0.5, 0$ y 0.5.

5.4. Explicación del modelo

Como el modelo pretende representar la realidad, es de suma importancia que se pueda explicar su funcionamiento ya que así se pueden identificar las variables o factores más importantes del fenómeno y las características que distinguen una clase de otra.

Por otro lado, cuando se puede explicar un modelo es más fácil encontrar mejoras y volverlo más confiable para las personas que no tienen conocimientos técnicos puesto que deja ser una "caja negra" en la que solo entran datos y salen predicciones.

5.4.1. Shap Values

Los valores SHAP (véase apéndice C) son usados en los algoritmos complejos de aprendizaje de máquina para entender las decisiones que toma el clasificador. La potencia de estos valores es que indican la importancia de cada variable en la clasificación de cada observación. En Python, existe un módulo llamado *shap* que permite interpretar los valores SHAP de los modelos de aprendizaje de máquina de una manera sencilla y visual. En particular, el *Force Plot* permite ver la importancia de cada variable en cada observación: las variables que "empujan" a la observación hacia la clase positiva se marcan en color rojo y las que lo hacen hacia la clase negativa en azul, el *base value* es el valor que se predeciría si no se conociera ninguna variable, es decir, el valor esperado μ, el tamaño de cada barra representa el valor SHAP de esa variable (importancia) de tal manera que la suma de los valores SHAP del conjunto de variables V más el *base value* será igual al *output value* que coincide con la distancia al hiperplano separador d como se muestra en la ecuación

$$\sum_{i \in V} SHAP(i) + \mu = d. \tag{5.6}$$

La figura 5.11 muestra el *Force Plot* de la observación positiva del día 26/12/2018 con un valor SHAP de 0.97 donde 7 variables contribuyen hacia el lado positivo y 6 al lado negativo y 36 son iguales a 0. La variable más importante fue el *cierre del precio anterior* seguida por el *promedio del precio de 2 días previos*. Por el contrario, en la figura 5.12 tenemos la observación del 01/07/2019 clasificada como Negativa en la que 5 variables contribuyeron al lado negativo y 5 al positivo, siendo el *cierre del precio anterior* y el *promedio del precio de dos días previos* las variables más importantes al igual que en la observación positiva pero en sentidos opuestos.

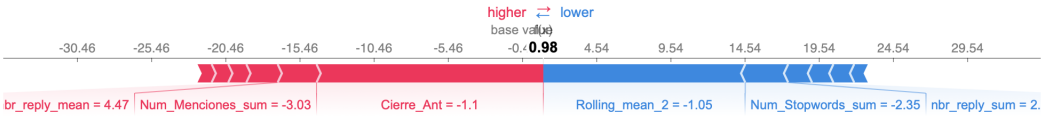


Figura 5.11: Force Plot de la observación correspondiente al 26/12/2018 clasificada como positiva

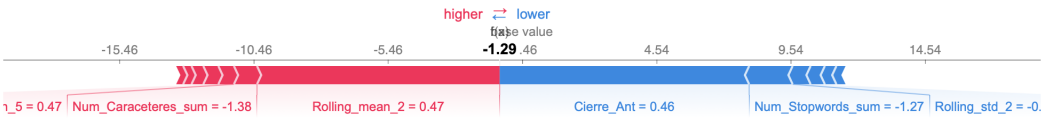


Figura 5.12: Force Plot de la observación correspondiente al 01/07/2019 clasificada como negativa

Como la importancia de las variables no es la misma en todas las observaciones, existen otras visualizaciones que ayudan a resumir o generalizar la importancia de cada variable. El *summary plot* del módulo *shap* de Python

provee un gráfico en el que se pueden observar los valores SHAP de todas las observaciones del conjunto de entrenamiento. En este gráfico se ve el impacto (positivo o negativo) de cada predictor sobre la variable objetivo. Las variables están acomodadas por nivel de importancia (entre mas arriba, mas importante), el eje X muestra si el valor está asociado a una predicción más alta o mas baja y el color indica si la variable es alta (en rojo) o baja (en azul).

En el *summary plot* de la figura 5.13 podemos notar que el **cierre del precio anterior** es la variable más importante del modelo seguida por el **promedio del precio de los dos días anteriores** y el **número de *stopwords*** mientras que las demás variables tienen poca importancia según su valor SHAP. Otra puntualización del gráfico es que el cierre del precio del día anterior está inversamente relacionado con la variable objetivo, por el contrario, las variables con segundo, tercer, cuarto y quinto lugar de importancia tienen relación directa con la variable objetivo, es decir, mientras más grande es la covariable más alto el valor SHAP.

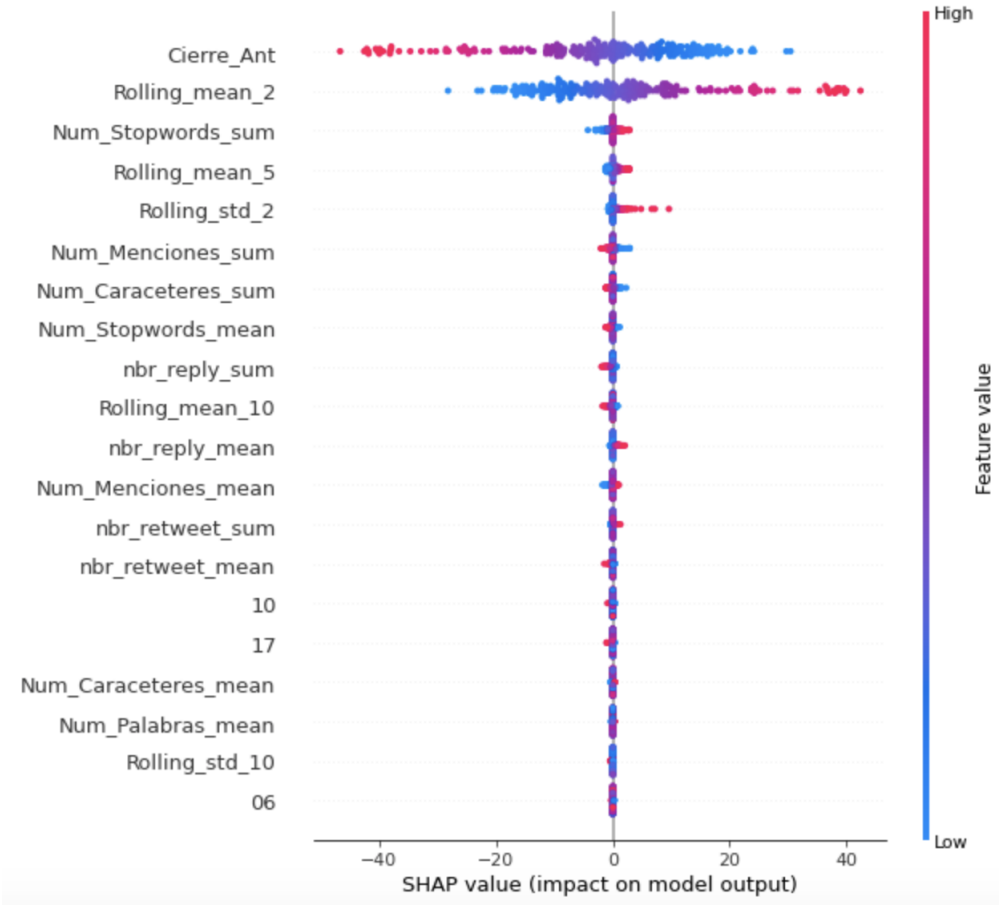


Figura 5.13: Valores SHAP del modelo de dirección de precios basado en las variables provenientes de las opiniones de Twitter.

El *summary plot* puede romper con hipótesis planteadas antes de desarrollar un modelo. En este caso, se podría pensar que un cierre anterior bajo está relacionado con la clase negativa, no obstante, el modelo indica que hay una relación inversa entre esta covariable y la predicción. También podríamos pensar que el **número de menciones** estaría relacionado con la clase positiva, sin embargo, el modelo deja ver que mientras más grande es esta variable, más pequeño es el valor SHAP. La potencia del uso de los valores SHAP es que ayudan a llegar a conclusiones a las que no se puede llegar directamente después de ajustar el modelo.

A pesar de la complejidad de calcular variables provenientes del texto, es de notarse que cuatro de las cinco variables más importantes para el modelo provienen de la información financiera. Además, la variable de **Sentimiento** no figura dentro de las veinte variables más importantes lo cual indica que la ganancia de incluirla en el modelo es marginal.

El código fuente programado para este trabajo está disponible en un repositorio de GitHub[1] con fines de consulta para aquellos interesados en este estudio.

[1] https://github.com/CarolSanchezGaribay/direccion_precios_IPC_NLP.git

Capítulo 6

Conclusiones y trabajo futuro

6.1. Conclusiones y discusiones

Del presente trabajo se puede concluir que las opiniones de los usuarios en las redes sociales en conjunto con la información histórica de precios permite tener una noción de la dirección del mercado en el futuro inmediato. El modelo arrojó un AUC promedio de 0.75, lo cual quiere decir que el indicador es 25 puntos porcentuales más alto que una clasificación aleatoria, por lo tanto, esta diferencia puede ser valiosa a la hora de definir estrategias de inversión en el mercado de valores mexicano. Por otro lado, los estudios previos de NLP sobre este tema se han concentrado en la predicción de movimientos bursátiles del mercado accionario de Estados Unidos basados en el idioma inglés. Este trabajo se ha basado en un análisis cualitativo y cuantitativo de opiniones en español para predecir el movimiento del mercado mexicano alcanzando un grado de predictibilidad mejor que el azar.

El planteamiento del problema representó un gran reto técnico porque se debía modelar el comportamiento de una variable objetivo y la información que se encontraba en la literatura sobre este tipo de problemas era escasa. Se evaluaron varias opciones hasta llegar a la que se propuso en el capítulo 4. Por otro lado, el lenguaje natural y en específico el idioma español tiene muchas particularidades y su interpretación no es tarea fácil, por lo tanto, varios análisis como el de secuencia de caracteres ASCII ("emojis") y el sarcasmo tuvieron que dejarse fuera del alcance de este trabajo.

El análisis exploratorio de los datos previo al desarrollo del modelo tuvo un papel fundamental en el descubrimiento de la viabilidad del trabajo. El cálculo de correlaciones, distribuciones y medidas de tendencia de las variables permitió tener un indicio de la predictibilidad de ellas en los modelos, por ejemplo, en la construcción del modelo de sentimientos era claro que la frecuencia de ciertas palabras era más alta en la clase positiva que en la negativa y esto nos dio la señal de que el modelo podría funcionar.

La clasificación de tuits para el modelo de sentimientos fue una tarea ardua pero necesaria ya que se necesitaba tener datos de entrenamiento que estuvieran clasificados correctamente cuya fuente fuese la misma red social. Para resolver el problema se tuvo que recurrir a una intensa búsqueda de bases de datos de instutuciones confiables y, además, se clasificaron manualmente 10,000 tuits para complementar el conjunto de datos.

Las herramientas y tecnologías actuales (como el lenguaje de programación Python) facilitan el desarrollo de modelos de aprendizaje automático, no obstante, es importante conocer la justificación matemática de esos

algoritmos para llegar a las conclusiones correctas. Los algoritmos se implementaron utilizando los estándares de escritura de código en Python (PEP-8) permitiendo tener un código fácil de leer por desarrolladores distintos al autor.

Finalmente, se sabe que existe cierta desconfianza en el medio financiero acerca de la capacidad de los modelos de aprendizaje automático en la predicción de precios de acciones. Los analistas financieros trabajan con modelos basados en los estados de resultados de las compañías o modelos estadísticos clásicos como las series de tiempo y no en el impacto de los factores externos (como las redes sociales). Por este motivo, la puesta en producción de los modelos de aprendizaje automático sigue siendo muy limitado en el sector.

6.2. Trabajo futuro

Así como otros modelos estadísticos, los modelos de aprendizaje automático deben ser actualizados frecuentemente ya que el comportamiento de los mercados varía en el tiempo y puede no ser representativa para modelar los mercados actuales. Por esta razón, en este capítulo se proponen mejoras que ayuden a tener un mejor desempeño para la puesta en producción del modelo construido.

6.2.1. Volumen y frecuencia de datos

Una de las formas más comunes para mejorar el rendimiento de los modelos es incrementar la dimensión del conjunto de entrenamiento, es decir, complejizar el algoritmo a través del uso de más variables provenientes de otras fuentes como sitios web de noticias, otras redes sociales, etc. o integrar más información histórica, es decir, incrementar el periodo de observación de entrenamiento. Ambas formas tienen sus pros y sus contras por lo que deben volver a calcularse las métricas de validación del modelo para ponerse en producción. Por ejemplo, agregar variables que no incrementen el rendimiento del modelo provocará que el algoritmo sea incapaz de generalizar, mientras que agregar información historica de mucho tiempo atrás puede no representar el comportamiento más reciente de los mercados y empeorar el ajuste.

Hoy en día toda la información proveniente de las redes sociales es actualizada instantaneamente ya que hay millones de usuarios que interactuan todo el tiempo en ellas. De la misma manera, el precio de las acciones tiene cambios intantaneos de dirección que pueden impactar severamente si el volumen de transacción de los inversionistas es alto, esto conduce a pensar que se podría usar la información inmediata anterior para hacer predicciones, terminando en una técnica conocida como *scalping* que se trata de hacer transacciones bursátiles en cuestión de minutos e incluso segundos. Esta técnica se caracteriza por tener ganancias marginales pero es ampliamente utilizada por los traders actuales ya que se pueden diseñar *bots* que ejecuten las transacciones automáticamente.

La utilización de más datos de entrenamiento o de información en tiempo real puede requerir del uso de otro tipo de tecnologías de *streaming*, por ejemplo la ingesta de datos mediante Flume o Kafka y el almacenamiento en bases de datos no relacionales como MongoDB o Elasticsearch. Por este motivo, debe definirse claramente el alcance de este tipo de modelos.

6.2.2. Indicador de oportunidad de venta

Si se plantea una estrategia de compra de acciones mediante el uso de un algoritmo de aprendizaje automático, suena razonable que exista otra estrategia para la venta de los activos así que puede definirse esa variable objetivo como

$$y_d = \begin{cases} 1 & \text{si } 1.005 * \sum_i p_{d-1,i} \geq \sum_i p_{d,i} \\ 0 & e.o.c. \end{cases} . \tag{6.1}$$

Contrastando con la figura 4.4 del capítulo 4 y tomando como ejemplo un punto de corte en la variación del precio de $-0.5\,\%$, la figura 6.1 muestra las oportunidades de venta segun el indicador definido en la ecuación 6.1.

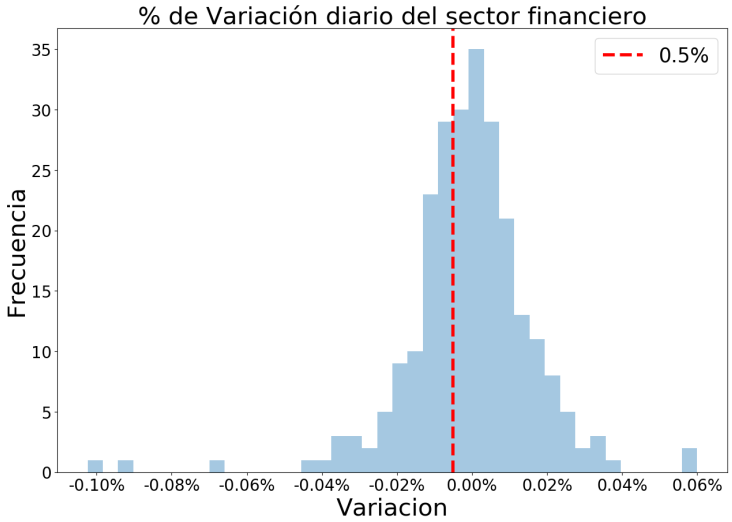


Figura 6.1: Variación porcentual del precio de un día vs el precio inmediato anterior.

Como la variación negativa de precios tiene mayor sesgo, hay más días en los que hay oportunidades de venta que de compra según el indicador construido.

6.2.3. Opiniones sarcásticas

En procesamiento del lenguaje natural y particularmente en el análisis de sentimientos, las opiniones pragmáticas dificultan el análisis, ya que los modelos de aprendizaje automático siguen reglas de sintaxis y semántica, mientras que las opiniones pragmáticas se basan en el contexto social y cultural como los tabúes y los modales. Por este motivo, se deben construir algoritmos que modelen la estructura del lenguaje y su relación con el significado expresado.

En su tesis de maestría, Mika Hämäläinen [11] desarrolla un modelo de aprendizaje automático basado en el idioma español (con una precisión de $64.8\,\%$) que identifica opiniones sarcásticas dentro de un texto a través del reconocimiento de rasgos característicos de este tipo de sentencias. Este modelo puede complementarse con el modelo de sentimientos construido en el presente trabajo para robustecer la variable de sentimiento que se usó.

Apéndice A

Dualidad de Wolfe

Algunos problemas de maximización/minimización requieren de optimizar una función sobre alguna superficie (o intersección de superficies). Una de las técnicas más usadas para para resolver este tipo de problemas es llamado método de *multiplicadores de Lagrange*.

El problema consiste en encontrar el máximo o mínimo de una función $f(x_1, ..., x_n)$ denominada *función objetivo* sobre un conjunto definido por una función de restricción $g(x_1, ..., x_n) = c$ que normalmente es una superficie $n-1$-dimensional.

Definición A.1 *Una curva de nivel de una función $f : \mathbb{R}^n \to \mathbb{R}^n$ es la curva con ecuación $f(x_1, ...x_n) = c$ tal que c es una constante en el rango de f.*

Intuitivamente, si f está en un punto máximo o mínimo a lo largo de la curva de restricción $g = c$ entonces la curva de nivel de f debe ser tangente a la curva de restricción y, por lo tanto, ∇f es paralela a ∇g en este punto.

Proposición A.1 *Sean $f : \mathbb{R}^n \to \mathbb{R}^n$ y $g : \mathbb{R}^n \to \mathbb{R}$ de clase C^1 (con derivadas parciales continuas) y $\mathbf{x_0} \in \mathbb{R}^n$ un mínimo local de f sobre el conjunto de restricción $g(\mathbf{x}) = c$. Esto es, $g(\mathbf{x_0}) = c$, y existe $\epsilon > 0$ tal que $\forall \mathbf{x} \in \mathbb{R}^n$ tal que $g(\mathbf{x}) = c$ y $||\mathbf{x} - \mathbf{x_0}|| < \epsilon$ tenemos que*

$$f(\mathbf{x_0}) \leq f(\mathbf{x}).$$

Suponiendo que $\nabla f(\mathbf{x_0}) \neq \mathbf{0}$, entonces existe $\lambda \in \mathbb{R}^n$ tal que

$$\nabla f(\mathbf{x_0}) = \lambda \nabla g(\mathbf{x_0}).$$

donde λ es el multiplicador de Lagrange.

Considerando un problema de optimización

$$\min_x f(x) \tag{A.1}$$

sujeto a

$$g(x) \geq 0, \tag{A.2}$$

donde f y g son funciones continuamente diferenciables, tenemos dos casos:

1) Cuando $g(x) = 0$ el problema se resuelve con los multiplicadores de Lagrange tal que

$$\frac{\partial f}{\partial x} - \lambda \frac{\partial g}{\partial x} = 0 \tag{A.3}$$

además, $\lambda > 0$ puesto que si no fuera así, se podría disminuir $f(x)$ en dirección contraria a $\frac{\partial f}{\partial x}$ sin salir del conjunto factible definido por $g(x) \geq 0$.

2) Cuando $g(x) > 0$, se requiere estar en un máximo de $f(x)$ para que $\frac{\partial f}{\partial x} = 0$ y $\frac{\partial f}{\partial x} - \lambda \frac{\partial g}{\partial x} = 0$ con $\lambda = 0$.

En cualquier caso, el sistema de ecuaciones de las condiciones de Karush-Kuhn-Tucker, se conservan:

$$\begin{gathered} \lambda g(x) = 0, \\ \lambda \geq 0 \text{ y} \\ g(x) \geq 0. \end{gathered}$$

Si suponemos ahora que $f(x)$ y $g(x)$ son convexas, y se define

$$L(x, \lambda) = f(x) - \lambda g(x) \tag{A.4}$$

entonces la ecuación A.3 es equivalente a

$$\frac{\partial L}{\partial x} = 0$$

y para cualquier $\lambda \geq 0$, L es convexa y, por lo tanto, tiene un solo mínimo (ínfimo). Además, para cualquier x, L es lineal en λ.

Se define

$$h(\lambda) = \text{mín}_x \, L(x, \lambda)$$

Proposición A.2 (Dualidad débil) *Consideremos el siguiente modelo lineal conocido como* ***primal***

$$max z = \mathbf{c}^T \mathbf{x}$$

sujeto a

$$\begin{gathered} \mathbf{A}\mathbf{x} \leq \mathbf{b} \\ \mathbf{x} \geq \mathbf{0} \end{gathered}$$

y el modelo ***dual***

$$min v = \mathbf{b}^T \mathbf{y}$$

sujeto a

$$\begin{gathered} \mathbf{A}^T \mathbf{y} \geq \mathbf{c} \\ \mathbf{y} \geq \mathbf{0} \end{gathered}$$

se cumple que $z = \mathbf{c}^T \mathbf{x} \leq \mathbf{b}^T \mathbf{y} = v$

En otras palabras, la proposición anterior nos indica que el valor máximo de la función objetivo primal es una cota inferior del mínimo de la función objetivo dual. Y, reciprocamente, el valor mínimo de la función objetivo dual es una cota superior del máximo de la función objetivo primal.

El mínimo de un conjunto de funciones lineales es concavo y su máximo corresponde a la función lineal con derivada igual a cero. Por lo tanto, $h(\lambda)$ también tiene un único máximo sobre $\lambda \geq 0$. Ya sea que:

1) el máximo de h se encuentre cuando $\lambda = 0$, en cuyo caso

$$h(0) = \text{mín}_x\, L(x, 0) = min_x f(x)$$

y se tenga un mínimo global de f

2) el máximo de h ocurra en

$$\frac{\partial L}{\partial \lambda} = g(x) = 0$$

y estemos en el límite del conjunto factible definido por g.

En cualquier caso, el problema

$$max_\lambda h(\lambda)$$

sujeto a

$$\lambda \geq 0$$

es equivalente a

$$max_{\lambda,x} L(x, \lambda)$$

sujeto a

$$\begin{array}{c} \lambda \geq 0, \\ \frac{\partial L}{\partial x} = 0 \end{array}$$

lo cual es conocido como el problema de dualidad de Wolfe que tiene una restricción no necesariamente lineal por lo que el problema de optimización es no convexo, sin embargo, la dualidad débil se mantiene.

Apéndice B

Valores SHAP

En regresión lineal, darle interpretabilidad a un modelo es trivial puesto que el efecto de cada variable es el tamaño del cambio multiplicado por el valor de la variable (debido a la linealidad); sin embargo, en modelos más complejos, se requiere una solución distinta como la que se propone con los valores SHAP que fueron creados por Lundberg y Lee en el año 2016 y tienen la finalidad de dar interpretabilidad a las predicciones de modelos complejos de aprendizaje de máquina. Estos valores estan basados en un método de Teoría de Juegos llamado valor de Shapley y nombrado así en honor a Lloyd Shapley quien lo introdujo en 1953 y que plantea la distribución equitativa de riqueza en juegos cooperativos.

B.0.1. Valores de Shapley

Dado un grupo M de m jugadores y $v : 2^M \to \mathbb{R}$ una función tal que, si S es una coalición de jugadores, $v(S)$ denota la súma de la ganancia que los miembros de S pueden obtener de dicha cooperación. La cantidad justa que el jugador i obtiene durante el juego (v, M) es:

$$\phi_i(v) = \sum_{S \subseteq M \setminus \{i\}} \frac{|S|!(m - |S| - 1)!}{m!}(v(S \cup \{i\}) - v(S)). \tag{B.1}$$

La interpretación de la fórmula anterior es que, dada la contribución $v(S \cup \{i\}) - v(S)$ del jugador i y las diferentes permutaciones con que se puede formar la coalición, la ganancia $\phi_i(v)$ será el promedio ponderado de dicha contribución en todos los escenarios posibles.

El valor de Shapley se considera como la única distribución "justa"de ganancias ya que cumple con propiedades deseables que se enumeran a continuación:

- Eficiencia: La ganancia total se distribuye entre los jugadores:

$$\sum_{i \in M} \phi_i(v) = v(M) \tag{B.2}$$

- Simetría: Si i y j son dos jugadores equvalentes tales que $v(S \cup \{i\}) = v(S \cup \{j\})$ para cada subconjunto S que no contiene i ni j, entonces

$$\phi_i(v) = \phi_j(v) \tag{B.3}$$

- Linealidad: Si dos juegos cooperativos con funciones de ganancia v y w son combinados, entonces la ganancia total corresponde a la ganancia marginal de v y w

$$\phi_i(v + w) = \phi_i(v) + \phi_i(w) \tag{B.4}$$

también, $\forall a \in \mathbb{R}$

$$\phi_i(av) = a\phi_i(v)\ \forall i \in M \tag{B.5}$$

- Jugador Nulo: Un jugador i es nulo en v si $v(S \cup i) = v(S)$

B.0.2. Valores SHAP

Como se mencionó antes, los valores SHAP estan basados en los valores Shapley. La tabla B.1 muestra la equivalencia entre dichos valores para su aplicación en modelos de aprendizaje de máquina.

Nomenclatura	Teoría de Juegos	Modelos
f	Juego	Modelo
M	Jugadores	Variables
S	Subconjunto de jugadores	Subconjunto de variables
i	Jugador específico	Variable específica
x		Observación a explicar

Tabla B.1: Tabla de equivalencia entre los valores Shapley y los valores SHAP.

Por ejemplo, si para una compañía del sector de consumo, con 100,000 operaciones y $1B transaccionado, se predice un precio por acción de $100 y el precio promedio es de $80, los SHAP responderán a la pregunta de cuánto contribuye cada característica a esta predicción comparado con el promedio. La respuesta podría ser que el sector contribuyó $5, las operaciones $10 y el importe $5 para llegar a los $20 de diferencia entre la predicción y el promedio.

Bibliografía

[1] Tibshirani, R (1996), Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*

[2] Tibshirani, R., Hastie, T. y Friedman, J. (2009), The elements of statistical learning: data mining, inference and prediction (2a ed.). *Springer*.

[3] Vapnik, V., (1999), The nature of statistical learning theory (2a ed.). *Springer*.

[4] Wasserman, L., (2004), All of statistics: A concise course in statistical inference. *Springer*.

[5] Lehmann, E.L. y Casella, G. (1998), Theory of point estimation (2a ed.). *Springer*.

[6] Lang, S. (1987), Calculus of several variables (3a ed.). *Springer*.

[7] Clark, A., Fox, C. y Lappin, S. (2010), The Handbook of Computational Linguistics and Natural Language Processing. *Wiley-Blackwell*.

[8] Lundberg, S. y Lee, S. (2017), A Unified Approach to Interpreting Model Predictions.

[9] Shapley, L. (1953), A Value for n-person Games. In Contributions to the Theory of Games, volumen II. *Princeton University Press*

[10] Escandell, M. V. (1996), Introducción a la pragmaática. *Ariel*.

[11] Hämäläinen, M. (2016), Reconocimiento automático del sarcasmo: ¡Esto va a funcionar bien! (tesina de máster). *Universidad de Helsinki*.

[12] McNamara, T. y Roever, C. (2006), Language testing: The social dimension. *Blackwell*.

[13] Villena-Román, J., Lana-Serrano, S., Martínez-Cámara, E., González-Cristobal, J.C. (2013), TASS - Workshop on Sentiment Analysis at SEPLN. Procesamiento del Lenguaje Natural, 50. http://journal.sepln.org/sepln/ojs/ojs/index.php/pln/article/view/4657