

Implementación de un algoritmo de *machine learning* supervisado para la predicción del comportamiento de las importaciones de teléfonos móviles celulares en Colombia

Diego Armando Martínez Flórez

Rodrigo Alejandro Meneses Castro



Ingeniería de Producción

Facultad Tecnológica

Universidad Distrital Francisco José de Caldas

Bogotá D.C.

2021

Implementación de un algoritmo de *machine learning* supervisado para la predicción del comportamiento de las importaciones de teléfonos móviles celulares en Colombia

Diego Armando Martínez Flórez - 20172377012

Rodrigo Alejandro Meneses Castro - 20172377080

Director: Ing. Javier Parra Peña

Trabajo de grado presentado como requisito para optar al título de Ingenieros de Producción

Modalidad: Monografía



Ingeniería de Producción

Facultad Tecnológica

Universidad Distrital Francisco José de Caldas

Bogotá D.C.

2021

Nota de aceptación

Firma del director

Firma del jurado

Bogotá, 2021

Contenido

1. Generalidades	6
1.1. Problema	6
1.2. Objetivos	7
1.2.1. General.....	7
1.2.2. Específicos	7
1.3. Alcance	7
1.4. Justificación	8
2. Marco teórico	10
2.1. Importaciones	10
2.1.1. Instituciones que intervienen en la importación de teléfonos.....	10
2.1.2. Importaciones de teléfonos móviles celulares y de otras redes inalámbricas	11
2.2. Inteligencia computacional	14
2.2.1. Machine learning	14
2.2.2. Aplicaciones del machine learning.....	16
2.2.3. Tipos de aprendizaje automático	18
2.2.3.1. Aprendizaje supervisado.....	18
2.2.3.1.1. Regresión lineal simple.....	19
2.2.3.1.2. Regresión lineal múltiple.....	20
2.2.3.1.3. Árboles decisión de regresión.....	29
2.2.3.1.4. Máquinas de vectores de soporte de regresión.	30
2.2.3.2. Aprendizaje no supervisado.....	31
2.2.3.3. Aprendizaje por refuerzo.	32
2.2.4. Lenguaje de programación Python	33
2.2.4.1. Anaconda.	34
2.2.4.2. Spyder.	35
2.2.4.3. Bibliotecas.....	36
3. Metodología	39

3.1. Preparación de los datos	40
3.1.1. Variables estudiadas.....	42
3.1.2. Cargue de conjunto de datos	50
3.2. Representación de los datos	51
3.3. Modelado y aprendizaje	58
3.3.1. Multicolinealidad	58
3.3.2. Homocedasticidad.....	60
3.4. Evaluación	63
3.5. Paso a producción	67
3.5.1. Predicción del modelo Año 2021.....	73
4. Resultados	75
4.1. Modelo 1 (Operaciones)	76
4.2. Modelo 2 (Diario)	78
4.3. Modelo 3 (Semanal)	80
4.4. Modelo 4 (Mensual)	84
5. Conclusiones	89
6. Trabajos futuros	92
Bibliografía	93
Lista de Tablas	96
Lista de Figuras	96
Lista de Ecuaciones	99
Anexos	100

1. Generalidades

1.1. Problema

En la actualidad se cuenta con un alto flujo de información y datos que ha generado la necesidad de desarrollar herramientas que permitan su adecuado procesamiento y análisis para la toma de decisiones en situaciones prácticas. Frente a este desafío, y debido a la complejidad que poseen tales escenarios de resolución de problemas, se hace necesario que los modelos tradicionales utilizados para tal fin se desarrollen en el marco de las tendencias tecnológicas de la actualidad; es así que han surgido ramas como el *machine learning* —aprendizaje de máquina por su terminología en inglés—, las cuales se basan en la creación de técnicas que incorporan los principios estadísticos básicos e introducen nuevas funciones en términos de automatización y manejo de alto volumen de datos, mejorando con ello los tiempos y la confiabilidad en la toma de decisiones para cualquier entorno que se quiera considerar (Martin, 2017).

A propósito de esto, uno de los problemas más dificultosos de la actualidad es el análisis de datos de operaciones de comercio exterior, pues debido a las condiciones del mercado y a ciertos factores económicos y políticos se ha generado un conjunto de variables macroeconómicas que dificultan el examen de este tipo de información bajo un modelo tradicional. Precisamente, es este el caso de los teléfonos móviles celulares, cuya importación ha crecido en una tasa promedio del 17.35 % en los últimos 10 años (véase *Tabla 1*), correspondientes a 89.646 embarques que implican organizar una gran cantidad de datos particulares como seriales, valor de impuestos, países de origen, valor de los equipos, descripciones, entre otros, dando como resultado información muy extensa para ser estudiada con rigurosidad. Por todo ello, y con el ánimo de hacer frente a este desafío, en el proyecto que aquí se presenta se buscó predecir el comportamiento de las importaciones de telefonía celular para los años 2020 y 2021 por medio de la implementación

de un modelo de *machine learning* (ML) supervisado, empleando el lenguaje de programación *Python* que actualmente está encabezando la lista de los lenguajes de programación con mayor versatilidad en el manejo de datos y en el desarrollo de arquitecturas del lado del servidor.

1.2. Objetivos

1.2.1. General

Implementar un algoritmo de *machine learning* supervisado para la predicción del comportamiento de las importaciones de teléfonos móviles celulares en Colombia para los años 2020 y 2021.

1.2.2. Específicos

- Analizar y probar algoritmos de entrenamiento de *machine learning* supervisados que se adapten a problemas de predicción.
- Evaluar el algoritmo seleccionado mediante los resultados obtenidos para corroborar la efectividad del modelo.
- Elaborar una predicción del comportamiento de las importaciones en Colombia de teléfonos móviles celulares con base al (o los) algoritmo(s) seleccionado(s) y evaluado(s).

1.3. Alcance

En este proyecto se emplearon los datos de importación en Colombia de la subpartida 8517120000 —la cual dentro de los términos arancelarios abarca los productos de teléfonos móviles celulares y los de otras redes inalámbricas— que abarca los años 2009 a 2019 y que fueron publicados por la DIAN. Además, se utilizó un modelo de *machine learning* supervisado por

medio del uso de bibliotecas del lenguaje de programación *Python* en el entorno de trabajo “Spyder” de Anaconda.

1.4. Justificación

En los últimos 10 años los teléfonos móviles celulares se han convertido en un elemento indispensable para todos los colombianos, por lo cual es natural que la importación de este tipo de productos haya tenido un crecimiento acelerado: el país pasó de 2.383.569 millones de unidades importadas en 2009 a 10.935.236 en 2019; además, durante este periodo de tiempo han ingresado al territorio colombiano más de 85.277.947 equipos celulares, lo que da cuenta del carácter indispensable que se le ha otorgado a su adquisición y uso en el país. En efecto, los avances en cuanto a tecnología, prestaciones, conectividad y redes móviles, así como las medidas de exoneración de arancel y la reciente exclusión del Impuesto al Valor Agregado (IVA) para los celulares que no superen los 22 puntos de la Unidad de Valor Tributario (UVT) que se ha dado a nivel mundial, han contribuido al crecimiento de las importaciones (véase *Tabla 1*), generando además un mercado interno para la distribución y venta de equipos, tanto como en lo que refiere a la venta de accesorios, servicios de reparación y mantenimiento, por lo que muchos colombianos dependen económicamente de su demanda.

Tabla 1. *Total transacciones de importación de la subpartida 8517120000*

AÑO	CANTIDAD DE IMPORTACIONES	PORCENTAJE DE CRECIMIENTO
2009	2812	
2010	4195	49,2%
2011	5038	20,1%
2012	5661	12,4%
2013	6589	16,4%
2014	7880	19,6%
2015	9431	19,7%
2016	10369	9,9%
2017	12301	18,6%
2018	12152	-1,2%
2019	13218	8,8%
TOTAL	89646	

Fuente: elaboración propia con base en los datos suministrados por la DIAN

Además, estudiar el comportamiento de la importación de estos productos resulta importante para los mercados que se han generado en torno suyo, ya que hay diversos factores económicos como la volatilidad del dólar que han ocasionado que su tráfico internacional disminuya, por lo que el método que se empleó permite evaluar las diversas variables que pueden influir en tal demanda, adaptarse a los cambios que puedan llegar a tener estas con el fin de disminuir al máximo la incertidumbre y el error, así como a la tendencia y la naturaleza de los datos.

Para el caso del presente proyecto se utilizó un modelo de aprendizaje supervisado debido a la conveniencia con el tipo de datos que se recopilaban ya que estos tenían unos resultados que estaban dados en función de las variables de entrada, lo cual generó la necesidad de estudiar la relación que existía entre estos dos conjuntos. Esto hizo que se hiciera dispensable la utilización del modelo de aprendizaje no supervisado que por el contrario no requiere de insumo datos comparativos para el proceso de aprendizaje, lo cual lo hace útil para la implementación en otro tipo de problemas.

2. Marco teórico

A continuación se presentarán conceptos básicos con respecto al tema de importaciones, en el cual se abordará sobre el contexto de las importaciones en el país haciendo hincapié en la importación de teléfonos móviles celulares de acuerdo con la subpartida emitida por el organismo de aduanas. Así mismo se incluye un segundo capítulo en el cual se realiza un repaso sobre la teoría de la inteligencia computacional, haciendo énfasis en el machine learning, en los tipos de aprendizaje y en las técnicas estadísticas que se involucran en este, para finalmente cerrar con el desglose del lenguaje de programación utilizado y los entornos de trabajo de este.

2.1. Importaciones

2.1.1. Instituciones que intervienen en la importación de teléfonos

El ente que regula todas las operaciones de comercio exterior en el país es el Ministerio de Comercio, Industria y Turismo, el cual “promueve el desarrollo económico y el crecimiento empresarial, impulsa el comercio exterior y la inversión extranjera y fomenta el turismo, fortaleciendo el emprendimiento, la formalización, la competitividad, la sostenibilidad y el posicionamiento de las empresas en el mercado local e internacional” (MINCIT, 2020) y cuya estructura misional se encuentra contenida en el Decreto 210 de 2003. En definitiva, entre otras cosas esta entidad se encarga de asesorar al gobierno nacional en los lineamientos que debe adoptar para el establecimiento de políticas relacionadas con el comercio exterior en la búsqueda de aumentar la competitividad y la industria del país.

También es importante mencionar al Banco de la República, el cual tiene una función fundamental en las operaciones de comercio exterior y específicamente en las importaciones de teléfonos móviles celulares al intervenir el mercado de divisas y regular la inflación, pues ello

puede afectar el poder adquisitivo que tienen los colombianos para obtener todo tipo de productos (Lozano y otros, 2019) e influir en un indicador que en materia de comercio exterior es clave y atañe al volumen de importaciones: la tasa representativa del mercado o TRM.

Otra entidad relevante es la Dirección de Impuestos y Aduanas Nacionales (DIAN) que tiene dos ejes que son fundamentales para el país: el primero es la recaudación y administración de impuestos como el tributo a las ventas, a la renta y a otros que no tenga asignados otra entidad (DIAN, 2020); el segundo es el eje que resulta más relevante para el presente trabajo, pues tiene que ver con su papel de ente para las operaciones de comercio exterior incluyendo el recaudo de impuestos como el arancel, así como verificando que todos los usuarios aduaneros cumplan con la normatividad y los procedimientos establecidos, imponiendo las respectivas sanciones en caso de detectar irregularidades.

2.1.2. Importaciones de teléfonos móviles celulares y de otras redes inalámbricas

El Decreto 2685 de 1999 define la actividad de importación como “la introducción de mercancías de procedencia extranjera al territorio aduanero nacional [así como la] la introducción de mercancías procedentes de Zona Franca Industrial de Bienes y de Servicios, al resto del territorio aduanero nacional”. Precisamente, es este el caso de los teléfonos móviles celulares — pertenecientes a la subpartida 8517120000— que se distribuyen en el mercado colombiano, pues ellos provienen prácticamente en su totalidad del exterior, principalmente de países como China, Vietnam, México y Estados Unidos. De hecho, por su alta demanda y por la actual problemática que tiene el país con respecto a su hurto, en el 2015 se creó una norma específica para su nacionalización por medio del Decreto 2025 en donde se prohíbe su importación por medio de tráfico postal o envíos urgentes, además de permitir que un viajero pueda registrar una máximo de 3 equipos; esto deriva en que todos los equipos deban ser declarados por medio de importación

ordinaria y ser registrados en las bases de datos publicadas por la DIAN. Asimismo, los números de IMEI (*International Mobile Equipment Identity*) deben ser registrados en las bases de datos del Ministerio de Tecnología de la Información y las Comunicaciones (MINTIC) y contar con el certificado de homologación.

Ahora bien, para la importación de productos al territorio nacional se debe pagar un impuesto al Estado denominado *arancel de aduanas*, el cual tiene dos objetivos principales: recaudar fondos para la nación y proteger a los productores nacionales (Nieto, 2016) para evitar la llegada de artículos importados a un menor costo y que los consumidores nacionales prefieran adquirir estos en desmedro de la producción interna. Asimismo, el Estado también puede regular estos aranceles para incentivar la adquisición de artículos que sean de vital necesidad y no sean producidos en el país como es el caso de algunos suministros y equipos médicos, tanto como para la importación de productos que sirvan para incentivar la manufactura del país y con ello mejorar el Producto Interno Bruto (PIB). Con todo, al convertirse los teléfonos móviles celulares —más puntualmente los *smartphones* o teléfonos inteligentes— en herramientas de acceso a las Tecnologías de la Información y las Comunicaciones (TIC), y por ese camino a diversos campos del conocimiento, se decretó en el año 2017 un cobro de arancel de 0% para estos productos bajo el decreto 1563 de 2017; además, si su valor no supera los 22 UVT, les será aplicada la exclusión de IVA, según la medida puesta en circulación bajo el decreto único reglamentario 1625 de 2016 y solo si los equipos cuentan con las siguientes características:

- Cuentan con teclado completo, táctil o físico
- Operan sobre sistemas operativos actualizables
- Tienen capacidad de procesamiento y cómputo
- Permiten la navegación en Internet

- Tienen conectividad WIFI
- Tienen acceso a tiendas de aplicaciones y soportan las aplicaciones hechas por terceros

Para determinar el arancel de los productos se debe seguir la estructura establecida en el Decreto 2153 de 2016, la cual se divide en 21 secciones y 98 capítulos; en esta, los teléfonos móviles celulares se encuentran clasificados en la sección XVI bajo el capítulo 85¹ y su partida arancelaria es la 8517.12.0.000 correspondiente a estos y otras redes inalámbricas. Justamente, esta clasificación es muy importante para el presente trabajo ya que se trata de una nomenclatura que permite individualizar técnicamente el producto de estudio con otro tipo de equipos de recepción o emisión de frecuencias en las bases de datos consultadas.

Por su parte, para la liquidación de los impuestos que debe pagar un importador al Estado se debe determinar el valor en aduanas y este se debe liquidar en dólares americanos (USD), por lo que cualquier importación que se encuentre en otra moneda debe liquidarse con la tasa de cambio vigente (Art. 15 del Decreto 1165 2019). El valor en aduanas contiene todos los gastos de la mercancía para la importación, entre ellos el Valor FOB (*Free on Board*²) en donde se incluyen los costos que debe asumir el importador por costo de producto, embalaje, transporte local y agenciamiento en el país de origen (Legiscomex, 2010), con lo que este valor no solo está sujeto al precio del artículo a importar sino que también puede aumentar o disminuir dependiendo de los costos logísticos y la regulación aduanera del país donde se compre; por demás, sumado a este

¹ Tal sección se denomina *Máquinas y aparatos, material eléctrico y sus partes* y el respectivo capítulo *Máquinas, aparatos y material eléctrico, y sus partes; aparatos de grabación o reproducción de sonido, aparatos de grabación o reproducción de imagen y sonido en televisión, y las partes y accesorios de estos aparatos*. Además, el tema está recogido en la partida 85.17 cuyo nombre es *Teléfonos, incluidos los teléfonos móviles celulares) y los de otras redes inalámbricas; los demás aparatos de emisión, transmisión o recepción de voz, imagen u otros datos, incluidos los de comunicación en red con o sin cable (tales como redes locales (LAN) o extendidas (WAN)*.

² Término perteneciente a los términos comerciales internacionales.

valor FOB se halla el valor del transporte internacional y el costo del seguro de la carga. Así pues, con estos tres rubros definidos, y realizando la respectiva conversión a dólares americanos, se determinan los tributos aduaneros que se deben pagar para realizar la importación. Al valor FOB de la mercancía, más el valor del seguro y el flete se denomina valor CIF (*Cost, Insurance and Freight*³).

2.2. Inteligencia computacional

La inteligencia computacional es un concepto que nace de la necesidad de estudiar el comportamiento de agentes inteligentes; es decir, cualquier elemento de un sistema que se encargue de ejecutar algún tipo de acción. De esta manera, se puede decir que un agente de esta clase “es un sistema que actúa de manera inteligente: lo que hace es apropiado para su circunstancia y su objetivo” (Poole, Mackworth y Goebel, 1998) y permite que se pueda someter a entornos y objetivos cambiantes enfocados en el estudio de sistemas naturales o artificiales que, en últimas, son el propósito principal de la inteligencia computacional. Ella, por cierto, viene a ser una rama de la inteligencia artificial y busca especializarse en la creación de programas que en cierta manera resulten inteligentes, tales como el estudio de elementos de la lógica difusa, el aprendizaje, la evolución y la adaptación a entornos cambiantes; es así como surge el *machine learning*, especialidad que se encarga de estudiar las técnicas de desarrollo de aprendizaje enfocado a las computadoras⁴.

2.2.1. *Machine learning*

Como rama de la inteligencia computacional, a través del *machine learning* se desarrollan técnicas que posibiliten que las máquinas aprendan. Frente a esto, Nilsson (1998) afirma que una

³ Término perteneciente a los términos internacionales de negociación

⁴ *Machine learning* podría traducirse como aprendizaje automático o aprendizaje de máquinas.

máquina aprende “cada vez que cambia su estructura, programa o datos —en función de sus entradas o en respuesta a información externa— de tal manera que mejore su rendimiento futuro esperado” (p. 1), por lo que, al requerir del análisis de datos, se tiende a comparar con la estadística inferencial. A este respecto, pese a que no es posible negar que el *machine learning* se apoya en diversos insumos de la estadística, existe una clara distinción entre ambas pues este último se enfoca más en solucionar el problema de la complejidad computacional tanto como en la detección de patrones de comportamiento, mientras que la primera no. Además, también genera un agregado que tiene que ver con la automatización de procesos: si bien la estadística se vuelve necesaria, con el *machine learning* se puede generar el valor agregado de la automatización de procesos por medio de métodos algorítmicos. Un ejemplo como el empleado por Shwartz (Shwartz y David, 2014) en torno al aprendizaje animal viene bien para comprender esto: a un roedor se le pone un trozo de alimento al que habrá de dar un pequeño mordisco y, de acuerdo con su experiencia previa con el sabor y el efecto psicológico de peligro y enfermedad, tendrá dos opciones: le podrá generar una sensación de seguridad por la cual podrá seguir tomando trozos del alimento o desistir y alejarse del mismo. Incluso, Alpaydin plantea la utilidad en los modelos de *machine learning* en un ejemplo más aplicado (Alpaydin, 2010): si se requiere generar un algoritmo que pueda filtrar los correos electrónicos que representan spam, y se conoce que la salida debe ser del tipo Si / No, se deben evaluar las entradas que permitan determinar si el correo electrónico representa spam; sin embargo, estas no son claras ni se encuentran definidas, para lo cual se debe evaluar la información por medio de palabras claves o contenido que pueda permitir al algoritmo cumplir con su tarea.

Con todo, estas comparaciones son importantes ya que los modelos de *machine learning*, al igual que la rata del experimento y el algoritmo para filtrar el correo spam, emplean la experiencia y más específicamente la información suministrada para resolver los problemas que

son resueltos por estos métodos. En suma, como lo expresa Pineda (2017), se trata de crear nuevos algoritmos para diversos problemas que puedan generar modelos sin la intervención o asistencia humana, lo que permite resolver una gran variedad de conflictos que de la manera tradicional no sería posible solventar ya que pueden ser difíciles de caracterizar.

Ahora bien, pese a que la idea del *machine learning* no es nueva —ante todo por los desarrollos tecnológicos con máquinas con una mayor capacidad de procesamiento y de almacenamiento de datos— se ha vuelto relevante, entre otras cosas debido a la disponibilidad de una gran cantidad de data de diversos campos que ha hecho que muchas empresas adopten este tipo de algoritmos para resolver problemas de la vida real (*Ibid.*, 2017). También es importante mencionar que el *machine learning* se ha hecho más extensivo debido a que en la actualidad muchos algoritmos se encuentran disponibles gratuitamente a través de comunidades en línea, contando con una mayor capacidad de almacenamiento y de acceso a la publicación de bibliotecas que facilitan el trabajo a los desarrolladores y personas del común que quieren incursionar en este campo (Hurwitz y Kirsch, 2018); algunos de estos, de hecho, fueron empleados en el presente trabajo para el análisis de la información, como por ejemplo el sistema de gestión de paquetes *Anaconda* del lenguaje de programación *Python*, o bien el recurso dispuesto por Google denominado *Colab*. En general, estas herramientas permiten el procesamiento de los datos y el uso de los algoritmos sin tener conocimientos avanzados en programación, lo que ayuda al analista a emplear el algoritmo adecuado e interpretar los resultados para la solución de problemas.

2.2.2. Aplicaciones del *machine learning*

El crecimiento generalizado del *machine learning* se ha dado a partir de aportes de diversas áreas, lo que ha redundado en que ellas mismas se beneficien del uso de este: no es entonces raro encontrar que un determinado campo le haya brindado algunos aportes a la construcción del ML

y que, a un tiempo, se haya apoyado en él para avanzar en metodologías que optimizan la ejecución de procesos, como pueden ser:

- La **estadística**, en la que se ve a menudo el uso de estimadores para encontrar un valor a partir del conjunto de puntos de una muestra, de donde es posible colegir que “los métodos estadísticos para tratar estos problemas pueden considerarse casos de aprendizaje automático porque las reglas de decisión y estimación dependen de un corpus de muestras extraídas del entorno del problema” (Pineda Cortés, 2017). Asimismo, es de allí que el *machine learning* se ha apoyado para fundamentar algunos de sus algoritmos y, a su vez, la estadística se ha aprovechado de estos para ser utilizados en algunas aplicaciones de la estadística en general, permitiendo la mejora en la ejecución de estos e incluso permitiéndole aumentar su capacidad de análisis
- Los **modelos cerebrales**, que refieren al estudio del comportamiento del cerebro y en donde también ha dado pautas interesantes al momento de desarrollar algoritmos de *machine learning*, sobre todo en el estudio de modelos no lineales como las *redes neuronales*
- La **teoría de control adaptativo**, en donde surgen a menudo cambios que obligan a las máquinas a adaptarse a estos para poder ejecutar su función de manera adecuada; es así como se han adaptado algunas aplicaciones del *machine learning* a que aprendan a reconocer entradas sensoriales con las que cuentan estos sistemas para poder dar respuesta oportuna
- Los **modelos evolutivos**, pues el estudio de la biología también ha permitido generar aportes al *machine learning* tales como la identificación de patrones evolutivos en las especies para poder imitar el comportamiento de estos en la utilización de algoritmos;

así pues, aunque no los copian de manera idéntica, sí intentan manejar ciertas similitudes que han ayudado en la solución de problemas de cualquier campo que haya sido intervenido con *machine learning*.

2.2.3. Tipos de aprendizaje automático

En el campo del *machine learning* se puede evidenciar la prevalencia de tres tipos de aprendizaje, los cuales han sido el resultado de agrupar diversas técnicas en conjuntos que manejan similitud en su forma de proceder: el *aprendizaje supervisado*, el *aprendizaje no supervisado* y el *aprendizaje reforzado*.

2.2.3.1. Aprendizaje supervisado.

El aprendizaje supervisado se conoce principalmente por tener datos de entrenamiento y datos deseados que son los de prueba. Básicamente su objetivo central tiene que ver con “aprender un modelo a partir de datos de entrenamiento etiquetados, que nos permite hacer predicciones sobre datos futuros o no vistos”. (Raschka y Mirjalili, 2019). Es aquí donde cobran importancia los datos de prueba, pues son estos los que posibilitan determinar lo acertado que resulta el modelo para la realización de predicciones; en últimas los datos de entrenamiento se tienen que parecer a los de testeo para poder dar aplicación real del modelo. Este modelo de aprendizaje tiene como punto de partida una función que puede cobrar dos tipos de valores: los numéricos o etiquetas de clase, siendo conocidos los primeros en problemas de predicción y comunes al momento de realizar regresiones, contando con un número de variables predictivas —es decir, se pueden dar a entender como las explicativas— y una variable de respuesta continua —resultado o destino deseado—; por su parte, las etiquetas de clase son conocidas en problemas de clasificación por su característica textual que permite generar una asignación que resulta legible para el código a

programar, estas clasificaciones pueden ser de tipo binario o de múltiples valores. En suma, los algoritmos más habituales que aplican para el aprendizaje supervisado son:

- Árboles de decisión
- Clasificación de *Naïve Bayes*
- Regresión por mínimos cuadrados
- Regresión Logística
- *Support Vector Machines* (SVM)
- Métodos “Ensemble” (conjuntos de clasificadores)

En esta parte se relacionan aquellos que específicamente se enfocan a la regresión ya que es el tipo de algoritmo que se emplea para el presente proyecto.

2.2.3.1.1. Regresión lineal simple.

Este es uno de los algoritmos más empleados y fundamentales de regresión. Su principal objetivo es lograr la estimación o aproximación de un valor continuo (Y) basado en un conjunto de variables independientes (X), todo lo cual a través de una recta de la forma:

Ecuación 1. Ecuación de la recta

$$y(x) = a + bx$$

donde a es el término independiente de la recta que determina el punto de corte con el eje Y y b es la pendiente o inclinación que tendrá la recta. A este par de valores también se les denomina *pesos de la regresión* y su propósito es encontrar los valores de a y b que puedan generar un modelo que permita la estimación con mayor precisión basado en una serie de datos de entrenamiento iniciales.

La regresión lineal simple también se emplea para la determinación de la dependencia de una variable continua por medio de una variable explicativa. Sin embargo, para que este método

sea empleado con fines de predicción es muy importante asegurar que exista la relación de independencia entre la variable de ingreso que debe ser estudiada, ya que si no existe esta suposición el modelo arroja resultados poco fiables. Con todo, para determinar los coeficientes de la regresión se busca minimizar el error que genera el modelo, ante lo que el método más utilizado es el de mínimos cuadrados el cual tiene como ventajas:

- Que puede aplicarse para el procesamiento de una gran cantidad de datos ya que no requiere recursos computacionales avanzados y, además,
- Que su interpretación y análisis es sencillo de realizar.

En suma, este modelo tiene como ventajas que es muy fácil de entender y de modelar, además de que es útil para aplicar en relaciones poco complejas y de las cuales se tengan pocos datos. Por otro lado, este algoritmo no es útil para entender relaciones complejas donde haya una relación entre las variables que no sea lineal, además que es muy susceptible a reducir su efectividad por datos atípicos.

2.2.3.1.2. Regresión lineal múltiple.

En los modelos de regresión lineal simple se estudia una única variable dependiente para predecir una variable independiente. No obstante, en la práctica muchos de los problemas de regresión tienen diversas variables que influyen en la formulación de un modelo. En contraste, en la regresión lineal múltiple se busca predecir una variable dependiente que es afectada por múltiples variables independientes pero que tienen una relación lineal.

Ecuación 2. Regresión lineal múltiple

$$Y = w_0 + w_1x_1 + w_2x_2 + w_3x_3 + \dots w_nx_n$$

Como en el modelo de regresión lineal simple, las variables independientes van a tener un coeficiente que debe ser calculado y una variable independiente, pese a lo cual en este caso no se habla de una línea recta sino de un plano o un hiperplano cuya ecuación puede modelar el comportamiento de una serie de datos de entrenamiento minimizando el error asociado. Para calcular estos coeficientes se emplea un análisis similar al utilizado para la regresión lineal simple, involucrando el error cuadrático medio. Sin embargo, al ser un problema de mayor grado de complejidad por el hecho de emplear más variables se debe reescribir el planteamiento de las ecuaciones. De manera analítica, si se tiene el siguiente sistema de ecuaciones:

Ecuación 3. Planteamiento de ecuaciones

$$y_1 = w_0 + w_1x_{11} + w_2x_{12} + w_3x_{13} + \dots$$

$$y_2 = w_0 + w_1x_{21} + w_2x_{22} + w_3x_{23} + \dots$$

$$y_3 = w_0 + w_1x_{31} + w_2x_{32} + w_3x_{33} + \dots$$

$$y_4 = w_0 + w_1x_{41} + w_2x_{42} + w_3x_{43} + \dots$$

$$y_5 = w_0 + w_1x_{51} + w_2x_{52} + w_3x_{53} + \dots$$

Se puede descomponer el sistema de manera vectorial obteniendo una matriz X donde cada una de las columnas representa una variable independiente que afecta la variable de respuesta, y en cada una de las filas se van a encontrar los datos de medición.

Figura 1. Matriz X

$$\begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \\ x_{41} & x_{42} & x_{43} \\ x_{51} & x_{52} & x_{53} \end{bmatrix}$$

Fuente: Guareño (2013)

Además de un vector Y con las variables de respuesta y un vector W con cada uno de los parámetros que se deben determinar para el planteamiento del modelo.

Figura 2. Vector Y

$$\begin{bmatrix} y_{11} \\ y_{12} \\ y_{13} \\ y_{14} \\ y_{15} \end{bmatrix}$$

Fuente: Guareño (2013)

Figura 3. Vector W

$$\begin{bmatrix} w_0 & w_1 & w_2 & w_3 \end{bmatrix}$$

Fuente: (Guareño, 2013)

Con esta nueva notación el sistema de ecuaciones anterior se puede expresar de la siguiente manera:

Ecuación 4. Nueva representación del sistema de ecuaciones

$$Y = MX$$

Por otro lado, el error cuadrático medio se obtiene determinando la diferencia entre el valor real de los datos de entrenamiento para la variable dependiente con el valor de predicción que muestra el modelo, elevando al cuadrado este valor con el fin de penalizar los valores con mayor error asociado.

Ecuación 5. Error Cuadrático

$$MSE = \frac{1}{M} \sum_{i=1}^M (real_i - estimado_i)^2$$

Con la transformación de los términos de manera matricial se tiene la expresión del Error cuadrático medio vectorial:

Ecuación 6. Error cuadrático medio

$$(Y - XW)^T(Y - XW)$$

Al operar los paréntesis de la ecuación se obtiene la siguiente expresión:

Ecuación 7. Error cuadrático simplificado

$$Y^T Y - W^T X^T Y - Y^T X W + W^T X^T X W$$

Para encontrar el valor mínimo de esta función se calcula la derivada de la expresión igualada a cero:

Ecuación 8. Derivada del error cuadrático

$$-2X^T Y + 2X^T XW = 0$$

Al despejar la ecuación en términos del vector de Parámetros W se obtiene la siguiente fórmula:

Ecuación 9. Ecuación en términos de W

$$w = (X^T X)^{-1} X^T Y$$

Con esta expresión se puede de una manera analítica el determinar los coeficientes del modelo que generan el mínimo error asociado.

- **Multicolinealidad.** Uno de los principales supuestos de la regresión lineal múltiple es que los regresores empleados no pueden ser combinaciones lineales entre sí, ya que dada esta situación es imposible calcular los coeficientes debido a que la matriz $X^T X$ tendría determinante 0 y no se puede invertir, por lo cual la ecuación general presentada en el método de mínimos cuadrados no tendría solución. Esta relación entre los regresores donde su índice de correlación es igual a uno se denomina *multicolinealidad perfecta*; Stock y Watson indican que “cuando se presenta multicolinealidad perfecta, a menudo es el reflejo de un error lógico en la elección de los regresores o bien alguna otra característica particular no reconocida previamente del conjunto de datos” (2012). Así pues, la multicolinealidad perfecta es un problema que hace que los regresores planteados no sean válidos y por tanto el modelo no se pueda desarrollar. Sin embargo, en la práctica es poco frecuente que esta se presente y pueda ser tratada de manera sencilla al validar las relaciones entre los diversos regresores y evaluar el método de recolección de los datos eliminando si es necesario las variables que tienen colinealidad.

Sin embargo, también existe un escenario en el cual el coeficiente de correlación entre los regresores nos es igual a uno pero si se encuentran altamente correlacionados, esta situación se denomina multicolinealidad imperfecta, en este caso durante el cálculo por medio del método de mínimos cuadrados la matriz $X^T X$ tiene un valor diferente a 0 en su determinante, por lo cual la operación para generar los coeficientes se puede realizar, pero si las variables se encuentran muy relacionadas el valor de dicho determinante es muy cercano a 0 y, al invertir una matriz, esta debe ser dividida por su determinante lo cual generaría un problema de imprecisión en el cálculo de al menos uno de los coeficientes (Wooldridge, 2010). Además, la presencia de multicolinealidad puede generar una bondad de ajuste del modelo muy alto sin que la relación entre el valor t de algunas variables sea significativa y por lo tanto tengan la capacidad de realmente explicar el fenómeno estudiado dando lugar a un modelo que carece de veracidad (Novales Cinca, 1993).

Otra consecuencia importante de la presencia de multicolinealidad es la sensibilidad que tienen los coeficientes a los pequeños cambios generados en los datos. Así pues, tal presencia imperfecta puede ser generada por diversas razones como, por ejemplo, en análisis de Ciencias Sociales no se puede tratar ya que se encuentra intrínseca en el fenómeno a explicar (Gujarati y Porter, 2009). Si este no es el caso se puede generar colinealidad entre dos variables por su razón de cambio durante el tiempo, por ejemplo dos variables como la riqueza y el ingreso pueden tener un aumento en series de tiempo similares por la naturaleza de su comportamiento, por lo cual al calcular el coeficiente de correlación entre estas variables aumentará y se podrían presentar problemas de multicolinealidad (*Ibid.*, 2009). Otra causa para este problema es la elección de regresores que se encuentran directamente relacionados o donde alguno de los regresores sea el resultado de una operación relacionada con el anterior y que explican el mismo componente del fenómeno estudiado por lo cual alguno de los regresores no aporta a la descripción del modelo.

Ahora bien, para el proceso de Detección de la Multicolinealidad se procede de la siguiente manera:

- **Verificación de correlación entre parejas de variables:** una de las reglas clásicas para detectar multicolinealidad es verificar los coeficientes de correlación entre parejas de regresores, método que posibilita detectar posibles regresores que tengan un alto índice de colinealidad y que se relacionan entre sí. Sin embargo, este método no resulta totalmente concluyente si se está trabajando con más de dos regresores, ya que no permite detectar la colinealidad con respecto a otras variables y se podrían transformar, o bien tratar estos en conjunto y que sean significativos para el modelo (Wooldridge, 2010).
- **Regresiones auxiliares y análisis de Factor de Inflación de la Varianza:** debido a que la multicolinealidad se genera porque una o más de las variables regresoras son combinaciones lineales totales o parciales de las otras variables, se puede determinar la presencia de estas relaciones realizando la regresión de cada X_i sobre las variables X restantes y calcular el R^2 correspondiente a cada una de estas observaciones, a las cuales se les denomina *regresiones auxiliares* —auxiliares a la regresión principal de Y sobre las X . Es posible aplicar la regla de Klein que sugiere que la multicolinealidad solo es un problema que requiere tratamiento si la R^2 obtenida de una regresión auxiliar es mayor que la R^2 global (Gujarati y Porter, 2009).

Sin embargo, algunos autores emplean el Factor de Inflación de la Varianza (FIV) debido a que si el R^2 resultante de las regresiones auxiliares tiende a uno —es decir se incrementa la colinealidad—, el FIV también se aumenta y el límite puede ser infinito (Wooldridge, 2010); por lo tanto, el FIV es una medida de la multicolinealidad. Al

emplear este indicador puede surgir la siguiente pregunta: ¿qué valor del FIV hace que el regresor se convierta en un problema y genere multicolinealidad? Al respecto, Gujarati cita la regla práctica propuesta por Kleinbaum donde indica que un valor mayor a 10 en el FIV es un indicador de alta colinealidad —lo cual se logra con un R^2 mayor a 0.9. Para el cálculo del FIV se emplea la siguiente expresión:

Ecuación 10. Factor de Inflación de la Varianza

$$FIV = \frac{1}{1 - R_{xi}^2}$$

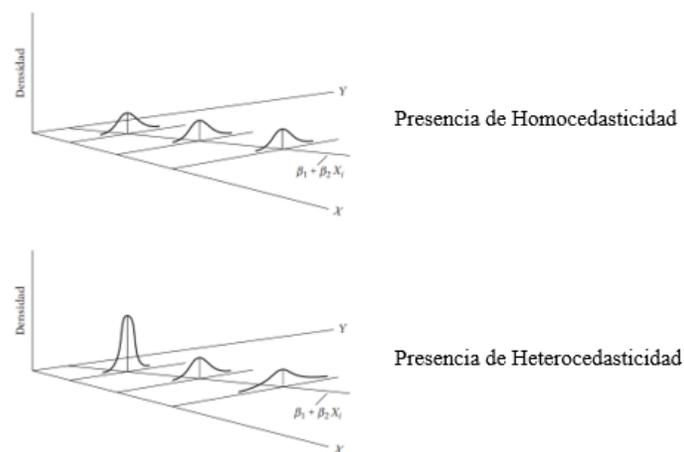
donde R_{xi}^2 corresponde al coeficiente de determinación de cada una de las regresiones auxiliares.

La multicolinealidad se puede tratar de diferentes maneras dependiendo de la naturaleza de los datos y de las variables; por ejemplo, si se tienen dos o más variables cuyo comportamiento es similar, es decir aumentan o decrecen a través de series de tiempo, se puede realizar una transformación reemplazando los datos originales por la variación entre el registro y su dato anterior con el objetivo de eliminar la tendencia similar que genera multicolinealidad y de esta manera llevar a cabo el análisis de regresión lineal. Asimismo, también se pueden omitir variables que teóricamente explican el mismo fenómeno y no resultan significativas al realizar el análisis de regresión lineal múltiple, para lo que se deben evaluar por separado las correlaciones entre grupos de variables y analizar aquellas que presenten una alta colinealidad entre sí, con el fin de determinar si son significativas en el estudio o si por el contrario alguna de ellas es redundante en el modelo generando un problema de multicolinealidad (Stock y Watson, 2012)

- **Homocedasticidad.** La homocedasticidad es un supuesto estadístico que se aplica para la implementación de modelos predictivos y “establece que la varianza del error no observable,

condicional sobre las variables explicativas, es constante” (Wooldridge, 2010). Así pues, esta terminará por no satisfacerse cuando la varianza de las variables explicativas cambié en distintos segmentos de la población tal como se aprecia en la Figura 6, la cual en este caso se denominará heterocedasticidad.

Figura 4. Homocedasticidad y heterocedasticidad



Fuente: Gujarati y Porter (2009)

Al momento de realizar la implementación de un modelo predictivo se puede contar con alguno de los dos supuestos: en caso de que el supuesto sea heterocedástico el problema radicaría en la imposibilidad de establecer intervalos de confianza y probar hipótesis con las pruebas t y F que suelen usarse comúnmente; y, en el caso de que se desconozca la presencia de heterocedasticidad al momento de efectuar la predicción, el problema radicaría en la existencia de una varianza que puede llegar a sobreestimar o subestimar el sesgo. En palabras de Gujarati y Porter, si se insiste en “los procedimientos de prueba usuales a pesar de la presencia de heteroscedasticidad, las conclusiones o inferencias que obtengamos pueden ser muy equivocadas” (2009).

Ahora bien, para detectar la existencia de homocedasticidad han surgido diversos tipos de pruebas que se pueden clasificar en dos grandes grupos: métodos formales e informales. Para efectos del presente proyecto se hace uso de una prueba que pertenece al conjunto de los primeros, cuyo nombre es *Breusch-Pagan*, la cual se puede realizar en 3 pasos que se enuncian a continuación (Wooldridge, 2010):

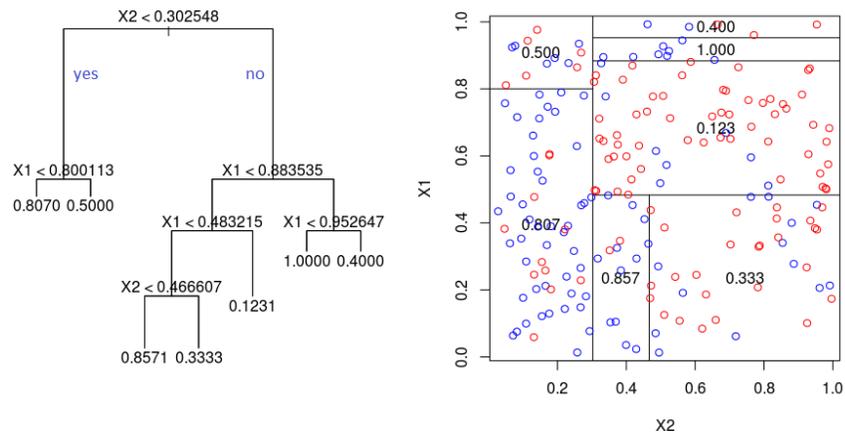
- Se estima el modelo de regresión lineal múltiple como de costumbre, obteniendo los residuos cuadrados para cada observación.
- Se ejecuta nuevamente una regresión en función de los residuos cuadrados obtenidos en el anterior paso. Se conserva el r cuadrado de esta regresión.
- Se forma el estadístico F y se calcula el p-valor. Estos dos últimos van a ser el insumo de la prueba de hipótesis, en caso de que el p-valor sea inferior al nivel de significancia elegido se rechaza la hipótesis nula de homocedasticidad; y si por el contrario la prueba de hipótesis se realiza en función del estadístico F , si su resultado es superior al nivel de significancia la hipótesis nula de homocedasticidad se acepta.

2.2.3.1.3. Árboles decisión de regresión.

Los árboles de decisión además de ser empleados para problemas de clasificación también pueden ser aplicados para aquellos en los cuales se quieren elaborar modelos de predicción en donde intervienen diversas variables independientes, preferiblemente binarias. De hecho, son muy empleados en situaciones en las cuales los datos no tienen una distribución definida. Además, este método tiene como ventaja que se pueden interpretar y modelar gráficamente las relaciones entre las variables independientes aun cuando intervienen tres o más de ellas, pues no se requiere un trabajo previo en el procesamiento de los datos de entrenamiento. Precisamente, su construcción

en muy similar a los árboles de decisión empleados para problemas de clasificación, en donde se divide el espacio de la característica en varias regiones rectangulares simples delimitadas por divisiones paralelas de ejes; así, a pesar de que teóricamente estas regiones pueden tomar cualquier forma para la simplificación en la computación y análisis del modelo, lo recomendable es que estas regiones sigan una forma rectangular como en el ejemplo de la Figura 1, pues ellas no deben estar superpuestas y su número ha de variar según los resultados de la clasificación de los datos durante el proceso de entrenamiento.

Figura 5. Árbol de decisión



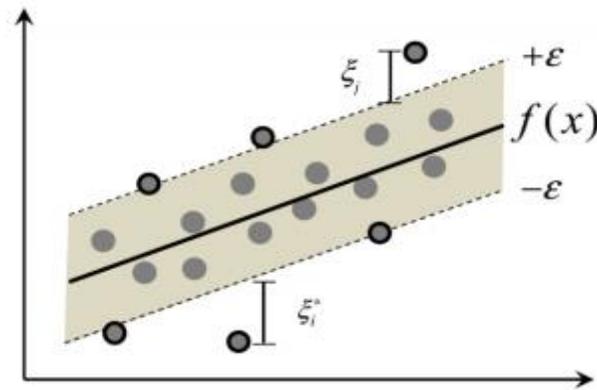
Fuente: Amat (2017)

2.2.3.1.4. Máquinas de vectores de soporte de regresión.

Dadas las ventajas en el uso de los Vectores de Soporte en Clasificación, también se han empleado este tipo de modelos para problemas de regresión. En este caso, a diferencia del problema de clasificación donde se busca una separación de los datos por un hiperplano óptimo, se pretende una función que se ajuste a los datos minimizando el margen de separación de los vectores de soporte y procurando que la mayoría de los datos de entrenamiento se encuentren

dentro del margen entre los vectores y la función de regresión obtenida (Pontil, Rifkin y Theodoros, 1998), tal como se da cuenta en la Figura 2.

Figura 6. Vectores de Soporte Regresión



Fuente: Guareño (2013)

Este algoritmo busca minimizar el error asociado del modelo empleando vectores perpendiculares entre los datos, generando así un espacio de separación donde se quiere albergar la mayor cantidad de datos de entrenamiento tal como se evidencia en la Figura 4, ya que los datos que se encuentran dentro de este margen no son tomados por el algoritmo en el cálculo del error. Por otro lado, los datos que se encuentran fuera del margen y que son denotados por la letra Épsilon son los empleados para el cálculo del error.

2.2.3.2. Aprendizaje no supervisado.

Mientras que el aprendizaje supervisado requiere de realizar un tipo de comparación entre datos de entrenamiento y datos deseados —que tienen el factor común de poseer etiquetas identificables en cada caso—, para el caso del aprendizaje no supervisado no es necesario contar con datos etiquetados ya que para cualquier conjunto de datos existentes este es tratado en su proceso de aprendizaje. En efecto, una de las características más significativas de este tipo de aprendizaje es que no requiere de un conocimiento previo y se realiza por medio de “un proceso

iterativo de análisis de datos sin intervención humana” (Hurwitz y Kirsch, 2018). Así pues, este tipo de algoritmos permite dar un primer vistazo del comportamiento de las entradas pues encuentra patrones de comportamiento en los cuales “se puede explorar la estructura de los datos para extraer información significativa sin la ayuda de una variable de resultado conocida o en función de recompensa” (Raschka y Mirjalili, 2019). Los tipos de algoritmo más habituales en aprendizaje no supervisado son:

- Algoritmos de *clustering*
- Descomposición en valores singulares (*singular value decomposition*)
- Análisis de componentes principales (*Independent Component Analysis*)

2.2.3.3. Aprendizaje por refuerzo.

Existe una tercera categoría en la cual se enmarcan los tipos de aprendizaje que se excluyen de los supervisados y no supervisados: se trata del aprendizaje por refuerzo o reforzado, y su principio de funcionamiento básico es la exposición conductual en la cual aparece un estímulo por cada acción ejecutada correctamente; es decir, existe la entrega de una recompensa luego del resultado de la evaluación de una acción por parte de una función determinada. Es importante resaltar que este se puede desagregar en varios tipos, pero grosso modo se enmarca en un sistema en donde

“el agente en aprendizaje reforzado intenta maximizar la recompensa mediante una serie de interacciones con el entorno. Cada estado puede estar asociado a una recompensa positiva o negativa, y una recompensa se puede definir como el logro de un objetivo general” (Raschka y Mirjalili, 2019).

2.2.4. Lenguaje de programación Python

Python es un lenguaje de programación interpretado “de muy alto nivel que permite expresar algoritmos de forma casi directa” (Marzal Varó, Gracia Luengo y García Sevilla, 2014), el cual tiene la particularidad de ser un lenguaje multiparadigma que se adapta fácilmente a distintas escalas en la programación como la orientación a objetos o la programación funcional; sus sentencias de programación tienen la ventaja de tener legibilidad en el código, lo que optimiza los tiempos en su depuración. Además, también “ofrece la potencia y la flexibilidad de los lenguajes compilados con una curva de aprendizaje suave” (WayBack Machine, 2021). Mirando hacia atrás, *Python* fue creado por Guido van Rossum a finales de la década de 1980 desde el Instituto Nacional de Investigación de Matemáticas y Ciencias de la Computación en los Países Bajos (Venners, 2003), y vino a ser el sucesor del lenguaje de programación ABC con el propósito de manejar excepciones y conectar con el sistema operativo *Amobrea*. Este lenguaje se ha vuelto muy popular en diversas áreas entre las que se encuentran:

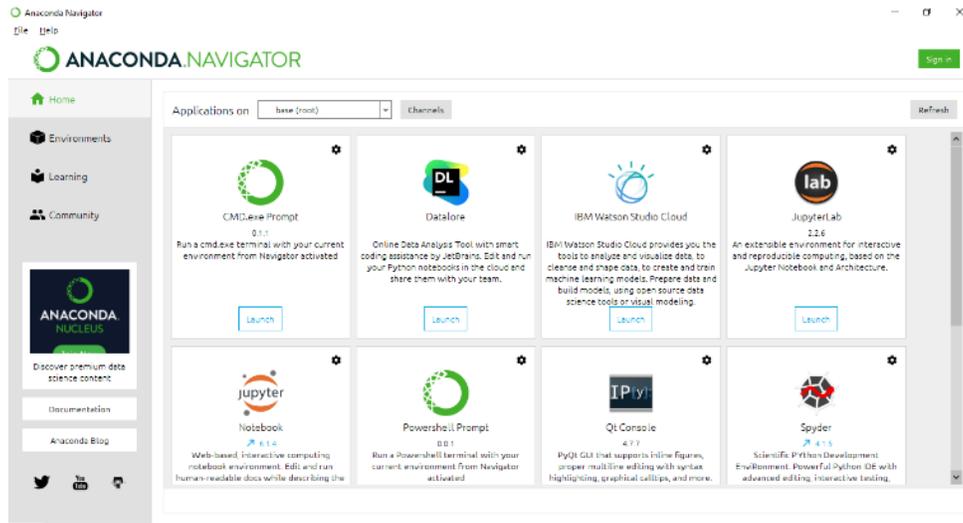
- Usos en la Inteligencia Artificial: gracias a su facilidad en escritura, tiene la ventaja de poder plasmar ideas complejas con pocas líneas de código. Por su gran difusión, este lenguaje ha dado paso a la creación de grandes comunidades que proporcionan nuevos avances y soporte continuo.
- Aplicaciones en Big data: tiene como ventaja la existencia de bibliotecas para el procesamiento de datos dentro del *cluster* HDFS, un sistema de ficheros de Hadoop muy conocido dentro del campo.
- Usos en la ciencia de datos: cuenta con bibliotecas que sirven de motores numéricos y permiten la manipulación de datos tabulares, matriciales y estadísticos, lo cual le ha brindado una gran gama de posibilidades dentro de este campo de estudio.

- Usos en el desarrollo web: gracias a su sintaxis, se ha caracterizado por construir más con menos líneas de código permitiendo la creación de prototipos de manera eficiente. Dentro de este campo es muy famoso el uso del entorno de trabajo Django para la creación de aplicaciones web dinámicas y seguras. (Soloaga, 2018)

2.2.4.1. Anaconda.

Es un paquete de programas de código abierto que se emplea para trabajar el área de ciencia de datos por medio del lenguaje de programación *Python* y que tiene una serie de aplicaciones y librerías que posibilitan desarrollar modelos de *machine learning* de una manera sencilla. *Anaconda* tiene diversas ventajas dentro de las que se encuentran la gran cantidad de entornos y paquetes que tiene a disposición del usuario para el análisis de datos, cuenta también con una capacidad para el procesamiento de datos a gran escala, documentación acerca de su uso, soporte para computación de alto rendimiento y la simplificación de la programación para proyectos robustos. Además, *Anaconda* cuenta con una interfaz sencilla que se puede ver en la Figura 7, la cual posibilita acceder a diversos entornos de trabajo como *Jupyter* y *Spyder* para desarrollar modelos de *machine learning* de una manera dinámica, también posee accesos a foros y comunidades en crecimiento que facilitan el aprendizaje del uso de este programa.

Figura 7. Interfaz Anaconda

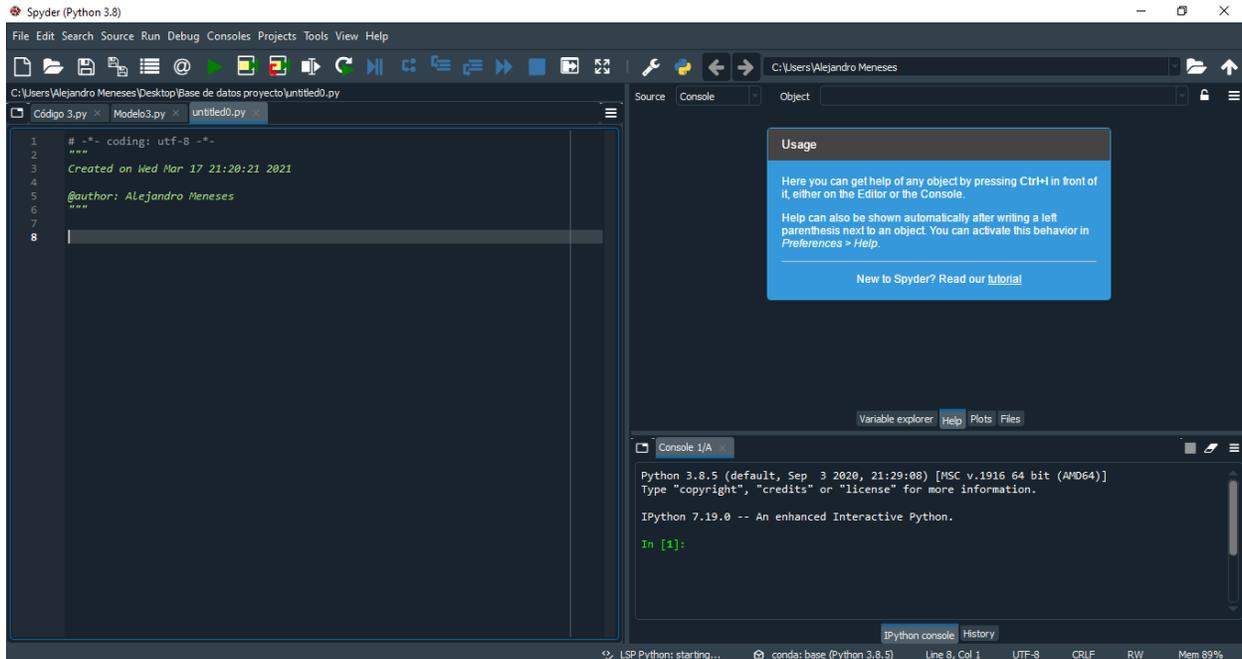


Fuente: captura propia

2.2.4.2. Spyder.

Es un entorno de trabajo de código abierto para programación científica en el lenguaje *Python* que posee herramientas para el desarrollo de proyectos en *machine learning* y propicia el análisis, la edición, la evaluación por medio de pruebas interactivas, depuración e introspección de bases de datos, lo cual lo convierte en una de las mejores opciones para el desarrollo de algoritmos dentro del paquete de *Anaconda*. En general, este entorno contiene un editor optimizado y herramientas para el análisis de código, lo cual permite detectar errores en la depuración de manera instantánea, además integra un visor que ayuda a acceder a las variables creadas durante la ejecución y la edición por medio de arreglos. Este entorno también incorpora un explorador de archivos que propicia la ejecución y cargue información de manera práctica. En resumen, la vista de este editor maneja una estructura matricial en la cual divide cada una de las áreas de trabajo tal como se muestra en la Figura 8.

Figura 8. Interfaz Spyder



Fuente: captura propia

2.2.4.3. Bibliotecas.

Una biblioteca o también llamada librería es un conjunto de archivos pertenecientes a un lenguaje de programación que contiene diversas funcionalidades. Para el caso del presente proyecto se utilizaron algunas de las librerías más comunes dentro del estudio de la ciencia de datos, entre los que se mencionan:

- **Pandas:** para facilitar el análisis de datos en *Python* se emplea el paquete de datos denominado *pandas* (Panel Data) que se especializa brindar herramientas para manipular bases de datos y cargarlos en formato CSV, además permite su procesamiento y visualización de manera ordenada en columnas y filas que pueden filtrarse de manera óptima. Esta librería permite estructurar los datos como arreglos, estos últimos son un conjunto de datos organizados de manera rectangular en forma de

tabla o matriz, que a diferencia de otra estructura denominada *dataset* posibilita etiquetar los datos por medio de series que funcionan como rotulo para facilitar la visualización y análisis. Además, estas series permiten valores alfanuméricos algo que no es permitido en los *dataset*. (Pandas, 2021)

- **Matplotlib:** es una librería de código abierto que da pie a la visualización de una serie de datos de manera fácil y con una gran variedad de estilos de gráficos y opciones de programación que se adapta a las necesidades del proyecto. Además, esta librería tiene dos módulos principales: el *pyplot* con el cual se trabajó en el presente proyecto para facilitar la graficación automática de los datos y con *pylab* que requiere una programación detallada que se implementa para usos particulares (Matplotlib, 2021)
- **Scikit-learn:** esta librería contiene una amplia variedad de algoritmos de *machine learning* de aprendizaje supervisado y de aprendizaje no supervisado que pueden ser empleados en el campo académico como en el comercial, además de ser una librería de código abierto y escrita en el lenguaje *Python*, *Scikit-learn* también se complementa con otros paquetes que complementan su uso y que propocian un mejor análisis y validación de los modelos empleados como *Pandas*, *Numphy* y *Matplotlib*. (Scikit-learn, 2021)
- **Numpy:** es una librería que tiene como principal función dar soporte a vectores y matrices multidimensionales con una gran cantidad de datos, además integra una gama de funciones matemáticas de alto nivel para lograr operar de manera óptima las bases de datos (Numpy, 2021).
- **Scipy:** es una librería de código abierto para *Python* que se encuentra diseñada para uso en áreas como las matemáticas, ciencias e ingenierías, debido a que incorpora una

gran cantidad de funciones y herramientas para optimización, álgebra lineal, integración, interpolación, procesamiento de imágenes y señales entre otras. Se trata de una librería que emplea arreglos generados por *Numpy*, por lo cual deben emplearse simultáneamente. (Pypi, 2021)

- **Seaborn:** es una librería de visualización de datos para *Python* que fue desarrollada con base a *matplotlib*. Sin embargo, esta librería ofrece una interfaz de alto nivel para crear gráficos más detallados y que se emplean para análisis exploratorio o descriptivo de datos. (Seaborn, 2021)
- **Statsmodels:** es una biblioteca para el lenguaje *Python* que permite generar estimadores y funciones para el análisis estadístico de datos generando diversas pruebas para la exploración de estos. Además, con esta librería se puede realizar la comprobación de los supuestos y condicionales de diversos algoritmos con el fin de validar pruebas estadísticas para la validez de estos. (Statsmodels, 2021)

3. Metodología

El método de investigación utilizado se asemeja al método estadístico, el que según Colomé y Femenia (2018) “consiste en tomar una parte del universo a estudiar y luego extrapolar las características de esa muestra a todo el universo”. Así pues, para el desarrollo del proyecto se plantearon cinco etapas, las cuales son comunes dentro de la implementación de un algoritmo de *machine learning*, a saber:

- **Preparación de los datos:** para esta primera etapa se realizó la obtención de los datos provenientes de la base de datos del Departamento de Impuestos y Aduanas Nacionales (DIAN) el cual dispone de un módulo en su página web para poder acceder al comportamiento de las importaciones en el país; adicionalmente, se contemplaron otras fuentes para ingresar otras variables explicativas. El segundo paso fue la limpieza de información a través de la depuración de los datos que no se incluyeron en el análisis, para finalmente consolidar la base de datos que alimentó el modelo.
- **Representación de datos:** Esta etapa se encargó de dar los insumos para un primer análisis exploratorio de los datos, el cual brindó una guía para determinar cuál era modelo por utilizar. Este consistió en generar una perspectiva visual para detectar el comportamiento de los datos, su correlación, detección de datos atípicos y la normalidad estadística. Para poder realizar una nueva intervención sobre la base de datos alimentada por el paso anterior, y así continuar con la fase de modelado.
- **Modelado y aprendizaje:** una vez ejecutado la etapa de representación de datos, se dio paso a la elección del modelo a utilizar, el cual requirió de un proceso de tratamiento de las posibles inconsistencias detectadas en el conjunto de datos para así lograr que se adaptaran

al modelo seleccionado. Una vez consolidado este procedimiento, se dio inicio al entrenamiento del modelo con los datos de 2009 a 2019.

- **Evaluación:** en esta etapa se sometieron a evaluación los resultados obtenidos en la fase de modelado y aprendizaje realizando una división de los datos en un conjunto de entrenamiento con los datos de año 2009 al 2018 y en otro de pruebas con los datos del año 2019, obteniendo una representación visual del comportamiento de la proyección y de los coeficientes de esta.
- **Paso a producción:** fue la última fase y consistió en alimentar el modelo con las variables regresoras del año 2020 y 2021, las cuales requirieron de una estimación ya que no se contaba con información a la fecha consultada para algunas variables, de esta manera se dio paso a la proyección de los años deseados.

A continuación se detallará el desarrollo de cada una de las etapas.

3.1. Preparación de los datos

Para la obtención de los datos de importaciones de aparatos celulares se ingresó a la página web de la DIAN, entidad que publica la información plasmada por los usuarios aduaneros en el formulario 500 —declaraciones de importación— con fines de estudio y análisis. Para determinar cuáles operaciones corresponden a los equipos objeto de estudio se filtró por la subpartida arancelaria 8517120000, a través de la cual se estos se clasifican para el pago de tributos. Así pues, se descargaron las bases de datos de las operaciones de importación desde el año 2009 al 2019 en formato Excel y para facilitar su examen y depuración fueron compilados en un único archivo y ordenados por año.

Ahora bien, es común que algunos usuarios que diligencian información cometan errores en algunos campos de la declaración de importación, por lo cual posteriormente deben presentar

una declaración ya sea de corrección o de legalización que subsane dichos errores. Para asegurar que la toma de la información sea libre de este tipo de inconsistencias se determinaron las operaciones que surgieron de este tipo de cambios por medio de la casilla 34, donde se referencian las declaraciones que fueron objeto de cambios y se eliminaron de la base de datos de estudio dejando únicamente los registros libres de errores detectados por los usuarios. Asimismo, se depuraron las operaciones que no generaron un reembolso al exterior por el pago de la importación, es decir las importaciones que corresponden a muestras sin valor comercial o garantías que no fueron tomadas en cuenta en este análisis. Posteriormente, se eliminaron las variables que no se emplearon en el presente proyecto ya que las bases de datos obtenidas de la DIAN muestran un total de 137 variables que corresponden a cada uno de los campos de la declaración de importación dejando únicamente 10 variables para el estudio. Debido a que la información se encuentra desglosada por declaración de importación se consolida la data por operación y posteriormente se consolida por un periodo específico de tiempo de la siguiente manera:

- Datos Modelo 1 (*Operaciones*), donde cada uno de los registros corresponde a una única operación por Documento de Transporte.
- Datos Modelo 2 (*Diario*), donde cada uno de los registros corresponde a la agrupación de operaciones por día.
- Datos Modelo 3 (*Semanal*), donde cada uno de los registros corresponde a la agrupación de operaciones por semana.
- Datos Modelo 4 (*Mensual*), donde cada uno de los registros corresponde a la agrupación de operaciones por mes.

Posteriormente se definieron 16 variables económicas y sociales externas de diferentes fuentes gubernamentales nacionales e internacionales para un total de 26 variables estudiadas, las cuales se describen enseguida.

3.1.1. Variables estudiadas

A continuación, se dará una breve descripción de las variables que van a cumplir el papel de dependientes e independientes dentro del modelo de regresión.

- **Fecha levante:** es la fecha en la cual la entidad aduanera otorga el número de levante de la carga con la cual se certifica que se han cumplido las obligaciones legales y el importador puede disponer de la mercancía para ser comercializada; este valor se encuentra disponible en la base de datos publicada por la DIAN mensualmente. Se incluye esta variable para estudiar la demanda de teléfonos móviles celulares a través del tiempo y agrupar los datos adecuadamente.
- **Cantidad de unidades importadas:** corresponde al número total de unidades físicas de la mercancía importadas por operación, dato que se obtiene directamente de la base de datos publicada por la DIAN mensualmente y se consigna en el Formulario 500 de la DIAN por cada operación de importación. Esta es la variable que se desea predecir en el modelo lo cual la convierte en la variable dependiente.
- **Cantidad de bultos:** es toda unidad de embalaje independiente y no agrupadas de mercancía acondicionadas para el transporte: pueden ser pallets, cajas, bolsas entre otras unidades y su información se consigna en el Formulario 500 de la DIAN por cada operación de importación.
- **Tasa de cambio:** corresponde a la tasa de cambio del último día hábil anterior a la presentación de la Declaración según lo dispone el Artículo 15 del Decreto 1165 de

2019; este dato es publicado por el Banco de la República y corresponde al promedio de las tasas de compra y venta de dólares del día. Además, determina tanto el valor de los tributos que se deben pagar por las importaciones como el costo que deben girar al exterior los importadores en el país a su proveedor en el exterior, por lo cual es una variable que puede afectar directamente el precio y la demanda de un producto importado.

- **Peso bruto en kilos:** es el peso de la mercancía incluyendo el material de embalaje y se consigna en el formulario 500 de la DIAN por cada operación de importación.
- **Peso neto en kilos:** es el peso de la mercancía declarada una vez es deducido el peso del embalaje y se consigna en el formulario 500 de la DIAN por cada operación de importación.
- **Valor CIF por unidad importada:** se trata del valor FOB más los costos por flete internacional y seguro de la carga por unidad importada. Este dato se encuentra disponible por operación y se obtiene dividiendo el valor CIF total declarado entre la cantidad de operaciones, los cuales se consignan en el Formulario 500 de la DIAN. Es, con todo, una variable importante para estudiar ya que teóricamente es el valor costo total por unidad en la operación de importación; es decir, el costo de cada equipo celular hasta el momento de su importación y que afecta significativamente el precio de venta al consumidor final.
- **Porcentaje de arancel:** porcentaje del impuesto de arancel que al momento de la importación debe cancelar el importador por el derecho a introducir teléfonos móviles celulares al país. Al respecto, a partir del año 2018 se declaró exenta de arancel la subpartida 8517120000 por el Decreto 1563 del año 2017; sin embargo, en los periodos

anteriores a la adopción de este documento se tenía fijado un porcentaje del 5%. Esta variable es importante para el estudio ya que afecta directamente el costo total que deben pagar los importadores y afecta el precio final al consumidor.

- **Porcentaje de IVA:** porcentaje del impuesto de valor agregado que al momento de la importación le corresponde al importador liquidar. Dicha proporción se encuentra regulada por el Decreto 1563 2017, en donde los teléfonos móviles celulares que no sobrepasen las 22 UVT se encuentran exentos de tal pago. Así las cosas, para el año 2021 el valor de las 22 UVT corresponde a \$798.776 COP, que es el precio de un teléfono celular de gama media baja. Para los teléfonos móviles celulares que superen este valor se debe liquidar un valor de IVA del 19%.
- **Cantidad de proveedores en el mercado:** se trata del número de proveedores que realizaron exportaciones de teléfonos móviles celulares en un periodo estudiado. Este valor se determina analizando mensualmente los proveedores que exportaron al país artículos de este tipo y se estudia para determinar la influencia de la oferta internacional del producto en el comportamiento de sus importaciones. El dato del exportador debe ser consignado en el Formulario 500 de la DIAN, en cada una de las operaciones de comercio exterior y en la declaración de cambio.
- **Índice de precios al consumidor (IPC):** según el Banco de la República “el índice de precios al consumidor (IPC) mide la evolución del costo promedio de una canasta de bienes y servicios representativa del consumo final de los hogares” (2021), es decir que permite medir la inflación de los productos que componen la canasta del hogar la cual se agrupa en diversas categorías incluyendo información y comunicación. Este indicador es importante ya que ayuda a medir la evolución de la economía del país y

realizar un diagnóstico de la capacidad adquisitiva de los hogares y su comportamiento general de consumo.

- **Cantidad de abonados de telefonía móvil por año:** tiene que ver con la cantidad de personas que adquirieron servicios de telefonía celular en el país por año, incluyendo planes prepago o pospago. Este es un indicador directo del uso de los teléfonos móviles celulares por parte de los colombianos, ya que la decisión de adquirir un teléfono celular puede depender de la infraestructura y los servicios que pueden ofrecer los operadores de telefonía en el país. Se podría decir que si este indicador crece también el consumo de teléfonos móviles celulares debe ascender proporcionalmente y podría depender de las facilidades que ofrecen los comercializadores de productos de comunicaciones en el país.
- **Población anual en el país:** se trata de los resultados de censos poblacionales y las estimaciones realizadas por el DANE anualmente, en los que se mide la cantidad de habitantes del territorio nacional colombiano. El crecimiento de la población puede determinar en gran medida la demanda de teléfonos móviles celulares en el país teniendo en cuenta que el uso de este tipo de tecnología se está volviendo cada vez más frecuente. El DANE publica anualmente una estimación de la cantidad de habitantes del país teniendo en cuenta la cantidad de muertes y nacimientos proyectados; sin embargo, la cifra del año 2018 sirvió para la realización de un ajuste a las proyecciones que se venían dando.
- **Salario mínimo legal vigente (SMLV):** es la compensación salarial mínima que devenga un trabajador en el país. Teniendo en cuenta que, según datos del DANE, para el año 2020 el 63.1 % de los colombianos devenga este valor, tal podría ser un buen

indicador de la capacidad adquisitiva que tiene la mayor parte de la población colombiana. Además, su variación anual es la base para el aumento del consumo de determinados productos y servicios, lo cual puede afectar la capacidad de adquisición de los hogares colombianos.

- **Tasa de empleo:** proporción de la población que se encuentra en edad de trabajar y que participa en el mercado laboral, la cual es calculada por el DANE mensualmente y deviene en indicador de la situación de la economía del país. Su estudio como variable del modelo es importante ya que la población que no tiene acceso a una remuneración por el empleo de su fuerza de trabajo difícilmente puede adquirir productos y servicios; en este caso los teléfonos móviles celulares. Así pues, se calcula tomando el coeficiente de las personas que se encuentran ocupadas y el total de la población que se encuentra en edad para ejercer actividades laborales.
- **Tasa de desempleo:** es la relación porcentual entre el número de personas que están buscando trabajo y el número de personas que hacen parte de la población económicamente activa o fuerza laboral. A diferencia de la tasa de empleo, donde se toma el total de la población que se encuentra en edad para trabajar, en este indicador se mide quiénes se encuentran en edad para trabajar y además quieren ingresar al mercado laboral. Además, este indicador se emplea para determinar las condiciones actuales del mercado y determinar los grupos que no tienen acceso a un empleo.
- **Hurto a personas:** cantidad de denuncias anuales interpuestas por la población del país por el delito de hurto. Esta estadística es publicada por la Policía Nacional de manera regional y se encuentra disponible para consulta en sus bases de datos. De hecho, se considera fundamental ya que la cantidad de hurtos hacia personas ha

aumentado considerablemente en los últimos años y generalmente en este tipo de asaltos uno de los elementos que más es robado tiene que ver con el teléfono celular, por lo cual este tipo de conductas delictivas puede incidir en la demanda y comercialización de teléfonos móviles celulares en el país.

- **Valor de balanza comercial:** es la diferencia que existe entre el total de las exportaciones e importaciones de un país y se mide en dólares FOB, además de ser un indicador importante acerca de cómo se encuentra la actividad de comercio exterior en el país y para el caso de estudio permite analizar la tendencia de las importaciones de bienes en un determinado periodo de tiempo.
- **Costo promedio mensual de equipos USD:** es el valor CIF promedio de las importaciones de teléfonos móviles celulares registradas en un periodo de tiempo determinado, y a través suyo se busca estudiar si el costo en general de los equipos importados al país influye en su demanda. El costo promedio mensual puede estar influenciado por diversos factores como un aumento en los insumos para la fabricación de los teléfonos, el lanzamiento de nuevos modelos, la competencia de distintos fabricantes a nivel internacional, entre otros.
- **Cantidad de países fabricantes de equipos:** relaciona la cantidad de países de origen que exportaron teléfonos móviles celulares hacia Colombia durante un periodo de tiempo, con lo cual se busca determinar la influencia en el aumento o disminución de países que son fabricantes de equipos celulares y estudiar si una oferta más amplia de países que exportan este tipo de productos hacia el país aumenta su demanda.
- **Producto Interno Bruto a precios constantes - Sector información y comunicaciones** (miles de millones de pesos colombianos): es el valor a precios del

mercado real sin contar con índices de inflación de la producción de bienes y servicios finales producidos en un país; en este caso, trimestralmente para el sector de la información y las comunicaciones. Con esta variable se pretende estudiar el impacto en la variación de la actividad del sector de información y comunicaciones, lo que incluye una gran cantidad de productos y servicios complementarios que pueden afectar la demanda de teléfonos móviles celulares en el país. Además, sin el efecto de la inflación se desea tener una medida nominal del comportamiento del sector.

- **Producto Interno Bruto a precios corrientes - Sector información y comunicaciones** (miles de millones de pesos): es el valor a precios del mercado real contando con los índices de inflación o deflación de la producción de bienes y servicios finales producidos en un país en este del sector de la información y las comunicaciones, cuya unidad a precios constantes busca evaluar el estado del sector, pero teniendo en cuenta también el efecto de la variación en los precios.
- **Coefficiente de GINI desigualdad país:** es una medida económica desarrollada por el Italiano Corrado Gini y publicada anualmente por el DANE; se emplea para determinar la desigualdad salarial entre los habitantes de un país en un periodo determinado de tiempo. A propósito, en los últimos años este coeficiente ha disminuido, teniendo un repunte en el año 2018 que posicionó al país dentro de los primeros en la lista a nivel Latinoamérica. Este es un indicador muy importante para determinar el comportamiento de la distribución del ingreso y puede ser una variable importante para determinar la capacidad adquisitiva durante un periodo de tiempo.
- **Incidencia de la pobreza anual:** porcentaje de personas por debajo de la pobreza monetaria fijada anualmente y que se calcula anualmente por el DANE, la cual

evidencia la evolución de la situación económica de la población del país. Este índice es importante ya que nos permite estimar un parámetro con la proporción de la población que pueda adquirir productos tecnológicos como teléfonos móviles celulares.

Para términos prácticos dentro del conjunto de datos se asigna la abreviatura contemplada en la Tabla 2 a cada una de las variables para facilitar la enunciación en el código.

Tabla 2. *Abreviatura de variables del modelo*

NOMENCLATURA	VARIABLE
FLEV	Fecha Levante
CNBUL	Cantidad de Bultos
CN	Cantidad
PN	Peso Neto en Kilos
PB	Peso Bruto en Kilos
TRM	Tasa de Cambio
CIFUNI	Valor CIF
ARAN	Porcentaje de Arancel
IVA	Porcentaje de IVA
PROVE	Cantidad de Proveedores en el Mercado
IPC	El índice de precios al consumidor (IPC)
ABONA	Cantidad De Abonados Telefonía Móvil Por Trimestre
POBLA	Población x Año
SMLV	Salario Mínimo
TASEMP	Tasa de Empleo

TASADESEM	Tasa de Desempleo
HURTO	Hurto a personas
BALAN	Balanza comercial Millones de Dólares
COSTO	Costo Promedio Mensual Equipos USD
COSTO	Costo Promedio Mensual Equipos USD
PAISES	Cantidad de Países Fabricantes Equipos
PIBPC	Producto Interno Bruto a precios constantes Sector Información y comunicaciones (Miles de Millones de Pesos Colombianos)
PIBPCN	Producto Interno Bruto a precios constantes Sector Información y comunicaciones (Miles de Millones de Pesos
PIBPCOR	Producto Interno Bruto a precios corrientes Sector Información y comunicaciones (Miles de Millones de Pesos)
GINI	Coefficiente de GINI Desigualdad País
POBRE	Incidencia de la Pobreza Anual

Fuente: elaboración propia

3.1.2. Cargue de conjunto de datos

Para realizar el cargue de la información se debe transformar el archivo en formato *CSV-UTF-8* para que pueda ser cargado a entorno de trabajo *Spyder* por medio de la biblioteca *Pandas* empleando la instrucción *pd.read_csv* como se puede ver en la siguiente imagen:

Figura 9. Código para el cargue de datos

```
# -- Cargue de datos --  
dataset = pd.read_csv("Datos_de_Entrenamiento_M4.csv", delimiter = ';')
```

Fuente: elaboración propia

3.2. Representación de los datos

En esta etapa se utilizaron diferentes bibliotecas para la realización del análisis exploratorio de los datos. Así, para encontrar una primera vista de su comportamiento se realizó el gráfico de parejas, el cual realiza una serie de gráficos de manera matricial en la cual a cada uno de los ejes le asigna el conjunto de variables del modelo. Para la realización de este se contempló el uso de la biblioteca *sns* específicamente con su función *pairplot*, la cual requiere de la declaración de una variable en la cual su entrada es el conjunto de datos. Tiene otros parámetros adicionales en los que se le pueden agregar características visuales; para este caso, se dio uso a la denominada *hue*, la cual permite determinar una variable determinante para asignar con un color específico la nube de puntos de cada gráfico, como se ve en la Figura 10.

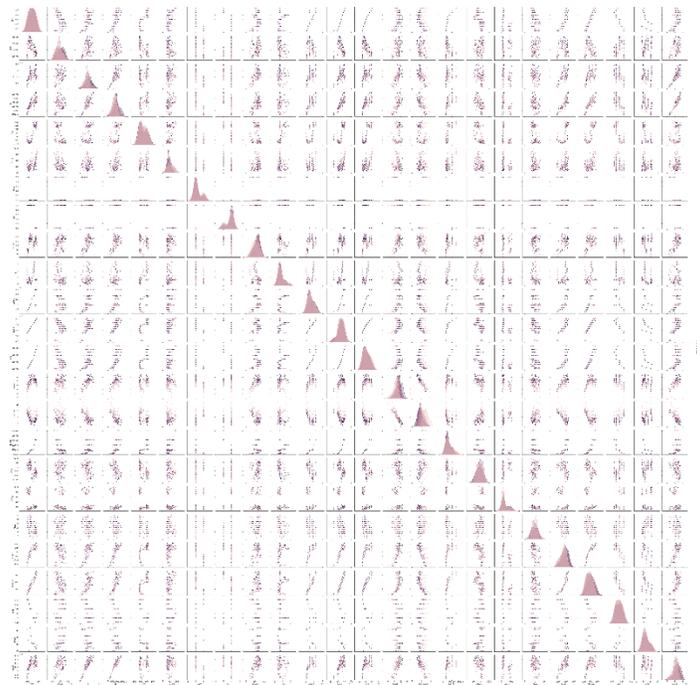
Figura 10. Declaración de variables

```
q = dataset.iloc[:, :-1].values  
w = dataset.iloc[:, 24].values  
g = sns.pairplot(dataset, hue="MES")
```

Fuente: elaboración propia

Como resultado a la declaración anterior se puede apreciar en la Figura 11 el resultado del gráfico de parejas.

Figura 11. Gráfico de parejas



Fuente: elaboración propia

Una segunda vista fue la del mapa de calor, el cual permite crear un gráfico bidimensional que al igual que el gráfico anterior en cada eje asigna todas las variables del modelo para que en su contenido le asigne un color específico según el grado de correlación que exista entre las variables. Para la declaración del código se necesita enunciar la función, asignándole una entrada con el *dataset* y especificándole el tipo de desagregación que para este caso es una correlación. Este código se puede apreciar en la Figura 12.

Figura 12. Declaración para el gráfico de calor

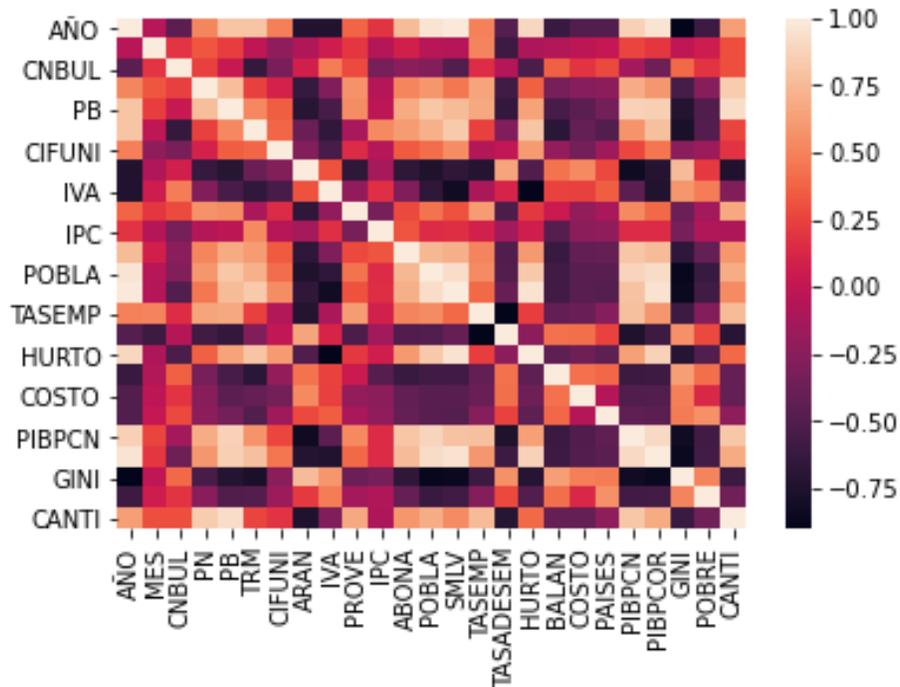
```
#Grafico de calor
sns.heatmap(dataset.corr())
```

Fuente: elaboración propia

Así, por ejemplo, se puede ver en la escala de colores que entre mayor grado de correlación exista entre las variables el tono va a ser más claro y si es al contrario el color va a tender a

oscurecerse. Tal es el caso de la intersección entre la variable PB y CANTI, la cual tiende a tener un coeficiente de correlación superior a 0.75.

Figura 13. *Gráfico de calor*



Fuente: elaboración propia

Con base al anterior gráfico también resultó necesaria la ejecución de un código que nos arrojará los coeficientes de correlación; este requería como entrada el conjunto de datos a estudiar y el orden el que se debían visualizar las variables, tal como se aprecia en la siguiente figura.

Figura 14. *Código para la declaración de coeficientes de correlación*

```
#Grafico Coeficientes correlación
corr = dataset.corr()
corr[['CANTI']].sort_values(by = 'CANTI',ascending = False).style.background_gradient()
```

Fuente: elaboración propia

Esto dio como resultado un arreglo matricial en donde en el contenido de cada uno de los registros daba el coeficiente de correlación tal como se puede apreciar en la siguiente figura

Figura 15. Gráfico de coeficientes de correlación

Index	AÑO	MES	CNBUL	PN	PB	TRM	CFUNI	ARAN	IVA	PROVE	IPC	ABONA	POBLA	SMLV	TASEMP	TASADESEM	HURTO	BALAN	COSTO
AÑO	1	-0.0464353	-0.443421	0.518441	0.799316	0.805957	0.493198	-0.740046	-0.740158	0.378997	0.188381	0.758433	0.958623	0.98664	0.497585	-0.484757	0.980156	-0.615203	-0.49585
MES	-0.0464353	1	0.191928	0.328264	0.233338	-0.00716636	-0.219218	-0.0731381	0.055252	0.198152	-0.0488792	0.0819496	-0.0448421	-0.0577911	0.516107	-0.594763	-0.0894991	-0.0671815	-0.0139158
CNBUL	-0.443421	0.191928	1	0.234315	0.0309408	-0.641769	-0.316614	0.0844929	0.480216	0.283185	-0.323862	-0.248479	-0.279101	-0.49841	0.154085	-0.0501837	-0.530775	0.361668	0.187389
PN	0.518441	0.328264	0.234315	1	0.77067	0.244592	0.0974617	-0.638183	-0.285697	0.570116	-0.0559259	0.517513	0.583682	0.468462	0.651838	-0.578766	0.368849	-0.329645	-0.250823
PB	0.799316	0.233338	0.0309408	0.77067	1	0.524172	0.357234	-0.707044	-0.544617	0.547937	-0.0152719	0.682747	0.819711	0.759611	0.66612	-0.646132	0.649895	-0.543817	-0.451979
TRM	0.805957	-0.00716636	-0.641769	0.244592	0.524172	1	0.380953	-0.390717	-0.652417	-0.113932	0.532592	0.620093	0.712817	0.833239	0.23455	-0.281367	0.004012	-0.690843	-0.409865
CFUNI	0.493198	-0.219218	-0.316614	0.0974617	0.357234	0.380953	1	-0.273119	-0.54011	0.147979	-0.0424193	0.34288	0.414284	0.547664	-0.0675775	-0.00169315	0.007473	-0.1918	-0.339654
ARAN	-0.740046	-0.0731381	0.0844929	-0.638183	-0.707044	-0.390717	-0.273119	1	0.311111	-0.671475	-0.118174	-0.668084	-0.751429	-0.659378	-0.718873	0.656477	-0.504229	0.440785	0.535073
IVA	-0.740158	0.055252	0.480216	-0.285697	-0.544617	-0.652417	-0.54011	0.311111	1	-0.205742	0.17312	-0.292533	-0.666417	-0.821142	-0.0954191	0.10866	-0.893366	0.252737	0.245866
PROVE	0.378997	0.198152	0.283185	0.570116	0.547937	-0.113932	0.147979	-0.671475	-0.205742	1	-0.331053	0.276384	0.440077	0.315323	0.613854	-0.518273	0.208751	0.0478018	-0.197677
IPC	0.188381	-0.0488792	-0.323862	-0.0559259	-0.0152719	0.532592	-0.0424193	-0.118174	0.17312	-0.331053	1	0.322619	0.145201	0.16863	0.8094031	-0.138005	0.0772215	-0.476297	-0.240047
ABONA	0.758433	0.0819496	-0.248479	0.517513	0.682747	0.620093	0.34288	-0.668084	-0.292533	0.276384	0.322619	1	0.744136	0.706069	0.515474	-0.503595	0.60446	-0.618567	-0.423969
POBLA	0.958623	-0.0448421	-0.279101	0.583682	0.819711	0.712817	0.414284	-0.751429	-0.666417	0.440077	0.145201	0.744136	1	0.929266	0.543782	-0.495937	0.827627	-0.570164	-0.468651
SMLV	0.98664	-0.0577911	-0.49841	0.468462	0.759611	0.833239	0.547664	-0.659378	-0.821142	0.315323	0.16863	0.706069	0.929266	1	0.402306	-0.399492	0.953263	-0.573302	-0.459129
TASEMP	0.497585	0.516107	0.154085	0.651838	0.66612	0.23455	-0.0675775	-0.718873	-0.0954191	0.613854	0.0094031	0.515474	0.543782	0.402306	1	-0.903903	0.22735	-0.41416	-0.38954
TASADESEM	-0.484757	-0.594763	-0.0501837	-0.578766	-0.646132	-0.281367	-0.00169315	0.656477	0.10866	-0.518273	-0.138005	-0.503595	-0.499337	-0.399492	-0.903903	1	-0.231283	0.427868	0.421645
HURTO	0.980156	-0.0894991	-0.530775	0.368849	0.649895	0.004012	0.607473	-0.504229	-0.893366	0.208751	0.0772215	0.60446	0.827627	0.953263	0.22735	-0.231283	1	-0.441659	-0.369163
BALAN	-0.615203	-0.0671815	0.361668	-0.329645	-0.543817	-0.690843	-0.1918	0.440785	0.252737	0.0478018	-0.476297	-0.618567	-0.570164	0.573382	-0.41416	0.427868	-0.441659	1	0.430943
COSTO	-0.49585	-0.0139158	0.187389	-0.250823	-0.451979	-0.409865	-0.339654	0.535073	0.245866	-0.197677	-0.240047	-0.423969	-0.468651	-0.459129	-0.38954	0.421645	-0.369163	0.430943	1
PAISES	-0.495386	0.0141481	0.274127	-0.239415	-0.341237	-0.500096	-0.159758	0.275991	0.360818	-0.112128	-0.215795	-0.411668	-0.465527	-0.487032	-0.260165	0.250851	-0.435801	0.393883	0.9509996
PIBPCN	0.852593	0.259156	-0.131635	0.688406	0.866795	0.581483	0.264434	-0.824071	-0.445585	0.532702	0.144981	0.806694	0.884185	0.791866	0.780718	-0.751077	0.63194	-0.610644	-0.506873
PIBPCOR	0.957943	0.199889	-0.37447	0.593459	0.851701	0.783857	0.450539	-0.723728	-0.735284	0.404207	0.14477	0.764802	0.918654	0.949246	0.594542	-0.597958	0.86864	-0.598688	-0.496265
GINI	-0.099037	-0.0100537	0.401958	-0.549677	-0.720806	-0.770642	-0.24457	0.705543	0.50439	-0.382221	-0.327162	-0.666797	-0.067974	-0.050639	-0.597435	0.567991	-0.712733	0.620896	0.486448
POBRE	-0.60201	0.0625396	0.182919	-0.250074	-0.502759	-0.494108	-0.164242	0.214156	0.481727	-0.130729	-0.0799887	-0.406742	-0.620946	-0.581338	-0.270721	0.273863	-0.50068	0.419951	0.132045
CANTI	0.620764	0.295024	0.296875	0.84148	0.927572	0.209122	0.18684	-0.742297	-0.205819	0.677686	-0.0929662	0.590452	0.692668	0.542198	0.756692	-0.708038	0.390618	-0.403792	-0.400282

Fuente: elaboración propia

Los siguientes gráficos utilizados son los gráficos bidimensionales de nube de puntos, los cuales consisten en declarar en el eje x todos los valores de la variable dependiente que para el caso del presente proyecto son las cantidades de importaciones, y en el eje y cada una de las variables independientes. Para la realización de dichos gráficos se procedió con la utilización de la función *plt* de la biblioteca *matplotlib.pyplot*, la cual requiere de la declaración de cada una de las variables que están en el eje x y el eje y, así como de sus nombres y un título.

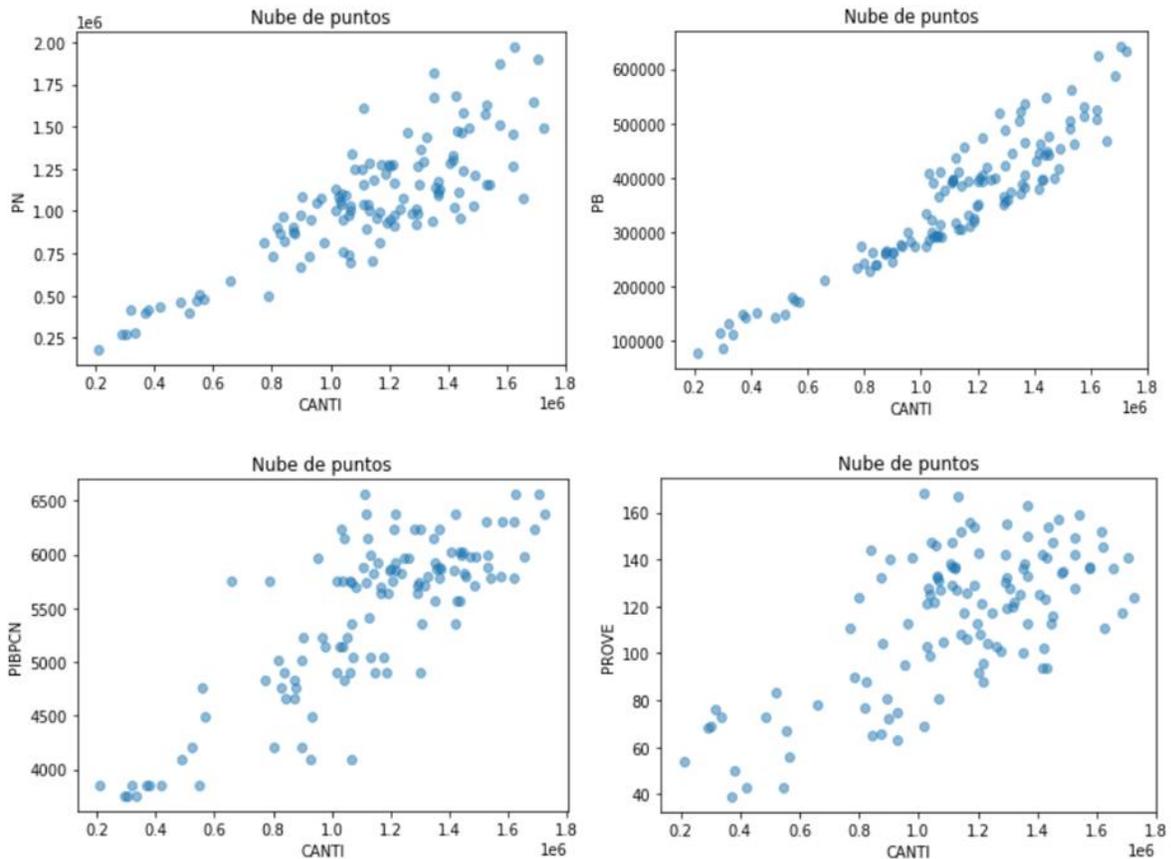
Figura 16. Código para la declaración de gráfico bidimensional de nube de puntos

```
plt.scatter(dataset['CANTI'],dataset['CNBUL'], alpha = 0.5);plt.title('Nube de puntos');plt.xlabel('CANTI');plt.ylabel('CNBUL');
```

Fuente: elaboración propia

Esto permitió encontrar en cada una de las variables la existencia de registros atípicos para su posible eliminación, este proceso se hizo directamente en la base de datos a importar para su posterior cargue a la etapa de entrenamiento. En la Figura 17 se pueden evidenciar cuatro variables ejemplo para este tipo de gráfico.

Figura 17. Gráfico de nube de puntos para cuatro variables



Fuente: elaboración propia

El ultimo gráfico implementado es el que se utilizó para detectar la normalidad estadística de los datos, para este fue necesaria la utilización de la función *distplot* de la biblioteca *sns* y de la función *figure* de la biblioteca *plt*, en las cuales se declaró únicamente como variable de entrada aquella a analizar. En la siguiente figura se podrá ver el código utilizado para evidenciar el comportamiento normal de la variable *PN*.

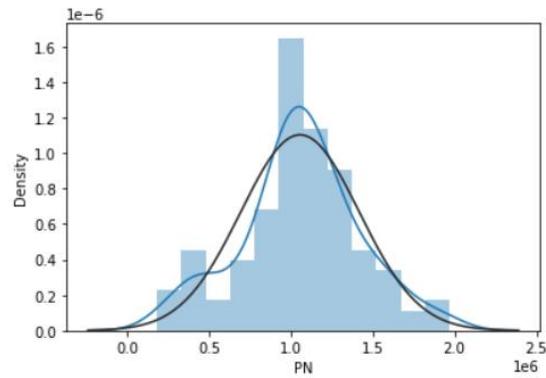
Figura 18. Código para la declaración de gráfico normalidad estadística

```
# ---Variable PN
sns.distplot(dataset['PN'], fit = norm);
fig = plt.figure()
res = stats.probplot(dataset['PN'], plot = plt)
```

Fuente: elaboración propia

Tal código dio como resultado una gráfica con la distribución normal de la variable, así como la posibilidad de lograr determinar qué se cumple con este principio. En la siguiente figura se podrá encontrar dicho gráfico para la variable *PN*.

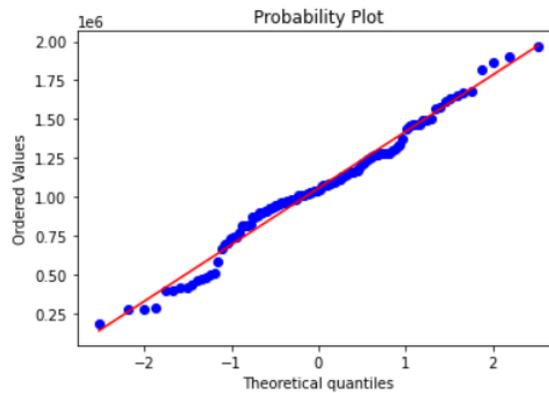
Figura 19. *Gráfico normal para la variable PN*



Fuente: elaboración propia

Asimismo, para complementar la revisión se utilizó un gráfico de probabilidades en donde se puede apreciar en el *eje x* los cuantiles teóricos y en el *eje y* los valores en los que puede incurrir la variable. Ello se puede apreciar para la variable *PN* en la Figura 20.

Figura 20. *Gráfico de probabilidades de variable PN*



Fuente: elaboración propia

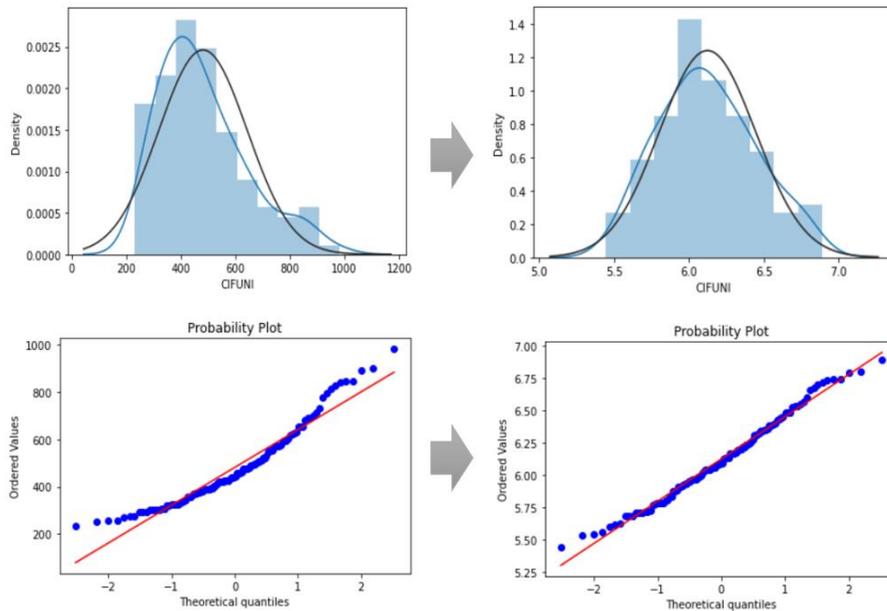
Con base en estos dos últimos gráficos se detectaba si existía la necesidad de realizar normalización; así, en caso de que alguna variable lo requiriera, se procedía con el código de la siguiente figura para someter a cada una de las variables contenidas en el conjunto de datos a una función logarítmica la cual permitiera la normalización de dichos registros. Esto se puede apreciar en la Figura 21 en donde se aprecia la variable CIFUNI en el antes y el después de someter los datos al proceso de normalización.

Figura 21. Código para la normalización de variables

```
dataset['CIFUNI'] = np.log(dataset['CIFUNI'])
sns.distplot(dataset['CIFUNI'], fit = norm);
fig = plt.figure()
res = stats.probplot(dataset['CIFUNI'], plot = plt)
```

Fuente: elaboración propia

Figura 22. Antes y después de la normalización estadística



Fuente: elaboración propia

3.3. Modelado y aprendizaje

Una vez eliminados los datos atípicos y hecha la normalización en la anterior etapa, se procede con la etapa de modelado y aprendizaje. Para la ejecución de esta resultó necesario visualizar el comportamiento de los datos para ver si era necesaria otra fase de tratamiento, por lo que se procedió a realizar un procedimiento para detectar multicolinealidad y homocedasticidad.

3.3.1. Multicolinealidad

Para este proceso se aplicó el método de *factor de inflación de la varianza*, el cual requiere de la verificación de correlación entre parejas de variables. Así pues, se tomaron todas las variables regresoras y se hizo un procedimiento de predicción entre estos conjuntos: por ejemplo, para la ejecución de la predicción de la primera variable se tomó un arreglo con una variable *a1* y *b1* como se muestra en la Figura 23, en donde la primera declaración corresponde a la variable a predecir y la segunda a las variables predictoras. Este comando arroja el R^2 para la predicción de dicha variable tal como se aprecia en la Figura 24 y se repite de la misma forma como se muestra en la Figura 25 para el resto del conjunto de variables.

Figura 23. Código para realiza la predicción en función de la primera variable

```
b1 = q[:,[0]]
a1 = q[:,[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23]]
regression = LinearRegression()
regression.fit(a1, b1)
a1_pred = regression.predict(a1)
c1 = regression.score(a1, b1); c1
```

Fuente: elaboración propia

Figura 24. R cuadrado para la predicción de la primera variable

```
Out[4]: 0.9995836801995759
```

Fuente: elaboración propia

Figura 25. Código para realiza la predicción en función de la segunda variable

```
b2 = q[:,[1]]
a2 = q[:,[0,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23]]
regression = LinearRegression()
regression.fit(a2, b2)
a2_pred = regression.predict(a2)
c2 = regression.score(a2, b2); c2
```

Fuente: elaboración propia

Al finalizar cada una de las regresiones para todas las variables se procede con la creación de un arreglo con todos los resultados para cada uno de los comandos:

Figura 26. Arreglo con todos los R cuadrado del modelo

Inde	Type	Size	
0	float64	1	0.9995836801995759
1	float64	1	0.875461131653791
2	float64	1	0.8082925137852695
3	float64	1	0.7481311395830519
4	float64	1	0.9224608521340966
5	float64	1	0.968935730570093
6	float64	1	0.6231353004786184
7	float64	1	0.9145001335362217
8	float64	1	0.982964676396777
9	float64	1	0.8384060109121921
10	float64	1	0.8747614832883975
11	float64	1	0.8621920988704254
12	float64	1	0.957302917618968
13	float64	1	0.9995620301529541
14	float64	1	0.9075751333904012
15	float64	1	0.8983965753649925
16	float64	1	0.9795423146323244
17	float64	1	0.744363596530689
18	float64	1	0.5692676137323304
19	float64	1	0.5903006649743667
20	float64	1	0.9920560249880543
21	float64	1	0.9951991113153589
22	float64	1	0.9790667618029218
23	float64	1	0.8483550646824053

Fuente: elaboración propia

Es así como se procede con la identificación y eliminación de aquellas variables que presentan un alto grado de correlación entre sí y que no resultan significativas para el modelo.

3.3.2. *Homocedasticidad*

Una vez depuradas las variables que presentan un alto índice de correlación en el procedimiento de multicolinealidad se procede con la detección de homocedasticidad, para la cual se utilizó la prueba de *Breusch-Pagan* cuyo atributo de entrada tiene que ver con la ejecución de una regresión con las variables resultantes del proceso anterior, esto con el fin de obtener los residuos cuadrados para cada observación. Es así como se toma el arreglo de variable q y w , tal como se puede apreciar en la Figura 27.

Figura 27. *Código para la primera regresión del supuesto de homocedasticidad*

```
regression = LinearRegression()
regression.fit(q, w)
#haciendo la predicción de los resultados con el conjunto de testing
w_pred = regression.predict(q)
regression.score(q, w)
```

Fuente: elaboración propia

Con este resultado se procede a la creación de una columna de números uno, para poder identificar los coeficientes, el error estándar, el valor t y el p-valor —dicho comando se puede apreciar en la Figura 28. Este último va a ser el insumo para identificar aquellas variables que estén por encima del estadístico de prueba que para el caso del presente proyecto es 0.05.

Figura 28. *Creación de columna de unos*

```
X = np.append(arr = np.ones((118,1)).astype(int), values = q, axis = 1)
#corremos OLS
regression_OLS = sm.OLS(endog = w, exog = X.tolist()).fit()
regression_OLS.summary()
```

Fuente: elaboración propia

Como resultado del comando anterior se obtuvo la tabla de la Figura 28, en la cual se pueden apreciar aquellas variables que tienen un p-valor superior al estadístico de prueba y que fueron eliminadas para poder comprobar finalmente la hipótesis de homocedasticidad.

Figura 29. Resultado de los p-valor para cada variable

	coef	std err	t	P> t
const	-4.295e+08	1.33e+08	-3.231	0.002
x1	2.148e+05	6.62e+04	3.245	0.002
x2	6752.6869	3415.000	1.977	0.051
x3	0.2308	0.051	4.553	0.000
x4	0.1067	0.023	4.734	0.000
x5	2.2846	0.122	18.698	0.000
x6	9.7157	42.849	0.227	0.821
x7	7.3287	41.096	0.178	0.859
x8	-2.978e+04	3.29e+04	-0.906	0.367
x9	6.008e+04	7.36e+04	0.816	0.417
x10	346.3480	336.298	1.030	0.306
x11	-7416.6566	6928.661	-1.070	0.287
x12	-0.0049	0.002	-2.876	0.005
x13	0.0148	0.010	1.445	0.152
x14	-6.6700	1.969	-3.387	0.001
x15	-5248.8284	6881.357	-0.763	0.448
x16	-7269.6738	8846.362	-0.822	0.413
x17	1.1296	0.431	2.622	0.010
x18	19.0953	12.539	1.523	0.131
x19	-19.1437	21.908	-0.874	0.384
x20	-856.9624	2614.992	-0.328	0.744
x21	-55.8662	61.630	-0.906	0.367
x22	0.3882	64.850	0.006	0.995
x23	1.535e+06	1.76e+06	0.874	0.384
x24	9247.2024	4923.806	1.878	0.064

Fuente: elaboración propia

Para la eliminación de las variables basta con volver a declarar el arreglo que contiene a todos los regresores dejando en su interior solo las variables que van a ser utilizadas para la prueba de hipótesis, tal como se muestra en la Figura 30.

Figura 30. Declaración del arreglo sin las variables eliminadas

```
q = q[:, [0, 2, 3, 4, 10, 12, 14, 15, 21]]
```

Fuente: elaboración propia

Una vez declarado el arreglo anterior se procede con una nueva línea de código que se encarga de entrenar a las variables q y w , para poder obtener un arreglo con los valores pronosticados denominado w_pred . Este último va a servir para declarar la variable z que será la encargada de calcular los residuos cuadrados los cuales van a servir para entrenar una nueva regresión como la mostrada en la Figura 31, y a partir de ello generar un nuevo resumen.

Figura 31. *Regresión en función de la diferencia de cuadrados*

```
z = (w_pred - w)*(w_pred - w)

regression = LinearRegression()
regression.fit(q, z)

z_pred = regression.predict(q)

X = np.append(arr = np.ones((118,1)).astype(int), values = q, axis = 1)
#corremos OLS para verificar el F-estadístico
regression_OLS = sm.OLS(endog = z, exog = X.tolist()).fit()
regression_OLS.summary()
```

Fuente: elaboración propia

En el código anterior, una vez se tenga entrenada la regresión en función de la variable z , se procede con la obtención del resumen mostrado en la Figura 32, el cual arrojó el valor del estadístico F que será el encargado de someterse a las hipótesis de la Figura 33; así, como para este caso el valor arrojado fue de 0.156, se procede a aceptar la hipótesis nula de inexistencia de heterocedasticidad.

Figura 32. *Resumen de la regresión*

```
=====
                    OLS Regression Results
=====
Dep. Variable:          y      R-squared:                0.111
Model:                  OLS    Adj. R-squared:           0.037
Method:                 Least Squares  F-statistic:              1.502
Date:                   Tue, 16 Mar 2021  Prob (F-statistic):      0.156
Time:                   14:37:23    Log-Likelihood:          -2802.0
No. Observations:      118         AIC:                     5624.
Df Residuals:          108         BIC:                     5652.
Df Model:               9
Covariance Type:       nonrobust
=====
```

Fuente: elaboración propia

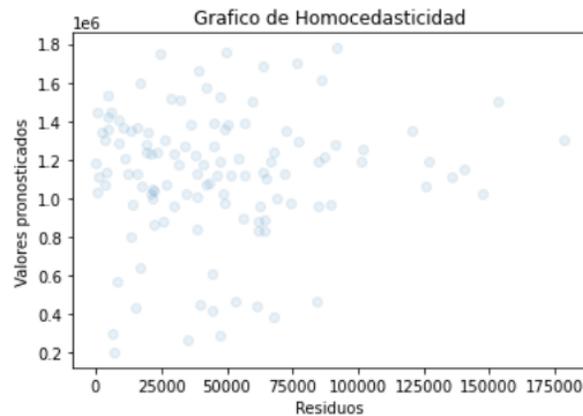
Figura 33. Planteamiento de hipótesis de homocedasticidad

H_0 : No existe heterocedasticidad H_1 : Existe heterocedasticidad H_0 Se acepta sí: $P > 0.05$ H_0 Se rechaza sí: $P < 0.05$
--

Fuente: elaboración propia

De igual forma se procede a realizar el gráfico de residuos donde se evidencia el comportamiento de los datos, afirmando de nuevo la presencia de homocedasticidad.

Figura 34. Gráfico de homocedasticidad



Fuente: elaboración propia

Ya con la comprobación de los supuestos de homocedasticidad y multicolinealidad, y con el entrenamiento del modelo en relación con todas las variables suministradas, se procede a obtener el arreglo mostrado en la siguiente etapa para la evaluación del modelo.

3.4. Evaluación

Para validar los resultados que el modelo de regresión lineal múltiple ha generado se hizo necesaria la evaluación de los valores pronosticados por dicho modelo y compararlos con los valores reales en un periodo de tiempo específico. Tras comprobar que las variables estudiadas no poseen

multicolinealidad y que las varianzas son homocedásticas, se realizó la evaluación del desempeño del modelo; para esto se dividió el conjunto de datos en dos matrices diferentes: una para los valores de entrenamiento y otra para los valores de evaluación. En este caso los datos de entrenamiento corresponden a los registros comprendidos entre el año 2009 y 2018, y los datos de evaluación son los registros del año 2019. Por ello, fueron declaradas cuatro nuevas variables para cada una de las matrices: los valores de $X(q_{train})$ y $Y(w_{train})$ que sirven para generar el nuevo modelo, evaluar los coeficientes y precisión, así como los valores de $X(q_{test})$ y $Y(w_{train})$ que son las variables de entrada para realizar la predicción de prueba. Con todo, tras declarar las variables se realiza el entrenamiento de los datos por medio del comando *regression.fit* con $X(q_{train})$ y $Y(w_{train})$ declaradas anteriormente, tal como se muestra en la Figura 35.

Figura 35. División de variables para el entrenamiento del modelo

```
q_train = q[:-49,:]
q_test = q[517:,:]
w_train = w[:-49]
w_test = w[517:]

regression = LinearRegression()
regression.fit(q_train, w_train)
#haciendo la predicción de los resultados con el conjunto de testing
w_pred = regression.predict(q_test)
regression.score(q_test, w_test)

regression_OLS.summary()
```

Fuente: elaboración propia

Posteriormente, se generaron los valores pronosticados por el entrenamiento de los datos por medio del comando *regresión.predit* y son almacenados en la variable w_{pred} los cuales son comparados con los valores reales del año 2019. Además, se elaboró la tabla de resultados de la regresión por medio del comando *regression_OLS.summary*.

Figura 36. Resultados de la regresión para el modelo 3

OLS Regression Results						
Dep. Variable:	y	R-squared:	0.966			
Model:	OLS	Adj. R-squared:	0.966			
Method:	Least Squares	F-statistic:	1584.			
Date:	Sat, 13 Mar 2021	Prob (F-statistic):	0.00			
Time:	19:14:43	Log-Likelihood:	-6386.6			
No. Observations:	566	AIC:	1.280e+04			
Df Residuals:	555	BIC:	1.284e+04			
Df Model:	10					
Covariance Type:	nonrobust					
	coef	std err	t	P> t	[0.025	0.975]
-----	-----	-----	-----	-----	-----	-----
const	2.155e+05	5.54e+04	3.889	0.000	1.07e+05	3.24e+05
x1	-633.7921	75.631	-8.380	0.000	-782.351	-485.233
x2	0.3262	0.018	18.586	0.000	0.292	0.361
x3	0.1370	0.009	15.087	0.000	0.119	0.155
x4	2.1221	0.045	46.904	0.000	2.033	2.211
x5	216.8153	35.130	6.172	0.000	147.811	285.819
x6	-0.0059	0.001	-4.446	0.000	-0.009	-0.003
x7	-8215.2373	1139.716	-7.208	0.000	-1.05e+04	-5976.553
x8	1363.8942	418.909	3.256	0.001	541.053	2186.735
x9	14.3561	4.022	3.569	0.000	6.456	22.256
x10	1731.4192	562.731	3.077	0.002	626.076	2836.763
-----	-----	-----	-----	-----	-----	-----
Omnibus:	1.201	Durbin-Watson:	1.328			
Prob(Omnibus):	0.548	Jarque-Bera (JB):	1.020			
Skew:	0.051	Prob(JB):	0.600			
Kurtosis:	3.181	Cond. No.	3.23e+09			
-----	-----	-----	-----	-----	-----	-----

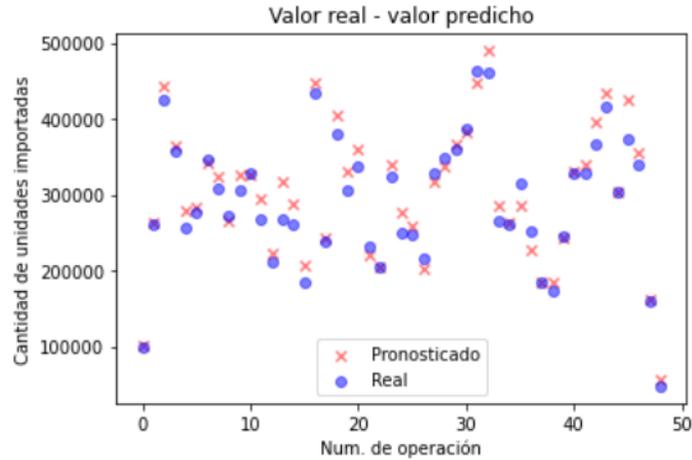
Fuente: Elaboración propia

Como se puede ver en la Figura 36, los resultados de la regresión muestran un coeficiente de determinación del 96.6% (*R-squared* en la gráfica), lo cual comprueba que el modelo logró explicar adecuadamente el comportamiento de las importaciones y debido a que durante el análisis de inflación de la varianza se trató la multicolinealidad, los errores estándar para las variables son menores a los obtenidos inicialmente. Además, se obtuvieron unos valores *t* inferiores al 0.005 en todas las variables explicativas demostrando que todas estas variables son significativas con unos errores estándar reducidos.

Posteriormente, se graficaron los valores pronosticados por el modelo contra los valores reales de importación para el año 2019, observando un grado de ajuste de los datos adecuado y con variaciones menores entre sí. Además, los valores pronosticados tienen una tendencia y

comportamiento similar a los valores reales durante todo el año, comportamiento que se puede apreciar en la Figura 37.

Figura 37. Valor real - Valor predicho



Fuente: elaboración propia

En la Tabla 3 se evalúan, respecto, de las primeras 13 semanas, los valores reales contra la predicción generada por el modelo, en donde se observan porcentajes de error muy bajos que oscilan entre el 0.2 % y el 9.67%.

Tabla 3. Resultado para la predicción semanal año 2019

Semana	Cantidad Unidades 2019 Real	Cantidad Unidades 2019 Proyectado	% Variación
1	100426	102622	2,19%
2	262157	264090	0,74%
3	424931	443961	4,48%
4	357241	363892	1,86%
5	256664	279003	8,70%
6	277831	284578	2,43%
7	345595	342977	-0,76%
8	308011	325364	5,63%
9	271567	266993	-1,68%
10	307421	325868	6,00%
11	327668	326826	-0,26%
12	268462	294433	9,67%
13	213126	222329	4,32%

Fuente: elaboración propia

3.5. Paso a producción

Tras la evaluación de los resultados que generó el modelo en datos cuyos registros son conocidos, y al evidenciar que las salidas tienen un comportamiento similar a las importaciones reales con un nivel de error mínimo y constante a través del tiempo, se procedió a realizar el pronóstico de la cantidad de operaciones de importación para un periodo de tiempo desconocido con las variables regresoras significativas que el análisis de regresión lineal múltiple indicó, las cuales son:

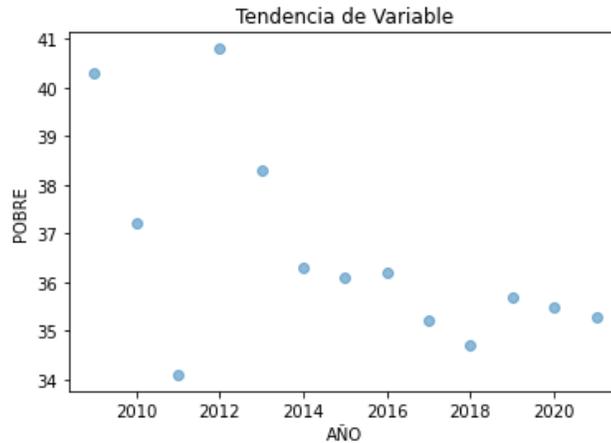
- Índice de Pobreza
- Producto Interno Bruto del sector de telecomunicaciones
- Cantidad de países exportadores
- Tasa de desempleo
- Población

- Cantidad de proveedores de teléfonos móviles celulares
- Peso Neto
- Peso Bruto
- Cantidad de bultos

Hay que apuntar, sin embargo, que los valores de estas variables solo son conocidas en su mayoría para el año 2020, para determinar el comportamiento de estos datos durante el año 2021 se hace necesario realizar la proyección por métodos de series de tiempos. Por lo tanto, para realizar dichas proyecciones y tener las entradas que requiere el modelo para generar el pronóstico del año 2021 se evalúa el comportamiento de cada una de las variables gráficamente para determina el método que mejor se adapta a los datos y generar el pronóstico. A continuación, se relaciona el método utilizado para cada variable:

- **Índice de Pobreza:** Este índice, que se publica anualmente, ha tenido un comportamiento inestable a lo largo del tiempo, ya que es afectado por diversos factores económicos, políticos y sociales. Sin embargo, en los últimos 8 años ha presentado una tendencia de reducción con excepción únicamente en el año 2019. Por este motivo, para pronosticar el comportamiento del año 2020 —ya que se publica en los últimos meses de este— se empleó el método de media móvil simple obteniendo la tendencia mostrada en la Figura 38.

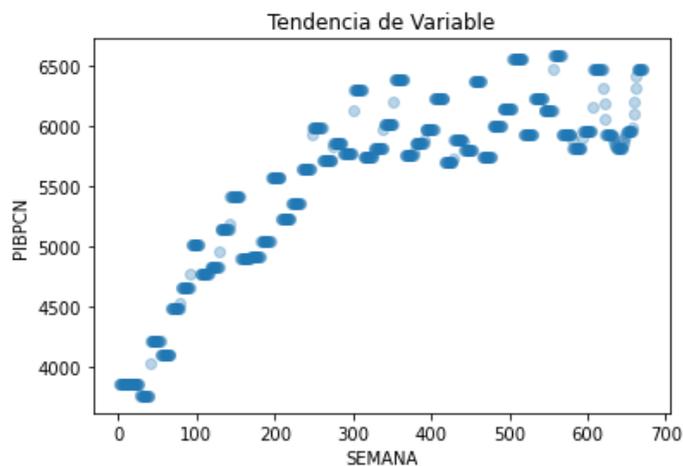
Figura 38. *Tendencia Índice de Pobreza*



Fuente: elaboración propia

- **Producto Interno Bruto del sector de telecomunicaciones:** esta variable presenta un comportamiento logarítmico por el aumento en infraestructura que han tenido que realizar los sectores privados y públicos debido a la implementación de nuevas tecnologías al principio de la década, que a través del tiempo se ha ido estabilizando, teniendo un crecimiento constante como se proyectó durante el año 2021 empleando la regresión logarítmica.

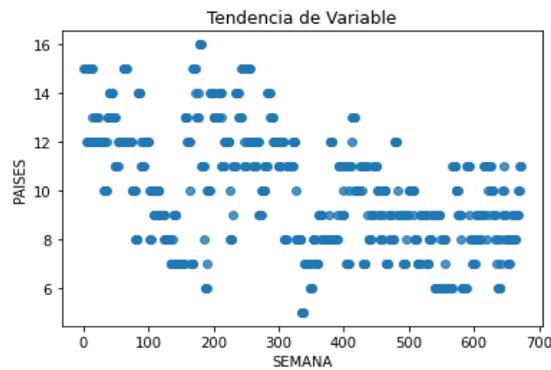
Figura 39. *Tendencia Producto Interno Bruto*



Fuente: elaboración propia

- **Cantidad de países exportadores:** la cantidad de países exportadores es una variable que no tiene una variación cíclica corta; es decir, los datos pueden variar rápidamente y en un rango muy corto, por lo cual se emplea el promedio móvil simple con un número de periodos N de tres.

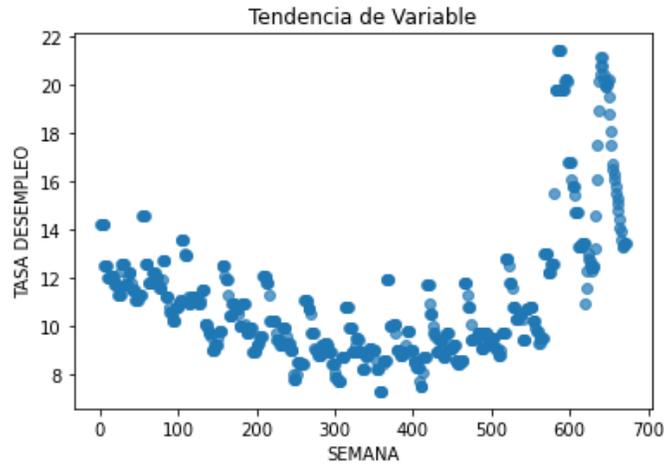
Figura 40. *Tendencia de cantidad de países exportadores*



Fuente: elaboración propia

- **Tasa de desempleo:** Esta variable también presenta un comportamiento cíclico a lo largo de los últimos 8 años; sin embargo, por la pandemia generada por el covid-19 durante el 2020 este índice se incrementó exponencialmente, por lo cual se empleó el método de media móvil, con un número de periodos N de 8, dándole un rango más amplio ya que este índice se calcula mensualmente, tal como se puede ver en la Figura 41.

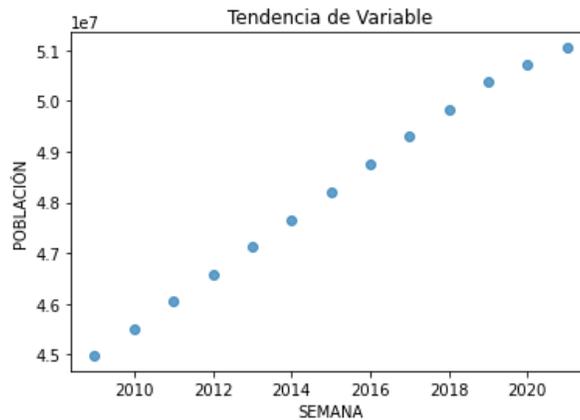
Figura 41. *Tendencia Tasa de Desempleo*



Fuente: elaboración propia

- **Población:** esta variable, cuya estimación se publica anualmente, presentó un comportamiento lineal constante a lo largo del tiempo, por lo cual dar cuenta del valor proyectado en el año 2021 se realizó una regresión lineal simple ya que es el método que mejor se adapta a esta tendencia.

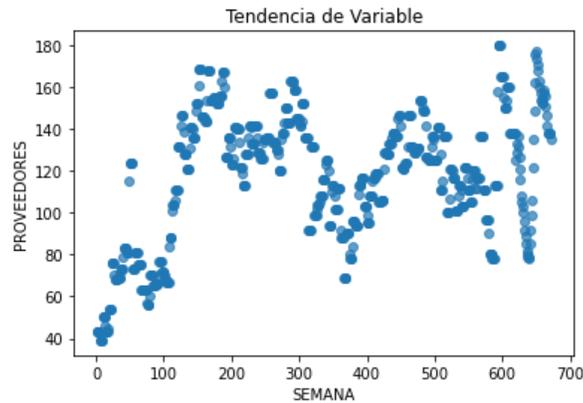
Figura 42. *Tendencia población*



Fuente: elaboración propia

- **Cantidad de proveedores:** esta variable, la cual se calcula semanalmente, posee una tendencia cíclica pero que no está muy definida, por lo cual se empleó el método de media móvil con un número de periodos N de 3.

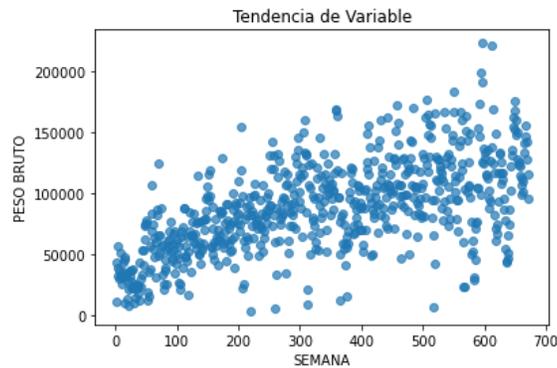
Figura 43. *Tendencia cantidad de proveedores*



Fuente: elaboración propia

- **Peso Bruto:** esta variable que se obtiene semanalmente tiene una tendencia lineal ascendente debido al aumento en la demanda de teléfonos móviles celulares, así como a la cantidad de embarques, por lo que para su proyección en el año 2021 se empleó el método de Regresión Lineal simple.

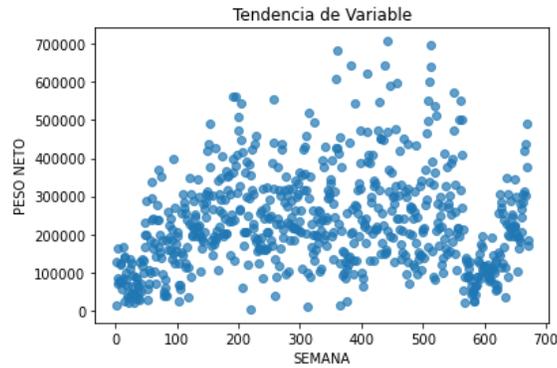
Figura 44. *Tendencia peso bruto*



Fuente: elaboración propia

- **Peso Neto:** esta variable posee un comportamiento logarítmico por la disminución en las dimensiones de los equipos móviles. Sin embargo, se observaron algunos datos atípicos que pueden sesgar el resultado. Pero, a pesar de este sesgo, se determinó que la regresión logarítmica es el mejor método para realizar un pronóstico más acertado.

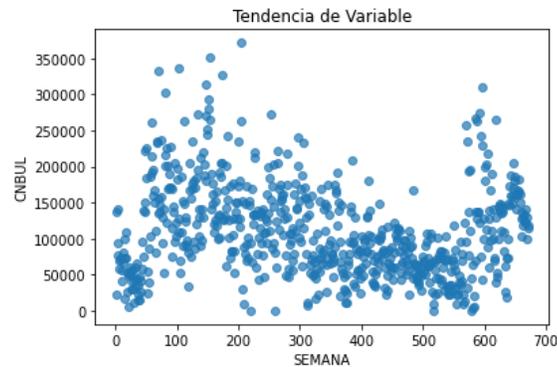
Figura 45. *Tendencia Peso Neto*



Fuente: elaboración propia

- **Cantidad de bultos:** tal variable posee un comportamiento estacional con un ciclo muy amplio, por lo que se decidió emplear el método de media móvil con un número de periodos N de 10 para mantener la tendencia ascendente de los datos en este ciclo.

Figura 46. *Tendencia cantidad de bultos*



Fuente: elaboración propia

3.5.1. *Predicción del modelo Año 2021*

Con las variables de entrada que requiere el modelo, las cuales fueron estimadas por medio de métodos de serie de tiempo, se procedió a cargar la data en el programa por el comando `pd.read_csv` con los datos obtenidos y se generó un arreglo para ingresar los datos al modelo obtenido y así generar las predicciones deseadas.

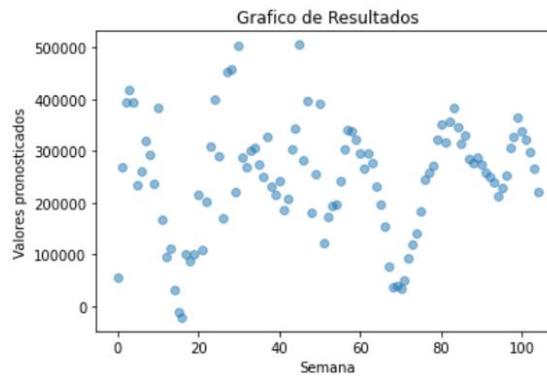
Figura 47. Código para importar los datos para la predicción

```
datasetpred = pd.read_csv("Datos_de_Entrenamiento_M3_2020y2021.csv", delimiter =  
q_pred = datasetpred.iloc[:, :].values
```

Fuente: elaboración propia

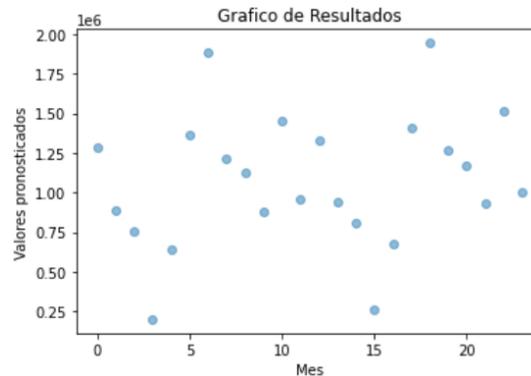
Se realiza un gráfico con los resultados obtenidos para la cantidad de teléfonos móviles celulares que serán importados durante el horizonte de tiempo observado. En dichos gráficos se aprecia que los datos predichos mantienen una estructura similar a la obtenida en la etapa de evaluación.

Figura 48. Gráfico de resultados para la predicción de semana



Fuente: elaboración propia

Figura 49. Gráfico de resultados para la predicción de mes



Fuente: elaboración propia

4. Resultados

Como resultado de implementación de código expuesto en el capítulo de metodología se puede apreciar el diferencial de la utilización de *Python* para la implementación del algoritmo de *machine learning* en cada una de las etapas de desarrollo. Es así como en la etapa de preparación de datos se declaró una arquitectura que permite la adaptabilidad a cualquier conjunto de datos que mantenga un formato común y una estructura estándar, esto permitió la interacción con cuatro conjuntos de datos con diferentes series de tiempos, también se dejó una estructura por medio de la utilización de librerías que permitió declarar a las variables regresoras y de respuesta como un objeto que puede ser fácilmente utilizado para cualquier operación, mejorando los tiempos de cómputo.

Así mismo en la etapa de representación de datos se estructuró el código de manera tal que pueda ejecutarse con diversas entradas, creando gráficos que se ajusten a la estructura del insumo. Dicho esto se pudo ver como utilizando la misma línea de código para los cuatro modelos, se obtuvo gráficos similares que se adaptaron a cada conjunto de datos. Permitiendo la automatización de esta tarea para poder proceder con el análisis de cada escenario.

Por otro lado en la fase de modelado y aprendizaje se implementó un método que garantizó el uso para cada uno de los modelos, generando los coeficientes de evaluación para cada una de las pruebas estadísticas, dando paso a la eliminación de las variables que no cumplieran dichas pruebas.

De igual forma para la fase de evaluación se procede con el entrenamiento de cada uno de los modelos, permitiendo la utilización de las mismas líneas de código para cada ejecución, continuando con la generación de la tabla con las estadísticas del modelo, para de esta manera

poder tomar la decisión con respecto al entrenamiento del modelo y su pertinencia para el paso a producción.

Finalmente se procede con la línea de código de paso a producción que se realiza con el modelo resultante de la evaluación, dejando establecido el algoritmo que permite realizar el cargue de las variables regresoras del nuevo periodo establecido.

Al someter cada uno de los modelos planteados en la etapa de preparación de datos a las fases de representación, modelado, aprendizaje y evaluación, se obtienen cuatro respuestas diferentes para cada uno. A continuación, se enuncian los resultados para cada uno de los modelos.

4.1. Modelo 1 (Operaciones)

El objetivo de este modelo fue realizar la predicción por operación de la cantidad de unidades importadas. Al someterlo a la prueba de multicolinealidad, se encontraron las variables de la Tabla 4 que resultaban poco relevantes y que presentaban un alto índice de colinealidad, las cuales se suprimieron del estudio.

Tabla 4. *Variables eliminadas en Multicolinealidad Modelo 1*

Nomenclatura	Variable
AÑO	Año
TRM	Tasa de Cambio
SMLV	Salario Mínimo
PIBPCOR	Producto Interno Bruto a precios corrientes Sector Información y comunicaciones (Miles de Millones de Pesos)
GINI	Coficiente de GINI Desigualdad País

Fuente: elaboración propia

Posteriormente, se realizó el proceso de homocedasticidad, en el cual se depuraron otras variables que no cumplían con el estadístico de prueba, y que al correr nuevamente para validar la hipótesis mencionada en la Figura 33 de la prueba de *Breush-pagan* daba para la negación de la hipótesis nula. Dicho coeficiente se puede apreciar en la Figura 50.

Figura 50. Resultado de prueba de homocedasticidad para Modelo 1

```
=====
Dep. Variable:          y      R-squared:          0.291
Model:                  OLS    Adj. R-squared:     0.291
Method:                 Least Squares  F-statistic:       2366.
Date:                   Tue, 16 Mar 2021  Prob (F-statistic): 0.00
Time:                   |      21:37:12  Log-Likelihood:    -1.4326e+06
No. Observations:      80594  AIC:               2.865e+06
Df Residuals:          80579  BIC:               2.865e+06
Df Model:              14
Covariance Type:      nonrobust
=====
```

Fuente: elaboración propia

A pesar de no haber cumplido con los supuestos mencionados anteriormente, se procedió a realizar la fase de evaluación, en la cual se obtiene un coeficiente de determinación del 87.9% como se puede apreciar en la Figura 51.

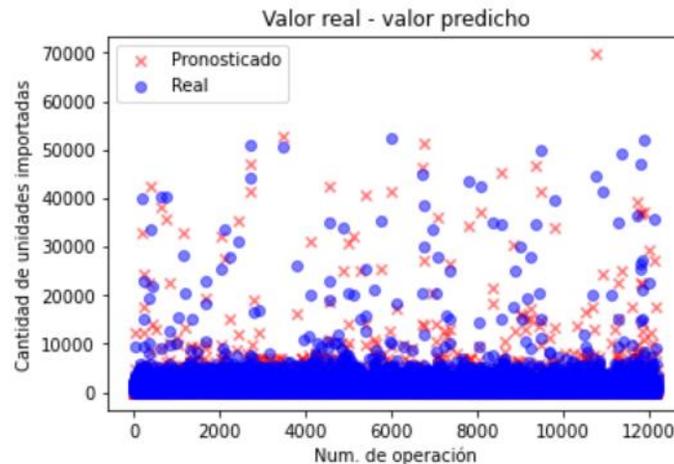
Figura 51. Resultado de coeficiente de determinación Modelo 1

```
=====
                        OLS Regression Results
=====
Dep. Variable:          y      R-squared:          0.879
Model:                  OLS    Adj. R-squared:     0.879
Method:                 Least Squares  F-statistic:       3.545e+04
Date:                   Tue, 16 Mar 2021  Prob (F-statistic): 0.00
Time:                   |      21:41:33  Log-Likelihood:    -5.8601e+05
No. Observations:      68398  AIC:               1.172e+06
Df Residuals:          68383  BIC:               1.172e+06
Df Model:              14
Covariance Type:      nonrobust
=====
```

Fuente: elaboración propia

De la misma manera, se procedió a graficar los resultados obtenidos en la Figura 52.

Figura 52. Resultados predicción Modelo 1



Fuente: elaboración propia

Como se puede apreciar, los resultados no son claros ya que mantienen un comportamiento distorsionado y no tienen una utilidad práctica debido a que no están dados en función de una unidad de tiempo.

4.2. Modelo 2 (Diario)

Con este modelo se buscó realizar predicciones en función del número de día del año. Para este caso se tomaron los datos suministrados en la etapa de procesamiento que venían agrupados por número de día. Se realizó entonces la intervención con las mismas etapas de la metodología: para el supuesto de multicolinealidad dio para eliminación las mismas variables suprimidas en el modelo 1 (véase *Tabla 4*), mientras que para la comprobación del modelo de homocedasticidad se procedió a eliminar las variables que no cumplieran con el estadístico de prueba, lo cual arrojó el coeficiente de la Figura 53 que no cumplía con la hipótesis nula de homocedasticidad planteada en la Figura 33; es decir, los datos no se comportaban de manera homocedastica, por lo cual los resultados podían llegar a ser poco fiables.

Figura 53. Resultado de prueba de homocedasticidad para Modelo 2

```
=====  
Dep. Variable:          y      R-squared:                0.156  
Model:                  OLS    Adj. R-squared:           0.154  
Method:                 Least Squares    F-statistic:              72.27  
Date:                   Tue, 16 Mar 2021    Prob (F-statistic):       1.50e-109  
Time:                   22:27:40      Log-Likelihood:           -62440.  
No. Observations:      3141      AIC:                      1.249e+05  
Df Residuals:          3132      BIC:                      1.250e+05  
Df Model:               8  
Covariance Type:       nonrobust  
=====
```

Fuente: elaboración Propia

A pesar de lo anterior, se procedió con la etapa de evaluación del modelo en la cual la regresión de prueba arrojó un coeficiente de determinación del 95%, tal como se puede apreciar en la Figura 34.

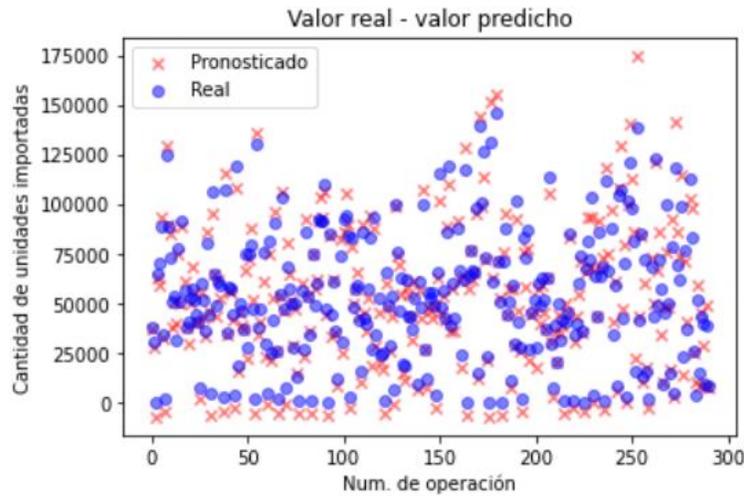
Figura 54. Resultado de coeficiente de determinación Modelo 2

```
=====  
Dep. Variable:          y      R-squared:                0.950  
Model:                  OLS    Adj. R-squared:           0.950  
Method:                 Least Squares    F-statistic:              6814.  
Date:                   Tue, 16 Mar 2021    Prob (F-statistic):       0.00  
Time:                   22:32:52      Log-Likelihood:           -29361.  
No. Observations:      2849      AIC:                      5.874e+04  
Df Residuals:          2840      BIC:                      5.879e+04  
Df Model:               8  
Covariance Type:       nonrobust  
=====
```

Fuente: elaboración Propia

De igual manera, esto dio como resultado lo mostrado en la Figura 55, en donde se pueden apreciar los valores pronosticados contra los valores reales, además de que visualmente existen unos datos que presentan una alta variación.

Figura 55. Resultados predicción Modelo 2



Fuente: elaboración Propia

4.3. Modelo 3 (Semanal)

En este modelo se procedió con la predicción de los datos con una escala de tiempo semanal, la cual es sometida a cada una de las etapas de la metodología, arrojando para el supuesto de multicolinealidad la eliminación de las variables mencionadas en la Tabla 5.

Tabla 5. Variables eliminadas en multicolinealidad Modelo 3

Nomenclatura	Variable
AÑO	Año
TRM	Tasa de Cambio
IVA	Porcentaje de IVA
SMLV	Salario Mínimo
HURTO	Hurto a personas
PIBPCOR	Producto Interno Bruto a precios corrientes Sector Información y comunicaciones (Miles de Millones de Pesos)
GINI	Coefficiente de GINI Desigualdad País

Fuente: elaboración propia

Por otro lado, en la comprobación del modelo de homocedasticidad se procedió a eliminar las variables que no cumplieran con el estadístico de prueba, lo que arrojó el coeficiente de la figura 56 que no cumplía con la hipótesis nula de homocedasticidad planteada en la Figura 33.

Figura 56. *Resultado de prueba de homocedasticidad para Modelo 3*

```

=====
Dep. Variable:          y      R-squared:                0.094
Model:                 OLS    Adj. R-squared:           0.077
Method:                Least Squares  F-statistic:              5.735
Date:                  Tue, 16 Mar 2021  Prob (F-statistic):       3.20e-08
Time:                  22:50:03    Log-Likelihood:           -12163.
No. Observations:     566      AIC:                      2.435e+04
Df Residuals:         555      BIC:                      2.440e+04
Df Model:              10
Covariance Type:      nonrobust
=====

```

Fuente: elaboración propia

En este caso se procedió con la etapa de entrenamiento en la cual se obtiene un coeficiente de determinación del 96.6% como se puede apreciar en la Figura 57.

Figura 57. *Resultado de coeficiente de determinación Modelo 3*

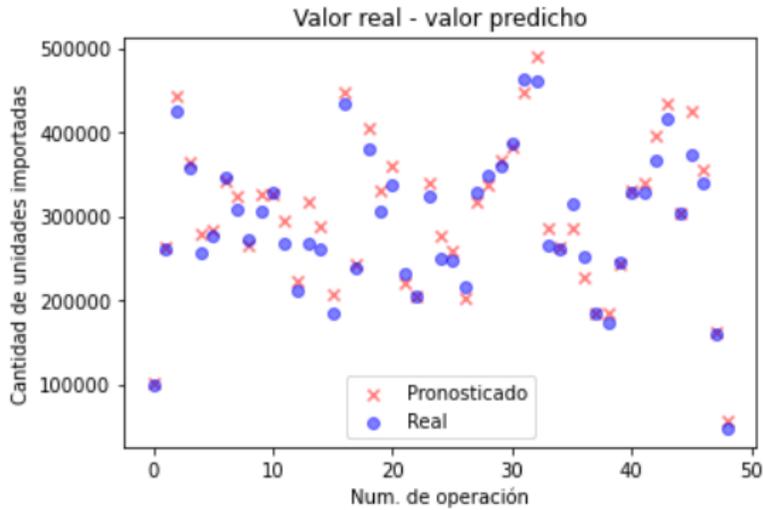
OLS Regression Results						
=====						
Dep. Variable:	y	R-squared:	0.966			
Model:	OLS	Adj. R-squared:	0.966			
Method:	Least Squares	F-statistic:	1448.			
Date:	Wed, 17 Mar 2021	Prob (F-statistic):	0.00			
Time:	22:51:42	Log-Likelihood:	-5838.6			
No. Observations:	517	AIC:	1.170e+04			
Df Residuals:	506	BIC:	1.175e+04			
Df Model:	10					
Covariance Type:	nonrobust					
=====						
	coef	std err	t	P> t	[0.025	0.975]

const	1.735e+05	6.26e+04	2.773	0.006	5.06e+04	2.97e+05
x1	-613.6799	82.574	-7.432	0.000	-775.909	-451.451
x2	0.3210	0.018	17.515	0.000	0.285	0.357
x3	0.1317	0.010	13.566	0.000	0.113	0.151
x4	2.1690	0.050	43.116	0.000	2.070	2.268
x5	202.9617	35.906	5.653	0.000	132.419	273.504
x6	-0.0058	0.001	-4.121	0.000	-0.009	-0.003
x7	-7089.1560	1323.661	-5.356	0.000	-9689.705	-4488.607
x8	1012.0053	442.128	2.289	0.022	143.372	1880.639
x9	15.9201	4.141	3.844	0.000	7.784	24.056
x10	2210.0824	592.155	3.732	0.000	1046.698	3373.467
=====						
Omnibus:	0.769	Durbin-Watson:	1.307			
Prob(Omnibus):	0.681	Jarque-Bera (JB):	0.580			
Skew:	-0.006	Prob(JB):	0.748			
Kurtosis:	3.164	Cond. No.	3.43e+09			
=====						

Fuente: elaboración propia

Esto arrojó el siguiente gráfico de resultados, en el cual se puede apreciar un alto grado de ajuste de la variable pronosticada contra la real.

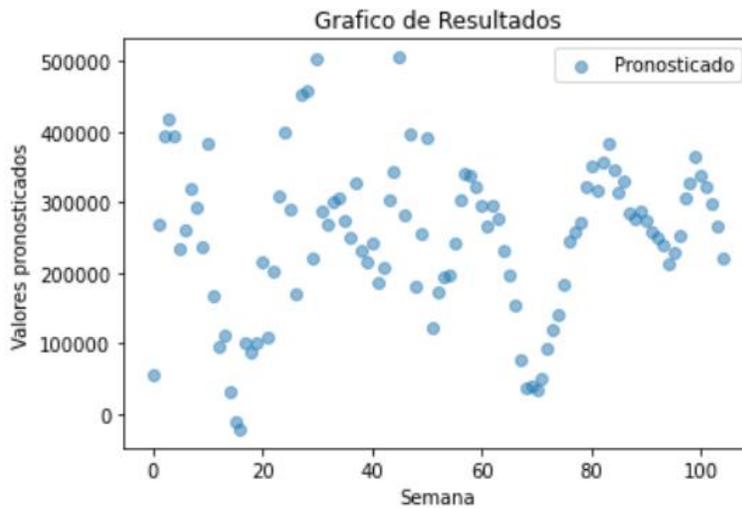
Figura 58. Resultados predicción Modelo 3



Fuente: elaboración propia

Es así como se procede a alimentar este modelo con los datos proyectados en la etapa de producción para los años 2020 y 2021 con las variables que se ven en la Tabla 6, arrojando el siguiente comportamiento

Figura 59. Resultados predicción Modelo 3 año 2020 y 2021



Fuente: elaboración propia

Tabla 6. *Variables de entrenamiento Modelo 3*

Variables de entrenamiento
AÑO
SEMANA
CNBUL
PN
PB
PROVE
POBLA
TASADESEM
PAISES
PIBPCN
POBRE

Fuente: elaboración propia

4.4. Modelo 4 (Mensual)

Para este modelo se tuvo en cuenta el conjunto de datos que tiene la agrupación por serie de tiempo mensual. Así, igual a los anteriores modelos, este se sometió a cada una de las etapas establecidas en la metodología, arrojando para el caso de multicolinealidad la necesidad de eliminar las variables que se muestran en la Tabla 7.

Tabla 7. *Variables eliminadas multicolinealidad Modelo 4*

Nomenclatura	Variable
ARAN	Porcentaje de Arancel
TASEMP	Tasa de Empleo

Fuente: elaboración propia

Una vez realizada la eliminación de variables del supuesto de multicolinealidad, se procedió a validar la homocedasticidad del modelo, eliminando al igual también aquellas variables que no cumplen con el estadístico de prueba. Por tanto, con este procedimiento se obtiene el coeficiente 0.1 de la Figura 60, el cual cumple con la hipótesis nula de la Figura 30, lo que indica que el modelo no es heterocedástico.

Figura 60. Resultado de prueba de homocedasticidad para Modelo 4

```

=====
Dep. Variable:          y      R-squared:              0.105
Model:                 OLS    Adj. R-squared:         0.031
Method:                Least Squares  F-statistic:           1.414
Date:                  Tue, 16 Mar 2021  Prob (F-statistic):    0.191
Time:                  23:28:32  Log-Likelihood:        -2717.3
No. Observations:     118      AIC:                   5455.
Df Residuals:         108      BIC:                   5482.
Df Model:              9
Covariance Type:      nonrobust
=====

```

Fuente: elaboración propia

Ahora bien, una vez comprobados los dos supuestos mencionados anteriormente se procedió con la etapa de entrenamiento que arroja un coeficiente de determinación del 98.3% como se puede apreciar en la Figura 61.

Figura 61. Resultado de coeficiente de determinación Modelo 4

```

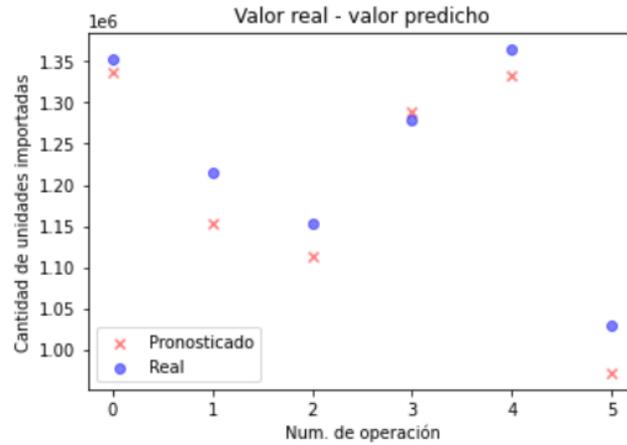
=====
                        OLS Regression Results
=====
Dep. Variable:          y      R-squared:              0.983
Model:                 OLS    Adj. R-squared:         0.981
Method:                Least Squares  F-statistic:           654.9
Date:                  Wed, 17 Mar 2021  Prob (F-statistic):    5.22e-86
Time:                  22:57:25  Log-Likelihood:        -1359.7
No. Observations:     112      AIC:                   2739.
Df Residuals:         102      BIC:                   2767.
Df Model:              9
Covariance Type:      nonrobust
=====
                coef      std err          t      P>|t|      [0.025      0.975]
-----
const      -3.937e+08   3.86e+07   -10.210   0.000   -4.7e+08   -3.17e+08
x1          1.973e+05   1.94e+04    10.181   0.000   1.59e+05   2.36e+05
x2           0.2431    0.046       5.242   0.000    0.151    0.335
x3           0.1078    0.024       4.560   0.000    0.061    0.155
x4           2.4343    0.106      22.980   0.000    2.224    2.644
x5          -0.0045    0.002      -2.964   0.004   -0.007   -0.001
x6          -6.2146    0.847      -7.341   0.000   -7.894   -4.536
x7           0.6419    0.579       1.108   0.270   -0.507    1.791
x8          34.3635   10.442       3.291   0.001   13.652   55.075
x9          1.355e+04   2719.785    4.981   0.000   8152.645   1.89e+04
=====
Omnibus:                1.240   Durbin-Watson:          0.922
Prob(Omnibus):           0.538   Jarque-Bera (JB):       1.322
Skew:                    0.207   Prob(JB):               0.516
Kurtosis:                 2.665   Cond. No.:               4.42e+11
=====

```

Fuente: elaboración propia

De igual manera, este arrojó el siguiente gráfico de resultados para esta etapa de entrenamiento:

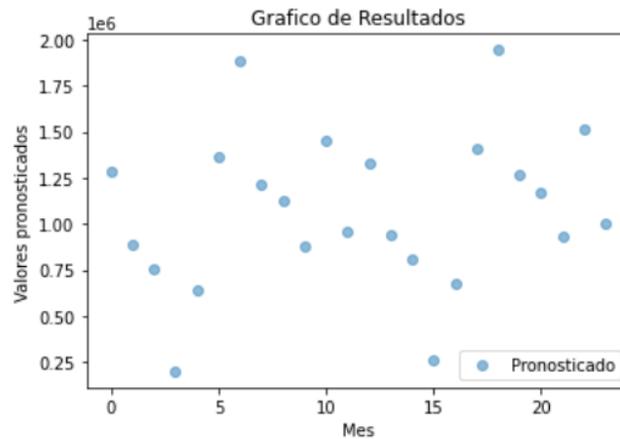
Figura 62. Resultados predicción Modelo 3



Fuente: elaboración propia

Con el modelo entrenado se procedió a alimentarlo con los datos proyectados en la etapa de producción para los años 2020 y 2021, en escala de tiempo mensual y con las variables que se registran en la Tabla 8, lo que dio como resultado el comportamiento observado en la Figura 63.

Figura 63. Resultados predicción Modelo 4 año 2020 y 2021



Fuente: elaboración propia

Tabla 8. *Variables de entrenamiento Modelo 4*

Variables de entrenamiento
AÑO
CNBUL
PN
PB
ABONA
SMLV
HURTO
BALAN
POBRE

Fuente: elaboración propia

Los resultados por mes para los años pronosticados se ven reflejados en las Tablas 9 y 10.

Tabla 9. *Resultados pronósticos mensuales 2020*

Año	Mes	Cantidad de unidades importadas
2020	1	1.284.960
	2	887.155
	3	751.982
	4	200.813
	5	639.320
	6	1.363.930
	7	188.540
	8	121.130
	9	1.128.770
	10	881.049
	11	1.451.170
	12	955.905

Fuente: elaboración propia

Tabla 10. *Resultados pronósticos mensuales 2021*

Año	Mes	Cantidad de unidades importadas
2021	1	1.332.320
	2	937.708
	3	808.534
	4	261.292
	5	680.426
	6	1.404.140
	7	1.943.580
	8	1.264.330
	9	1.172.440
	10	929.090
	11	1.518.010
	12	1.003.570

Fuente: elaboración propia

5. Conclusiones

- El *machine learning* a pesar de tener bases en la estadística, logra optimizar procesos gracias a la implementación de automatización, procedimientos para el manejo de volúmenes de datos e integración con fuentes emisoras de datos. Adicionalmente cuenta con alta adaptabilidad a los cambios que el entorno o las situaciones vayan dándole a los problemas estudiados gracias a la facilidad de modificación en la arquitectura del código.
- La implementación de modelos de *machine learning* en el lenguaje de programación *Python* tiene la ventaja con respecto a otros softwares o incluso a otros lenguajes de programación de ofrecer un catálogo de herramientas amplio para el tratamiento de altos volúmenes de datos permitiendo optimizar los procesos de cómputo y programación, además permite la actualización permanente gracias a las mejoras que la comunidad vaya generando en torno a las herramientas disponibles.
- La fase de preparación juega un papel importante, ya que la manipulación incorrecta de los datos y la selección errónea de variables puede llevar a equivocaciones futuras en la implementación del modelo, presentándose falencias de estructura y asignación de tipo de variables.
- Las variables que en la etapa de preparación de datos aparentan tener un alto grado de incidencia para la predicción no siempre resultan relevantes para la implementación del modelo.
- Existen datos que no tienen una fuente de origen confiable o que no cuentan con una metodología factible para su recolección, por lo que pueden llegar a afectar el comportamiento real de la variable estudiada y llevar a ignorar un aspecto que impacte de manera drástica el comportamiento de la variable estudiada.

- Para la implementación de algoritmos de *machine learning* predictivos es necesario que exista una unidad fija de medida como una escala de tiempo o un parámetro constante a lo largo del modelo, de manera que permita el modelado confiable de las variables regresoras.
- La representación es el primer encuentro con los datos: es en ese momento en donde se genera un insumo para la identificación de necesidad de tratamientos y un panorama general de cuál podría ser el algoritmo adecuado. En esto es importante que la elección de los tipos de representación sean las adecuadas según las variables insumo.
- Es importante la revisión y tratamiento de datos atípicos que pueden ser producto de errores en la alimentación de las bases datos consultadas.
- La estadística juega un papel importante dentro de la implementación de algoritmos de *machine learning*, ya que es el campo de estudio que suministra la fundamentación teórica para la validación de supuestos y la implementación de modelos matemáticos.
- La validación de los supuestos de homocedasticidad permite determinar la confiabilidad del modelo en todas las proyecciones inducidas, de tal modo que se garantice la consistencia en función de la cantidad de datos suministrados y a lo largo del tiempo de uso.
- Resulta buena práctica la validación de un modelo por medio de la división del conjunto de datos en un periodo de tiempo establecido, pues de esta manera es posible validar los valores pronosticados contra los reales encontrando así el coeficiente de determinación del modelo.
- Cuando se obtiene el resultado de la proyección resulta una buena práctica la comparación gráfica de la tendencia de los valores pronosticados contra el comportamiento de los históricos.
- Se pudo encontrar que la compra de telefonía móvil tiene sus picos de demanda a principio, mediano y fin de año, lo que genera un comportamiento similar a una tendencia estacional.

- Los coeficientes resultantes de las variables cambian según el modelo de manera significativa de acuerdo con la serie de tiempo contemplada.
- *Python*, en conjunto con los espacios de trabajo suministrados por *Anaconda*, otorga una serie de facilidades para la implementación de modelos de *machine learning*. Además, cuenta con una comunidad de desarrolladores y académicos que le están aportando constantemente a la evolución del lenguaje y a la optimización de los flujos de trabajo.
- Para la implementación completa de un proceso de *machine learning* resulta necesaria la integración de un gran conjunto de tecnologías que permitan la adaptación total del flujo de trabajo. Sin embargo, el alcance del presente proyecto contempló la fase analítica del proceso de implementación.

6. Trabajos futuros

Con el desarrollo del presente proyecto se abren las puertas a diferentes frentes de acción para el análisis de operaciones de comercio exterior, dentro de las cuales se pueden incluir análisis de mercados nacionales e internacionales para estudiar operaciones de importación y de exportación.

Queda abierta la mejora de poder crear un enlace a un servidor que se pueda conectar a las entidades que se encargan de publicar la información, de manera tal que suministre datos con cierta frecuencia de tiempo y permita tener proyecciones en tiempo real para una óptima toma de decisiones. Así mismo se puede pensar en la utilización de alguna herramienta de inteligencia de negocios que permita mirar el estado de cada una de las variables regresoras implementadas.

La investigación puede tener nuevas extensiones realizando la inclusión de otras variables que tienen interacción con el objeto de estudio y también se puede realizar el estudio en función de un sector en específico y no en una gama de productos como el caso del presente proyecto.

Finalmente se deja abierta la posibilidad de utilizar otras técnicas de la inteligencia computacional que permitan estructurar otros tipos de arquitectura para la ejecución de regresiones. También se puede pensar en un desarrollo orientado a la experiencia de usuario en donde se pueda manipular de manera dinámica la tendencia de las variables regresoras (por ejemplo, poder manipular manualmente el comportamiento de una variable como la inflación).

Bibliografía

- Alpaydin, E. (2010). *Introduction to machine learning*. Cambridge: Massachusetts Institute of Technology.
- Banco de la republica. (2021). Obtenido de <https://www.banrep.gov.co/es/estadisticas/indice-precios-consumidor-ipc>
- Camelo, S. (2014). *El Problema de Clasificación en Aprendizaje de Máquinas*.
- Colomé, D. y Femenia, P. (2018). *Metodología de investigación para cursos de posgrado en ingeniería*. San Juan: Ediciones Plaza.
- DIAN. (2020). *DIAN*. Obtenido de <https://www.dian.gov.co/dian/entidad/Paginas/Presentacion.aspx>
- Guareño, M. (2013). *Support Vector Regression: Propiedades y Aplicación*. Sevilla: Universidad de Sevilla.
- Gujarati, D. y Porter, D. (2009). *Econometría. Quinta edición*. Mexico: McGraw-Hill.
- Harrington, P. (2012). *Machine Learning in Action*. Nueva York: Manning.
- Hurwitz, J. y Kirsch, D. (2018). *Machine Learning For Dummies*. Hoboken: IBM Limited Edition.
- Legiscomex. (2010). *Legiscomex*. Obtenido de <https://www.legiscomex.com/Documentos/incoterms-2011-fob>
- Lozano, I., Arias, F., Bejarano, J., González, A., Granger, C., Hamann, F., . . . Rodríguez, D. (2019). *La política fiscal y la estabilización*. Bogotá: ESPE.
- Martin, E. (28 de Noviembre de 2017). *El país*. Obtenido de https://elpais.com/tecnologia/2017/11/28/actualidad/1511866764_933798.html
- Marzal Varó, A., Gracia Luengo, I. y García Sevilla, P. (2014). *Introducción a la programación con Python 3*. Castellón de la Plana: Universidad Jaume.

Matplotlib. (28 de 01 de 2021). Obtenido de <https://matplotlib.org/stable/index.html>

Ministerio de Comercio, I. y. (2020). *Mincomercio*. Obtenido de <https://www.mincit.gov.co/ministerio/organizacion/mision-vision-objetivos-normas-principio-etico>

Ministerio de comercio, industria y turismo. (16 de Octubre de 2015). *MINTIC*. Obtenido de https://www.mintic.gov.co/portal/604/articles-13720_documento.pdf

Ministerio de hacienda. (28 de Diciembre de 1999). *Sig*. Obtenido de https://www.sic.gov.co/recursos_user/documentos/normatividad/Dec2685_1999.pdf

Nieto, V. (2016). Una nota sobre la evolución de la estructura arancelaria de Colombia 2002 - 2014. *Tiempo y economía*, 73-113.

Nilsson , N. (1998). *Introduction to machine learning*. California: Stanford.

Novales Cinca, A. (1993). *ECONOMETRÍA Segunda edición*. Madrid: McGraw-Hill.

Numpy. (2021). Obtenido de <https://numpy.org/doc/stable/contents.html>

Pandas. (02 de 03 de 2021). Obtenido de <https://pandas.pydata.org/docs/index.html>

Pineda Cortés, L. (2017). *La Computación en México por especialidades académicas*. México: Academia Mexicana de Computación.

Pontil, M., Rifkin, R. y Theodoros, E. (1998). *From Regression to Classification in Support Vector*. Massachusetts: Massachusetts Institute of Technology.

Poole, D., Mackworth, A. y Goebel, R. (1998). *Computational Intelligence*. New York: Oxford University Press.

Pypi. (2021). Obtenido de <https://pypi.org/>

Raschka, S. y Mirjalili, V. (2019). *Python Machine Learning*. Marcombo.

Scikit-learn. (2021). Obtenido de <https://scikit-learn.org/stable/index.html>

Seaborn. (2021). Obtenido de <https://seaborn.pydata.org/index.html>

Shwartz, S. y David, S. (2014). *Understanding Machine Learning*. New York: Cambridge University Press.

Soloaga, A. (19 de 10 de 2018). *Akademus*. Obtenido de <https://www.akademus.es/blog/programacion/principales-usos-python/>

Statsmodels. (2021). Obtenido de <https://www.statsmodels.org/stable/index.html#>

Stock, J. y Watson, M. (2012). *Introducción a la Econometría, 3.ª edición*. Madrid: PEARSON EDUCACIÓN.

Venners, B. (13 de 6 de 2003). *Artima*. Obtenido de <https://www.artima.com/articles/the-making-of-python>

WayBack Machine. (01 de 2021). Obtenido de <https://web.archive.org/>

Wooldridge, J. (2010). *Introducción a la econometría. Un enfoque moderno, 4a. edición*. Mexico: Cengage Learning.

Lista de Tablas

Tabla 1. Total transacciones de importación de la subpartida 8517120000	9
Tabla 2. Abreviatura de variables del modelo	49
Tabla 3. <i>Resultado para la predicción semanal año 2019</i>	67
Tabla 4. <i>Variables eliminadas en Multicolinealidad Modelo 1</i>	76
Tabla 5. <i>Variables eliminadas en multicolinealidad Modelo 3</i>	80
Tabla 6. Variables de entrenamiento Modelo 3	84
Tabla 7. Variables eliminadas multicolinealidad Modelo 4	84
Tabla 8. Variables de entrenamiento Modelo 4	87
Tabla 9. Resultados pronósticos mensuales 2020	87
Tabla 10. Resultados pronósticos mensuales 2021	88

Lista de Figuras

Figura 1. Matriz X.....	22
Figura 2. Vector Y	22
Figura 3. Vector W	22
Figura 4. Homocedasticidad y heterocedasticidad.....	28
Figura 5. Árbol de decisión.....	30
Figura 6. Vectores de Soporte Regresión	31
Figura 7. Interfaz Anaconda	35
Figura 8. Interfaz Spyder	36
Figura 9. Código para el cargue de datos.....	51

Figura 10. Declaración de variables.....	51
Figura 11. <i>Gráfico de parejas</i>	51
Figura 12. Declaración para el gráfico de calor.....	52
Figura 13. Gráfico de calor.....	53
Figura 14. Código para la declaración de coeficientes de correlación.....	53
Figura 15. Gráfico de coeficientes de correlación.....	54
Figura 16. Código para la declaración de gráfico bidimensional de nube de puntos.....	54
Figura 17. Gráfico de nube de puntos para cuatro variables.....	55
Figura 18. Código para la declaración de gráfico normalidad estadística.....	55
Figura 19. Gráfico normal para la variable PN.....	56
Figura 20. Gráfico de probabilidades de variable PN.....	56
Figura 21. Código para la normalización de variables.....	57
Figura 22. Antes y después de la normalización estadística.....	57
Figura 23. Código para realiza la predicción en función de la primera variable.....	58
Figura 24. R cuadrado para la predicción de la primera variable.....	58
Figura 25. Código para realiza la predicción en función de la segunda variable.....	59
Figura 26. Arreglo con todos los R cuadrado del modelo.....	59
Figura 27. Código para la primera regresión del supuesto de homocedasticidad.....	60
Figura 28. Creación de columna de unos.....	60
Figura 29. <i>Resultado de los p-valor para cada variable</i>	61
Figura 30. Declaración del arreglo sin las variables eliminadas.....	61
Figura 31. Regresión en función de la diferencia de cuadrados.....	62
Figura 32. Resumen de la regresión.....	62

Figura 33. Planteamiento de hipótesis de homocedasticidad.....	63
Figura 34. Gráfico de homocedasticidad	63
Figura 35. División de variables para el entrenamiento del modelo.....	64
Figura 36. Resultados de la regresión para el modelo 3	65
Figura 37. Valor real - Valor predicho.....	66
Figura 38. <i>Tendencia Índice de Pobreza</i>	69
Figura 39. Tendencia Producto Interno Bruto	69
Figura 40. Tendencia de cantidad de países exportadores	70
Figura 41. <i>Tendencia Tasa de Desempleo</i>	70
Figura 42. Tendencia población.....	71
Figura 43. Tendencia cantidad de proveedores.....	72
Figura 44. Tendencia peso bruto.....	72
Figura 45. Tendencia Peso Neto	73
Figura 46. Tendencia cantidad de bultos	73
Figura 47. Código para importar los datos para la predicción	74
Figura 48. Gráfico de resultados para la predicción de semana	74
Figura 49. Gráfico de resultados para la predicción de mes	74
Figura 50. Resultado de prueba de homocedasticidad para Modelo 1	77
Figura 51. <i>Resultado de coeficiente de determinación Modelo 1</i>	77
Figura 52. Resultados predicción Modelo 1	78
Figura 53. Resultado de prueba de homocedasticidad para Modelo 2	79
Figura 54. Resultado de coeficiente de determinación Modelo 2.....	79
Figura 55. Resultados predicción Modelo 2	80

Figura 56. Resultado de prueba de homocedasticidad para Modelo 3	81
Figura 57. <i>Resultado de coeficiente de determinación Modelo 3</i>	81
Figura 58. <i>Resultados predicción Modelo 3</i>	82
Figura 59. Resultados predicción Modelo 3 año 2020 y 2021	83
Figura 60. Resultado de prueba de homocedasticidad para Modelo 4	85
Figura 61. Resultado de coeficiente de determinación Modelo 4.....	85
Figura 62. Resultados predicción Modelo 3	86
Figura 63. Resultados predicción Modelo 4 año 2020 y 2021	86

Lista de Ecuaciones

Ecuación 1. Ecuación de la recta	19
Ecuación 2. Regresión lineal múltiple	20
Ecuación 3. Planteamiento de ecuaciones	21
Ecuación 4. Nueva representación del sistema de ecuaciones.....	23
Ecuación 5. Error Cuadrático.....	23
Ecuación 6. Error cuadrático medio.....	23
Ecuación 7. Error cuadrático simplificado.....	23
Ecuación 8. Derivada del error cuadrático.....	24
Ecuación 9. Ecuación en términos de W	24
Ecuación 10. Factor de Inflación de la Varianza	27

Anexos

Anexo 1. Índice de variables

Anexo 2. Editor de Código Modelo 1

Anexo 3. Graficas Modelo 1

Anexo 4. Resultados Consola Modelo 1

Anexo 5. Editor de Código Modelo 2

Anexo 6. Gráficas Modelo 2

Anexo 7. Resultados consola Modelo 2

Anexo 8. Editor de Código Modelo 3

Anexo 9. Graficas Modelo 3

Anexo 10. Resultados consola Modelo 3

Anexo 11. Editor de Código Modelo 4

Anexo 12. Graficas Modelo 4

Anexo 13. Resultados consola Modelo 4

Anexo 14. Datos Utilizados