

TFM

TRABAJO DE FIN DE MÁSTER

Análisis de las ventas, compras y stock

Unamuno Acha, Isabel

2023 - 2024

Máster Universitario en

Análisis de Datos para la Inteligencia de Negocio /

Business Analytics



TRABAJO DE FIN DE MÁSTER

ANÁLISIS DE INVENTARIO

TRABAJO DE FIN DE MÁSTER PRESENTADO EN: Mondragon Unibertsitatea

PARA LA OBTENCIÓN DEL TÍTULO DE: Máster Universitario en Análisis de Datos para la Inteligencia de Negocio / Business Analytics

AUTORA: Isabel Unamuno Acha

DIRECTOR/A: XXX

TUTOR/A: Jon Perez Visaires

ORGANIZACIÓN EN LA QUE HA REALIZADO EL PROYECTO: Datua

FECHA DE DEFENSA: Bilbao, a 05 de 07 de 2024



El autor/la autora del Trabajo de Fin de Máster autoriza a la Facultad de Empresariales de Mondragón Unibertsitatea, con carácter gratuito y con fines exclusivamente de investigación y docencia, los derechos de reproducción y comunicación pública de este documento siempre que: se cite el autor/la autora original, y el uso que se haga de la obra no sea comercial.



Reconocimiento – NoComercial – CompartirIgual (by-nc-sa): No se permite un uso comercial de la obra original ni de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.

DECLARACIÓN DE ORIGINALIDAD

Yo Isabel Unamuno Acha

Declaro que este Trabajo de Fin de Máster es original, fruto de mi trabajo personal, y que no ha sido previamente presentado para obtener otro título o calificación profesional.

Las ideas, formulaciones, imágenes, ilustraciones tomadas de fuentes ajenas han sido debidamente citadas y referenciadas.

RESUMEN

Este Trabajo de Fin de Máster se centra en el diseño y creación de un cuadro de mando interactivo para una empresa mayorista de vinos y licores, con el objetivo de optimizar la gestión de compras, ventas y stock mediante el uso de Power BI. El proyecto abarca desde la limpieza y procesamiento de datos, utilizando Python y herramientas como Pandas y NumPy, hasta la implementación de un sistema de visualización eficiente y accesible para los directivos de la empresa.

La limpieza de datos realizada en Python asegura la integridad y precisión de los conjuntos de datos utilizados, facilitando el análisis y la toma de decisiones más confiables. El proceso incluye la identificación y corrección de valores nulos, la eliminación de duplicados, y la normalización de formatos inconsistentes.

La creación del modelo ha implicado el diseño y estructuración de diversas tablas para organizar y almacenar la información de manera eficiente. Utilizando herramientas avanzadas de bases de datos, se han definido y relacionado entidades clave mediante la creación de tablas normalizadas. Este proceso ha permitido asegurar la integridad referencial y optimizar el acceso a los datos. Además, se han implementado índices y claves primarias para mejorar la velocidad de las consultas, facilitando así una gestión más eficaz de grandes volúmenes de datos y la preparación del conjunto de datos para análisis y visualización avanzada.

El cuadro de mando desarrollado ofrece una visión integral del rendimiento de la empresa, permitiendo a los usuarios navegar fácilmente entre diferentes secciones y acceder a información detallada según sus necesidades. Inicialmente, el análisis se ha realizado con datos de enero de 2016, recopilados en archivos CSV locales. Sin embargo, se plantea una futura integración con SAP, que permitirá manejar mayores volúmenes de datos de forma más eficaz y en tiempo real. Esta migración no solo mejorará la escalabilidad y precisión de los análisis, sino que también facilitará la toma de decisiones estratégicas basadas en datos actualizados.

En conclusión, este proyecto piloto ha demostrado el valor de una gestión de datos robusta y bien estructurada, proporcionando una plataforma interactiva que mejora significativamente la toma de decisiones y posiciona a la empresa para enfrentar futuros retos en un entorno de negocios cada vez más competitivo y orientado a los datos.

LABURPENA

Master Amaierako Lan hau ardoen eta likoreen handizkako enpresa batentzat aginte-koadro interaktibo bat diseinatu eta sortzean oinarritzen da; Power BI erabiliz erosketen, salmenten eta stockaren kudeaketa optimizatzeko helburuarekin. Proiektuaren helburu nagusiak datuak garbitzea eta prozesatzea eta aginte panela diseinatzea dira. Horretarako, lehen helbururako Python, Pandas eta NumPy bezalako tresnak erabili dira, eta bigarrenenerako, enpresako zuzendariak erraz ikusteko moduko bistaratze-sistema eraginkor bat, hau da Power BI.

Python-en egindako datuen garbiketak, erabilitako datu-multzoen osotasuna eta zehaztasuna ziurtatzen ditu, analisiak eta erabaki fidagarriagoak hartzea erraztuz. Prozesuaren barruan sartzen dira balio nuluak identifikatzea eta zuzentzea, bikoizketak ezabatzea eta funtsik gabeko formatuak normalizatzea.

Eredua sortzeko, hainbat taula diseinatu eta egituratu dira, informazioa eraginkortasunez antolatu eta biltegitratzeko. Datu-baseen tresna aurreratuak erabiliz, funtsezko erakundeak definitu eta erlazionatu dira, taula normalizatuak sortuz. Prozesu horri esker, integritate erreferentziala ziurtatu eta datuetarako sarbidea optimizatu da. Gainera, lehen mailako indizeak eta gakoak ezarri dira kontsulten abiadura hobetzeko, eta, horrela, datu-bolumen handien kudeaketa eraginkorragoa eta datu-multzoa analisi eta bistaratze aurreraturako prestatzea erraztu da.

Garatutako aginte-koadroak enpresaren errendimenduaren ikuspegi integrala ematen du, eta erabiltzaileei aukera ematen die atal batetik bestera erraz nabigatzeko eta beharren arabera informazio zehatza eskuratzeko. Hasiera batean, azterketa 2016ko urtarileko datuekin egin da, tokiko CSV artxiboetan bilduta. Hala ere, etorkizunean SAP-ekin integratzea planteatzen da, datu-bolumen handiagoak modu eraginkorragoan eta denbora errealean maneiatu ahal izateko. Migrazio horrek analisiaren zehaztasuna hobetzeaz gain, datu eguneratuetan oinarritutako erabaki estrategikoak hartzea ere erraztuko du.

Laburbilduz, proiektu pilotu honek datuen kudeaketa sendo eta ondo egituratuaren balioa erakutsi du, erabakiak hartzea nabarmen hobetuko duen plataforma interaktibo bat eskainiz. Gainera, gero eta lehiakorragoa eta datuetara bideratuagoa den negozio-ingurune batean, etorkizuneko erronkei aurre egiteko ongi kokatzen du enpresa.

ABSTRACT

This master's Thesis focuses on the design and creation of an interactive dashboard for a wine and spirits wholesale company, with the aim of optimizing the management of purchases, sales and stock using Power BI. The project ranges from data cleaning and processing, using Python and tools such as Pandas and NumPy, to the implementation of an efficient and accessible visualization system for the company's managers.

The data cleaning performed in Python ensures the integrity and accuracy of the data sets used, facilitating more reliable analysis and decision making. The process includes identifying and correcting null values, removing duplicates, and normalizing inconsistent formats.

The creation of the model involves the design and structuring of several tables to organize and store the information efficiently. Using advanced database tools, key entities have been defined and related through the creation of standardized tables. This process has ensured referential integrity and optimized data access. In addition, indexes and primary keys have been implemented to improve the speed of queries, thus facilitating more efficient management of large volumes of data and preparation of the dataset for advanced analysis and visualization.

The developed dashboard provides a comprehensive view of the company's performance, allowing users to easily navigate between different sections and access detailed information according to their needs. Initially, the analysis has been performed with data of January 2016, collected in local CSV files. However, a future integration with SAP is planned, which will allow handling larger volumes of data more efficiently and in real time. This migration will not only improve the scalability and accuracy of analytics but will also facilitate strategic decision making based on up-to-date data.

In conclusion, this pilot project has demonstrated the value of robust and well-structured data management, providing an interactive platform that significantly improves decision making and positions the company to face future challenges in an increasingly competitive and data-driven business environment.

ÍNDICE

RESUMEN.....	3
LABURPENA.....	4
ABSTRACT	5
1. INTRODUCCIÓN.....	11
1.1. PRESENTACIÓN DE LA EMPRESA Y PROBLEMÁTICA	11
1.2. MARCO TEÓRICO.....	11
1.3. OBJETIVOS	16
1.4. DESCRIPCIÓN DE LA METODOLOGÍA EMPLEADA	17
1.5. PLANIFICACIÓN DEL PROYECTO	18
2. DESARROLLO	19
2.1. ANÁLISIS DE LOS DATOS.....	19
2.1.1. <i>Entrada de datos:</i>	19
2.1.2. <i>Análisis exploratorio</i>	20
2.1.3. <i>Limpieza y procesamiento</i>	21
2.2. MODELADO DE DATOS	24
2.2.1. <i>Modelo semántico</i>	24
2.2.2. <i>Relaciones entre tablas de hechos y dimensiones</i>	25
2.2.3. <i>Definición de medidas y KPIs</i>	28
2.3. VISUALIZACIÓN	29
2.3.1. <i>Estructura del informe</i>	29
2.3.2. <i>Especificación del diseño del informe analítico</i>	30
2.4. APLICACIÓN DE MODELOS PARA LA PREDICCIÓN DE LA DEMANDA	34
2.1. CONTROL DE VERSIONES EN GIT	42
3. RESULTADOS	45
3.1. NAVEGACIÓN ENTRE PÁGINAS	45
3.2. DIMENSIONES DE FILTRO	46
3.3. PÁGINAS DEL INFORME	46
3.3.1. <i>Página de inicio</i>	47
3.3.2. <i>Página de resumen</i>	47
3.3.3. <i>Página de ventas</i>	48
3.3.4. <i>Página de compras</i>	49
3.3.5. <i>Página de stock</i>	50

3.3.6.	<i>Página de tiendas</i>	51
4.	CONCLUSIONES Y DISCUSIÓN	52
5.	LINEAS FUTURAS	53
6.	BIBLIOGRAFÍA	54
7.	ANEXOS	57

ÍNDICE DE FIGURAS

FIGURA 1: PLANIFICACIÓN GANTT DEL PROYECTO	18
FIGURA 2: MODELO ESTRELLA; FUENTE: [19].....	24
FIGURA 3: DESCRIPCIÓN DEL USO DE POWER BI PARA CREAR ANÁLISIS CONTROLADOS POR DATOS; FUENTE: [23].....	27
FIGURA 4: DISEÑO DEL MODELO	28
FIGURA 5: DISTRIBUCIÓN DE COMPRAS DIARIAS	34
FIGURA 6: DISTRIBUCIÓN DE VENTAS DIARIAS	35
FIGURA 7: DISTRIBUCIÓN DE VENTAS DIARIAS AJUSTADAS	36
FIGURA 8: SERIE TEMPORAL DE VENTAS Y COMPRAS	37
FIGURA 9: PREDICCIÓN DE LAS UNIDADES VENDIDAS	38
FIGURA 10: PREDICCIONES DE LAS UNIDADES VENDIDAS MEDIANTE RANDOM FOREST	40
FIGURA 11: DESCOMPOSICIÓN DE LA SERIE TEMPORAL	41
FIGURA 12: PREDICCIÓN DE LAS VENTAS DE LOS PRÓXIMOS 14 DÍAS	42
FIGURA 13: CONTROL DE VERSIONES MEDIANTE LA INTEGRACIÓN DE GITHUB EN VISUAL STUDIO CODE	44
FIGURA 14: NAVEGACIÓN ENTRE PÁGINAS	46
FIGURA 15: DIMENSIONES DE FILTRO	46
FIGURA 16: PÁGINA DE INICIO	47
FIGURA 17: PÁGINA DE RESUMEN	48
FIGURA 18: VISTA GRÁFICA DE LA SUBPÁGINA DE VENTAS DENTRO DEL APARTADO DE VENTAS	49
FIGURA 19: SUBPÁGINA DE PROVEEDOR DENTRO DEL APARTADO DE COMPRAS	50
FIGURA 20: PÁGINA DE STOCK	50
FIGURA 21: PÁGINAS DE LAS TIENDAS	51

INDICE DE TABLAS

TABLA 1: ANÁLISIS EXPLORATORIO DE LOS DATOS	21
TABLA 2: METRICAS DE EVALUACIÓN DE LOS MODELOS	42

INDICE DE ECUACIONES

ECUACIÓN 1: MEAN ABSOLUTE ERROR.....	38
ECUACIÓN 2: ROOT MEAN SQUARE ERROR.....	38

1.INTRODUCCIÓN

1.1. Presentación de la empresa y problemática

El entorno empresarial actual se caracteriza por un creciente control basado en datos. Tanto pequeñas como grandes empresas utilizan los datos para decisiones en ventas, contratación, objetivos y otras áreas. Sin embargo, para el usuario promedio sin conocimientos de análisis de datos o estadísticas, puede ser difícil comprender y visualizar estos datos de manera clara. Eso es lo que le ocurre a la siguiente empresa. Es un mayorista de vinos y licores que opera en múltiples ubicaciones en Estados Unidos. Con aproximadamente 80 establecimientos y unas ventas anuales superiores a 450 millones de dólares, la empresa ha experimentado dificultades significativas para gestionar el alto volumen de datos sobre las compras, ventas y stock. La empresa maneja ventas y costos de bienes vendidos que alcanzan los cientos de millones de dólares, lo que hace que las hojas de cálculo tradicionales resulten insuficientes para gestionar el vasto volumen de datos, que incluye millones de registros de ventas, compras e inventario. Esta complejidad requiere un enfoque sofisticado para un análisis eficaz. Estos problemas han llevado al equipo directivo a buscar soluciones para analizar y optimizar sus prácticas de gestión. El objetivo es diseñar un cuadro de mando que permita controlar las compras, ventas y el stock mediante una interacción rápida y comprensible, así como obtener información valiosa sobre clientes y proveedores. Esto permitirá a la empresa tomar decisiones de valor de un vistazo, mejorando la eficiencia operativa y la capacidad de respuesta en un entorno altamente competitivo.

1.2. Marco teórico

En el entorno empresarial actual, la aplicación de algoritmos de *Machine Learning (ML)* para la gestión de inventarios se ha convertido en una herramienta esencial. Estas técnicas avanzadas permiten a las empresas predecir con mayor precisión la demanda futura y ajustar sus niveles de stock de manera eficiente. Al optimizar los costos de almacenamiento y reducir las pérdidas asociadas con la escasez de productos, las empresas pueden no solo mejorar su rentabilidad, sino también elevar la satisfacción del cliente al asegurar la disponibilidad continua de los productos más demandados. La implementación de algoritmos de *ML* en la gestión de inventarios proporciona una ventaja competitiva significativa. Estas herramientas permiten a las empresas operar de forma más eficiente, responder mejor a las fluctuaciones del mercado y adaptarse rápidamente a las necesidades cambiantes de los consumidores. Por ello, en el siguiente apartado, primero de todo se ha resumido el concepto de *ML* y se han analizado cuáles son los diferentes algoritmos que se emplean actualmente para la predicción de la

demanda y la mejora en la gestión de *stock* [1].

Los algoritmos de *ML* permiten identificar patrones complejos en grandes volúmenes de datos, infiriendo sus propias reglas para reconocer patrones similares en nuevos conjuntos de datos. Esto da lugar a la creación de sistemas inteligentes que mejoran de forma autónoma mediante la observación de datos. Estos sistemas pueden aprender a predecir comportamientos, detectar similitudes o anomalías de manera automática, y tomar decisiones adecuadas basadas en los datos analizados. Los modelos de *ML* analizan datos reales que representan el proceso que se desea mejorar o automatizar. Este proceso, conocido como entrenamiento de algoritmos, permite que el sistema extraiga conclusiones relevantes y aprenda a realizar tareas sin necesidad de programación previa, simplemente observando y analizando los datos proporcionados. En los proyectos de *ML*, hay tres pasos que se deben seguir, que son los siguientes.

- Revisión de datos: Implica examinar los datos disponibles para determinar cuáles son útiles, asegurando su correcto almacenamiento y formatos adecuados. Es crucial limpiar los datos eliminando registros antiguos, incompletos o erróneos para garantizar la calidad del *dataset*.
- Organización de datos: Fundamental para facilitar el aprendizaje automático. Se seleccionan datos que reflejen las decisiones que el sistema deberá tomar automáticamente, estructurándolos de manera que optimicen el proceso de entrenamiento del modelo.
- Entrenamiento y la validación del modelo: Durante esta fase, el modelo analiza los datos para identificar patrones y relaciones, ajustando sus parámetros para poder detectar estas características automáticamente en futuros datos. Este proceso permite al modelo configurar su propio sistema de detección basado en la información observada [2].

Tras conocer los conceptos generales del *ML*, se ha analizado como se aplican en la gestión de inventarios actualmente. Las redes neuronales son ampliamente utilizadas para predecir el comportamiento del inventario. Los modelos de *backpropagation* (BPNN) destacan por su estructura simple y capacidad de aprendizaje. Por un lado, se han mencionado algunos científicos que han desarrollado varios artículos que se han considerado interesantes, entre ellos:

- Sustrova, que desarrolló varios modelos de redes neuronales para optimizar la cantidad de stock, seleccionando el modelo con menor error cuadrático medio (MSE) y un coeficiente de determinación (R^2) cercano a uno [3].

- Shoujing Zhang, combinó BPNN con clustering mejorado y PCA para gestionar el inventario de piezas de repuesto en mantenimiento de camiones, evitando el impacto subjetivo de los gestores de inventario [4].
- Praveen y otros, utilizaron el modelo de regresión XGBoost para predicciones de demanda en pequeñas y medianas empresas, logrando reducir el stock y el capital dedicado [5].

Por otro lado, se han recogido algunos de los algoritmos que se han considerado interesantes y se ha explicado la función de ellas brevemente.

- Red Neuronal Artificial multicapa con Embeddings: Los *embeddings* son una técnica para convertir variables categóricas en representaciones continuas, que se utilizan como entradas en un modelo de red neuronal para aprendizaje supervisado. Estas representaciones permiten que las categorías similares se ubiquen más cerca entre sí y se ajustan durante el entrenamiento para minimizar la pérdida del modelo. Al emplear embeddings, se pueden obtener predicciones más precisas en comparación con el uso exclusivo de las unidades de stock [1].
- Gradient Boosting Regressor: Este estimador construye un modelo aditivo por etapas y permite optimizar funciones de pérdida diferenciables arbitrarias. En cada etapa se ajusta un árbol de regresión sobre el gradiente negativo de la función de pérdida dada [6].
- Random Forest Regressor: Un bosque aleatorio es un meta estimador que ajusta una serie de árboles de decisión regresores a varias submuestras del conjunto de datos y utiliza el promedio para mejorar la precisión predictiva y controlar el sobreajuste [7].
- Decision Tree Regressor: Un árbol de decisión es un tipo de algoritmo de aprendizaje supervisado que se usa comúnmente en el aprendizaje automático para modelar y predecir resultados basados en datos de entrada. Es una estructura en forma de árbol donde cada nodo interno prueba el atributo, cada rama corresponde al valor del atributo y cada nodo hoja representa la decisión o predicción final. El algoritmo del árbol de decisión entra en la categoría de aprendizaje supervisado. Se pueden utilizar para resolver problemas tanto de regresión como de clasificación [8].
- Light Gradient Boosting Machine Regressor: Es un marco de código abierto ampliamente utilizado para el refuerzo por gradiente. Destacado por su capacidad para manejar conjuntos de datos de gran tamaño, ofrece un rendimiento superior en comparación con otros marcos de refuerzo por gradiente. Una de las características distintivas de

Light Gradient Boosting Machine (Light GBM) es su método de muestreo unilateral basado en el gradiente para dividir los árboles. Este enfoque reduce significativamente el uso de memoria y mejora la precisión del modelo. Además, Light GBM emplea un crecimiento por hojas en lugar de crecimiento por niveles, lo que lo hace considerablemente más rápido que los métodos tradicionales de crecimiento en profundidad [9].

Volviendo al entorno empresarial, vivimos en un momento de rápidos cambios en el área tecnológica, donde todas las empresas deben poder reaccionar rápidamente. Cualquier organización necesita adaptarse cada vez más rápido a un mundo en constante cambio. Además, cada día aumenta la complejidad de la cadena de suministro. Por lo tanto, no hay otra manera de que las empresas sobrevivan y se adapten rápidamente a esos cambios que automatizando procesos en su gestión de suministros. Si bien no se pueden resolver todos los desafíos simplemente añadiendo tecnología, una estrategia de cadena de suministro moderna requiere una pila de tecnología actualizada. Las cadenas de suministro, tradicionalmente lineales y predecibles, enfrentan una importante fragmentación de la demanda debido al crecimiento del comercio digital y nuevos modelos de cumplimiento. Esta evolución está haciendo que las cadenas actuales se vuelvan obsoletas. Aunque nadie puede prever con certeza el futuro de la cadena de suministro, es claro que la inteligencia artificial jugará un papel fundamental para impulsar su resiliencia. Se espera una mayor visibilidad de extremo a extremo con sistemas más interconectados y un mayor uso de la IA y el *ML* para la previsión de la demanda y prácticas más sostenibles, respondiendo a las demandas de los consumidores. En un futuro cercano, las cadenas de suministro podrían ser altamente autónomas, con sistemas impulsados por IA que gestionen la mayoría de los procesos, desde la adquisición hasta la entrega [10].

El *ML* ha demostrado tener un gran potencial en varios ámbitos, incluida la configuración de optimización del control de inventario. La optimización de inventario es fundamental para la gestión eficiente de los mayoristas, independientemente de su tamaño. Implica manejar una gran cantidad de productos de manera regular con el objetivo de reducir costos operativos y aumentar las ventas. Una parte esencial de este proceso es el control de inventario, que implica decidir cuándo y cuánto pedir de un artículo en particular para mantener un equilibrio óptimo entre oferta y demanda. Para ello, es esencial optimizar las políticas de inventario, ajustándolas según parámetros como el periodo de revisión, el tiempo de entrega y el nivel de servicio objetivo para cada artículo. El *ML*, al aprovechar algoritmos avanzados para analizar datos, puede manejar patrones de demanda complejos, identificar tendencias y ajustar pronósticos según múltiples variables y restricciones. Esto mejora la precisión de la previsión de la demanda y optimiza los niveles de inventario en consecuencia. Con su capacidad para aprender

y adaptarse a partir de datos, el *ML* es más dinámico y flexible que el software tradicional de gestión de inventario, especialmente en la previsión y optimización de la demanda. Esto permite a las empresas determinar puntos de reabastecimiento óptimos, niveles de existencias de seguridad y estrategias de asignación de inventario, lo que resulta en decisiones de gestión de inventario más precisas, una mejor gestión de la cadena de suministro, y una reducción del riesgo de desabastecimiento o exceso de inventario.

Implementar *ML* en la gestión de inventario ofrece una serie de beneficios significativos: Por un lado, la mejora de la precisión de pronósticos, ya que los algoritmos permiten determinar niveles óptimos de inventario considerando factores como tiempo de entrega, estacionalidad y limitaciones de costos. Identifican el equilibrio adecuado entre costos de mantenimiento y desabastecimientos, optimizando los niveles de inventario a través del análisis de datos históricos, ciclos de producción y pronósticos de ventas. Por otro lado, la reducción de costos y pérdidas, es decir, analizan datos históricos de ventas, tendencias del mercado y factores externos para pronosticar con precisión la demanda de los clientes. Permite generar pronósticos de demanda más precisos, optimizando los niveles de inventario, reduciendo desabastecimientos y evitando excesos de inventario. Por último, también optimiza el ciclo de vida del producto. Los algoritmos de *ML* pueden generar probabilidades asociadas con diferentes niveles de demanda, permitiendo una gestión más eficiente de los productos. Por ejemplo, en una tienda de comestibles, pueden analizar fechas de vencimiento, patrones de demanda y datos históricos de ventas para optimizar los niveles de stock y minimizar el desperdicio debido a la caducidad del producto [11].

Estas son varias consideraciones y pasos clave para implementar *ML* en la gestión de inventario.

- La evaluación de necesidades, es decir, reflexionar sobre objetivos claros y específicos, como evitar el exceso de existencias, prevenir desabastecimientos o mejorar la previsión de la demanda.
- La selección de modelos y algoritmos, seleccionando modelos como regresión lineal, árboles de decisión o redes neuronales según la complejidad del problema y los datos disponibles.
- Entrenar los modelos con datos históricos, evaluando su rendimiento con métricas adaptadas a los objetivos de gestión de inventario, y ajustar y optimizar el modelo para alcanzar un rendimiento óptimo.
- La integración con sistemas existentes, colaborando estrechamente entre analistas de datos, profesionales de IT y expertos en gestión de suministros para integrar el modelo

en el sistema de gestión de inventario.

- Alinear los resultados del modelo con los procesos existentes y garantizar una integración fluida en la toma de decisiones.
- Establecer un proceso de monitoreo y mantenimiento constante, dado que los modelos de ML son dinámicos y requieren actualizaciones periódicas.

Empresas líderes como Amazon, Walmart y Nike han implementado con éxito el *ML* en la gestión de inventario, obteniendo resultados significativos. En el caso de Amazon, por ejemplo, predice la demanda de productos considerando diversas variables como tendencias de búsqueda, datos históricos de ventas y condiciones climáticas. Esto le permite mantener niveles óptimos de inventario, evitando excesos o faltantes de stock. Para el caso de Walmart, emplea un sistema de gestión de inventario impulsado por IA para proporcionar a los clientes lo que necesitan, cuando lo necesitan y al costo esperado. Combina datos históricos con análisis predictivos para colocar estratégicamente los artículos en centros de distribución y tiendas, optimizando la experiencia de compra. Y por último, Nike, está construyendo una cadena de suministro digital a nivel mundial para atender directamente a los consumidores a escala [11].

1.3. Objetivos

El objetivo de este proyecto es desarrollar un cuadro de mando integral que permita visualizar la información clave en tiempo real para mejorar las prácticas de gestión de inventarios, control de compras y ventas, y seguimiento de clientes y proveedores potenciales en la empresa. Este *dashboard* centralizará los datos esenciales, facilitando la toma de decisiones informadas y estratégicas para optimizar las operaciones y aumentar la eficiencia de la gestión empresarial. Con esta herramienta, se busca asegurar una mejor planificación, reducir costos, mejorar el servicio al cliente y maximizar los ingresos. Las tareas del proyecto se dividen en dos objetivos principales:

- Realizar un preprocesamiento efectivo de los datos de inventario utilizando Python para garantizar la calidad y la integridad de los datos antes de su análisis.
 - Implementar scripts de Python que limpien, formateen y preparen los datos para su análisis.
 - Utilizar bibliotecas de Python como *Pandas* y *NumPy* para realizar el preprocesamiento de datos de manera eficiente.
 - Garantizar que los datos estén libres de errores y sean coherentes para un

análisis preciso.

- Crear un informe interactivo en Power BI que aborde los problemas de gestión identificados y proporcione una visión clara del estado de las ventas, compras e inventario en todo momento.
 - Desarrollar un informe en Power BI que incluya visualizaciones dinámicas, métricas clave y recomendaciones para mejorar la gestión de inventario.
 - Utilizar las capacidades de Power BI para crear visualizaciones efectivas y personalizadas que aborden los problemas específicos de gestión.
- Implementar un control de versiones utilizando GitHub para gestionar y rastrear cambios en el código y los recursos del proyecto de análisis de inventario.
 - Crear un repositorio en GitHub y realizar *commits* periódicos para registrar los cambios en el código y los archivos relacionados con el proyecto.

1.4. Descripción de la metodología empleada

En términos de herramientas de software, este proyecto sigue una metodología centrada en el uso de Python para el procesamiento de datos. Se emplea un entorno *Jupyter Notebook* (.ipynb) para llevar a cabo todas las fases del tratamiento de datos, incluyendo carga, preprocesamiento y modelado. Además, para la creación del cuadro de mando, se utiliza Power BI. Estas herramientas han sido seleccionadas debido a su idoneidad para un proyecto nuevo que se desarrolla completamente desde cero. En relación con las diferentes etapas que abarca este proyecto, se pueden distinguir las siguientes:

- Carga de datos: En la fase inicial del proyecto, los datos se recopilan inicialmente en archivos en formato CSV o XLSX. Estos archivos se transforman en *DataFrames* de Pandas utilizando Python dentro del entorno *Jupyter Notebook*.
- Preprocesamiento de los datos: Durante esta etapa inicial, se lleva a cabo el procesamiento de los datos con el objetivo de asegurar su calidad y prepararlos adecuadamente para las tareas de análisis. Esto incluye la limpieza de los datos, el tratamiento de valores nulos y la normalización de datos cuando es necesario.
- Análisis de los datos: En la fase de análisis de datos, además de comprender los datos disponibles, se han generado nuevas tablas que añaden valor al informe y se han creado columnas adicionales a partir de los datos existentes para enriquecer la información disponible.

- Creación del informe: Las bases de datos almacenadas en Excel han sido integradas en Power BI, donde se ha procedido a crear un informe dinámico y claro. Este proceso asegura la correcta visualización de los datos de manera accesible y comprensible para los usuarios.
- Control de versiones: Durante todo el proyecto se ha hecho uso de *Git* para mantener una correcta gestión de las versiones.
- Conclusiones: una vez completado todo el proceso se extraen las conclusiones definitivas.

1.5. Planificación del proyecto

Una vez conocido el proyecto y definidos los objetivos a realizar, se han diseñado los pasos y tiempos necesarios para llevarlo a cabo. Para ello se ha utilizado el diagrama de Gantt, que corresponde al cuadro de Henry Gantt; creado a principios del siglo XX con el objetivo de representar el desarrollo de las actividades de un proyecto. Permite identificar los recursos que se utilizarán en cada actividad y establecer su duración. De esta forma se podrán evitar espacios de tiempo inútiles [12]. Siendo un proyecto de 3 meses, es muy importante tenerlo bien planificado para que cada acción se lleve a cabo en la fecha prevista. Toda la planificación se puede ver en la *Figura 1*.

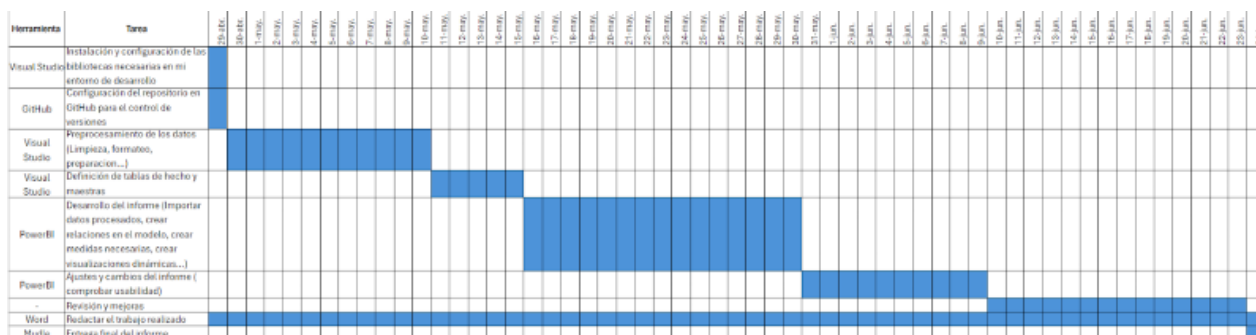


Figura 1: Planificación Gantt del proyecto

2.DESARROLLO

Para llevar a cabo el proyecto se ha cogido la información de una plataforma llamada *Kaggle* que proporciona un entorno colaborativo para el aprendizaje automático y la ciencia de datos. Esta página ofrece una variedad de recursos y herramientas en los que, entre otros, se encuentran bases de datos públicos [13].

En el desarrollo del proyecto se han dado una serie de pasos interrelacionados que van desde la entrada de datos hasta la visualización de los resultados. En el primer paso se han leído los datos, lo que implica la recopilación de información relevante. A continuación, se ha seguido con la preparación de datos, incluyendo la limpieza y consolidación de datos brutos para transformarlos en una forma adecuada para el análisis; verificando continuamente para garantizar la calidad y relevancia de los datos. Luego, en la exploración de datos se ha enriquecido la base de datos con información adicional, como por ejemplo columnas calculadas, para proporcionar una perspectiva más completa. Por último, se ha generado el cuadro de mando donde se organizan y presentan los resultados del análisis de manera comprensible y útil para compartir los conocimientos adquiridos.

2.1. Análisis de los datos

En esta fase se ha realizado lo que sería el proceso ETL, es decir, extracción de datos desde origen, carga de datos, transformación de datos y la nomenclatura de tablas y columnas. Todo este proceso se ha realizado en Python como se ha mencionado anteriormente, y el notebook completo se encuentra en el archivo llamado *procesamiento* en el siguiente enlace: [Procesamiento de los datos en Python](#)

2.1.1. Entrada de datos:

A la hora de leer los datos y procesarlos, se ha optado por emplear el entorno de desarrollo integrado *Visual Studio Code*, aprovechando su funcionalidad y facilidad de uso en la programación en Python, un lenguaje ampliamente utilizado en el ámbito del análisis y ciencia de datos. En la fase inicial del proceso, se han importado las bibliotecas y paquetes necesarios para la manipulación, análisis y visualización de datos. Entre estas herramientas esenciales, se incluyen las siguientes:

- Pandas, una biblioteca de manipulación y análisis de datos que proporciona estructuras de datos flexibles y eficientes, así como funciones para la lectura y escritura de datos en diversos formatos [14].

- NumPy, una biblioteca fundamental para la computación numérica en Python, que ofrece soporte para matrices y funciones matemáticas de alto nivel [15].
- Matplotlib, una biblioteca ampliamente utilizada para la visualización de datos en dos dimensiones, que permite la creación de gráficos estáticos, interactivos y animados [16].
- Seaborn, una biblioteca basada en Matplotlib que proporciona una interfaz de alto nivel para la creación de gráficos estadísticos atractivos e informativos [17].
- PyExcel, una biblioteca de Python diseñada para facilitar la lectura, escritura y manipulación de hojas de cálculo en diferentes formatos, como Excel (.xlsx, .xls) [18].

La elección de estas herramientas se sustenta en su capacidad para facilitar las tareas de lectura, procesamiento y análisis de datos, así como en su versatilidad y potencia para generar visualizaciones claras y efectivas que permitan comprender y comunicar adecuadamente los resultados obtenidos durante el análisis de datos. Dicho esto, se han cargado los seis archivos XLSX con los que se dispone para llevar adelante el trabajo. Estos archivos corresponden a operaciones comerciales del ejercicio cerrado en 2016. El conjunto de datos incluye las siguientes bases de datos: Inventario inicial del 2016, inventario final del 2016, facturas de compra correspondientes al 2016, precios de compra, datos de compras y datos de ventas. Una vez cargados los datos se han impreso las primeras 5 líneas de todos los datos para obtener una visión general de los datos.

2.1.2. Análisis exploratorio

Se ha realizado una función que devuelve una tabla, como la que se encuentra en la *Tabla 1*, donde se han recogido detalles clave sobre los archivos leídos. Entre ellos, el nombre de la columna, el tipo de dato, el número de filas, el número de valores únicos de cada columna, los valores no informados y el porcentaje de valores no informados frente al total. Así con la información clave de cada tabla se han sacado varias conclusiones para después proceder a la limpieza y preprocesamiento de los datos.

	Columna	TipoDato	Filas	Valores Unicos	Missings	Missings (%)
0	InventoryId	object	224489	224489	0	0.000
1	Store	int64	224489	80	0	0.000
2	City	object	224489	67	1284	0.572
3	Brand	int64	224489	9653	0	0.000
4	Description	object	224489	8732	0	0.000
5	Size	object	224489	47	0	0.000
6	onHand	int64	224489	548	0	0.000
7	Price	float64	224489	354	0	0.000
8	endDate	object	224489	1	0	0.000

Tabla 1: Análisis exploratorio de los datos

Por un lado, se ha observado que la columna *InventoryId* tiene los mismos registros únicos que los totales, lo que sugiere que podrían servir como claves primaria y foránea para unir las tablas. Por otro lado, las tablas contienen valores no informados que son necesarios identificar y tratar adecuadamente. Además, las columnas de tamaño muestran incoherencias en todas las tablas debido a la variación de unidades de medida. Por ello, es necesario convertirlas a una unidad única y coherente. Aparte de la columna de volumen, en alguna de las tablas también existe la columna *Size*, la cual contiene el mismo dato, por lo que se puede eliminar una de las dos columnas. En cuanto al formato de las fechas se ha observado que utilizan diferentes formatos en cada una de las tablas por lo que es conveniente unificar y ponerlos de la misma manera. Por último, en la columna *VendorName* se pueden ver espacios o signos tras la palabra que se pueden eliminar.

2.1.3. Limpieza y procesamiento

En el apartado anterior se han mencionado cuales son los puntos que se deben abordar para llevar a cabo la limpieza y el procesamiento de los datos, por lo que a continuación se han producido dichos cambios.

2.1.3.1. Tratamiento de *Missings*

La integridad y la calidad de los datos son aspectos fundamentales en cualquier análisis de datos. Los valores no informados, comúnmente conocidos como *missing values*, representan una preocupación significativa en el contexto del análisis de datos, ya que pueden distorsionar los resultados y conducir a interpretaciones erróneas si no se manejan adecuadamente. Por lo tanto, es crucial detectar y abordar de manera efectiva estos valores faltantes. Los valores no informados pueden surgir debido a una variedad de razones, que van desde errores en la entrada de datos hasta fallos en la recopilación o transferencia de información.

Independientemente de su origen, la presencia de valores no informados puede comprometer la validez y la fiabilidad de cualquier análisis realizado sobre los datos afectados. Durante el análisis del conjunto de, se han detectado valores no informados en cuatro tablas. A continuación, se describen las acciones tomadas para abordar estos valores faltantes:

- Tabla inventario final: En la columna *City* se ha identificado ausencia de valores. Tras un análisis detallado, se ha observado que todos los nombres de ciudades están presentes excepto para una ubicación de tienda específica. Es decir, todos los nombres faltantes en la columna ciudad pertenecen al número de tienda 46. Siendo esto así, se ha observado en la tabla inventario inicial donde se encuentra dicha tienda obteniendo el nombre TYWARDREATH como respuesta. Por lo que se ha decidido reemplazar los valores nulos por esa ciudad.
- Tabla compras: Se han encontrado solo tres valores faltantes en la columna *Size*, lo que representa un porcentaje mínimo del total de registros. Por lo tanto, se ha optado por eliminar estas filas para preservar la integridad de los datos restantes.
- Tabla ventas: En la columna *Approval* se muestra un alto porcentaje de valores faltantes, lo que representa el 93% del total de registros. Dada la magnitud de esta ausencia de información, se ha tomado la decisión de eliminar completamente esta columna para evitar cualquier sesgo o distorsión en el análisis posterior.
- Tabla precio compras: Se han identificado valores faltantes en las columnas *Description*, *Size* y *Volume*. Dado que estos valores no informados representan registros individuales, se ha decidido eliminar las filas correspondientes para mantener la coherencia y la integridad de los datos restantes.

Una vez tratados los *missings* de cada tabla se ha creado una función que devuelve la cantidad de valores no informados para cada tabla y se ha obtenido un valor de 0 para cada una de ellas.

2.1.3.2. Tratamiento de duplicados

Mediante un bucle y el método *duplicated* en Python, se ha realizado una verificación para determinar la presencia de duplicados dentro del conjunto de datos representado. Se ha guardado la información en una variable llamada *duplicados*, la cual captura el resultado booleano, proporcionando una indicación clara sobre la presencia o ausencia de duplicados en los datos. En este caso ninguna de las tablas cuenta con valores duplicados, por lo que no se ha aplicado ninguna modificación.

2.1.3.3. Irregularidades en los datos

Se han observado irregularidades significativas en las entradas de datos, especialmente en la columna *Size*, que exhibe variaciones en todas las tablas de datos. Estas discrepancias se refieren a la representación de información de volumen, expresada en diferentes unidades y formatos, como litros, mililitros, onzas, paquetes (*pk*) y combinaciones de estas unidades. Esto implica la necesidad de estandarizar estas medidas en una unidad de volumen única y coherente. Para abordar esta cuestión, se ha llevado a cabo un análisis exhaustivo de los recuentos únicos en la columna *Size*. Por un lado, aquellos elementos que exhiben un solo recuento y no presentan un patrón discernible han sido asignados a una tasa equivalente estándar. Por otro lado, aquellos elementos con múltiples recuentos y un patrón discernible han sido transformados de acuerdo con estos patrones identificados. Este enfoque metodológico busca garantizar la consistencia y la comparabilidad de los datos de volumen a lo largo de las diferentes tablas, lo que es esencial para facilitar un análisis coherente. La estandarización de las unidades de volumen contribuye a mitigar posibles sesgos o distorsiones en los resultados del análisis, permitiendo así una interpretación precisa y fiable de los datos. Además, para garantizar la coherencia y la fiabilidad en el manejo de datos temporales, se ha creado una función que se encarga de estandarizar las fechas en varios *DataFrames*, convirtiéndolas al formato día/mes/año. Asimismo, se ha creado otra función para eliminar los espacios finales en las columnas especificadas.

Por último, se ha identificado una discrepancia en los nombres de las ciudades presentes en las tablas de inventario inicial e inventario final, donde los nombres no corresponden a ubicaciones geográficas reales. Para abordar esta irregularidad, se ha realizado un proceso de corrección mediante el reemplazo de los nombres de las ciudades no reales por aquellos que sí lo son. Para llevar a cabo esta corrección, se han extraído todos los nombres de ciudades presentes en las tablas y se han reemplazado con una lista de ciudades geográficamente válidas. Para complementar la información, se ha añadido una nueva columna llamada estado que indica el estado de cada ciudad. Esta información adicional se ha considerado apropiada para utilizarla en el informe añadiendo una visualización de un mapa geográfico. Además, se ha observado que el identificador de inventario (*InventoryId*) estaba influenciado por el nombre de la ciudad. Por lo tanto, se ha procedido a recrear este identificador combinando el número de tienda, el nombre de la ciudad y el número de marca, separados por un guion. Así, se ha garantizado que los identificadores de inventario reflejan de manera precisa la relación entre la tienda, la ciudad y la marca, proporcionando así una representación más fiel de la realidad en el conjunto de datos. Para terminar, se han impreso otra vez las primeras cinco líneas de todas las tablas y se ha asegurado que se ha completado la limpieza y procesamiento de los datos, para así empezar con la siguiente fase.

2.2. Modelado de datos

Una vez limpiados y procesados los datos, se ha comenzado con el modelado de datos. Esta es una parte fundamental en un proyecto BI cuyo propósito es organizar y estructurar los datos de manera que permite realizar el análisis necesario al cliente. En este proyecto se quieren mostrar las ventas, compras y stock de la empresa, por lo que para realizar el modelado de datos es importante identificar todas las métricas a analizar, para posteriormente proceder a definir el modelo de datos de la forma óptima.

2.2.1. Modelo semántico

Este método organiza los datos de manera lógica, específica y fácilmente comprensible. Puede entenderse como una forma de dar orden y contexto a los datos. El modelo semántico organiza los elementos de datos describiendo las relaciones entre ellos. Un buen modelo semántico ofrece las siguientes ventajas: una exploración de datos más rápida, agregaciones más fáciles de crear, informes más precisos, dedicar menos tiempo a crear el informe y mejor seguimiento a futuro [19]. A la hora de crear el modelo, se ha optado por el modelo estrella, en el que los datos se organizan en tablas de hechos, que constituyen con elementos medibles y en tablas de dimensiones, donde se recoge la información de referencia. Cada hecho está rodeado por sus dimensiones asociadas en un patrón similar a una estrella como se puede ver en la *Figura 2*. También existe el esquema de copo de nieve, que se parece al esquema de estrella, pero incluye capas adicionales de dimensiones asociadas, lo que hace que la ramificación del patrón sea más compleja.

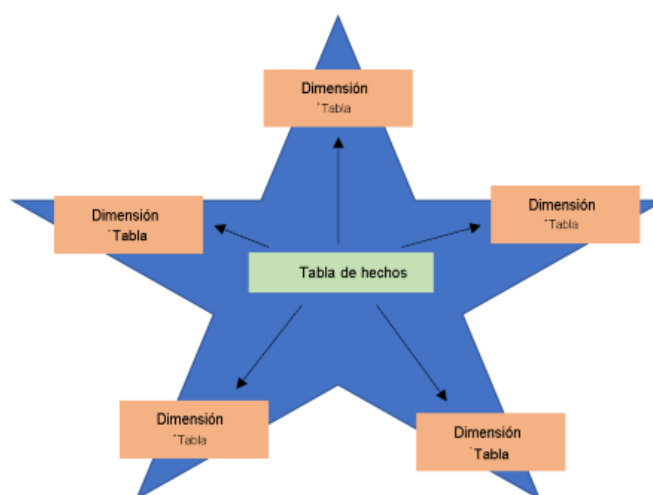


Figura 2: Modelo estrella; Fuente: [19]

Como se ha comentado anteriormente, el esquema estrella se fundamenta en una tabla de hechos que contiene uno o más datos que en lo posible deben ser medibles y una o más tablas de dimensiones. Los hechos, con frecuencia datos agregados, pueden identificarse como medidas tomadas de la intersección de todas las dimensiones. Estas contienen valores de datos de eventos o de observación y pueden contener varios valores repetidos. Los atributos en las dimensiones son, por lo general, textuales y discretos, con el propósito de establecer restricciones específicas en las consultas. Estas suelen contener detalles sobre los datos de las tablas de hechos como por ejemplo productos, ubicaciones, clientes y proveedores. Estas tablas están conectadas a la tabla de hechos a través de columnas de clave. Las tablas de dimensiones se usan para filtrar y agrupar los datos de las tablas de hechos [20].

La construcción de una base de datos bajo este esquema requiere de una rígida definición de hechos y dimensiones que anticipen la consulta de datos bajo patrones preestablecidos [21].

2.2.2. Relaciones entre tablas de hechos y dimensiones

Al tener varias tablas, las relaciones entre ellas son necesarias para calcular los resultados de forma precisa y mostrar la información correcta en los informes. En caso de que se hayan nombrado correctamente las columnas, es decir, poniendo el mismo nombre a aquellas que se quieren relacionar, la característica de detección automática es capaz de crear las relaciones automáticamente. Sin embargo, es necesario revisar todas las relaciones y realizar cambios en los casos que sea necesario. Además, al crear o editar una relación, se puede configurar más de una opción. La opción cardinalidad puede tener una de las siguientes configuraciones:

- Varios a uno (*:1): Indica que una columna de una tabla puede contener múltiples instancias de un valor, mientras que la tabla relacionada, frecuentemente denominada tabla de búsqueda, tiene solo una instancia de ese valor.
- Uno a uno (1:1): La columna de una tabla contiene una única instancia de un valor específico, y la tabla relacionada también contiene una única instancia de ese mismo valor.
- Uno a varios (1: *): En una relación de uno a varios, la columna de una tabla contiene una sola instancia de un valor, y la tabla relacionada puede contener múltiples instancias de ese mismo valor.
- Varios a varios (:): Los modelos compuestos permiten establecer relaciones de varios a varios entre tablas, eliminando la necesidad de valores únicos en las tablas. Esto también facilita evitar soluciones alternativas previas, como la creación de nuevas tablas únicamente para establecer relaciones.

La opción dirección de filtro cruzado también es muy importante a la hora de crear las relaciones que puede tener unas de las siguientes opciones

- Ambos: Esta configuración indica que, para efectos de filtrado, ambas tablas se tratan como si fueran una sola tabla. Se utiliza cuando una única tabla rodeada de muchas tablas de búsqueda.
- Único: Esta es la dirección predeterminada más común, donde las opciones de filtrado en las tablas conectadas afectan solo a la tabla en la que se agregan los valores [20].

Por último, recalcar la importancia de las claves primarias y foráneas en el contexto de Power BI para definir y gestionar las relaciones entre las tablas de un modelo de datos. Una clave primaria es una columna o un conjunto de columnas en una tabla cuyos valores identifican de forma exclusiva una fila de la tabla. Una base de datos relacional está diseñada para imponer la exclusividad de las claves primarias permitiendo que haya sólo una fila con un valor de clave primaria específico en una tabla. En este caso las claves primarias se han nombrado indicando un *id* por delante. En cuanto a la clave foránea, es una columna o un conjunto de columnas en una tabla cuyos valores corresponden a los valores de la clave primaria de otra tabla. Para poder añadir una fila con un valor de clave foránea específico, debe existir una fila en la tabla relacionada con el mismo valor de clave primaria. Es decir, la llave primaria de la tabla de hechos es una concatenación de las llaves primarias de cada una de las dimensiones [22].

Por lo mencionado anteriormente, a la hora de crear el modelo y crear las tablas de dimensiones, se ha tenido en consideración qué columna es la primaria, es decir, la que indica valores únicos. En este caso se ha visto que ni el producto, ni la marca eran únicos, y que la combinación de los dos es la que crea productos únicos, ya que hay más de una marca para algunos productos. Por ello se ha creado una nueva columna uniendo la marca y el producto que valdrá como clave primaria. Lo mismo ha ocurrido con las tiendas y las ciudades, ya que en ciertas ciudades hay más de una tienda, por lo que se ha creado una nueva columna con la unión de las dos para crear la tabla de dimensiones y que sea única. A la hora de crear tanto las tablas de dimensiones como las tablas de hechos se ha utilizado la Python. Para crearlas, primero, se han seleccionado las columnas relevantes de las tablas que se tienen y se han almacenado en un nuevo *DataFrame*. Luego, para los casos que se ha visto que aporta valor al informe, se han añadido nuevas columnas calculadas. Por ejemplo, en la maestra de productos se ha creado una columna llamada categoría donde se agrupan los productos por diferentes rangos de precios. A continuación, se han renombrado las columnas de la tabla ya que estas están en inglés y se les ha dado un nombre claro para todos los casos. En este paso se ha tenido en cuenta que se mantenga la misma nomenclatura en todos los nombres de las

columnas. También se han reordenado las columnas para mejorar la legibilidad y se les ha asignado un formato específico en el caso que fuera necesario. El resultado final es un *DataFrame* que comienza por D, en caso de que la tabla sea de dimensión o por H, si la tabla es de hechos, estructurado y categorizado, listo para ser utilizado en análisis posteriores. Los *DataFrames* mencionados se han exportado al formato *XLSX* para luego insertarlos en el Power BI. En el Anexo A se muestran todas las tablas creadas y la descripción de las columnas de cada tabla para tener una visión general de las tablas ingestadas en Power BI.

El anterior paso ha sido el último que se ha realizado en Python, por lo que a continuación se ha comenzado con el uso de Power BI. Entre varias herramientas existentes para la visualización de los datos se ha seleccionado Microsoft Power BI, una colección de servicios de software, aplicaciones y conectores que operan en conjunto para transformar orígenes de datos no relacionados en información coherente, interactiva y visualmente atractiva. Ya sea un simple libro de Microsoft Excel o una colección de almacenes de datos híbridos locales o basados en la nube, permite conectar fácilmente con los orígenes de datos, limpiar y modelar los datos sin afectar al origen subyacente, visualizar lo más relevante y compartirlo con los destinatarios deseados [23].

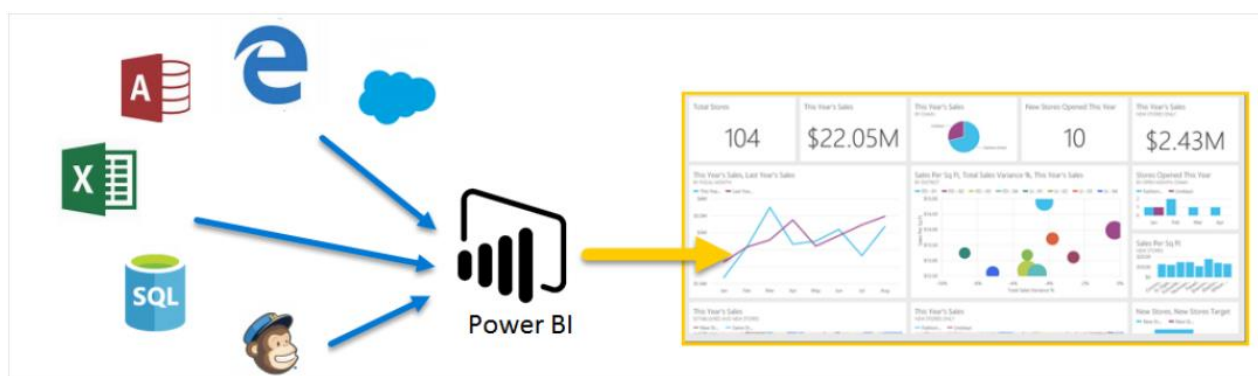


Figura 3: Descripción del uso de Power BI para crear análisis controlados por datos; Fuente:[23]

Al conectar una hoja de cálculo de Excel, es importante que los datos estén en una tabla plana y que cada columna contenga el tipo de datos correcto, como texto, fecha, número o moneda. Además, debe haber una fila de encabezado y no deben existir columnas o filas que muestren totales, ya que las operaciones totales se gestionan en Power BI durante la creación de los objetos visuales [23]. A la hora de realizar la carga de datos al servicio Power BI, este permite crear informes que se conectan a archivos CSV ubicados en el equipo. Por lo tanto, es así como se ha conectado el conjunto de datos que se ha obtenido tras realizar el procesamiento de ellos en Python. Una vez conectadas todas las tablas, se ha procedido a crear el modelo semántico. Para ello, se ha tomado en cuenta lo mencionado anteriormente y en la *Figura 4* se observa cómo se ha creado el modelo y las relaciones. En todos los casos excepto en uno

la cardinalidad se ha definido uno a varios con la dirección de filtro única.

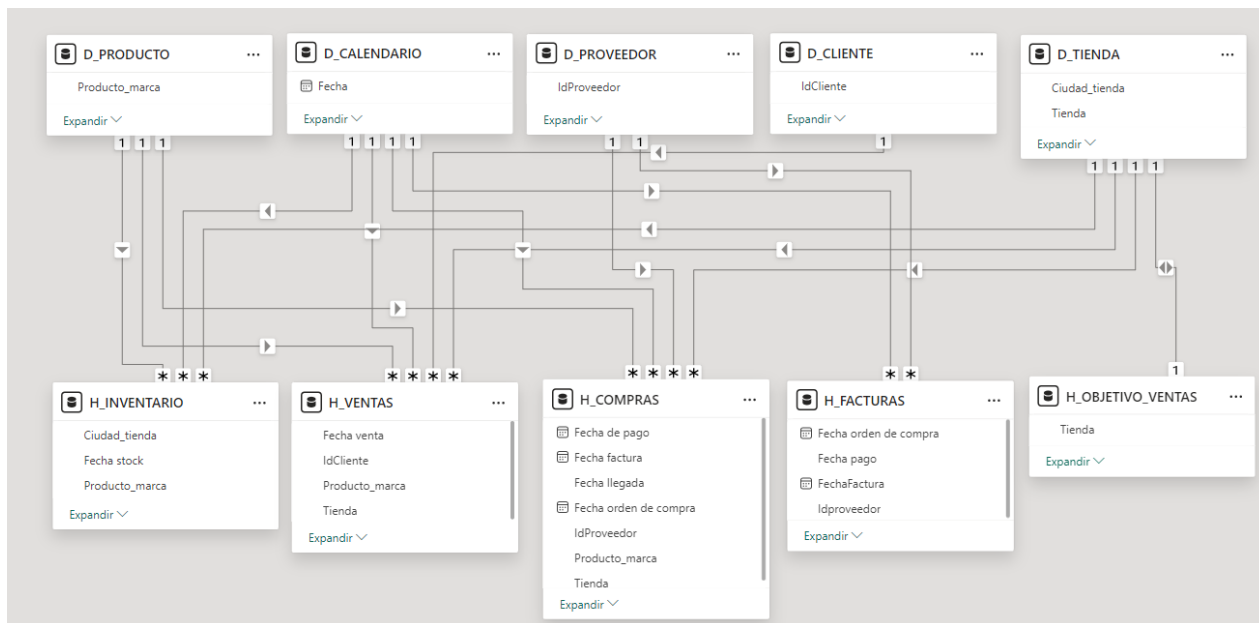


Figura 4:Diseño del modelo

2.2.3. Definición de medidas y KPIs

Para la realización de medidas y *Key Performance Indicator (KPI)* se ha utilizado expresiones de análisis de datos, llamados *Data Analysis Expression (DAX)*, un lenguaje de expresiones de fórmulas que se usa en *Analysis Services*, *Power BI* y *Power Pivot* en Excel. Las fórmulas *DAX* abarcan funciones, operadores y valores para realizar cálculos avanzados y consultas en los datos de las tablas y columnas relacionadas de los modelos de datos tabulares. Las fórmulas *DAX* se usan en medidas, columnas calculadas, tablas calculadas y seguridad de nivel de fila.

2.2.3.1. Medidas

Las medidas son fórmulas de cálculo dinámico en las que los resultados cambian en función del contexto. Las medidas se usan en informes en los que se pueden combinar y filtrar datos del modelo mediante varios atributos, como un informe de *Power BI*. Las medidas se crean con la barra de fórmulas *DAX* del diseñador de modelos. Una fórmula en una medida puede usar las funciones de agregación estándar creadas automáticamente con la característica de autosuma (como *COUNT* o *SUM*), aunque también se puede definir una fórmula propia con la barra de fórmulas *DAX*. Al definir una fórmula para una medida en la barra de fórmulas, una característica de información sobre herramientas muestra una vista previa de cuáles serían los resultados para total en el contexto actual, pero de lo contrario no se generan los resultados inmediatamente en ninguna parte. La razón por la que no se pueden ver los resultados del cálculo inmediatamente es que el resultado de una medida no se puede determinar sin el contexto. Se ejecuta una consulta distinta por cada celda de los resultados. Es decir, cada

combinación de encabezados de fila y de columna de una tabla dinámica, o cada selección de segmentación de datos y filtros de un informe de Power BI, genera un subconjunto de datos diferente sobre el que se calcula la medida [24]. En este informe se han creado varias medidas, las cuales se han organizado por carpetas para mantener un orden claro de cada una de ellas. En el Anexo B se encuentran las funciones que se han creado.

2.2.3.2. Tablas calculadas

Una tabla calculada es un objeto calculado, basado en una expresión de fórmula, que se deriva de todas las tablas (o parte de ellas) del mismo modelo. En lugar de consultar y cargar valores en las columnas de la nueva tabla desde un origen de datos, una fórmula *DAX* define los valores de la tabla. Las tablas calculadas admiten relaciones con otras tablas, tienen tipos de datos y formato, y pueden pertenecer a una categoría de datos [24].

La tabla D_CALENDARIO donde se registran las fechas para después relacionarla con las tablas de hecho, se ha creado utilizando *DAX*, mediante la función *CALENDAR*. Esta función devuelve un intervalo de fechas contiguo en función de fechas de inicio y de finalización que se especifican como argumentos en la función. La fecha de inicio se elige como la más temprana del modelo semántico y la fecha de finalización es la última [25]. En este caso se han cogido los datos desde 2015 a 2017 ya que el análisis se basa en los datos de 2016. Una vez creada la tabla se cuenta con una columna de fechas que se puede usar. A través de la columna fecha, utilizando las funciones de *DAX*, se han obtenido otras columnas como la del año, el número del mes, la semana del año y el día de la semana.

2.3. Visualización

Una vez conectados los datos, diseñado el modelo y creado las medidas necesarias, se ha comenzado a crear el informe. En el lado derecho se encuentran los paneles de visualizaciones, filtros y campos. Los datos de la tabla del libro de Excel aparecen en el panel campos, mostrando el nombre de la tabla, en la parte superior, y los encabezados de columna como campos individuales debajo.

2.3.1. Estructura del informe

Estructuralmente, el informe de Power BI se conecta a un único modelo semántico y tiene varias páginas de informe. En cada página, se han diseñado objetos de informe, entre los que se incluyen los siguientes:

- Objetos visuales, que son visualizaciones de datos del modelo semántico
- Elementos, que proporcionan interés visual, pero no usan datos del modelo semántico.

Entre ellos se incluyen cuadros de texto, botones, formas e imágenes.

2.3.2. Especificación del diseño del informe analítico

Tener datos correctos y seleccionar los objetos visuales adecuados es crucial, pero también es importante que el informe sea visualmente atractivo. Un buen diseño debe guiar al consumidor para encontrar y comprender rápidamente las respuestas a sus preguntas. Al observar un informe, los consumidores pasan por un proceso automático e inconsciente para comprender lo que ven. Los informes y cuadros de mando eficaces deben permitir a los usuarios responder rápidamente a sus preguntas sobre datos para que puedan centrarse en sus principales tareas y responsabilidades empresariales. Para ello se recomienda seguir la regla 3-30-300, un enfoque directo y práctico para producir diseños de informes eficientes estructurando los informes de una manera funcionalmente jerárquica. Esta regla se define por el tiempo que deberían tardar los usuarios en obtener cierta información o realizar ciertas tareas en un informe. Es decir, en 3 segundos, los usuarios deben obtener una visión general de las preguntas y áreas más importantes. En 30 segundos, los usuarios deben filtrar y hacer *zoom* para identificar los periodos y categorías en los que centrarse. Y en 300 segundos, los usuarios deben poder obtener detalles a la carta para fundamentar sus decisiones y acciones. Los segundos de esta regla pretenden ilustrar los conceptos de forma concreta; son objetivos aproximados y no literales [26].

Por ello, es fundamental cumplir con los principios básicos de diseño de informes para comunicar eficazmente el significado de los datos, combinando ciencia y arte. El diseño del informe comienza determinando el número, secuencia y propósito de las páginas, evitando combinar temas u objetivos opuestos en la misma página. Luego, se especifica cada diseño de página con los objetos de informe pertinentes [27].

2.3.2.1. Espacio, márgenes y equilibrio

El espacio es esencial para un diseño eficaz, ya que ayuda a reducir el desorden y a aumentar la legibilidad. Este se aplica a los márgenes de la página del informe, es decir, el área de borde que rodea cada página, y entre los objetos del informe cuando consta de varios grupos de objetos relacionados. Los objetos de informe deben estar enmarcados por un área de borde con un espaciado coherente y los márgenes deben ser iguales a la izquierda y a la derecha, pero pueden variar en la parte superior e inferior. El equilibrio, también, en la disposición de objetos es fundamental para la estabilidad y la estructura del diseño de un informe, refiriéndose al peso distribuido en la página mediante la colocación de objetos de tamaños iguales o diferentes. En este caso se ha jugado con un equilibrio asimétrico.

2.3.2.2. Alineación y tamaño del objeto visual

Cuando hay varios objetos visuales en la página del informe, deben estar correctamente alineados, es decir, los bordes y los espaciados de los objetos visuales deben ser coherentes. Además, generalmente, se debe ubicar la información más importante en la esquina superior izquierda de la página y organizar los elementos de izquierda a derecha y de arriba abajo. Con el objetivo de crear una conexión visual y evitar el desorden, los objetos relacionados deben agruparse lógicamente, ya que la alineación visualmente atractiva de los objetos puede transmitir más energía e interés que simplemente centrar o colocar aleatoriamente los objetos de informe. Por último, el tamaño de un objeto visual indica su importancia ya que los consumidores de informes suelen enfocarse primero en los objetos más grandes.

2.3.2.3. Color

El color debe emplearse con moderación y propósito, evitando su uso excesivo para no distraer al lector. En el informe, se ha utilizado una gama de colores azules que se alinean con la paleta corporativa, asegurando que los datos sean el foco principal del informe. Los colores llamativos se han reservado para resaltar excepciones y el contraste se ha usado para combinar dos objetos opuestos.

2.3.2.4. Coherencia en el diseño de informes

Es fundamental mantener la coherencia al diseñar y configurar objetos de informe. Esta coherencia debe reflejarse en todos los aspectos del diseño del informe, incluyendo el espaciado, los márgenes, el tamaño, la alineación y, especialmente, las opciones de formato del objeto. Un buen uso de la repetición también crea asociación y coherencia y ayuda a reforzar el diseño de un informe asociando objetos de informe relacionados. En el informe se ha utilizado la repetición para analizar las compras y las ventas.

2.3.2.5. Selección de objetos visuales en informes

El objetivo principal de la visualización de datos es comunicar información de manera clara y efectiva a los consumidores del informe. Por ello, es crucial seleccionar el tipo de objeto visual más adecuado para cumplir con los requisitos específicos. Se debe seleccionar un objeto visual que sea estéticamente atractivo y, al mismo tiempo, maximice la utilización del espacio disponible en la página.

Los gráficos de barras o columnas se han utilizado para datos que contienen varias categorías, ya que estos son efectivos para visualizar proporciones en múltiples dimensiones, como la categoría de producto o el análisis por estados. Para visualizar la evolución tanto de las ventas

como de las compras a lo largo del tiempo, se ha utilizado el gráfico de líneas, presentando la fecha en el eje X ordenado de los períodos más antiguos a los más recientes. Los objetos visuales proporcionales representan los datos como partes de un todo, comunicando eficazmente la distribución de un valor en una dimensión. Este tipo de gráfico se ha utilizado para visualizar las unidades vendidas por categoría. Los valores numéricos, han sido representados mediante objetos visuales tipo tarjeta, ya que destacan información crucial de manera rápida y directa. Las tablas y matrices pueden comunicar eficazmente una gran cantidad de información detallada. Las tablas tienen un número fijo de columnas que pueden mostrar datos agrupados o resumidos. Por otro lado, las matrices permiten organizar datos en grupos tanto en columnas como en filas. Estos se han utilizado para mostrar mayor detalle de la información. Además, las matrices ofrecen una excelente experiencia para la navegación jerárquica, permitiendo a los usuarios explorar profundamente en columnas y filas para descubrir datos detallados de interés. Las opciones de formato disponibles para tablas y matrices ofrecen un alto grado de control sobre el estilo y la presentación de los datos de cuadrícula. Asimismo, la aplicación de opciones de formato condicional, como colores de fondo, colores de fuente o iconos, ha realzado visualmente los valores con indicadores visuales, facilitando así la interpretación de informes y equilibrando la presentación en la página. Se ha aplicado una regla en la página de tiendas donde si las ventas son mayores que el objetivo el fondo se muestra en verde, y del contrario, el fondo se muestra en rojo. Por último, como el modelo semántico contiene información geoespacial, se ha visualizado mediante objetos visuales de mapa. Power BI ofrece varios objetos visuales principales diseñados para mapas, cada uno con diversas opciones de formato que, aplicadas correctamente, pueden resaltar datos geoespaciales de manera efectiva [28].

2.3.2.6. Formato y configuración de visualizaciones

Power BI proporciona diversas opciones para personalizar el aspecto de las visualizaciones seleccionadas, como los colores y el formato del texto. Las opciones de formato disponibles varían según el tipo de visualización seleccionada. Entre las más comunes se encuentran el título, fondo y borde.

- En la sección título, se agrega o edita el título de la visualización para describir claramente los datos presentados. Se puede ajustar el tamaño del texto, la fuente, el color y la alineación del título.
- En la sección fondo, se establece un color de fondo para la visualización. Es recomendable utilizar un fondo claro para asegurar que los datos sean legibles.
- La sección borde permite definir un borde alrededor del objeto visual para destacarlo

en el lienzo y mejorar su claridad visual, en el informe se ha optado por poner el borde redondeado.

En el panel formato, también se ajustan los colores y las etiquetas de valores de datos específicos. En la sección colores de datos, se configuran los colores que representan diferentes campos de datos dentro de la visualización. Se ha mantenido consistencia en la elección de colores para la coherencia del informe. También existe la opción de información sobre herramientas que permite agregar detalles adicionales que aparecen al mantener el ratón sobre la visualización [28].

2.3.2.7. Trabajo con marcadores

Los marcadores ofrecen una forma efectiva de transformar un informe en una experiencia analítica guiada, maximizando el espacio de página y facilitando interacciones intuitivas. Pueden variar desde acciones simples, como restablecer filtros, hasta comportamientos complejos que alteran objetos visuales o exploran en profundidad. En el informe se ha hecho uso de marcadores para cambiar la visualización de gráfica a tabla. Para ello, se han añadido botones y se han configurado sus acciones. A continuación, se ha explicado brevemente como se han configurado los marcadores definiendo los estados que capturan y el ámbito de los objetos visuales afectados. Por defecto, un nuevo marcador captura todos los tipos de estado, pero se puede optar por deshabilitar cualquiera de ellos según sea necesario [28].

- Estado Datos: Captura las configuraciones de consulta que afectan al modelo semántico, como segmentaciones y criterios de ordenación. También incluye la profundidad de los detalles de los objetos visuales afectados por la consulta.
- Estado Mostrar: Se refiere a la visibilidad de los objetos en el informe, incluyendo objetos visuales, cuadros de texto, botones, formas e imágenes. Permite ocultar o mostrar objetos según el marcador aplicado.
- Estado Página actual: Decide si el marcador dirige al usuario a la página marcada o si aplica la página actual.

El ámbito del marcador determina a qué objetos visuales se aplica:

- Todos los objetos visuales: Aplica el marcador a todos los objetos del informe, incluso aquellos que están ocultos.
- Objetos visuales seleccionados: Limita el marcador a los objetos visuales específicamente seleccionados al momento de configurar el marcador.

2.4. Aplicación de Modelos para la predicción de la demanda

Después de revisar varios modelos de *ML* en el marco teórico, se ha procedido a implementar un modelo simple para la licorería siguiendo un enfoque práctico. Para ilustrar este proceso, se han seleccionado modelos previamente mencionados para predecir la demanda utilizando datos históricos disponibles. El código completo se encuentra en el siguiente repositorio [*Modelo de predicción*](#), en el archivo llamado modelo de predicción.

Antes de crear el modelo de predicción, se han visualizado los datos de las ventas y compras. Para la visualización de las compras, se han agrupado los datos a nivel diario y se ha calculado la suma de las compras diarias, proporcionando así una nueva serie temporal que resume las compras día a día de los meses de enero y febrero. En la *Figura 5* se puede observar como las compras siguen un patrón semanal, donde, por un lado, existe un día sin compras, que podría ser el domingo, ya que es el día de descanso. Por otro lado, hay un día a la semana donde aumentan las compras significativamente, esto refleja la práctica de la empresa de recibir grandes volúmenes de productos para optimizar costos logísticos y aprovechar descuentos por volumen.

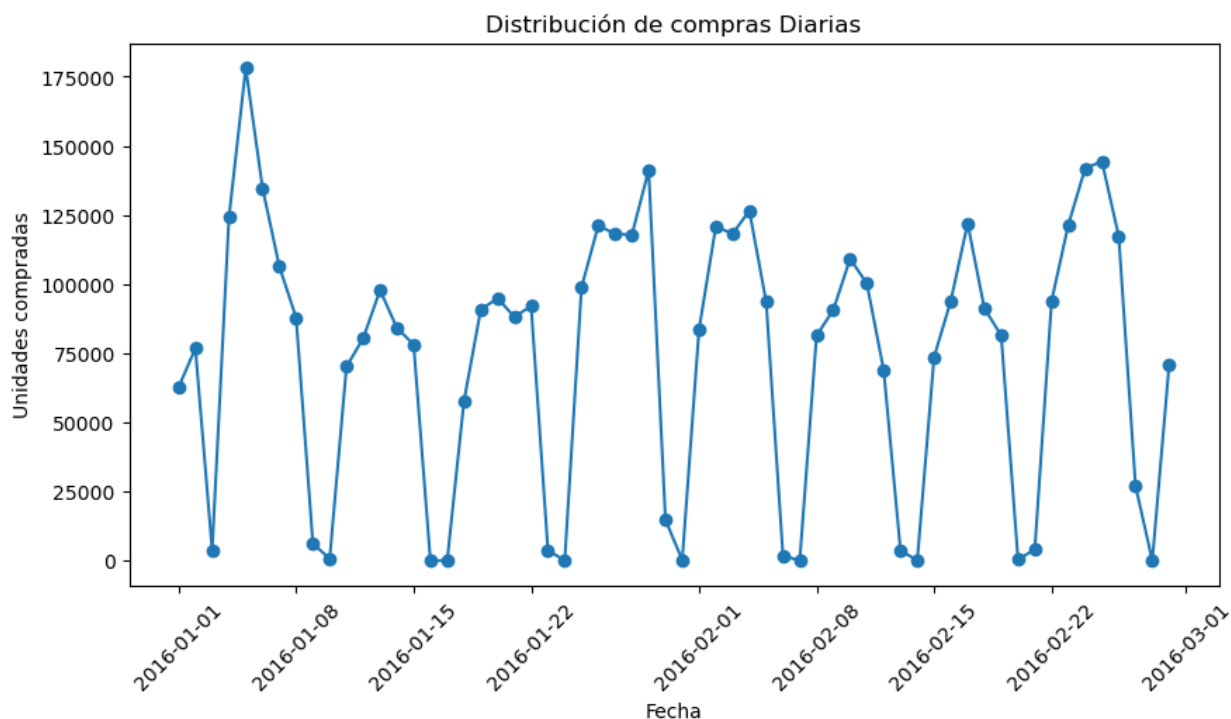


Figura 5: Distribución de compras diarias

Se ha realizado el mismo procedimiento para analizar las ventas, es decir, se han agrupado los datos a nivel diario y se ha calculado la suma de las ventas diarias como se observa en la *Figura 6*. En este caso, se ha observado un salto llamativo entre el último día de enero y el

primero de febrero, concluyendo que hay algún error en la recogida de datos; por lo que se ha decidido hacer un ajuste en los datos de febrero. Para ello, se ha comparado el valor de un pico de enero con un pico de febrero y se han multiplicado todos los datos de febrero por esa proporción.

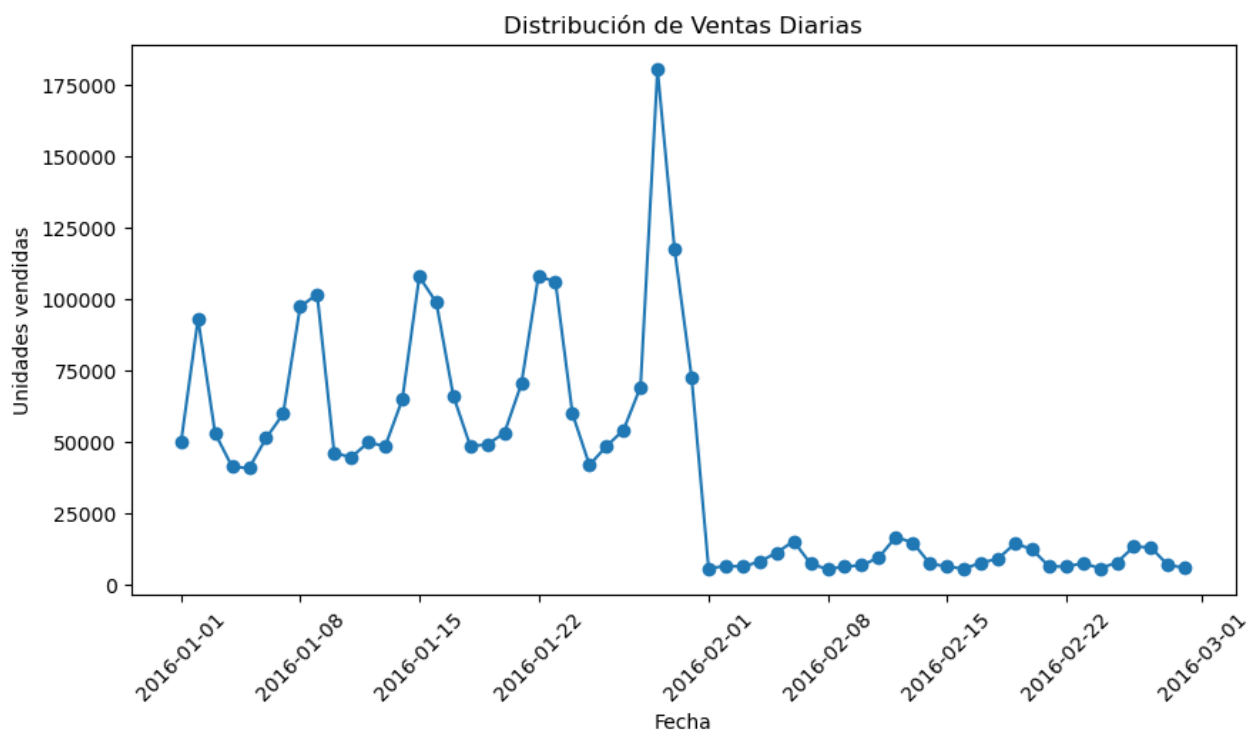


Figura 6: Distribución de ventas diarias

Una vez realizado el ajuste que se ha mencionado anteriormente, se ha obtenido la distribución que se muestra en la *Figura 7* pudiendo concluir que las ventas también siguen un patrón semanal, donde un día a la semana aumentan considerablemente las ventas y otro día o dos se reducen.

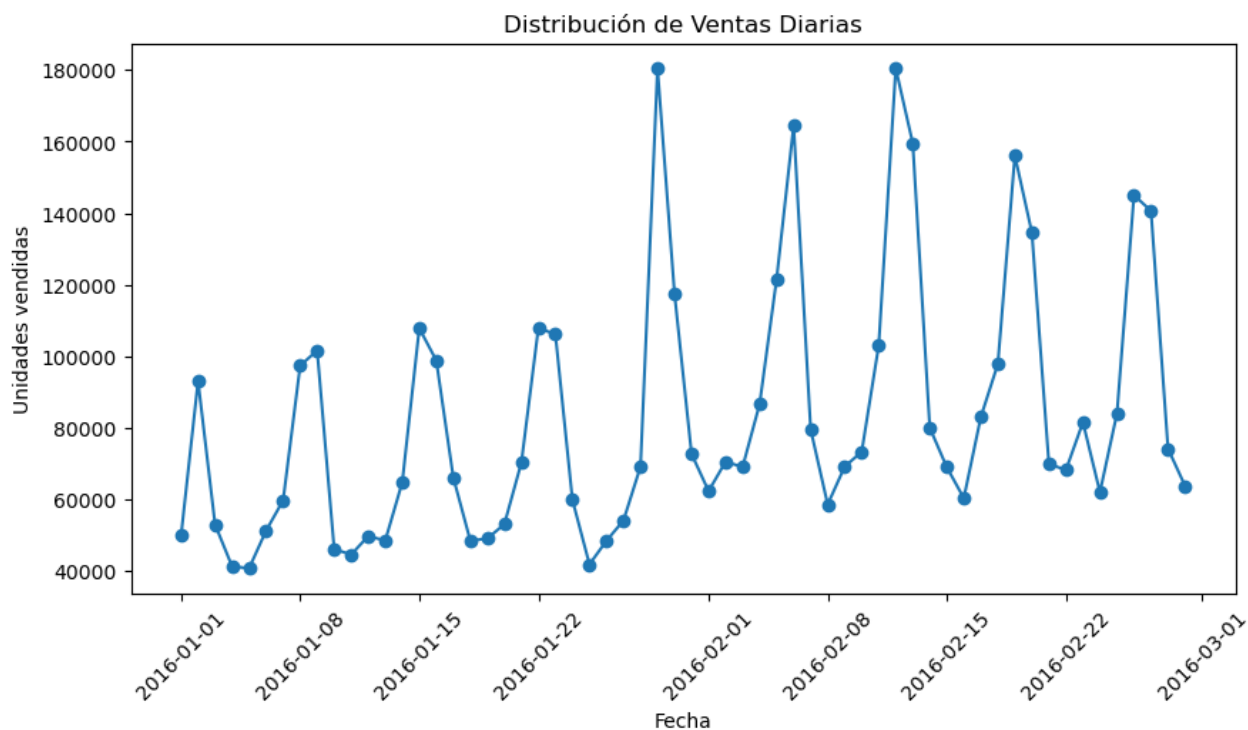


Figura 7: Distribución de ventas diarias ajustadas

A continuación, se ha realizado la consolidación de datos, donde se han combinado los datos de ventas y compras en un único *dataset* para facilitar el análisis. En ambos casos, se han cogido los datos desde el 01-01-2016 hasta el 29-02-2016 y se han visualizado en la *Figura 8*. El gráfico revela un patrón donde los días con altas compras tienen ventas relativamente bajas y viceversa. Esta dinámica es una consecuencia de la estrategia de gestión de almacén, para que exista una organización entre los camiones y además exista un equilibrio óptimo entre la oferta y la demanda.

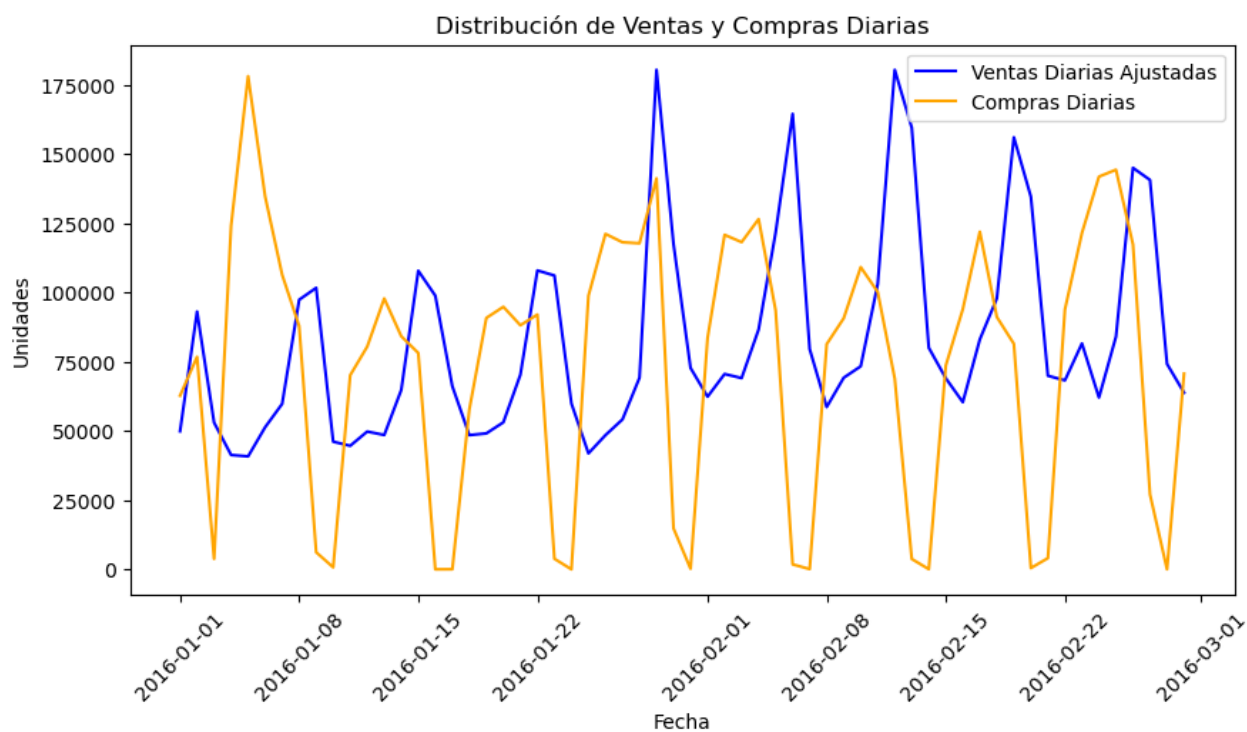


Figura 8: Serie temporal de ventas y compras

Una vez analizados los datos y solucionado las incongruencias, se ha seleccionado el modelo de predicción, comenzando con un enfoque sencillo: la regresión lineal. Seguido se ha entrenado el modelo utilizando los datos disponibles y se ha validado su desempeño. En esta fase, primero se han dividido los datos en conjuntos de entrenamiento y validación, luego se ha entrenado el modelo seleccionado con los datos de entrenamientos, y se ha realizado la predicción de las ventas de los próximos 14 días como se ve en la *Figura 9*.

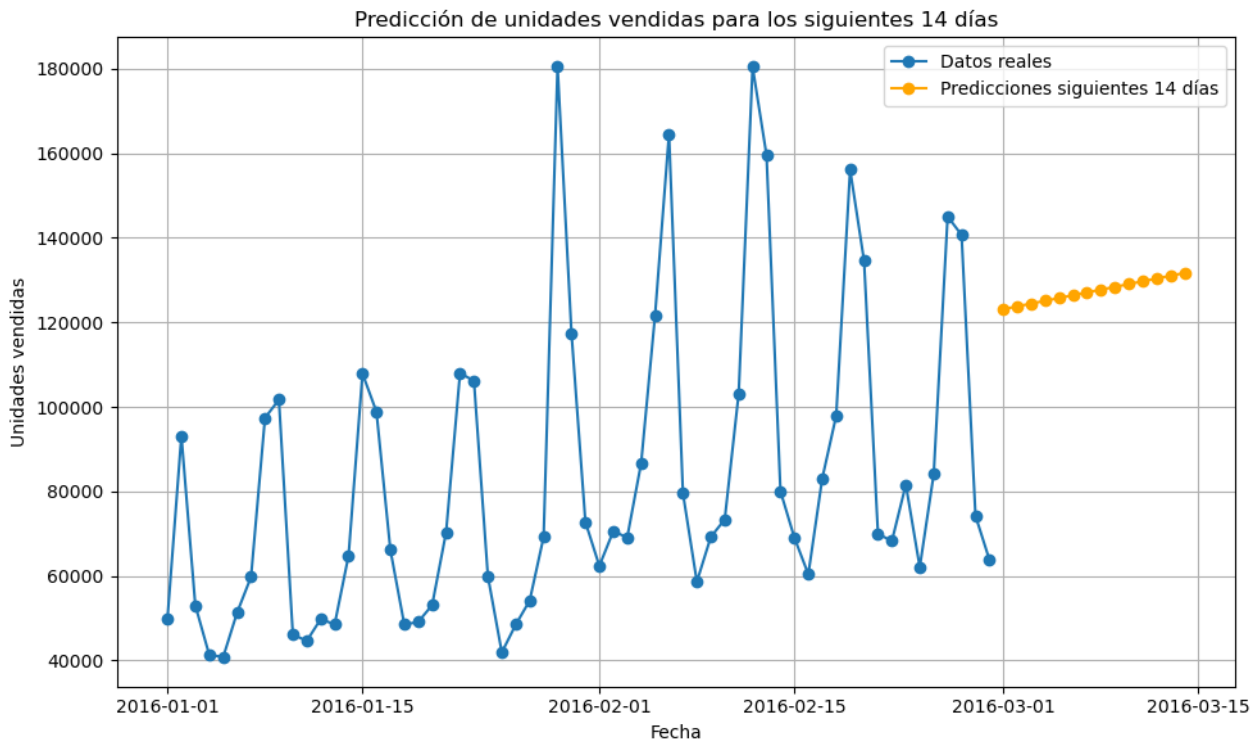


Figura 9: Predicción de las unidades vendidas

Por último, se ha evaluado el modelo utilizando las métricas de evaluación *MAE* (*Mean Absolute Error*) y *RMSE* (*Root Mean Square Error*). La métrica de evaluación *MAE* es una medida de la precisión de un modelo de predicción. Representa la media de las diferencias absolutas entre las predicciones del modelo y los valores reales observados (*Ecuación 1*). Un valor *MAE* más bajo indica un modelo más preciso. El *RMSE* es una medida comúnmente utilizada para evaluar la precisión de un modelo de regresión o de predicción. Se calcula como la raíz cuadrada de la media de los errores al cuadrado entre los valores predichos por el modelo y los valores observados reales (*Ecuación 2*). Este mide la diferencia promedio entre los valores predichos y los valores reales en la misma escala que los datos originales. Cuanto más bajo sea el valor de *RMSE*, mejor será la capacidad predictiva del modelo.

$$MAE = \frac{\sum_{i=1}^n |y_i - x_i|}{n}$$

Ecuación 1: Mean Absolute Error

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

Ecuación 2: Root Mean Square Error

En el modelo de regresión lineal realizado, el valor MAE obtenido ha sido 25.698, lo que significa que, en promedio, las predicciones del modelo están desviadas de los valores reales por aproximadamente 25.698 unidades de la cantidad y el valor RMSE obtenido ha sido 32.271. Al tener una escala de datos muy grande, se ha calculado el porcentaje de error en un día de ventas promedio, siendo el 31,06% en la métrica MAE y 39% en la métrica RMSE. Así, se ha concluido que es un porcentaje bastante alto por lo que se ha intentado mejorar realizando otro modelo de aprendizaje automático más avanzado como *Random Forest*.

Este modelo ofrece la capacidad de manejar relaciones no lineales y capturar la complejidad de los datos de manera más efectiva. En este caso, primero de todo se han importado las bibliotecas necesarias: *RandomForestRegressor* para el modelo, *GridSearchCV* para la búsqueda de hiperparámetros, y *train_test_split* para dividir los datos en conjuntos de entrenamiento y prueba. A continuación, se han definido las características (x) y la variable objetivo (y) para el modelo y se ha dividido el conjunto de datos en conjuntos de entrenamiento y prueba con una proporción del 20% de los datos reservados para pruebas y un estado aleatorio fijado en 42. Para obtener el resultado más óptimo, se ha definido un diccionario de hiperparámetros para probar diferentes combinaciones de valores y se ha realizado una búsqueda con validación cruzada. Después, se ha ajustado el modelo con los mejores hiperparámetros encontrados durante la búsqueda y se han realizado las predicciones en el conjunto de prueba utilizando el mejor modelo. En la *Figura 10* se encuentran las predicciones de las próximas dos semanas.

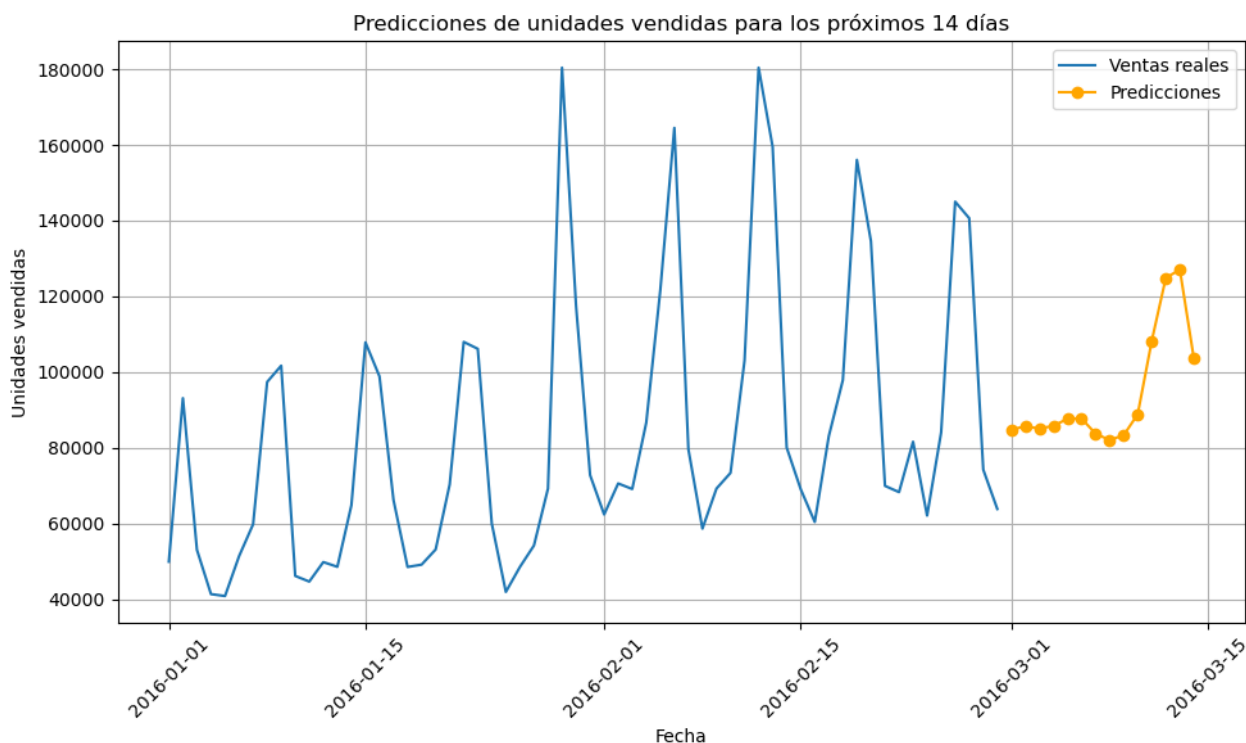


Figura 10: Predicciones de las unidades vendidas mediante Random Forest

Por último, se han calculado el error absoluto medio (MAE) y el error cuadrático medio ($RMSE$), donde los resultados no han mostrado una mejora con el enfoque de regresión lineal anterior, siendo el valor MAE 29.343 y el valor $RMSE$ 34.602.

Tras no encontrar una mejora en el resultado, se ha realizado un tercer estudio, donde además de explorar el rendimiento del modelo, también se ha considerado la aplicación de un enfoque alternativo utilizando un modelo ARIMA. El modelo ARIMA (*Autoregressive Integrated Moving Average*) es una técnica utilizada en series temporales para modelar y predecir datos. Es especialmente útil, como en este caso, cuando los datos muestran un comportamiento no estacionario, es decir, sus propiedades estadísticas cambian con el tiempo. Sin embargo, antes de proceder con el modelo ARIMA, se ha realizado la descomposición de la serie como se observa en la *Figura 11*, para entender mejor los componentes que contribuyen en la variabilidad de los datos. En la primera gráfica se observa la serie original donde se muestra la evolución completa de las ventas diarias. En la segunda gráfica se identifica que las ventas están aumentando en un horizonte a largo plazo, es decir, que tiene una tendencia al alza. En la gráfica de estacionalidad, se muestran los patrones repetitivos y predecibles en las ventas, lo que puede ayudar a identificar ciclos como picos y valles estacionales. Por último, la gráfica de residuos indica las irregularidades y las variaciones aleatorias que no se explican por la tendencia ni por la estacionalidad.

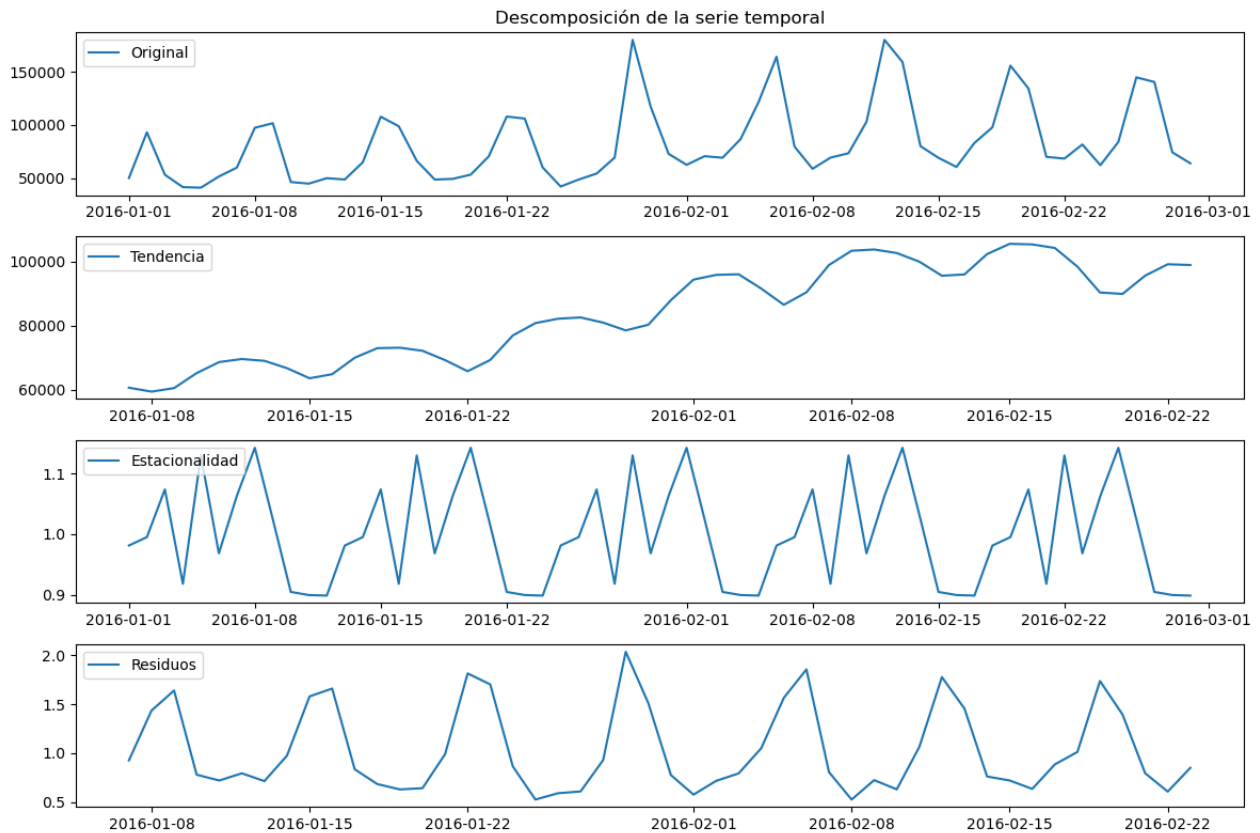


Figura 11:Descomposición de la serie temporal

Tras analizar la serie detalladamente, se ha procedido a ajustar y a entrenar un modelo ARIMA a la serie temporal para realizar las predicciones de los próximos 14 días, visualizándolas en la *Figura 12*. En este caso se ha obtenido un valor MAE de 14.537 y un valor RMSE de 17.334, los cuales son considerablemente mejores.

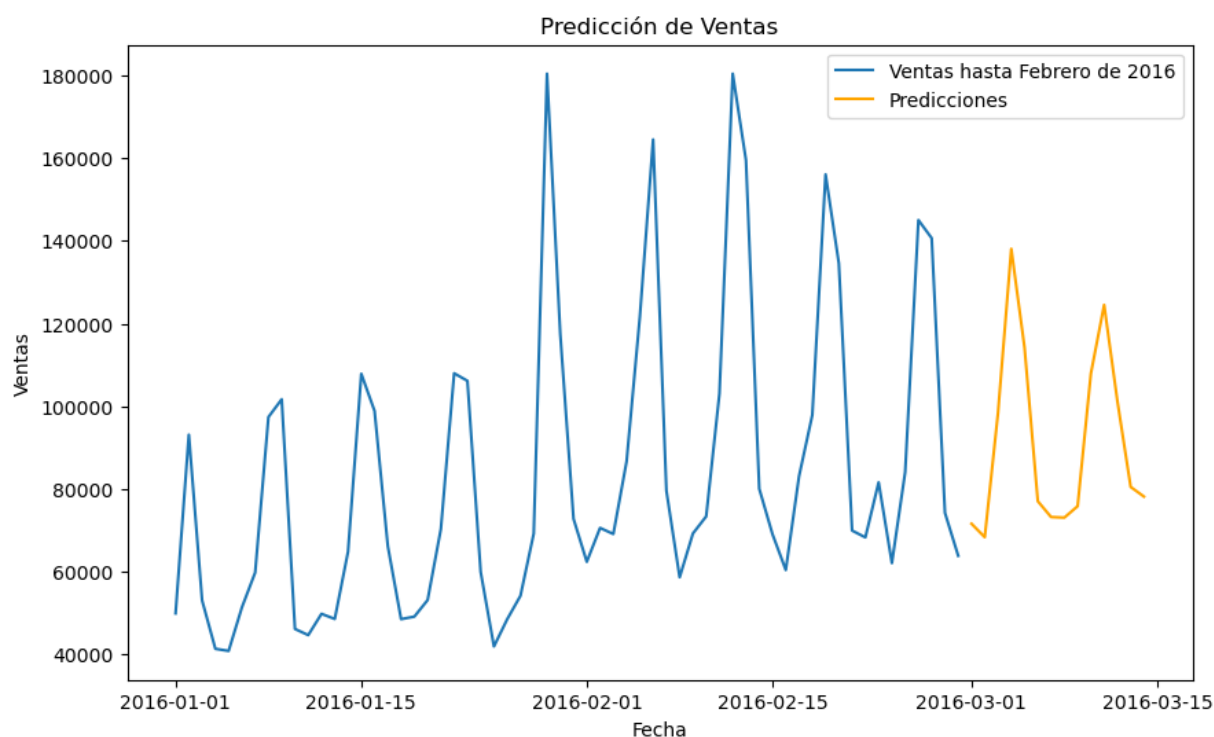


Figura 12: Predicción de las ventas de los próximos 14 días

Por último, se han recogido en la Tabla 2 los resultados obtenidos en cada uno de los modelos. Los resultados indican que el modelo ARIMA ha logrado capturar de manera efectiva la relación entre las variables predictoras y la variable objetivo, superando notablemente el desempeño del modelo de regresión lineal y Random forest anterior. Esto respalda la elección del modelo como un enfoque más adecuado para la tarea de predicción en este contexto específico.

	MAE	RMSE	% error promedio MAE	% error promedio RMSE
Regresión lineal	25.698,27	32.271,98	31,06%	39,00%
Random forest	29.343,22	34.602,12	34,76%	40,99%
ARIMA	14.537,77	17.334,05	15,40%	18,36%

Tabla 2: Métricas de evaluación de los modelos

2.1. Control de versiones en GIT

Git es un sistema avanzado y distribuidor de control de versiones que permite rastrear el progreso de un proyecto a lo largo del tiempo mientras evoluciona, registrando quién ha hecho qué cambios y por qué. Además, facilita el trabajo en paralelo de varios participantes, permitiendo que todos trabajen en el proyecto simultáneamente. GitHub es un servidor de alojamiento en línea para proyectos basados en Git, permitiendo la colaboración entre usuarios.

Funciona como un repositorio remoto donde se almacenan los archivos de un proyecto y registra su desarrollo de manera remota. Facilita compartir proyectos, registrar el desarrollo y proporciona seguridad en la nube. En proyectos colaborativos, GitHub se considera la copia principal del proyecto, siendo el punto central de interacción entre Git y los colaboradores [29].

Aunque el trabajo colaborativo no haya sido relevante en este caso específico, el uso de Git ha sido beneficioso para registrar y gestionar de manera efectiva las versiones del proyecto. Ha ayudado a mantener un flujo de trabajo organizado, a tener un registro claro de los cambios realizados y a facilitar futuras actualizaciones y mejoras en el proyecto. El repositorio creado se encuentra en el siguiente enlace: [Repositorio de github - TFM](#)

En primer lugar, Git ha proporcionado un historial detallado de todos los cambios realizados en el código a lo largo del tiempo. Esto es invaluable para entender cómo ha evolucionado el proyecto, qué decisiones se tomaron en cada paso del camino y por qué se tomaron. Incluso para un desarrollador individual, tener esta información a mano puede ser útil para rastrear errores, probar diferentes enfoques y mejorar la eficiencia en el desarrollo. Además, Git ha ofrecido la capacidad de crear ramas para experimentar con nuevas características o realizar cambios importantes sin afectar el código principal. Esto permite una mayor libertad y flexibilidad en el proceso de desarrollo, ya que los cambios pueden probarse y revisarse antes de fusionarse con la rama principal del proyecto. Otro beneficio es la posibilidad de revertir a versiones anteriores en caso de que surjan problemas o se necesite retroceder a una versión estable anterior. Esto proporciona una red de seguridad invaluable y tranquilidad al saber que es posible recuperar el proyecto en caso de emergencia.

En este proyecto, se ha trabajado con la integración de *Visual Studio Code (VS Code)* en GitHub. La integración efectiva de GitHub en *VS Code* comienza con la configuración y autenticación, garantizando una conexión segura entre el entorno de desarrollo local y los repositorios en GitHub. Una vez instalado *VS Code*, y creado una cuenta de GitHub, se han vinculado ambos mediante la generación de un token de acceso personal en GitHub, el cual se introduce en *VS Code*. Tras configurar y autenticar ambas cuentas, se han realizado las operaciones básicas como la creación de un nuevo repositorio y la clonación de repositorios existentes directamente desde la interfaz de *VS Code*. También se han gestionado cambios utilizando la interfaz gráfica de *VS Code*, incluyendo el *staging* y *commit* de cambios, así como el *push* y *pull* de cambios hacia y desde el repositorio remoto en GitHub [30]. En la siguiente *Figura 13* se muestra cómo se han realizado *commits* periódicos para mantener un eficiente control de las versiones.

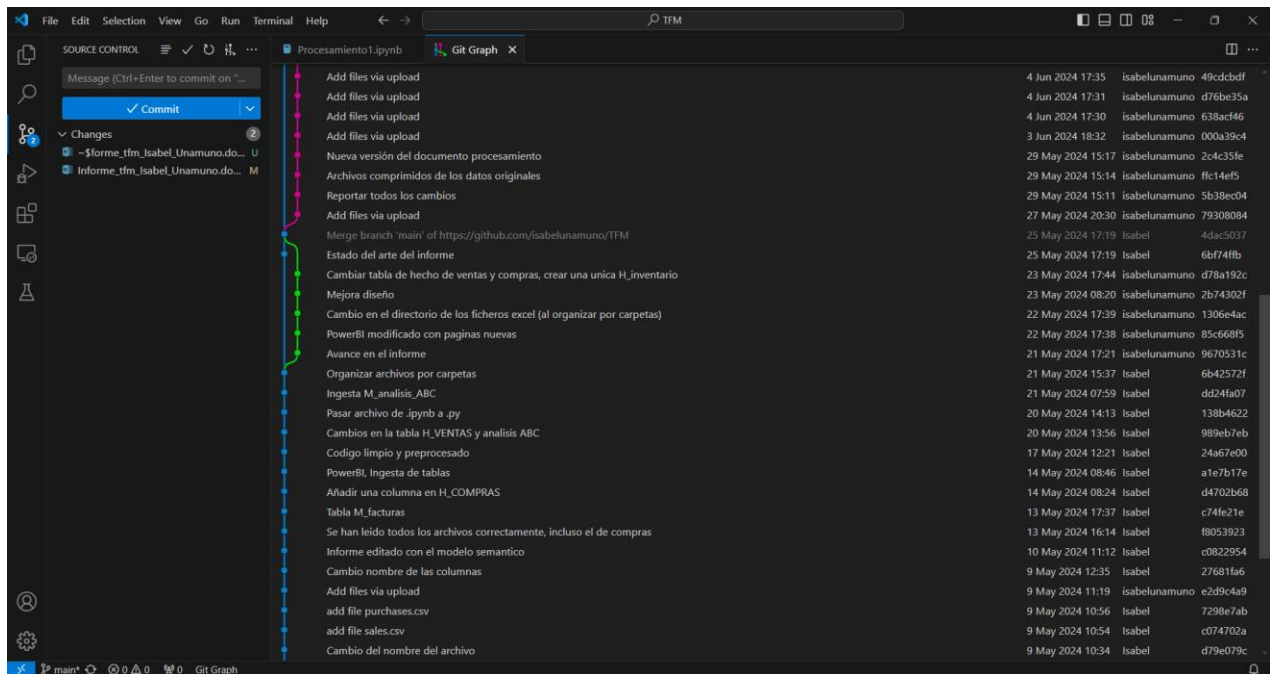


Figura 13:Control de versiones mediante la integración de Github en Visual Studio Code

3.RESULTADOS

Tras un exhaustivo proceso de limpieza y transformación de datos, se ha logrado desarrollar un informe interactivo que cumple con los objetivos establecidos para el proyecto. El proceso de limpieza de datos ha sido fundamental para garantizar la calidad y precisión de la información, eliminando inconsistencias, datos duplicados, y valores atípicos que podrían afectar el análisis final. El resultado final ha sido un informe interactivo que no solo presenta los datos de manera clara y comprensible, sino que también permite al cliente interactuar con la información de una manera que se alinea con sus necesidades y objetivos específicos para la toma de decisiones. Dado que se trata de un proyecto piloto, se ha decidido elaborar el informe utilizando los datos de enero de 2016, con la intención de incluir todos los datos futuros posteriormente.

A la hora de realizar el informe se ha trabajado con Power BI Desktop y una vez terminado, se ha publicado en la web a través de Power BI Service, con los beneficios que esto con lleva. El beneficio principal en este caso ha sido la accesibilidad, ya que los informes son accesibles desde cualquier lugar con conexión a Internet. Esto facilita que los usuarios puedan ver los informes sin necesidad de estar en la misma red local o tener instalado Power BI Desktop. Dicho esto, el informe final se encuentra en el siguiente enlace: [Cuadro de mando - Gestión de ventas, compras y stock](#), además de tenerlo en el repositorio.

3.1. Navegación entre páginas

El informe realizado consta de seis páginas principales que son la de inicio, resumen, ventas, compras, stock y tiendas. La navegación de estas páginas se realiza mediante la barra vertical a mano izquierda. Además, en los apartados de ventas y compras hay otro menú de navegación entre páginas que se encuentra en la parte superior. Estas subpáginas están relacionadas con su apartado y en el caso de las ventas, por ejemplo, se proporciona el resumen de la información de las ventas, el análisis de las ventas, el margen obtenido y la información sobre el cliente. Ambas navegaciones están indicadas con una flecha en la *Figura 14*.

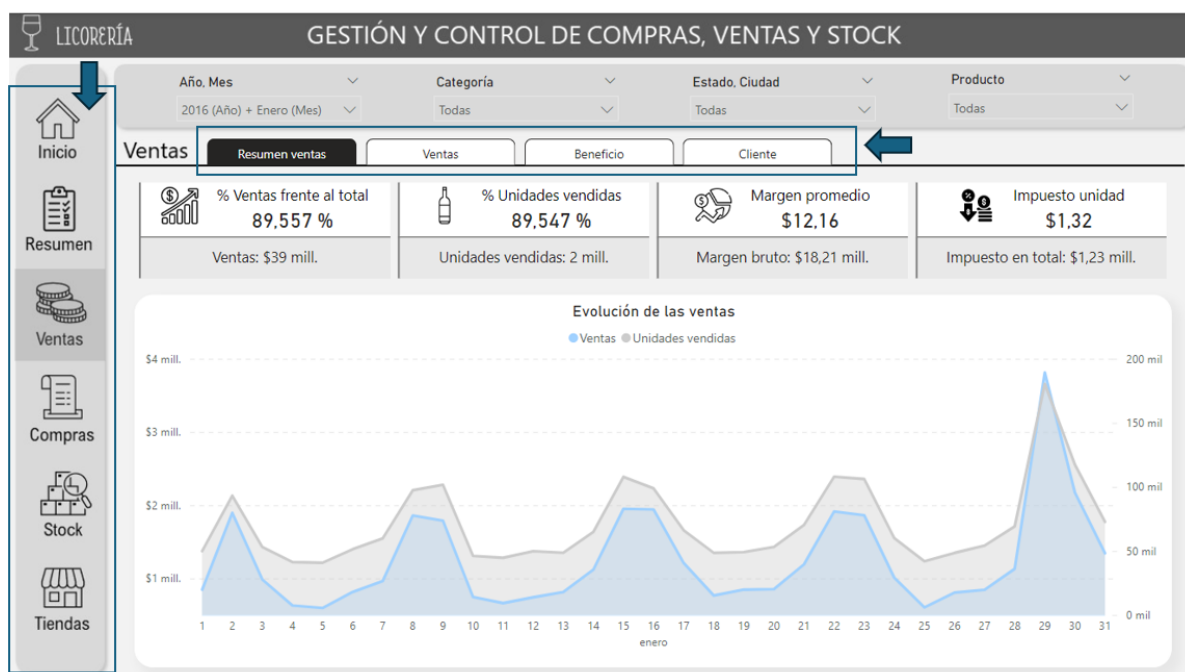


Figura 14: Navegación entre páginas

3.2. Dimensiones de filtro

En la parte superior de cada hoja se encuentran los campos de filtro de dimensiones que se aplican a cada página del informe como se observa en la *Figura 15*. Estos varían dependiendo de la página ya que en cada una se muestra información diferente. Entre las dimensiones de filtro mencionadas se encuentra el año, que en este caso se simplifica al 2016, pero en caso de que se siga utilizando dicho informe en la empresa el objetivo es que aparezcan todos los años hasta la fecha. También se puede filtrar por la categoría del producto, siendo estas las cinco categorías definidas, el producto, el estado y la ciudad donde se encuentra la tienda, el nombre del cliente y el nombre del proveedor.



Figura 15: Dimensiones de filtro

3.3. Páginas del informe

Como se ha mencionado anteriormente, el informe se compone de seis páginas principales. A continuación, se ha detallado la información presentada y la funcionalidad de cada una de ellas.

3.3.1. Página de inicio

La primera página es la de inicio, en la cual se encuentra únicamente la navegación de páginas como se observa en la *Figura 16*. Desde esta página se puede navegar a la hoja que desee el usuario haciendo *click* sobre el cuadrado correspondiente.



Figura 16: Página de inicio

3.3.2. Página de resumen

En la página de resumen se encuentra el resumen general de todo el cuadro de mando, donde se muestra la información que se ha considerado más importante tal y como se puede observar en la *Figura 17*. En la parte superior de la hoja, se encuentran las tarjetas que muestran la siguiente información:

- Las ventas y el coste de las compras del mes de enero tanto en dólares como en unidades, donde se puede ver la información filtrada por cualquiera de las dimensiones de filtro
- La cantidad total de stock disponible el 1 de enero, junto con su valor correspondiente. Es decir, proporciona tanto el número de unidades en existencia como su valoración monetaria, permitiendo una comprensión clara del capital inmovilizado en inventarios al inicio del periodo.
- El margen bruto, que se refiere a la resta entre las ventas y las compras con el porcentaje correspondiente.

En cuanto a los gráficos de la página, en la parte izquierda se encuentra un mapa donde se pueden ver las ventas por estado y por categoría y a mano derecha, en el gráfico de barras, se encuentra el porcentaje del beneficio para cada una de las categorías.

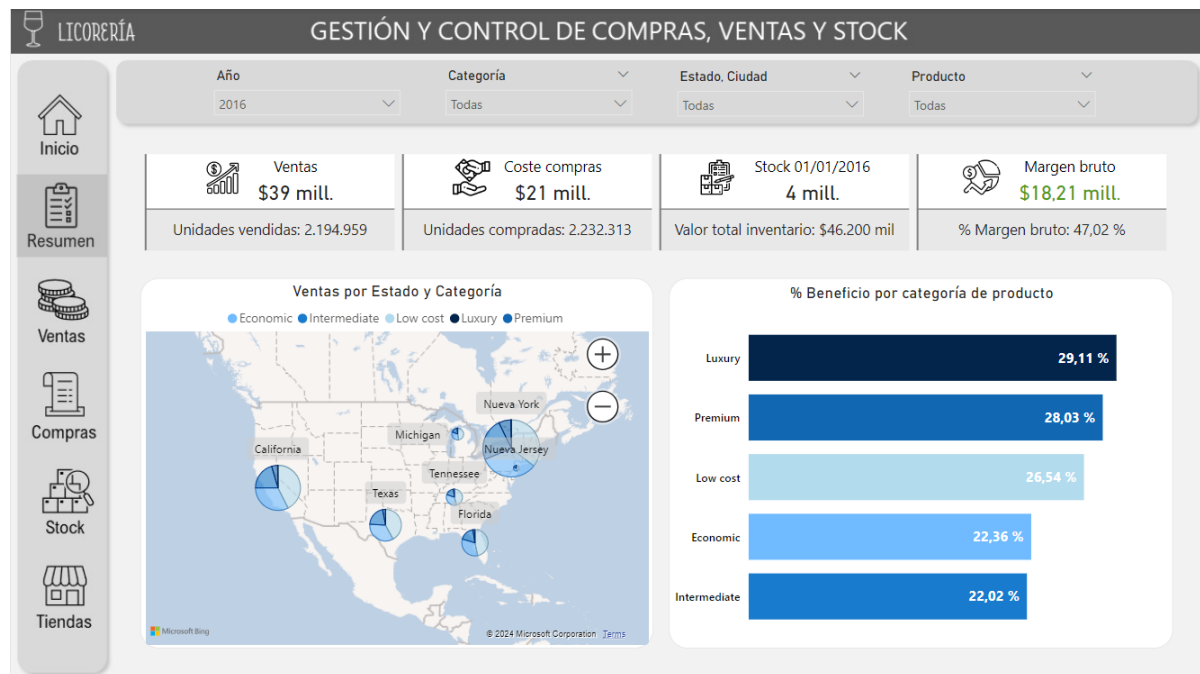


Figura 17: Página de resumen

3.3.3. Página de ventas

La página de las ventas comienza con un resumen donde se recoge la información más importante sobre ellas, como las unidades vendidas y las ventas realizadas en dólares con sus respectivos porcentajes y el margen promedio por categoría, además de la evolución de las ventas mediante un gráfico de líneas. A continuación, se encuentra el análisis de las ventas donde se observan las ventas realizadas por categoría de producto y por estado. En este apartado, la información se muestra tanto en gráficas como en una matriz, con la intención de ofrecer información más detallada en la matriz. Para ello, se han colocado dos iconos en la parte superior izquierda del informe, como se observa en la *Figura 18*, siendo estos el icono de la gráfica y de la tabla donde con un simple *click* se cambia la información que se muestra en pantalla. Esto se ha realizado mediante marcadores como se ha visto en el apartado Trabajo con marcadores. La siguiente página dentro de las ventas es la de beneficio, donde se ve a simple vista los productos que más beneficio bruto generan y se analiza las ventas de aquellos productos. La última subpágina dentro del apartado de ventas es la de cliente; en ella, existe la oportunidad de elegir el cliente deseado mediante un filtro y se obtiene la información detallada de dicho cliente.

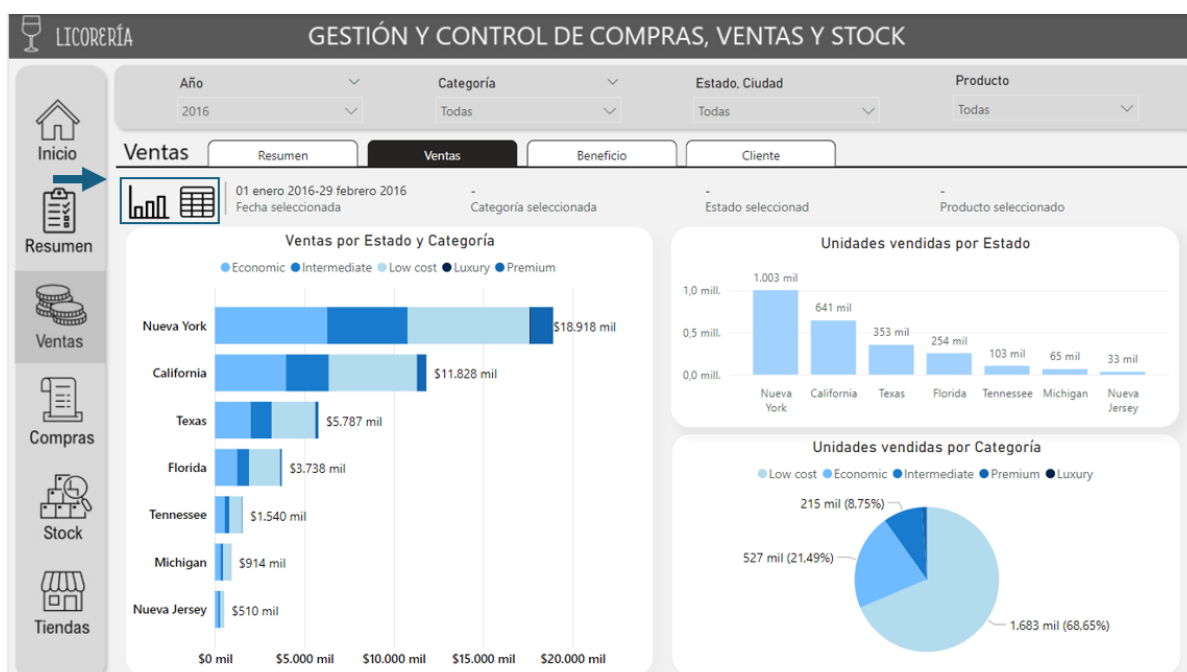


Figura 18: Vista gráfica de la subpágina de ventas dentro del apartado de ventas

3.3.4. Página de compras

En la página de compras se ha seguido la misma línea y estructura que en las ventas con el objetivo de acostumbrar al usuario y facilitar la navegación. Para ello, se ha creado una página de resumen donde se muestra la información más relevante de las compras como las unidades compradas y el coste, entre otras. La siguiente subpágina es la de compras, donde se observan los mismos gráficos que en las ventas, pero en este caso con las unidades vendidas y el coste de las compras por estado y categoría. También se ha habilitado la opción de ver la información mediante gráficos o mediante una matriz. Para ello, se han situado dos botones en el mismo sitio para poder realizar el cambio cuando el usuario lo desee. Por último, se ha realizado el análisis de los proveedores como se observa en la *Figura 19*, donde se puede ver información detallada filtrando por el proveedor deseado. Además, de un simple vistazo también se pueden ver cuáles son los proveedores potenciales de la empresa, ya que en la gráfica están ordenados por el coste de las compras en orden descendente.

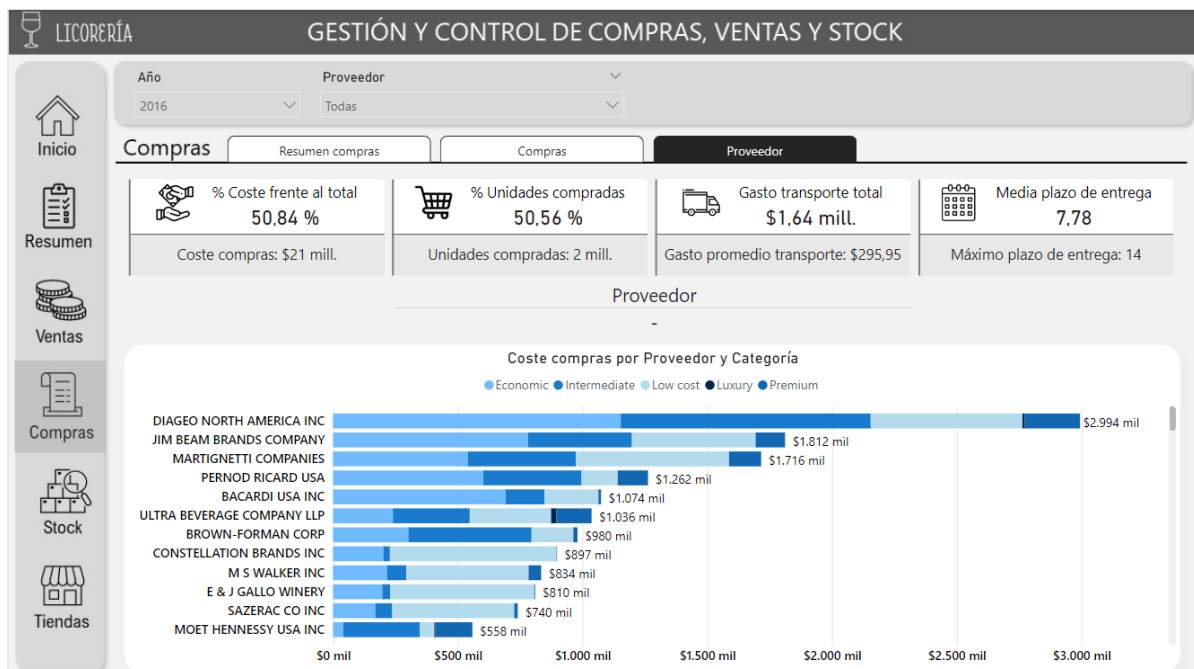


Figura 19: Subpágina de proveedor dentro del apartado de compras

3.3.5. Página de stock

Continuando en la navegación de páginas del informe, la siguiente muestra el análisis del stock. Por un lado, se ha mostrado mediante tarjetas, el stock disponible el 1 de enero de 2016, las compras y ventas de enero de ese mismo año, es decir, las entradas y salidas de material y el stock del 31 de enero de ese mismo año. También se ha mostrado el valor del inventario en dólares a principios de año y cuál sería el stock óptimo. Por otro lado, en este caso también se ha dado la opción de ver la información mediante gráficas o mediante la matriz, haciendo *click* en el botón indicado, gracias a los marcadores. En la siguiente *Figura 20* se pueden ver las dos opciones de visualizar los datos, es decir, mediante un gráfico de barras como el de la izquierda o mediante una matriz. En ambas páginas hay una relación entre los colores, ya que, se utiliza el mismo color para cada medida.

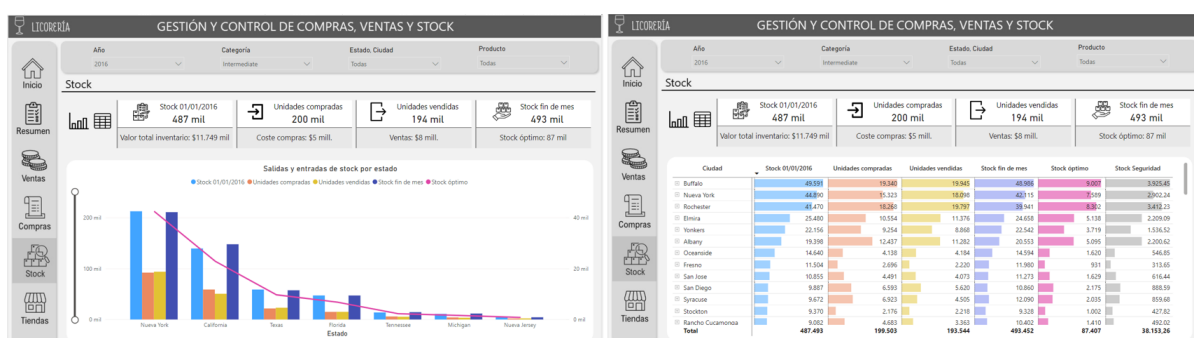


Figura 20: Página de stock

3.3.6. Página de tiendas

La última página es un análisis sobre las tiendas, es decir las ventas que se han realizado en cada una ellas, como se observa en la *Figura 21*. Además, se puede observar la situación de cada tienda, esto es, se han comparado las ventas con su objetivo y se ha mostrado en verde en caso de que haya superado el objetivo y en rojo en el caso contrario, con su respectivo porcentaje. Esta lógica también se mantiene en el mapa que se encuentra a mano izquierda del informe, donde se indica la ubicación de cada tienda y el estado de ellas. Se puede acceder a ellas y a su información haciendo *click* sobre el mapa. Por último, en las tarjetas se observa la información de la cantidad de tiendas que se encuentran, en caso de que se filtre por categoría, por estado o por tienda con la opción de dimensiones de filtro. También se puede ver las ventas en dólares, y el porcentaje de diferencia frente al objetivo.

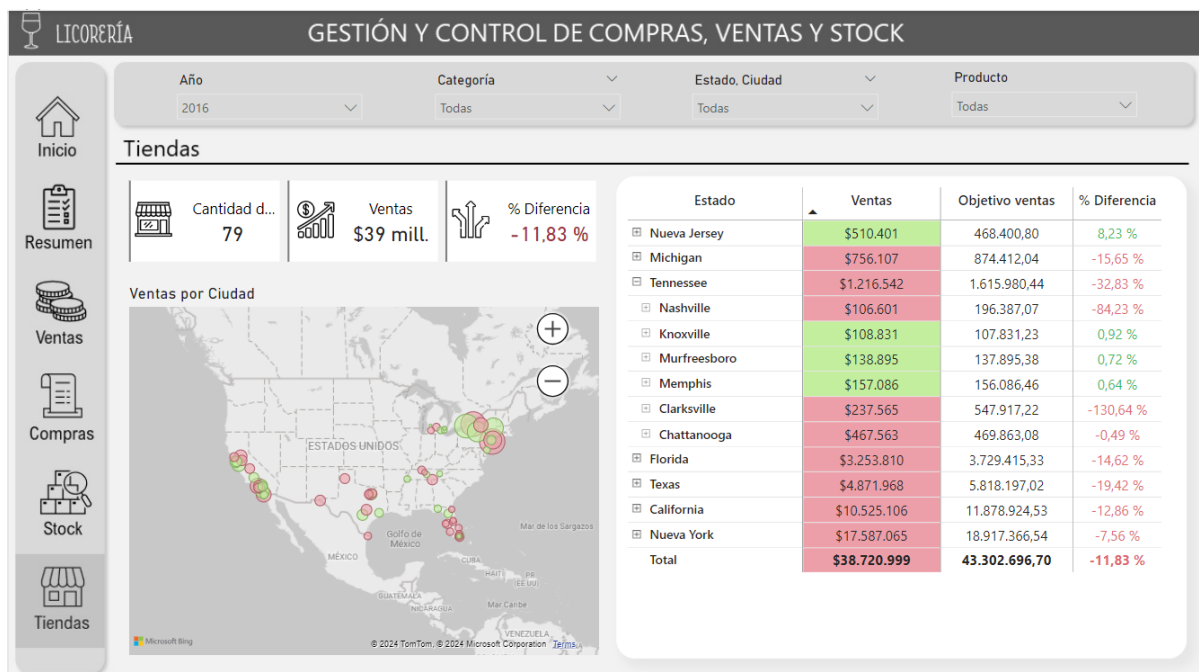


Figura 21: Páginas de las tiendas

4.CONCLUSIONES Y DISCUSIÓN

El proyecto de diseño y creación de un cuadro de mando para la gestión de compras, ventas y stock de una empresa mayorista de vinos y licores ha permitido abordar de manera efectiva las complejidades asociadas con el manejo de grandes volúmenes de datos. La utilización de Power BI como herramienta principal ha facilitado la visualización clara y accesible de los datos, proporcionando una plataforma interactiva que mejora significativamente la toma de decisiones estratégicas.

Primero de todo, se ha asegurado la calidad y consistencia de los datos, la cual se ha considerado un aspecto crítico. Para el preprocesamiento y limpieza de datos el uso de Python ha sido clave, garantizando así la precisión de los análisis.

A la hora de visualizar los datos, la implementación de Power BI ha permitido crear un cuadro de mando dinámico que simplifica la interpretación de datos complejos. Esto incluye la presentación de ventas, compras, y análisis de stock, lo que ofrece una visión integral del rendimiento de la empresa. Asimismo, el cuadro de mando interactivo ha permitido a los usuarios navegar fácilmente entre diferentes secciones y acceder a información detallada según sea necesario. Esta característica es esencial para usuarios con distintos niveles de conocimiento en análisis de datos. El informe también ha permitido al usuario un análisis detallado del stock, junto con el monitoreo de compras y ventas, el cual ha ayudado a identificar patrones y tendencias, además de la optimización de ellas. Esto facilita una gestión más eficiente del inventario, las adquisiciones y la distribución de productos, siendo fundamental para minimizar costos y maximizar la disponibilidad de mercancías. Por último, de cara a realizar predicciones de la demanda los próximos años se han implementado tres modelos diferentes, concluyendo que el modelo ARIMA es el más adecuado para la predicción de la demanda en este contexto específico, ofreciendo una mayor precisión y un mejor ajuste a los datos históricos.

En resumen, este proyecto ha demostrado el valor de una gestión de datos robusta y bien estructurada. La empresa no solo ha mejorado su capacidad de análisis y toma de decisiones, sino que también está mejor posicionada para enfrentar los retos futuros en un entorno de negocios cada vez más competitivo y orientado a los datos.

5.LINEAS FUTURAS

El éxito de este proyecto piloto sienta las bases para una implementación a mayor escala en la empresa. El análisis actual se ha realizado utilizando los datos correspondientes al mes de enero de 2016. Sin embargo, de cara a futuro, en caso de que la empresa apruebe el proyecto, se pretende extender el análisis a todos los años disponibles hasta la fecha. Esto permitirá observar y analizar la información de manera más completa y detallada, abarcando un período de tiempo más amplio y proporcionando una visión más profunda de las tendencias y patrones a lo largo de los años.

Además, actualmente, la conexión de datos en Power BI se realiza a través de archivos CSV almacenados localmente. No obstante, como el volumen de datos incrementará considerablemente al incluir todos los años, con el objetivo de mejorar la eficiencia y la escalabilidad, se propone migrar hacia una integración directa con *SAP*, el software utilizado por la empresa. En concreto la empresa utiliza *SAP HANA*, por lo que se deberán seguir los siguientes pasos. Power BI permite una conexión directa a la base de datos, pero para ello, primero de todo se debe instalar el conector de *SAP HANA*. A continuación, se debe configurar la conexión, introduciendo los detalles del servidor, la base de datos, y las credenciales de usuario. Por último, se deben importar los datos, y para ello, Power BI ofrece la opción de importar los datos a utilizar de consulta directa (*Direct Query*) para obtener los datos en tiempo real.

Finalmente, mencionar que la integración con *SAP HANA* también abre la puerta a futuras innovaciones en el análisis de datos, como la incorporación de algoritmos de *Machine Learning* para predicciones más precisas y automatización de procesos.

6. BIBLIOGRAFÍA

- [1] Jordi de Mas Jaumot, «Gestión del inventario en una empresa del sector farmacéutico mediante algoritmos de Machine Learning», 2021. [En línea]. Disponible en: https://openaccess.uoc.edu/bitstream/10609/129826/8/jde_masTFM0121memoria.pdf
- [2] I. de I. Conocimiento, «Machine Learning & Deep Learning», 2024, [En línea]. Disponible en: <https://www.iic.uam.es/inteligencia-artificial/machine-learning-deep-learning/>
- [3] T. Šustrová, «A Suitable Artificial Intelligence Model for Inventory Level Optimization Tereza Šustrová: A Suitable Artificial Intelligence Model for Inventory Level Optimization», vol. 25, n.º 1, pp. 48-55, 2016, [En línea]. Disponible en: <http://dx.doi.org/10.13164/trends.2016.25.48>
- [4] S. Zhang, X. Qin, S. Hu, Q. Zhang, B. Dong, y J. Zhao, «Importance Degree Evaluation of Spare Parts Based on Clustering Algorithm and Back-Propagation Neural Network», *Math. Probl. Eng.*, vol. 2020, 2020, doi: 10.1155/2020/6161825.
- [5] K. B. Praveen, P. Kumar, J. Prateek, G. Pragathi, y P. M. J, «Inventory Management System Using Machine Learning», *Int. J. Innov. Eng. Manag. Res.*, vol. 9, n.º 06, pp. 769-785, 2022, doi: 10.48047/ijiemr/v11/i06/51.
- [6] Scikit learn, «Gradient Boosting Regressor», 2024. <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingRegressor.html>
- [7] M. Pérez Ramis y A. P. Moreno Beltrán, «Aplicación de la Ciencia de Datos para la predicción de la Demanda», 2017.
- [8] «Decision tree in Machine Learning», 2024, [En línea]. Disponible en: <https://www.geeksforgeeks.org/decision-tree-introduction-example/>
- [9] Al Maverick, «Light Gradient Boosting Machine», 2023. <https://samanemami.medium.com/light-gradient-boosting-machine-b4f1b9e3f7d1>
- [10] Gonzalo Castillo, «Logística 4.0», *Innovación Digit. 360*, 2023, [En línea]. Disponible en: <https://www.innovaciondigital360.com/industria-4-0/supply-chain/logistica-4-0-que-es-caracteristicas-beneficios-y-tecnologia/>
- [11] EAE Business School, «Descubre la Optimización de Inventario con Machine Learning», 2024. <https://www.eaemadrid.com/es/blog/machine-learning>
- [12] «¿Qué es y para qué sirve un diagrama de Gantt? | Teamleader».

- <https://www.teamleader.es/blog/diagrama-de-gantt> (accedido 20 de abril de 2023).
- [13] Kaggle, «Inventory Analysis case study», 2023. <https://www.kaggle.com/datasets/bhanupratapbiswas/inventory-analysis-case-study>
- [14] P. D. A. Library, «Pandas», 2024. <https://pandas.pydata.org/>
- [15] «Numpy», 2024. <https://numpy.org/>
- [16] Matplotlib, «Visualization with python», 2024. <https://matplotlib.org/>
- [17] Seaborn, «Seaborn: Statistical data visualization», 2024, [En línea]. Disponible en: <https://seaborn.pydata.org/>
- [18] Pyexcel, «Let focus on data, instead of file formats», 2024. <https://docs.pyexcel.org/en/latest/>
- [19] Microsoft, «Diseño de un modelo semántico en PowerBI», 2023. <https://learn.microsoft.com/es-es/training/modules/design-model-power-bi/1-introduction>
- [20] Microsoft, «Crear y administras relaciones en PowerBI», 2024, [En línea]. Disponible en: <https://learn.microsoft.com/es-es/power-bi/transform-model/desktop-create-and-manage-relationships>
- [21] J. Quiroz, «El modelo relacional de bases de datos», *Boletín de Política Informática*, vol. 6, pp. 53-61, 2003.
- [22] IBM, «Primary and foreign keys», 2023, [En línea]. Disponible en: <https://www.ibm.com/docs/es/ida/9.1.2?topic=entities-primary-foreign-keys>
- [23] Microsoft learn, «Descripción del uso de Power BI para crear análisis controlados por datos», 2024. [En línea]. Disponible en: <https://learn.microsoft.com/es-es/training/modules/introduction-power-bi/2-describe-using-power-bi-build-data-driven-analytics>
- [24] Microsoft learn, «Información general sobre DAX», 2024, [En línea]. Disponible en: <https://learn.microsoft.com/es-es/dax/dax-overview>
- [25] Microsoft, «Creación de una tabla de fechas», 2024, [En línea]. Disponible en: <https://learn.microsoft.com/es-es/training/modules/design-model-power-bi/3-date-table>
- [26] Kurt Buhler, «Introducing the 3-30-300 rule for better reports», 2024, [En línea]. Disponible en: <https://www.sqlbi.com/articles/introducing-the-3-30-300-rule-for-better-reports/>
- [27] Microsoft learn, «Diseño de informes en Power BI», 2024, [En línea]. Disponible en: <https://learn.microsoft.com/es-es/training/modules/power-bi-effective-reports/>

- [28] «Mejora de los diseños de informes de Power BI para la experiencia del usuario», 2024, [En línea]. Disponible en: <https://learn.microsoft.com/es-es/training/modules/power-bi-effective-user-experience/>
- [29] J. Astigarraga y V. Cruz-Alonso, «iSe puede entender cómo funcionan Git y GitHub!», *Ecosistemas*, vol. 31, n.º 1, pp. 1-6, 2022, doi: 10.7818/ECOS.2332.
- [30] Github, «Uso de GitHub Codespaces en Visual Studio Code», 2024. <https://docs.github.com/es/codespaces/developing-in-a-codespace/using-github-codespaces-in-visual-studio-code>

7. ANEXOS

Anexo A: Descripción de las tablas del modelo

Tabla	Columna	Descripción
D_PRODUCTO	Producto_marca	Clave primaria. Es un denominador único de la tabla que se utiliza para relacionar con la tabla de hechos.
	Marca	Marca del producto
	Producto	Nombre del producto
	Precio compra unidad	El precio de compra de cada producto
	Precio venta unidad	El precio de venta de cada producto
	Margen bruto	El margen bruto por cada producto, es decir la resta entre el precio de venta y compra
	Volumen	El volumen en litros de cada producto
	Categoría	Se han dividido 5 categorías (Low cost, economic, intermediate, Premium y luxury) dependiendo del precio unidad de cada producto
D_CLIENTE	IdCliente	Clave primaria. Es un denominador único de la tabla que se utiliza para relacionar con la tabla de hechos
	Cliente	Nombre del cliente
D_PROVEEDOR	IdProveedor	Clave primaria. Es un denominador único de la tabla que se utiliza para relacionar con la tabla de hechos
	Proveedor	Nombre del proveedor
D_TIENDA	Ciudad_tienda	Clave primaria. Es un denominador único de la tabla, que se utiliza para relacionar con la tabla de hechos
	Tienda	Nombre de la tienda
	Ciudad	Nombre de la ciudad en la que se encuentra la tienda, ya que hay una única tienda por ciudad
	Estado	El nombre del estado de la ciudad donde se encuentra la tienda
D_CALENDARIO	Fecha	Clave primaria. Fecha completa construida mediante la función CALENDAR()
	Año	Año de la fecha
	NumMes	Número del mes de la fecha
	NumSemana	Número de la semana de la fecha indicada
	Día de la semana	Día de la semana correspondiente

	Mes	Mes correspondiente
	NumDía	Número del día correspondiente del mes
H_VENTAS	Tienda	Clave foránea. Código de la tienda que se relaciona con la tabla M_TIENDA
	Producto_marca	Clave foránea. Código del producto que se relaciona con la tabla M_PRODUCTO
	IdCliente	Clave foránea. Código del cliente que se relaciona con la tabla M_CLIENTE
	Fecha venta	Fecha de la venta de producto
	Unidades vendidas	Cantidad de ventas realizadas
	Precio venta unidad	Precio unitario del producto vendido
	Precio sin impuesto	Precio de venta considerando la cantidad vendida, pero sin tener en cuenta el impuesto
	Impuesto	Precio del impuesto
	Ventas	Columna calculada. Precio total considerando la cantidad y el impuesto
H_COMPRAS	Tienda	Clave foránea. Código de la tienda que se relaciona con la tabla D_TIENDA
	Producto_marca	Clave foránea. Código del producto que se relaciona con la tabla D_PRODUCTO
	IdProveedor	Clave foránea. Código del proveedor que se relaciona con la tabla D_PROVEEDOR
	Fecha orden de compra	Fecha de cuando se realizó la compra
	Fecha llegada	Clave foránea. Fecha de la llegada del producto a planta, que se relaciona con D_CALENDARIO
	Fecha factura	Fecha de la factura
	Fecha de pago	Fecha de pago
	Precio compra unidad	El precio unitario de compra
	Unidades compradas	La cantidad de compra
	Coste compras	El precio total de la compra, considerando la cantidad
	Plazo de entrega	Columna calculada, indica el plazo desde que se realizó la orden de compra hasta que llega el producto.
H_INVENTARIO	Ciudad_tienda	Clave foránea. Código de la tienda que se relaciona con la tabla D_TIENDA

	Producto_marca	Clave foránea. Código del producto que se relaciona con la tabla D_PRODUCTO
	Stock	Cantidad de stock el día 01/01/2016
	Fecha stock	Fecha cuando se contabilizó el stock
	Estado	Estado donde se almacena el stock
H_FACTURAS	IdProveedor	Clave foránea. Código del proveedor que se relaciona con la tabla D_PROVEEDOR
	Fecha factura	Fecha de llegada de la factura
	Numero orden de compra	Número de la orden de compra
	Fecha orden de compra	Fecha de cuando se emitió la orden de compra
	Fecha pago	Fecha de cuando se pago
	Unidades	Unidades vendidas
	Coste	Coste de la compra
	Transporte	Coste del transporte

Anexo B: Descripción de las medidas calculadas

Carpeta	Nombre	Descripción
Compras	% coste frente al total	Porcentaje del coste frente al coste total
	% unidades compradas frente al total	Porcentaje de las unidades compradas frente a las unidades compradas en total
	Coste compras	Suma del coste de las compras realizadas
	Coste total	La suma del precio de compra total en el periodo de enero y febrero, independientemente del filtro
	Máximo plazo de entrega	
	Media plazo de entrega	La media de días para entregar un producto desde que se ha realizado el pedido
	Precio compra unidad	La suma del precio de compra por unidad
	Transporte	Coste del transporte
	Unidades compradas	Suma de la cantidad de compras registrada
	Unidades compradas en total	Suma de la cantidad de unidades compradas, independientemente del filtro
Margen	% Beneficio	Beneficio bruto dividido entre las ventas
	% Margen bruto	Porcentaje de la diferencia entre las ventas y compras

		frente a las ventas
	Beneficio bruto	La suma del beneficio considerando todas las ventas
	Beneficio bruto unidad	El beneficio bruto por cada unidad vendida
	Margen bruto	La resta entre las ventas y las compras
	Margen promedio	El margen promedio por unidad
Stock	Stock 01/01/2016	La cantidad de stock por tienda el 1 de enero
	Stock fin de mes	La cantidad de stock por tienda el 31 de enero
	Stock optimo	Cálculo del stock óptimo
	Stock seguridad	Cálculo del stock de seguridad
	Valor inventario	Valor del inventario en dólares
Tiendas	% diferencia	Porcentaje de diferencia entre las ventas y el objetivo
	Cantidad de tiendas	Cantidad de tiendas que se encuentran en cada estado o ciudad
	Diferencia	Diferencia entre las ventas y el objetivo
	Objetivo ventas	Objetivo de ventas definido por la empresa para cada tienda
Ventas	% Unidades vendidas	Porcentaje de las unidades vendidas frente al total
	% Ventas frente al total	Porcentaje de las ventas realizadas frente a la facturación total
	Días con ventas	Cantidad de días con ventas registradas
	Facturación total	Facturación total de los meses de enero y febrero en dólares
	Impuesto en total	Suma del impuesto por venta de alcohol
	Impuesto unidad	Impuesto de cada producto por venta de alcohol
	Precio venta por unidad	Precio de compra por unidad
	Unidades vendidas	Cantidad de las unidades vendidas
	Ventas diarias	Cálculo de la división entre la cantidad de ventas en total y el número de días con ventas
	Unidades vendidas en total	Unidades vendidas en total el mese de enero y febrero
	Ventas	Precio de las ventas en dólares
Utilidad	Categoría seleccionada	Cálculo que indica la categoría que ha seleccionado el usuario en el filtro
	Ciudad seleccionada	Cálculo que indica la ciudad que ha seleccionado el usuario en el filtro
	Cliente seleccionado	Cálculo que indica el cliente que ha seleccionado el usuario en el filtro
	Producto seleccionado	Cálculo que indica el producto que ha seleccionado el usuario en el filtro

	Proveedor seleccionado	Cálculo que indica el proveedor que ha seleccionado el usuario en el filtro
	Fecha seleccionada	Cálculo que indica la fecha que ha seleccionado el usuario en el filtro
	Estado seleccionado	Cálculo que indica el estado que ha seleccionado el usuario en el filtro
	Color objetivo ventas	Cálculo que indica el color de fuente del número dependiendo de la condición; es decir, si las ventas son mayores que el objetivo en verde, y de lo contrario en rojo

TRABAJO DE FIN DE MÁSTER TRABAJO DE FIN DE MÁSTER TRABAJO DE FIN DE MÁSTER TRABAJO DE FIN DE MÁSTER TRABAJO DE FIN DE MÁSTER