

# Instituto Tecnológico y de Estudios Superiores de Occidente

Reconocimiento de validez oficial de estudios de nivel superior según acuerdo secretarial  
15018, publicado en el Diario Oficial de la Federación del 29 de noviembre de 1976.

Departamento de Matemáticas y Física  
**Maestría en Ciencia de Datos**



## Modelo Híbrido Prophet/XGBoost Para Predicción Demanda de Inventarios Retail Automotive Aftermarket

---

TRABAJO RECEPCIONAL que para obtener el GRADO de  
Maestro en Ciencia de Datos

Presenta:  
**Abraham Iñiguez Garcia**

Director:  
**Mtro. Arturo Silva Gálvez**

Tlaquepaque, Jalisco, 3 de junio de 2025



# Modelo Híbrido Prophet/XGBoost Para Predicción Demanda de Inventarios Retail Automotive Aftermarket

Abraham Iñiguez Garcia

## Resumen

La toma de decisiones fundamentada en datos se ha convertido en una necesidad imperativa para las empresas que buscan no solo mantenerse competitivas en el mercado actual, sino también impulsar un crecimiento sostenible a largo plazo. En un entorno empresarial cada vez más dinámico y complejo, confiar únicamente en la intuición o en experiencias pasadas ya no es suficiente. El análisis de datos permite a las organizaciones identificar patrones, predecir tendencias y tomar decisiones informadas que minimicen riesgos y maximicen oportunidades.

Este proyecto se enfoca en la predicción de demanda de inventario de empresas en el sector de venta de piezas automotrices retail con el objetivo de que las decisiones estratégicas de la dirección se basen en datos sólidos, permitiendo así una mayor precisión y efectividad. Al implementar un enfoque basado en datos, se busca optimizar la planificación de recursos, anticipar las necesidades del mercado y mejorar la rentabilidad de la empresa. De esta manera, la compañía estará mejor equipada para tomar decisiones informadas que promuevan un crecimiento sostenible y una ventaja competitiva en su sector.

Para el desarrollo de este proyecto, se utilizaron conjuntos de datos generados a partir de las bases de datos del sistema ERP productivo de una empresa perteneciente al ramo. Para el análisis de la predicción de demanda de inventario se implementó un algoritmo de series de tiempo llamado Prophet, así como algoritmo de machine learning XGBOOST que en conjunto crean un modelo híbrido donde tomamos las ventajas de ambos. [1].

El estudio comparó diversos modelos de predicción de demanda, demostrando que los enfoques modernos como Prophet y el Modelo Híbrido superan significativamente a los métodos tradicionales (ARIMA, ETS y Promedios Históricos), mostrando mayor precisión y adaptabilidad a fluctuaciones en los datos. XGBOOST y su versión mejorada presentaron un desempeño intermedio, superando a los modelos clásicos pero sin alcanzar el rendimiento de Prophet.

Se evidenció la baja efectividad del modelo de Promedios Históricos, actualmente utilizado por la empresa, destacando la necesidad de adoptar técnicas más avanzadas. Además, el procesamiento riguroso de los datos excluyendo períodos irregulares y ajustando por eventos atípicos como la pandemia fue clave para garantizar resultados robustos. Estos hallazgos resaltan la importancia de implementar modelos adaptativos en entornos con alta variabilidad en la demanda.

# Tabla de Contenidos

	<b>Página</b>
1 Introducción . . . . .	11
1.1. Contexto . . . . .	11
1.2. Justificación . . . . .	12
1.3. Problema . . . . .	13
1.4. Objetivos . . . . .	14
1.4.1. Objetivo General . . . . .	14
1.4.2. Objetivos específicos . . . . .	14
2 Metodología . . . . .	15
2.1. Descripción de los datos . . . . .	15
2.2. Análisis exploratorio . . . . .	16
2.3. Descripción de los modelos . . . . .	18
2.3.1. Prophet Para Predicción en Series de Tiempo . . . . .	19
2.3.2. XGBoost Para Detección De Relaciones . . . . .	20
2.3.3. Modelo Híbrido Prophet-XGBoost . . . . .	21
2.4. Descripción de las métricas . . . . .	21
2.5. Descripción de los experimentos o simulaciones . . . . .	22
2.5.1. Preprocesamiento . . . . .	22
2.5.2. Modelo Prophet . . . . .	23
2.5.3. Modelo XGBoost . . . . .	23
2.5.4. Modelo Híbrido Prophet-XGBoost . . . . .	24
2.5.5. Comparación contra modelos Arima, ETS y Promedios Históricos . . . . .	24
2.5.6. Resultados . . . . .	25
3 Resultados y discusión. . . . .	27
3.1. Resultados y discusión . . . . .	27
4 Conclusiones y trabajo futuro. . . . .	35
4.1. Conclusiones . . . . .	35
4.2. Trabajo futuro . . . . .	35

# Índice de figuras

	<b>Página</b>
2.1. Muestra de los primeros 15 renglones del conjunto de datos obtenido de un sistema ERP productivo. . . . .	15
2.2. Salidas de Stock (En piezas) a través del tiempo. . . . .	17
2.3. Salidas de Stock (En piezas) de Familias (pais origen de la pieza) a través del tiempo. . . . .	17
2.4. Salidas de Stock (En piezas) En Top 10 Grupos (Tipos de pieza) a través del tiempo. . . . .	18
3.1. Resultados de los pronósticos Prophet, XGBoost y Modelo Híbrido sin la configuración de días festivos. . . . .	27
3.2. Resultados de los pronósticos Prophet, XGBoost y Modelo Híbrido con la configuración de días festivos. . . . .	28
3.3. Resultados de los pronósticos de los diferentes modelos a nivel China. . . . .	29
3.4. Resultados de de los pronósticos de los diferentes modelos a nivel "Partes de transmisión". . . . .	30
3.5. Resultados de los pronósticos de los diferentes modelos a nivel Rodamientos, Conos y Tazas". . . . .	31
3.6. Resultados de los pronósticos de los diferentes modelos a nivel "Partes de Diferencial Meritor". . . . .	32



# Índice de tablas

	<b>Página</b>
2.1. Parámetros optimizables de XGBoost utilizando Gridsearch. . . . .	24
3.1. Métricas de error con la configuración de días festivos inactiva. . . . .	28
3.2. Comparación de modelos según métricas de error. . . . .	28
3.3. Métricas de rendimiento de los modelos para partes de transmisión. . . . .	30
3.4. Métricas de rendimiento de los modelos para rodamientos, conos y tazas. . . . .	32
3.5. Métricas de rendimiento de los modelos Partes de Diferencial Meritor. . . . .	33



*Dedicado a mi familia, mi fuente de apoyo incondicional; a mis maestros y asesores, quienes fueron mis guías en este camino.*



# 1 Introducción

## 1.1 Contexto

La industria de retail de piezas de automoviles abarca la venta de productos o servicios de forma directa a los consumidores finales, tanto para vehículos de uso personal como para vehículos comerciales. Este sector es crítico para el mantenimiento de los vehículos en circulación, ya que permite a los propietarios realizar reparaciones y mejoras sin tener que recurrir a concesionarios oficiales, y juega un papel vital en asegurar la disponibilidad de piezas y servicios esenciales, facilitando el mantenimiento y la mejora continua de los vehículos en circulación [2].

La gestión del inventario y la predicción de las ventas son fundamentales para mantener la rentabilidad y satisfacer la demanda del mercado. El exceso y la falta de inventario son dos problemas comunes que impactan negativamente a las empresas. El sobrestock genera costos adicionales de almacenamiento, riesgo de obsolescencia y deterioro de productos, además de inmovilizar capital que podría destinarse a otras inversiones. Adicionalmente, el desabastecimiento puede provocar la pérdida de ventas y afectar la satisfacción del cliente. La ausencia de piezas clave puede detener las operaciones y aumentar los costos al tener que recurrir a soluciones urgentes o adquisiciones a precios más altos. Optimizar el inventario para los clientes no sólo puede mantener las ventas, sino que también puede aumentarla [3].

La predicción de demanda es importante para cualquier empresa porque proporciona una base sólida para la toma de decisiones estratégicas y operativas. [4]. Cuando una empresa tiene disponibilidad constante de productos, los clientes confían en que siempre podrán encontrar lo que necesitan, lo que fomenta lealtad a la marca y reduce la probabilidad de que busquen opciones en la competencia. De igual manera, el tener un inventario suficiente permite a la empresa responder rápidamente a aumentos en la demanda capturando así más oportunidades de venta. Por otro lado las ventas de piezas de automóviles pueden estar sujetas a fluctuaciones regulares y estacionalidad, como aumentos en ciertas épocas del año o caídas inesperadas.

El avance de la tecnología ha permitido la integración de técnicas de Machine Learning y análisis de series de tiempo para abordar estos desafíos con mayor precisión. En este contexto, el algoritmo XGBoost ha ganado popularidad por su capacidad de manejar grandes volúmenes de datos, identificar patrones con relaciones difíciles de interpretar y realizar predicciones en diversos escenarios de negocio[5]. Patrones tales como el comportamiento de los consumidores[6] o el valor de ciertos productos en el mercado[7]. Por otro lado, los modelos de series de tiempo como Prophet son herramientas para capturar tendencias y patrones estacionales, lo que es crucial para la previsión de demanda en un entorno con fluctuaciones regulares [8].

Este estudio explora la implementación de un modelo híbrido que combina los algoritmos Prophet y XGBoost usando los datos generados desde el sistema planificación de recursos empresariales (**ERP** por sus siglas en inglés) de una empresa dedicada a la comercialización de piezas automotrices.

El objetivo es desarrollar un sistema predictivo que permita a la empresa anticipar la demanda de

productos específicos y optimizar la gestión de inventario, reduciendo costos y mejorando la satisfacción del cliente. A través de la combinación de estos algoritmos, se busca crear una solución integral que potencie la toma de decisiones estratégicas y operativas dentro de la organización.

Este documento se estructura de la siguiente manera:

- **Introducción:** En este capítulo, se presenta el contexto del problema relacionado con la gestión del inventario y la predicción de ventas en la industria automotriz. Se justifica la necesidad de implementar un sistema híbrido usando el algoritmo de serie de tiempo Prophet y el algoritmo de Machine Learning XGBoost, para mejorar la eficiencia operativa de la empresa. Además, se describen los objetivos principales del proyecto, incluyendo la optimización de inventarios en base a la predicción de demanda futura.
- **Metodología:** Este capítulo describe el enfoque metodológico seguido para llevar a cabo la investigación. Se incluye una descripción detallada de los datos utilizados, que abarca ventas históricas (registros de salida de inventario) de la empresa automotriz. Se realiza un análisis exploratorio de los datos para identificar patrones y comportamientos relevantes. Además, se explica la implementación del modelo de Series de tiempo Prophet para la descomposición de componentes que después serán usados en combinación con Machine Learning (XGBoost) para la predicción de inventarios. Posteriormente, se realiza una comparación entre los modelos previamente desarrollados y técnicas de pronóstico de series de tiempo, como ARIMA y ETS, para analizar su eficacia y precisión. También se detallan las métricas de evaluación utilizadas para medir la efectividad de los modelos.
- **Resultados y Discusión:** En este capítulo se presentan los resultados obtenidos tras la implementación de los modelos de Machine Learning y series de tiempo. Se discuten los hallazgos clave, como el rendimiento de los modelos en términos de precisión en la predicción de demanda y la mejora en la gestión del inventario. Además, se realiza un análisis sobre las implicaciones de estos resultados, destacando las ventajas y limitaciones de las soluciones propuestas.
- **Conclusiones:** En el capítulo final, se ofrecen conclusiones basadas en los resultados obtenidos en el análisis. Se discuten los beneficios de aplicar técnicas avanzadas como XGBoost y Prophet en la gestión de inventarios y la predicción de ventas. Finalmente, se proponen recomendaciones para futuras investigaciones en este campo, sugiriendo posibles mejoras en los modelos y la inclusión de nuevos enfoques para optimizar aún más los procesos de la empresa.

## 1.2 Justificación

El caso de estudio de este trabajo se enfoca en una empresa dedicada a la venta de piezas automotrices, donde las decisiones se toman principalmente con base en métodos como promedios de ventas o en la intuición empresarial. Esto a menudo genera problemas de sobrestock o desabastecimiento, lo que impacta negativamente en los costos operativos y en la satisfacción del cliente. La gestión del inventario y la predicción de ventas son dos de los desafíos más importantes en la operación de empresas dedicadas a la comercialización de piezas automotrices. Las fluctuaciones en la demanda, los ciclos de reposición de inventario y las tendencias del mercado requieren algoritmos avanzados[9] como modelos de Machine Learning o Series Temporales que permitan a las empresas anticiparse a posibles desajustes en la cadena de suministro y mejorar su eficiencia operativa. En el Machine Learning, un computador observa datos, construye un modelo basado en esos datos y utiliza ese modelo a la vez como una hipótesis acerca del mundo y una pieza de software que

puede resolver problemas [10]. Una Serie de tiempo es un conjunto de observaciones ordenadas en alguna dimensión, generalmente el tiempo [11].

El desarrollo de técnicas como Machine Learning y los modelos de series de tiempo ha abierto nuevas oportunidades para resolver estos problemas con mayor precisión y efectividad. XGBoost, un algoritmo de aprendizaje supervisado, ha demostrado su capacidad para analizar grandes volúmenes de datos y realizar predicciones más precisas en comparación con otros enfoques tales como Random Forest[12], Decision Tree[12], Catboost [12]. De manera complementaria, los modelos de series de tiempo como ARIMA y ETS son herramientas altamente eficaces para capturar patrones y tendencias a lo largo del tiempo, permitiendo predicciones más exactas sobre la venta futura que otros modelos de series de tiempo como MA y ARMA [13].

Este proyecto se justifica en la necesidad de implementar herramientas tecnológicas avanzadas que permitan a la empresa optimizar la gestión de inventarios y mejorar la precisión en la predicción de ventas. El uso de estas herramientas proporcionará una ventaja competitiva en el mercado automotriz, donde la eficiencia en la cadena de suministro y la capacidad de anticipar la demanda son determinantes para el éxito. Al aplicar técnicas de predicción basadas en machine learning y series de tiempo, la empresa podrá tomar decisiones más informadas. Este proyecto busca sentar las bases para una mayor innovación tecnológica dentro de la organización, lo que contribuirá a una mayor competitividad en el sector automotriz.

### 1.3 Problema

Por la simplicidad que representa, es común el uso de métodos básicos de pronóstico, lo que resulta insuficiente para capturar la complejidad y fluctuación de las ventas en un mercado dinámico como el de repuestos automotrices. Este enfoque no considera variaciones estacionales, tendencias de mercado, ni cambios en el comportamiento de los clientes, lo que provoca una toma de decisiones subóptima en la planificación del inventario [14].

El uso de promedios históricos tiene limitaciones críticas [15]:

- **Sobrestock:** Al no anticipar correctamente la demanda futura, la empresa frecuentemente mantiene un exceso de inventario, inmovilizando capital y generando costos innecesarios de almacenamiento, manejo y, en algunos casos, pérdidas por obsolescencia de productos.
- **Desabastecimiento:** Este enfoque puede provocar la falta de piezas necesarias para los clientes genera pérdidas de ventas, empeora la experiencia del cliente y afecta la rentabilidad de la empresa.
- **Falta de precisión en la predicción:** Los promedios simples no permiten prever fluctuaciones de la demanda causadas por factores estacionales o eventos inesperados en el mercado [16]. Esto genera predicciones inexactas que impiden ajustar de manera proactiva el stock a las necesidades reales del mercado.
- **Entregas a Tiempo:** Los proveedores pueden estar ubicados en el extranjero, lo que hace que los tiempos de entrega de las piezas varíen en tiempo dependiendo del proveedor. Anticipar correctamente la demanda es crítico, ya que cualquier error en la predicción podría ocasionar un desabastecimiento o un sobrestock.

La ausencia de un sistema avanzado que permita predecir con mayor precisión la demanda de productos y optimizar el inventario expone a la empresa a mayores riesgos financieros, operativos y de competitividad. El problema central es la falta de herramientas analíticas y predictivas que puedan proporcionar una visión más precisa y adaptativa del comportamiento del mercado. Este enfoque limitado afecta la gestión del inventario y a la eficiencia operativa general de la empresa, lo que impide su capacidad para adaptarse rápidamente a las demandas del mercado y maximizar su rentabilidad.

## 1.4 *Objetivos*

### 1.4.1 *Objetivo General*

Desarrollar un sistema predictivo basado en técnicas avanzadas de Machine Learning (XGBoost) y Prophet (Series de Tiempo) que permita a las empresas retail de piezas automotrices optimizar la gestión de su inventario en base de mejorar la precisión en la predicción de ventas.

### 1.4.2 *Objetivos específicos*

- Realizar un análisis exploratorio de los datos proporcionados por una empresa en el ramo retail automotriz para describir la base de datos, identificar patrones, tendencias, y relaciones entre las variables. Detectar valores atípicos y datos faltantes.
- Adaptar el modelo de Machine Learning XGBoost para predecir la demanda de inventario, evaluando su rendimiento. Definir las métricas de evaluación más apropiadas (como RMSE, MAE, etc.) para medir la precisión y efectividad de los modelos seleccionados en la predicción de la variable objetivo.
- Diseñar un conjunto de experimentos estructurados para entrenar y validar los modelos seleccionados, estableciendo parámetros y técnicas de validación como división de conjuntos de datos (train/test split), y ajuste de hiperparámetros para mejorar el rendimiento de los modelos.
- Realizar una comparación entre los modelos propuestos y la técnica actualmente implementada en nuestro caso de estudio (promedios mensuales históricos), junto con otras metodologías tradicionales de pronóstico de series de tiempo, para evaluar su eficacia y precisión relativa.
- Comparar los resultados obtenidos de los modelos entrenados utilizando las métricas definidas, identificando el modelo más efectivo y robusto para la predicción.
- Realizar un análisis de los errores de predicción y discutir los puntos fuertes y débiles de cada modelo, proporcionando recomendaciones sobre la implementación del modelo óptimo en un entorno real.
- Evaluar la viabilidad de implementación de los modelos desarrollados (en particular Prophet y el modelo híbrido) en empresas de retail de piezas automotrices, considerando factores como escalabilidad, costos computacionales y adaptabilidad a fluctuaciones de demanda reales. Este objetivo busca cerrar la brecha entre la validación teórica y la aplicación práctica en el sector objetivo.

## 2 Metodología

### 2.1 Descripción de los datos

Para este estudio, se empleará un conjunto de datos obtenido de un sistema ERP en operación en una empresa dedicada al sector retail de piezas automotrices. Estas fuentes proporcionan información detallada sobre las ventas en términos de unidades y valor monetario, abarcando el período comprendido entre septiembre de 2014 y septiembre de 2024. A continuación, se presenta una descripción detallada de cada uno de los campos incluidos en el conjunto de datos.

factura	fecha	cliente	cantidad	costo	subtotal	iva	importe	total	moneda	articulo	Factor US	TotalMXN	Familia	Bloque
42441	04-09-2019	10	5	56.95	284.75	45.56	284.75	330.31	DOLARES	MAT01	20	6606.2	4	28
42441	04-09-2019	10	5	52.75	263.75	42.2	263.75	305.95	DOLARES	MAT02	20	6119	4	28
42441	04-09-2019	10	5	39.95	199.75	31.96	199.75	231.71	DOLARES	MAT03	20	4634.2	4	33
42441	04-09-2019	10	5	38.95	194.75	31.16	194.75	225.91	DOLARES	MAT04	20	4518.2	4	33
42442	04-09-2019	1740	3	1443	4329	692.64	4329	5021.64	PESOS	MAT05	20	5021.64	4	9
42443	04-09-2019	1893	1	377.4	377.4	60.38	377.4	437.78	PESOS	MAT06	20	437.78	3	12
42444	04-09-2019	173	2	1.43	2.86	0.46	2.86	3.32	DOLARES	MAT07	20	66.4	1	9
42444	04-09-2019	173	12	2.35	28.2	4.51	28.2	32.71	DOLARES	MAT08	20	654.2	4	5
42444	04-09-2019	173	10	9.45	94.5	15.12	94.5	109.62	DOLARES	MAT09	20	2192.4	4	5
42444	04-09-2019	173	10	2.2	22	3.52	22	25.52	DOLARES	MAT10	20	510.4	4	5
42444	04-09-2019	173	5	11.99	59.95	9.59	59.95	69.54	DOLARES	MAT11	20	1390.8	4	9
42445	04-09-2019	1894	1	2140	2140	342.4	2140	2482.4	PESOS	MAT12	20	2482.4	3	12
42446	05-09-2019	159	20	11.49	229.8	36.77	229.8	266.57	DOLARES	MAT13	19.9	5304.743	4	3
42446	05-09-2019	159	5	14.99	74.95	11.99	74.95	86.94	DOLARES	MAT14	19.9	1730.106	4	30

Figura 2.1: Muestra de los primeros 15 renglones del conjunto de datos obtenido de un sistema ERP productivo.

- **Campos:** factura, fecha, cliente, cantidad, costo, subtotal, iva, importe, total, moneda, articulo, factor US - MXN, totalMXN, Familia y Bloque.
- **Cantidad:** 102,045 instancias.
- **Período:** Septiembre 2019 a Septiembre 2024

Donde tenemos las columnas:

**Factura:** Número de identificación único asignado a cada transacción de venta. Permite rastrear la operación dentro del sistema.

**Fecha:** Fecha en la que se realizó la venta o transacción, registrada en formato día-mes-año (DD-MM-AAAA).

**Cliente:** Código o identificación numérica asignada a cada cliente. Representa a quién se le vendieron los artículos.

**Cantidad:** Número de unidades de un artículo específico que fueron vendidas en la transacción correspondiente.

**Costo:** Precio unitario de cada artículo, en la moneda especificada (DÓLARES o PESOS), antes de aplicar impuestos.

**Subtotal:** Resultado de multiplicar la cantidad vendida por el costo unitario del artículo, sin incluir impuestos.

**IVA:** Impuesto al Valor Agregado aplicado a la transacción, calculado como un porcentaje del subtotal.

**Importe:** Valor neto de la transacción en la moneda original (ya sea DÓLARES o PESOS), excluyendo el IVA.

**Total:** Suma del subtotal más el IVA. Representa el total a pagar por el cliente en la moneda original de la transacción.

**Moneda:** Tipo de moneda en que se realizó la transacción, que puede ser en DÓLARES o PESOS.

**Artículo:** Código o identificador único asignado a cada producto o pieza vendida, lo que permite identificar el artículo específico en el sistema.

**Factor USD - MXN:** Tasa de conversión utilizada para convertir los valores de la transacción en dólares a pesos mexicanos. Varía ligeramente según la transacción.

**TotalMXN:** Total de la transacción convertido a pesos mexicanos, tomando en cuenta la tasa de conversión (Factor USD - MXN) si la transacción fue originalmente en dólares.

**Familia:** Código numérico asignado al origen del proveedor del producto (China, US, Canada, etc) .

**Bloque:** Código numérico asignado al grupo que pertenece el producto, un grupo engloba varios productos de características similares (Palancas de Cambio, Transmisión, Crucetas, Baleros, etc).

Dado que emplearemos técnicas de series de tiempo, la información original se agregará mensualmente en función de nuestro objetivo de pronóstico (familia o bloque), con el fin de obtener un formato de serie de tiempo adecuado para el entrenamiento de los modelos.

## 2.2 *Análisis exploratorio*

Es fundamental llevar a cabo un análisis exploratorio con el fin de comprender la estructura y las características particulares del conjunto de datos de ventas generado por el sistema ERP perteneciente a la empresa de retail de piezas automotrices.

En esta primera etapa, nos centramos en examinar la distribución de las variables dependientes objetivo, como la Cantidad y TotalMXN, a lo largo del tiempo para identificar sus características. Asimismo, analizaremos la distribución de la demanda de las distintas Familias y Grupos a través del tiempo. Para este análisis, emplearemos visualizaciones de datos, que nos permitirán identificar patrones, tendencias y estacionalidades en los datos. Para éste propósito emplearemos la herramienta de visualización de datos Power BI.

Éste análisis facilita una inspección visual directa de las propiedades, distribuciones y relación de las diferentes entidades que conforman el dataset.

En la figura 2.2 podemos observar el comportamiento de la demanda de piezas, donde se muestra un crecimiento sólido y constante a lo largo de los años, con algunas fluctuaciones estacionales. Desde 2019 hasta 2024, se observa un crecimiento notable en las ventas. A partir de 2021, las ventas muestran un incremento significativo año tras año, especialmente en 2023 y 2024, donde los niveles de venta superan claramente a los años anteriores. Mayo y Junio son meses de alto rendimiento a lo largo de casi todos los años, lo que podría estar vinculado a patrones estacionales o ciclos de compras importantes en la industria. Los meses

Serie de tiempo de demanda de piezas a través de los años

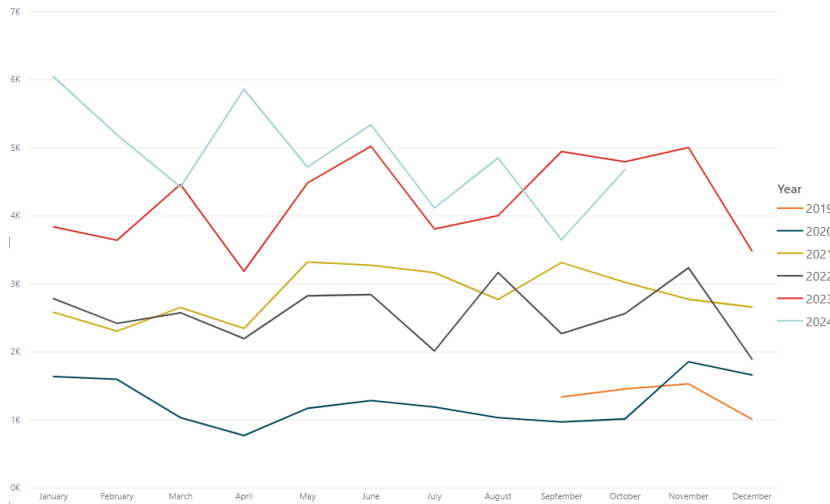


Figura 2.2: Salidas de Stock (En piezas) a través del tiempo.

de septiembre a noviembre también destacan como periodos de alta actividad, con picos recurrentes, lo que sugiere que estos meses son cruciales para las ventas. Los meses de marzo y abril, fluctuantes debido a los efectos de semana santa serán parte de nuestro objeto de estudio.

Serie de tiempo de demanda de piezas por familia a través de los años

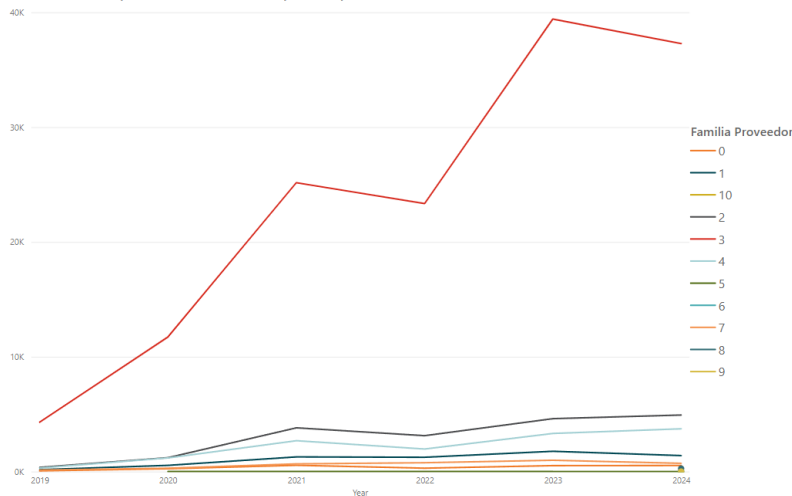


Figura 2.3: Salidas de Stock (En piezas) de Familias (pais origen de la pieza) a través del tiempo.

El gráfico 2.3 muestra la evolución de la demanda de piezas automotrices por familia de proveedores desde el año 2019 hasta 2024. A simple vista, destaca que la familia identificada como Familia 3 (línea roja, que representa proveedores asiáticos) presenta una tendencia de crecimiento significativo a lo largo de estos años. Desde 2019, esta familia muestra un incremento constante en la demanda, alcanzando su punto máximo en 2023, y luego disminuyendo ligeramente en 2024. Por otro lado, la mayoría de las demás familias presentan

una demanda estable y moderada, sin variaciones notables entre los años.

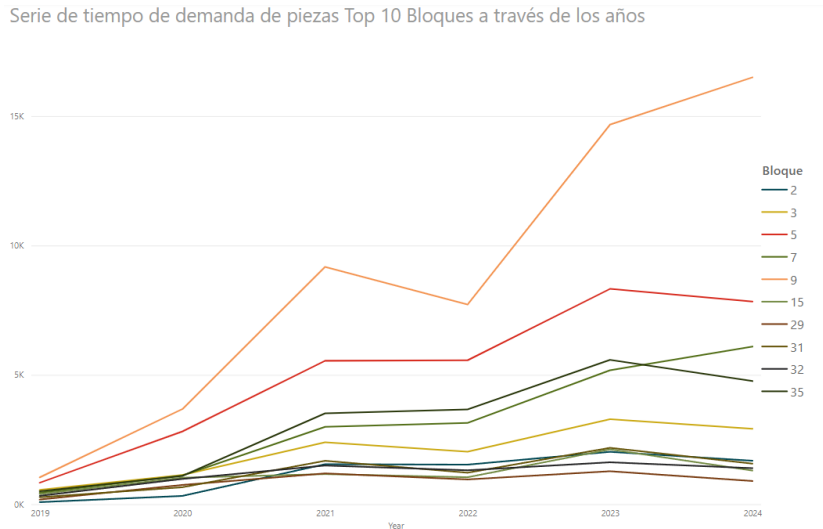


Figura 2.4: Salidas de Stock (En piezas) En Top 10 Grupos (Tipos de pieza) a través del tiempo.

La imagen 2.4 muestra una serie de tiempo de la demanda de piezas para los 10 bloques (tipo de pieza) más solicitados a través de los años, desde 2019 hasta 2024. En el eje vertical se representa la cantidad de demanda en miles (K), mientras que en el eje horizontal se representan los años. Cada línea de color diferente representa un bloque específico, identificado en la leyenda a la derecha con números del 2 al 35 (Para una mejor visibilidad se han tomado sólo el top 10 de bloques en demanda). La gráfica muestra que algunos bloques, como el bloque 9 (línea naranja) y el bloque 5 (línea roja), han experimentado un aumento significativo en la demanda a lo largo de los años, especialmente a partir de 2020. Otros bloques han tenido un crecimiento más moderado o incluso fluctuaciones en la demanda. Tomaremos los 3 bloques de mayor demanda como objeto de los experimentos.

### 2.3 Descripción de los modelos

Para llevar a cabo predicciones de la demanda de piezas basadas en los datos generados por el sistema ERP productivo de la empresa en cuestión, se ha elegido un modelo híbrido que integra dos enfoques principales: Prophet y XGBoost. Prophet, un modelo predictivo especializado en series de tiempo, es eficaz para capturar tendencias y estacionalidades. XGBoost, destaca por su capacidad para gestionar grandes volúmenes de datos e identificar relaciones entre variables. La combinación de estos modelos permite capturar una amplia gama de características, lo que otorga predicciones más precisas y fiables de la demanda futura. Se aplicarán métricas como  $R^2$  (coeficiente de determinación, que indica el grado de ajuste del modelo) y MSE (error

cuadrático medio, que mide la diferencia promedio entre los valores observados y los predichos) para evaluar la calidad de las predicciones.

### 2.3.1 Prophet Para Predicción en Series de Tiempo

Prophet es una biblioteca de código abierto desarrollada por Facebook para la predicción de series temporales[17]. Está diseñada para manejar datos con tendencias y estructuras estacionales de manera automática [17]. Prophet se enfoca a series temporales univariadas, ya que puede capturar tendencias no lineales, estacionalidad anual, semanal y diaria, así como efectos de días festivos. La implementación de Prophet se basa en un modelo aditivo que ajusta tendencias no lineales con estacionalidad anual, semanal y diaria, además de efectos de días festivos[17]. Prophet es robusto frente a datos faltantes y cambios en la tendencia, y generalmente maneja bien los valores atípicos[17].

Prophet utiliza un modelo aditivo para descomponer la serie temporal en componentes[17], como lo muestra la ecuación 2.1:

$$y(t) = g(t) + s(t) + h(t) + \epsilon_t \quad (2.1)$$

Donde:

- $y(t)$  es el valor de la serie temporal en el tiempo  $t$ .
- $g(t)$  es la función de tendencia, que modela los cambios a largo plazo en los datos.
- $s(t)$  es la función de estacionalidad, que captura los ciclos periódicos en los datos.
- $h(t)$  representa los efectos de las vacaciones.
- $\epsilon_t$  es el término de error, que captura la variabilidad no explicada por el modelo.

La tendencia  $g(t)$  puede ser modelada de diferentes maneras, como una tendencia lineal o logarítmica con puntos de cambio:

$$g(t) = (k + a(t)^T \delta)t + (m + a(t)^T \gamma) \quad (2.2)$$

Donde:

- $k$  es el cambio de crecimiento inicial.
- $m$  es el nivel inicial de la serie temporal.
- $\delta$  y  $\gamma$  son los cambios en la tasa de crecimiento y el nivel, respectivamente.
- $a(t)$  es una función indicadora que captura los puntos de cambio.

La estacionalidad  $s(t)$  se modela usando funciones trigonométricas:

$$s(t) = \sum_{i=1}^N \left[ A_i \sin \left( \frac{2\pi it}{P} \right) + B_i \cos \left( \frac{2\pi it}{P} \right) \right] \quad (2.3)$$

Donde:

- $A_i$  y  $B_i$  son los coeficientes de las funciones de seno y coseno.
- $P$  es el periodo de la estacionalidad.

Los efectos de vacaciones  $h(t)$  se modelan como variables dummy que representan los días festivos y eventos especiales.

### 2.3.2 XGBoost Para Detección De Relaciones

XGBoost (eXtreme Gradient Boosting) es una implementación de árboles de decisión impulsados por gradientes que es conocida por su velocidad y rendimiento[18]. Es una herramienta poderosa para problemas de aprendizaje supervisado, incluyendo clasificación y regresión[18]. XGBoost es altamente eficiente y flexible debido a su capacidad para manejar grandes volúmenes de datos y producir resultados de alta precisión[18]. XGBoost es una implementación de gradient boosting que utiliza árboles de decisión como base para construir modelos secuenciales que mejoran iterativamente el rendimiento del modelo[19]. La biblioteca soporta interfaces en Python, R, Java, y otros lenguajes, y ofrece características avanzadas como la regularización para evitar el sobreajuste[18].

El siguiente algoritmo de entrenamiento de XGBoost mejora las predicciones de un modelo iterativamente al construir y ensamblar múltiples árboles de decisión que corrigen los errores residuales de cada iteración anterior

---

#### Algorithm 1 Entrenamiento de XGBoost

---

- 1: Inicializar el modelo con predicciones constantes  $\hat{y}_i^{(0)}$  para todos los  $i$
  - 2: **for** cada iteración  $t$  en el número total de iteraciones **do**
  - 3:   Calcular el gradiente  $g_i^{(t)} = \frac{\partial L(y_i, \hat{y}_i^{(t-1)})}{\partial \hat{y}_i^{(t-1)}}$  para todos los datos  $i$
  - 4:   Calcular el hessiano  $h_i^{(t)} = \frac{\partial^2 L(y_i, \hat{y}_i^{(t-1)})}{\partial (\hat{y}_i^{(t-1)})^2}$  para todos los datos  $i$
  - 5:   Construir un nuevo árbol  $f^{(t)}$  minimizando la función objetivo regulada
  - 6:   Actualizar las predicciones  $\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + \eta f^{(t)}(x_i)$  para todos los  $i$
  - 7: **end for**
  - 8: Retornar el modelo final  $\hat{y}_i = \sum_t \eta f^{(t)}(x_i)$
- 

XGBoost utiliza la siguiente secuencia en su algoritmo:

**Inicialización:** XGBoost comienza creando un modelo inicial, normalmente un árbol de decisión simple, para predecir los resultados. Este modelo inicial puede estar muy lejos de ser perfecto.

**Cálculo de Errores (Gradientes):** XGBoost calcula el error residual (gradiente) entre las predicciones actuales y los valores reales. Este error residual es lo que el modelo tratará de corregir en las siguientes iteraciones.

**Construcción de Nuevos Árboles:** En cada iteración, se construye un nuevo árbol de decisión que trata de corregir los errores residuales del modelo anterior. Este nuevo árbol se entrena para predecir los errores residuales, no las respuestas originales.

**Actualización del Modelo:** Las predicciones del nuevo árbol se combinan con las predicciones del modelo

anterior para obtener nuevas predicciones. El peso de cada árbol se ajusta en función de su rendimiento.

**Iteraciones Múltiples:** Este proceso se repite muchas veces, añadiendo nuevos árboles en cada iteración y ajustando los pesos. La suma de todos los árboles (modelo ensamblado) constituye la predicción final.

**Regularización:** XGBoost incluye técnicas de regularización para evitar el sobreajuste. Esto se logra penalizando la complejidad de los árboles (por ejemplo, la profundidad del árbol).

### 2.3.3 *Modelo Híbrido Prophet-XGBoost*

Combinar Prophet y XGBoost en un modelo híbrido ofrece lo mejor de ambos mundos: la capacidad de Prophet para capturar componentes estacionales y efectos de tendencia a largo plazo, junto con la potencia predictiva y la flexibilidad de XGBoost para manejar relaciones complejas y no lineales en los datos.

#### **Justificación**

1. **Fortalezas de Prophet:** Prophet está enfocado en descomponer series temporales en componentes claros de tendencia, estacionalidad y eventos irregulares (como festivos). Esto lo hace ideal para datos con patrones estacionales claros y cambios de tendencia periódicos [17].

2. **Fortalezas de XGBoost:** XGBoost es un algoritmo de machine learning que puede manejar datos con alta dimensionalidad y relaciones complejas. Su capacidad para optimizar mediante boosting lo hace efectivo para capturar patrones no lineales y reducir el error de predicción [18].

3. **Complementariedad:** Al combinar ambos, Prophet puede descomponer la serie temporal en componentes básicos y proporcionar predicciones iniciales, mientras que XGBoost puede refinar estas predicciones corrigiendo los errores residuales y capturando patrones complejos que Prophet no puede modelar por sí solo [17, 18].

4. **Mejora del Rendimiento:** La naturaleza híbrida permite aprovechar la facilidad de interpretación y los componentes de tendencia de Prophet, al mismo tiempo que se beneficia de la precisión y capacidad de ajuste fino de XGBoost. Esto puede resultar en modelos más precisos y robustos.

5. **Flexibilidad:** Un modelo híbrido ofrece mayor flexibilidad para ajustarse a diferentes tipos de datos y escenarios. Por ejemplo, Prophet puede manejar directamente los efectos de los días festivos, mientras que XGBoost puede ajustarse dinámicamente a cualquier anomalía detectada.

En resumen, un modelo híbrido Prophet y XGBoost combina las fortalezas de ambos métodos, proporcionando un enfoque integral y robusto para la predicción de series temporales complejas.

### 2.4 *Descripción de las métricas*

Evaluar el desempeño de los modelos predictivos es crucial para garantizar la fiabilidad y precisión de las predicciones. El uso de métricas apropiadas permite no solo cuantificar los errores, sino también entender cómo se comporta el modelo en diferentes escenarios y cómo maneja la variabilidad de los datos. Las métricas nos proporcionan una visión clara y objetiva del rendimiento del modelo, ayudándonos a identificar áreas de mejora y a comparar de manera justa diferentes enfoques y algoritmos [20, 21]. Sin una evaluación rigurosa mediante métricas bien definidas, es imposible asegurar que un modelo no esté sobreajustado, subajustado o que no esté capturando correctamente los patrones relevantes en los datos. En resumen, las métricas son

herramientas esenciales que nos permiten validar la efectividad de los modelos y garantizar su robustez en el mundo real [22].

Para evaluar los modelos propuestos usaremos la métricas de error absoluto medio (MAE), error cuadrático medio (MSE), raíz error cuadrático medio (RMSE), error porcentual absoluto (MAPE) y el coeficiente de determinación (R2), donde:

**Error Absoluto Medio (MAE):** Mide la diferencia promedio entre las predicciones y los valores reales, proporcionando una interpretación directa del error en unidades reales 2.4. Fórmula:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (2.4)$$

[23]

**Error Cuadrático Medio (MSE):** Penaliza más los errores grandes debido a la elevación al cuadrado de las diferencias. Es útil para aplicaciones donde los errores grandes son especialmente indeseables 2.5. Fórmula:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (2.5)$$

[20]

**Raíz del Error Cuadrático Medio (RMSE):** Al ser la raíz cuadrada del MSE, proporciona una medida del error en las mismas unidades que los datos originales, lo que facilita la interpretación 2.6. Fórmula:

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (2.6)$$

[24]

**Error Porcentual Absoluto Medio (MAPE):** Mide el error relativo promedio en términos porcentuales, lo que facilita la comparación entre diferentes series temporales 2.7. Fórmula:

$$\text{MAPE} = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100 \quad (2.7)$$

[25]

## 2.5 Descripción de los experimentos o simulaciones

Para este proyecto se entrenaron 6 modelos diferentes alimentándolos con 37 meses de información para el conjunto de entrenamiento y 4 meses para el conjunto de prueba.

### 2.5.1 Preprocesamiento

Los experimentos inician con un preprocesamiento de datos, donde se extrajeron el año, el mes y el día a partir de la columna de fecha de factura, creando nuevas columnas en el DataFrame. Además, se convirtió una columna llamada 'period' en el índice del DataFrame con el objetivo de sumarizar la demanda por año y mes y así tener un formato de serie de tiempo para la información.

Posteriormente para el primer experimento, se filtraron los datos para la familia de Asia, que representa

la mayor parte de la demanda y es el principal objeto de estudio. Con el fin de obtener resultados más precisos y relevantes, se excluyeron los datos correspondientes al período de la pandemia, ya que este período introdujo irregularidades significativas en los patrones de demanda. Los datos se agruparon por período y mes, sumando la cantidad vendida, para obtener una visión consolidada y simplificada de las ventas históricas.

En la preparación de los datos para el análisis, se incluyó la definición de los días festivos de México que en nuestro caso de estudio tienen un impacto en la demanda (La compañía que proporcionó los datos tiene períodos largos de descanso en estas fechas), incluyendo Semana Santa y Festividades navideñas. Estos días festivos fueron utilizados como parámetros en el modelo Prophet, que los tuvo en cuenta para ajustar las fluctuaciones en la demanda, proporcionando así predicciones más precisas.

### 2.5.2 *Modelo Prophet*

El modelo Prophet se entrenó con los datos históricos desde el año 2021 y generó predicciones futuras para un período definido de 4 meses a partir de enero de 2024, es decir, se predice de febrero de 2024 a mayo de 2024. Prophet es un modelo basado en componentes aditivos que descompone la serie temporal en tres partes principales: tendencia, estacionalidad y efectos de días festivos. En este caso, se extrajeron características basadas en las predicciones de Prophet, como la tendencia (trend), que captura el crecimiento o decrecimiento general de la serie a lo largo del tiempo; los términos aditivos (additive terms), que incluyen la contribución de la estacionalidad y los días festivos; y la predicción (yhat), que es el resultado final del modelo que combina todos estos componentes.

Todas estas nuevas características, incluyendo la tendencia, los términos aditivos, y la predicción de Prophet, se integraron en el conjunto de datos para alimentar el modelo XGBoost.

### 2.5.3 *Modelo XGBoost*

Para la implementación de XGBoost. Como parte de los experimentos realizados, se diseñaron dos casos de prueba para evaluar el desempeño del modelo XGBoost. El objetivo de estos casos fue comparar los resultados obtenidos al utilizar un enfoque automatizado, donde la librería selecciona los parámetros de manera predeterminada, frente a un enfoque manual, en el cual se aplicaron técnicas de preprocesamiento y optimización de hiperparámetros para mejorar el rendimiento del modelo. En ambos casos, el conjunto de datos previamente mencionado se dividió en subconjuntos de entrenamiento y prueba, asegurando una evaluación robusta y reproducible. Asimismo, se prepararon las características (variables predictoras) y la variable objetivo (cantidad de demanda) para ambos conjuntos.

En el primer caso, el modelo XGBoost se entrenó utilizando los datos de entrenamiento sin aplicar técnicas adicionales de preprocesamiento u optimización. La librería se configuró para seleccionar automáticamente los parámetros predeterminados, lo que permitió evaluar el rendimiento base del modelo. Las predicciones se realizaron sobre el conjunto de prueba, utilizando siempre la demanda original como variable objetivo. Este enfoque sirvió como punto de referencia para comparar la efectividad de las mejoras implementadas en el segundo caso.

En el segundo caso, se implementó una metodología más rigurosa que incluyó pasos detallados de preprocesamiento y optimización de hiperparámetros. En primer lugar, se eliminaron las filas con valores faltantes (NaN) para garantizar la integridad del conjunto de datos. Posteriormente, se aplicó la técnica de estandarización

utilizando `StandardScaler()`, la cual transforma los datos para que tengan una media de 0 y una desviación estándar de 1. Este proceso es esencial cuando las características tienen diferentes escalas, con el objetivo de que el modelo converja más rápidamente y evita que variables con magnitudes mayores dominen el proceso de aprendizaje.

A continuación, se implementó el método de Análisis de Componentes Principales (PCA) para reducir la dimensionalidad del conjunto de datos, conservando el 95% de la varianza explicada. Esta técnica no solo simplifica el modelo, sino que también ayuda a eliminar redundancias y ruido en los datos, mejorando la eficiencia y el rendimiento del modelo. Luego, se realizó una búsqueda exhaustiva de los hiperparámetros más relevantes del modelo XGBoost, incluyendo `n_estimators` (número de árboles en el modelo), `max_depth` (profundidad máxima de cada árbol) y `learning_rate` (tasa de aprendizaje, que controla la contribución de cada árbol al modelo final). Estos parámetros se ajustaron mediante una búsqueda en cuadrícula (Grid Search) o validación cruzada, con el fin de encontrar la combinación óptima que maximizara la precisión del modelo. Los valores probados para cada hiperparámetro se detallan en la Tabla 2.1.

Modelo	Método	Parámetros
XGBoost	Gridsearch	- <code>n_estimators</code> : [100, 200, 300, 400, 500] - <code>max_depth</code> : [3, 5, 7, 9] - <code>learning_rate</code> : [0.01, 0.1, 0.2, 0.3]

Tabla 2.1: Parámetros optimizables de XGBoost utilizando Gridsearch.

Una vez identificada la mejor combinación de hiperparámetros, se procedió a entrenar el modelo con el conjunto de entrenamiento y a realizar predicciones sobre el conjunto de prueba.

#### 2.5.4 Modelo Híbrido Prophet-XGBoost

Finalmente dentro de nuestro algoritmo en python creamos un modelo adicional que promedia los resultados de XGBOOST Mejorado y Prophet. Este enfoque híbrido combina la capacidad de Prophet para modelar patrones temporales complejos con la flexibilidad y potencia de XGBoost para capturar relaciones no lineales y dependencias entre las características.

El proceso descrito genera un archivo en formato Excel que contiene los resultados del pronóstico correspondiente a un período de cuatro meses (febrero de 2024 a mayo de 2024) para cada uno de los modelos implementados. Posteriormente, se desarrolla un código en el lenguaje de programación R, el cual utiliza la misma información (que usó el código en python) para alimentar tres modelos adicionales: un modelo ARIMA, un modelo ETS y un modelo basado en el promedio de la demanda mensual.

#### 2.5.5 Comparación contra modelos Arima, ETS y Promedios Históricos

En el código desarrollado en R, se inicia con la lectura de la información histórica, la cual es agregada mensualmente para transformarla en una estructura de serie de tiempo. A partir de esta estructura, se visualiza la demanda histórica del inventario y se procede a realizar una descomposición STL (Seasonal and Trend decomposition using Loess) con el objetivo de identificar patrones estacionales y tendencias presentes en los datos.

En el código se implementan tres modelos principales:

- **Modelo ARIMA Automático:** Se ajusta un modelo ARIMA a la serie de tiempo y se generan pronósticos

para los siguientes cuatro meses.

- **Modelo ETS Automático:** Se ajusta un modelo ETS (Error, Trend, Seasonality) y se generan pronósticos para el mismo horizonte temporal.
- **Promedios Mensuales Históricos:** Se calculan los promedios mensuales de la demanda histórica, los cuales se utilizan para pronosticar la demanda futura. Este método corresponde al enfoque actualmente empleado en el caso de estudio.

### 2.5.6 Resultados

Una vez obtenidos los pronósticos mediante el código, el objetivo principal es comparar los resultados generados por los distintos modelos. Adicionalmente, el código en R está diseñado para leer los pronósticos previamente generados en Python, con el fin de calcular métricas de error que permitan evaluar el desempeño predictivo. Las métricas utilizadas incluyen:

- Error Absoluto Medio (MAE),
- Error Cuadrático Medio (MSE),
- Raíz del Error Cuadrático Medio (RMSE),
- Error Porcentual Absoluto Medio (MAPE).

Estas métricas se calculan tanto para los modelos de series de tiempo implementados en R como para los pronósticos generados en Python, lo que permite realizar una comparación integral entre los diferentes enfoques y evaluar su eficacia en la predicción de la demanda. Finalmente visualizamos las diferentes predicciones en un gráfico permitiendo así una comparación integral entre los diferentes enfoques y la evaluación de su desempeño predictivo.

Estos algoritmos se ejecutaron de manera iterativa para predecir la demanda correspondiente a cuatro categorías específicas: por un lado, la demanda proveniente de China, y por otro, la demanda de las tres piezas con mayor volumen de solicitudes, entre las cuales se incluyen los componentes de transmisión Fuller y Clark, los rodamientos (conos y tazas) y los elementos de diferencial Meritor.



## 3 Resultados y discusión

### 3.1 Resultados y discusión

Se realizaron experimentos en diversos niveles de granularidad de la información y configuraciones de días festivos. El primer experimento se llevó a cabo a nivel de importaciones generales procedentes de China, abarcando la totalidad de las piezas existentes en esta categoría. Posteriormente, se ejecutaron experimentos focalizados en los tres grupos principales de mayor demanda, los cuales incluyen: componentes de transmisión Fuller y Clark, rodamientos (conos y tazas) y elementos de diferencial Meritor.

En el primer experimento, se incorporaron pruebas con la configuración de días festivos habilitada y deshabilitada en el modelo Prophet, con el objetivo de evaluar el efecto de ajustar los días festivos correspondientes a la Semana Santa. Este experimento se centró en esta festividad debido a que la compañía generadora de la información no realiza operaciones durante este período, además de que la fecha de esta celebración varía entre marzo y abril dependiendo del año. En la Figura 3.1 se puede observar el comportamiento de los modelos con la configuración de días festivos deshabilitada, mientras que en la Tabla 3.1 se presentan las métricas resultantes de esta configuración cuando se encuentra habilitada.

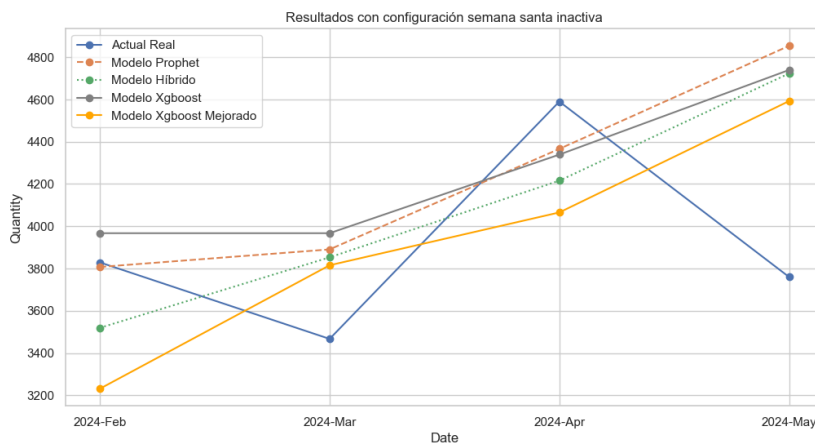


Figura 3.1: Resultados de los pronósticos Prophet, XGBoost y Modelo Híbrido sin la configuración de días festivos.

Por otro lado, al utilizar la propiedad de días festivos y alimentar el modelo Prophet con los datos históricos de las Semanas Santas desde el año 2019 hasta 2024, se obtienen los resultados mostrados en la Figura 3.2 y las métricas presentadas en la Tabla 3.2. En estos resultados se observa una mejora significativa en la mayoría de las métricas evaluadas, con excepción del modelo XGBoost. Además, visualmente se aprecia

Modelo	MAE	MSE	RMSE	MAPE
Modelo Prophet	440.68	356567.9	597.13	%11.68
Modelo XGBOOST	467.18	322927.1	568.26	%12.38
Modelo XGBOOST Mejorado	575.76	361658.5	601.38	%14.80
Modelo Híbrido	508.22	328092.7	572.79	%13.24

Tabla 3.1: Métricas de error con la configuración de días festivos inactiva.

que todos los modelos se ajustan mejor a las fluctuaciones de la demanda real, capturando tanto los valles como los picos de manera más precisa en comparación con los resultados obtenidos cuando no se incluye esta opción. En este último caso, las fluctuaciones del modelo no reflejan de manera adecuada el comportamiento de la demanda real.

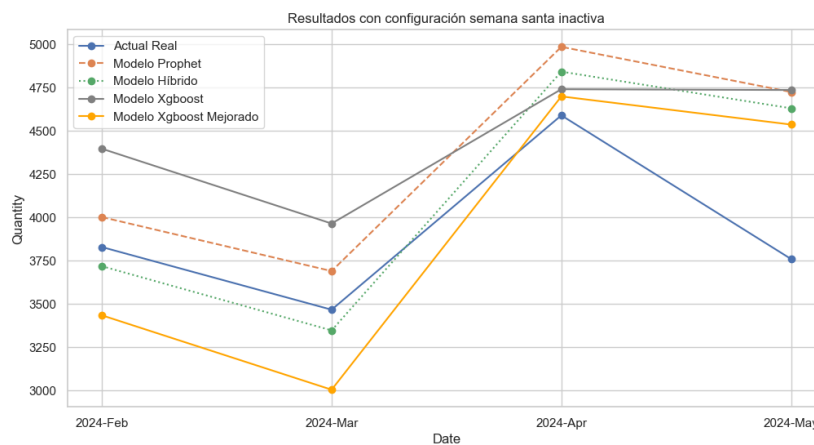


Figura 3.2: Resultados de los pronósticos Prophet, XGBoost y Modelo Híbrido con la configuración de días festivos.

Modelo	MAE	MSE	RMSE	MAPE
Modelo Prophet	439.33	291926.6	540.30	%11.31
Modelo XGBOOST	548.43	386844.3	621.96	%14.61
Modelo XGBOOST Mejorado	435.78	246216	496.20	%11.67
Modelo Híbrido	338.45	212221.4	460.67	%8.75

Tabla 3.2: Comparación de modelos según métricas de error.

A partir de lo anterior, se decidió utilizar la configuración de días festivos con el objetivo de capturar su efecto sobre la demanda. De esta manera, se llevó a cabo el experimento completo considerando los modelos mencionados, además de incorporar otros modelos de series de tiempo, como ARIMA y ETS automático. Los resultados obtenidos para toda la demanda proveniente de china se presentan en la Figura 3.3, junto con las métricas de error correspondientes, las cuales se detallan en la Tabla ??.

Al analizar la imagen 3.3, se observa que los modelos Prophet e Híbrido presentan un alto grado de ajuste a los valores reales, siguiendo de manera precisa tanto los picos como los valles de la serie de datos y

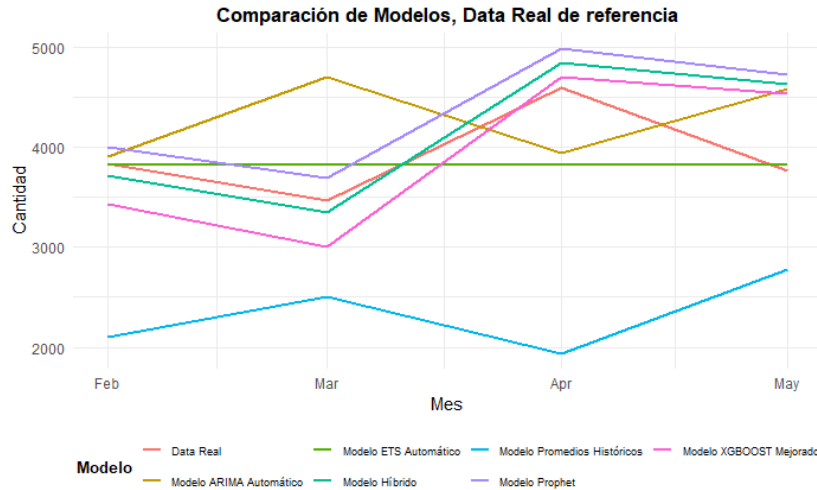


Figura 3.3: Resultados de los pronósticos de los diferentes modelos a nivel China.

replicando su fluctuación de forma consistente. Por otro lado, los modelos ARIMA y de Promedios Históricos muestran desviaciones considerables, lo que indica un menor nivel de precisión en la captura de la tendencia y variabilidad de los datos reales. Con base en los resultados de las métricas obtenidas para los diferentes modelos de pronóstico de demanda de inventario, se pueden hacer algunas observaciones clave. El Modelo Híbrido es el que muestra el mejor rendimiento en términos de las métricas de error, con un MAE bajo (338), un RMSE relativamente bajo (460.67) y un MAPE de solo 8.75 %, lo que sugiere que este modelo es muy preciso en sus predicciones. De manera similar, el Modelo ETS Automático también tiene un desempeño destacado, con un MAE de 298, un RMSE de 423 y un MAPE de 7.20 %, lo que lo coloca como otro modelo fuerte. El Modelo Prophet, por su parte, también tiene un rendimiento aceptable, con un MAE de 439 y un MAPE de 10.23 %, lo que lo convierte en una opción confiable, aunque no tan precisa como el híbrido o el modelo ETS. El Modelo ARIMA Automático, con un MAE de 693.53 y un MAPE de 18.34 % indica que no es adecuado para este conjunto de datos. Finalmente, el Modelo de Promedios Históricos, con un MAE de 1579 y un MAPE alarmantemente alto de 39.14 %, claramente no es una opción viable para predecir la demanda de inventario de manera precisa. En resumen para el experimento de pronóstico de demanda de China el Modelo Híbrido y el Modelo ETS Automático son los que muestran el mejor desempeño, con errores más bajos y predicciones más confiables.

En el segundo experimento, centramos nuestro análisis en la predicción de la demanda de las piezas denominadas "Partes de transmisión", las cuales también tienen su origen en China. En la imagen 3.4 se pueden observar los pronósticos generados por los distintos modelos evaluados. La tabla 3.3 presenta las métricas obtenidas para este caso de estudio en particular.

En la Figura 3.4 se observa que todos los modelos, a excepción del ETS automático, presentan un comportamiento similar, ajustándose a la varianza y mostrando picos y valles en los mismos meses. En particular, tanto Prophet como ARIMA demuestran un buen ajuste y capacidad predictiva para los meses de febrero y marzo. No obstante, para el mes de abril, todos los modelos subestimaron la demanda real. Posteriormente, en el mes de mayo, los modelos Híbrido y ARIMA logran realizar estimaciones más precisas,

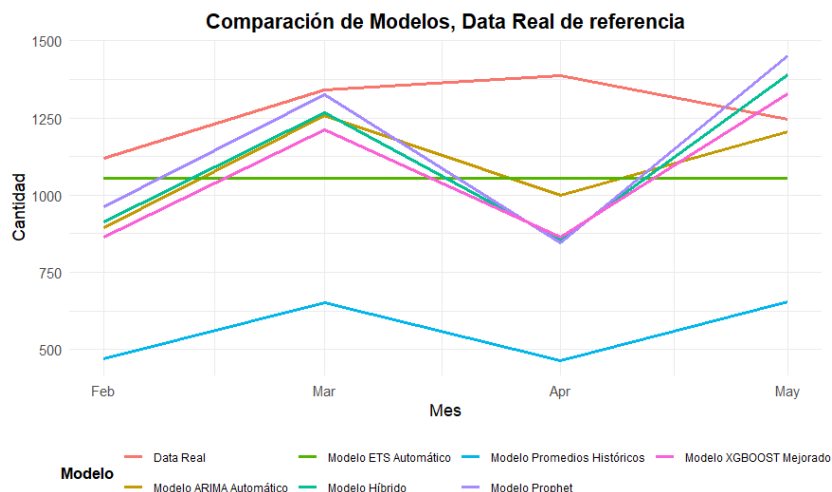


Figura 3.4: Resultados de de los pronósticos de los diferentes modelos a nivel "Partes de transmisión".

Modelo	MAE	MSE	RMSE	MAPE (%)
Modelo Prophet	230.60	89986.86	299.98	%17.75
Modelo XGBOOST	239.05	92501.30	304.14	%18.61
Modelo XGBOOST Mejorado	247.78	90597.50	300.99	%19.22
Modelo Híbrido	239.19	87879.47	296.44	%18.49
Modelo ARIMA Automático	182.83	51903.52	227.82	%14.28
Modelo ETS Automático	217.61	57863.64	240.55	%16.53
Modelo Promedios Históricos	713.83	525441.83	724.87	%55.92

Tabla 3.3: Métricas de rendimiento de los modelos para partes de transmisión.

Nota: MAE = Error Absoluto Medio, MSE = Error Cuadrático Medio, RMSE = Raíz del Error Cuadrático Medio, MAPE = Error Porcentual Absoluto Medio.

alineándose mejor con los valores observados. A partir de los resultados de las métricas, se puede inferir que el Modelo ARIMA Automático es el más preciso para predecir la demanda de partes de transmisión, con los menores errores en todas las métricas (MAE = 182.83, MSE = 51903.52, RMSE = 227.82, MAPE = 14.28 %), seguido por el Modelo ETS Automático, que también muestra un desempeño robusto (MAE = 217.61, MSE = 57863.64, RMSE = 240.55, MAPE = 16.53 %). El Modelo Híbrido y el Modelo Prophet presentan resultados moderados, con errores ligeramente superiores, pero aún competitivos. Por otro lado, los modelos basados en árboles (XGBOOST y XGBOOST Mejorado) no superan a los modelos clásicos de series de tiempo, con errores más altos (MAE 239-247, MSE 90597-92501, RMSE 300-304, MAPE 18-19 %). Finalmente, el Modelo de Promedios Históricos muestra el peor desempeño (MAE = 713.83, MSE = 525441.83, RMSE = 724.87, MAPE = 55.92 %), lo que confirma su inadecuación para capturar los patrones de la demanda. En conclusión, los modelos ARIMA y ETS son los más efectivos para este tipo de datos, superando a los enfoques basados en machine learning y promedios históricos.

En el tercer experimento, se realizó el pronóstico de la demanda para Rodamientos, Conos y Tazas. En la Figura 3.5 se observa el comportamiento de los modelos aplicados. En la gráfica, es evidente que en los modelos propuestos (Hbrido, Prophet, and XGBoost) muestran a mejor ajuste a los valles y picos de los datos reales, capturando de manera más precisa las variaciones en la demanda. Por el contrario, los modelos

ARIMA, ETS y promedios históricos no logran adaptarse adecuadamente a la información real, mostrando un desempeño inferior en términos de ajuste y precisión.

A pesar del buen ajuste que muestran los modelos Híbrido, Prophet y XGBoost a la variabilidad de la demanda, es importante destacar que existen pronósticos que no alcanzan una precisión total, como se observa en los meses de febrero y mayo. En estos periodos, las predicciones presentan desviaciones respecto a los valores reales, lo que sugiere que, aunque estos modelos capturan mejor las tendencias y fluctuaciones en comparación con otros enfoques, aún enfrentan desafíos para predecir con exactitud en ciertos momentos específicos. De las métricas obtenidas de este pronóstico representadas en la tabla 3.4. El Modelo XGBOOST Mejorado es el que presenta el mejor rendimiento general, con los valores más bajos en todas las métricas (MAE = 57.79, MSE = 5541.54, RMSE = 74.44, MAPE = 11.68 %), lo que indica que sus predicciones son las más precisas y cercanas a los valores reales. Le sigue en desempeño el Modelo Híbrido, que también muestra métricas relativamente bajas (MAE = 77.37, MSE = 7898.86, RMSE = 88.88, MAPE = 15.86 %), sugiriendo un buen ajuste a los datos. Por otro lado, el Modelo Promedios Históricos es el menos preciso, con errores significativamente más altos en todas las métricas (MAE: 163.25, MSE: 38142.8, RMSE: 195.30 y MAPE: 31.89 %), lo que refleja su limitada capacidad para capturar la variabilidad de la demanda. Los modelos Prophet, XGBOOST, ARIMA Automático y ETS Automático se sitúan en un rango intermedio, con errores moderados pero superiores a los del Modelo XGBOOST Mejorado y el Modelo Híbrido.

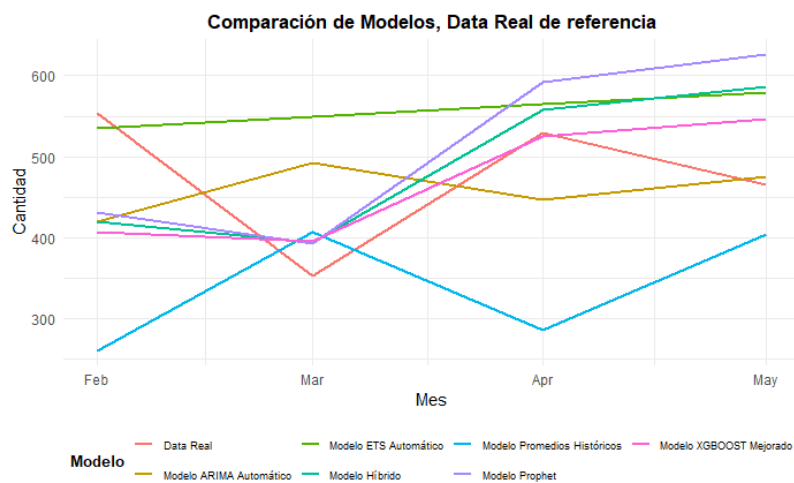


Figura 3.5: Resultados de los pronósticos de los diferentes modelos a nivel de odamentos, Conos y Tazas".

En el cuarto y último experimento, se llevó a cabo el pronóstico de la demanda para Partes de Diferencial Meritor. En la Figura 3.6 se observa un comportamiento muy similar al del experimento anterior, donde los modelos propuestos (Híbrido, Prophet y XGBoost) demuestran un mejor ajuste a los datos reales, capturando de manera más precisa las variaciones en la demanda. Por otro lado, los modelos ARIMA, ETS y Promedios Históricos continúan mostrando dificultades para adaptarse a la variabilidad de la demanda, lo que resulta en pronósticos con errores más significativos en comparación con los modelos propuestos. En las métricas obtenidas mostradas en la imagen 3.5 apreciamos que los Modelos Prophet y XGBOOST, tienen un rendimiento superior en comparación con los modelos estadísticos tradicionales, como el Modelo ARIMA Automático y el

Modelo	MAE	MSE	RMSE	MAPE (%)
Modelo Prophet	96.94	11676.88	108.06	%20.05
Modelo XGBOOST	94.01	12669.53	112.56	%18.44
Modelo XGBOOST Mejorado	57.79	5541.54	74.44	%11.68
Modelo Híbrido	77.37	7898.86	88.88	%15.86
Modelo ARIMA Automático	91.94	11145.94	105.57	%20.44
Modelo ETS Automático	91.35	13359.75	115.58	%22.61
Modelo Promedios Históricos	163.25	38142.86	195.30	%31.89

Tabla 3.4: Métricas de rendimiento de los modelos para rodamientos, conos y tazas.

Nota: MAE = Error Absoluto Medio, MSE = Error Cuadrático Medio, RMSE = Raíz del Error Cuadrático Medio, MAPE = Error Porcentual Absoluto Medio.

Modelo ETS Automático. El Modelo Prophet destaca por tener el menor error absoluto (MAE = 118.8446) y el menor error porcentual (MAPE = 20.00716%), lo que lo convierte en la mejor opción para minimizar errores absolutos y relativos. Por otro lado, el Modelo XGBOOST presenta el menor error cuadrático (MSE = 19283.03) y la menor raíz del error cuadrático (RMSE = 138.8633), siendo más adecuado para reducir errores grandes en las predicciones. En contraste, el Modelo Promedios Históricos es el menos preciso en todas las métricas, con valores significativamente más altos en MAE, MSE, RMSE y MAPE, lo que indica su ineficacia para tareas de predicción complejas. Los modelos híbridos y mejorados, como el Modelo Híbrido y el Modelo XGBOOST Mejorado, ofrecen un rendimiento intermedio, superando a los modelos estadísticos pero sin alcanzar el nivel de los mejores modelos basados en aprendizaje automático.

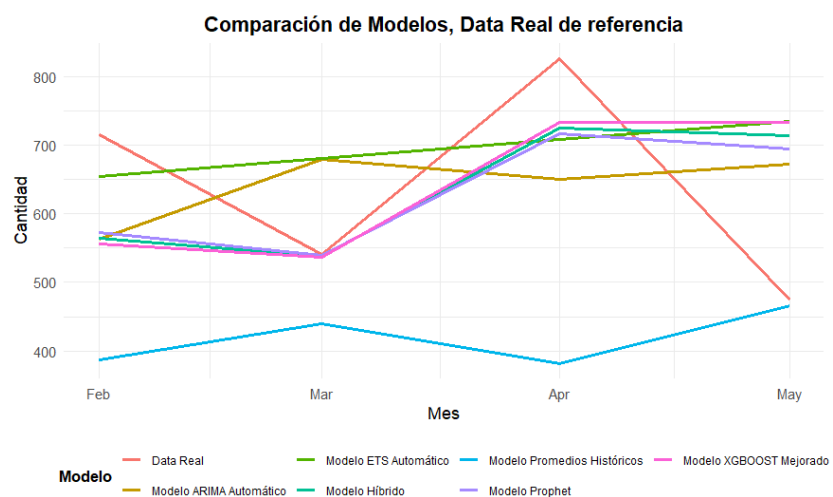


Figura 3.6: Resultados de los pronósticos de los diferentes modelos a nivel "Partes de Diferencial Meritor".

Los cuatro experimentos realizados permiten extraer conclusiones relevantes sobre el desempeño de los modelos de pronóstico aplicados a diferentes categorías de demanda. En primer lugar, se destaca que la inclusión de días festivos (Semana Santa) en el modelo Prophet mejoró significativamente la precisión, especialmente en la captura de fluctuaciones estacionales, como se evidenció en el primer experimento. Sin embargo, un hallazgo crítico fue que ningún modelo logró replicar adecuadamente el comportamiento real observado entre marzo y abril en el segundo experimento (Partes de transmisión), lo que sugiere la presencia

Modelo	MAE	MSE	RMSE	MAPE (%)
Prophet	118.84	20261.39	142.34	%20.00
XGBOOST	121.67	19283.03	138.86	%20.48
XGBOOST Mejorado	128.89	25315.56	159.11	%22.20
Híbrido	123.87	22660.14	150.53	%21.10
ARIMA Automático	166.29	28172.02	167.85	%27.45
ETS Automático	145.25	26348.00	162.32	%25.92
Promedios Históricos	220.92	79231.92	281.48	%30.05

Tabla 3.5: Métricas de rendimiento de los modelos Partes de Diferencial Meritor.

Nota: MAE = Error Absoluto Medio, MSE = Error Cuadrático Medio, RMSE = Raíz del Error Cuadrático Medio, MAPE = Error Porcentual Absoluto Medio.

de factores externos.

Por otro lado, el modelo ETS Automático mostró un comportamiento atípico en comparación con los demás, siendo menos consistente en su ajuste a la variabilidad de los datos. En contraste, el modelo Híbrido demostró robustez en múltiples escenarios, destacándose en la predicción de la demanda agregada y en categorías específicas como Rodamientos, donde el XGBoost Mejorado también sobresalió.

Finalmente, el modelo de Promedios Históricos) fue el menos preciso, subrayando la ventaja de enfoques más avanzados como Prophet y modelos híbridos. No obstante, la heterogeneidad en los resultados por categoría (p. ej., ARIMA tuvo un buen desempeño en Partes de transmisión pero no en Rodamientos) refuerza la necesidad de evaluar caso por caso y considerar la combinación de múltiples técnicas para mejorar la precisión en contextos con dinámicas complejas.



## 4 Conclusiones y trabajo futuro

### 4.1 Conclusiones

Recordemos que el modelo de promedios históricos se incluyó en el análisis debido a que representa la técnica actualmente utilizada para predecir la demanda de inventario en la compañía de donde se obtuvieron los datos. Sin embargo, en todos los casos evaluados, este modelo demostró un desempeño de baja calidad, con un margen de error significativamente alto en comparación con los demás enfoques.

En general, los modelos basados en técnicas como XGBOOST y Prophet son competitivos para la predicción de la demanda en los escenarios analizados, mientras que los enfoques tradicionales como ARIMA, ETS y Promedios Históricos tienden a ser menos precisos en los experimentos realizados. Esto resalta la importancia de utilizar métodos modernos y adaptativos para mejorar la precisión de las predicciones en contextos con alta variabilidad y complejidad en los datos. El Modelo Prophet y el Modelo Híbrido han demostrado consistentemente un rendimiento superior en la mayoría de los experimentos, presentando errores más bajos, una mayor precisión en las predicciones y un mejor ajuste a las fluctuaciones de la demanda real en comparación con los demás métodos evaluados. Por otro lado, los modelos XGBOOST y XGBOOST Mejorado han exhibido un desempeño intermedio, con métricas que, si bien no alcanzan los niveles de los modelos más robustos, superan consistentemente a los enfoques tradicionales como ARIMA, ETS y los Promedios Históricos. Estos resultados sugieren que, aunque poseen una capacidad aceptable para adaptarse a los patrones reales de demanda, podrían optimizarse mediante ajustes más finos en sus parámetros y un procesamiento más riguroso de los datos de entrada.

Cabe destacar que, durante el proceso de experimentación, fue necesario realizar un cuidadoso tratamiento de los rangos temporales para garantizar la validez de los resultados. En primer lugar, se depuraron los datos correspondientes al período inicial de operaciones, caracterizado por demandas altamente irregulares. Posteriormente, se consideraron los efectos atípicos causados por eventos externos, como la pandemia de COVID-19, a fin de evitar sesgos en las predicciones. Este enfoque metodológico permitió mejorar la robustez de los modelos y asegurar que las estimaciones reflejen de manera más fiel el comportamiento real de la demanda.

### 4.2 Trabajo futuro

En base a los experimentos realizados, se pudo observar que existe una relación directa entre la cantidad de información disponible y la precisión de las predicciones, evidenciándose que a mayor volumen de datos, menor es el error en los resultados. Cabe destacar que el primer experimento consideró la totalidad de la información relacionada con la demanda de partes importadas desde China. Posteriormente, se llevaron a cabo pruebas específicas con piezas ordenadas de acuerdo a su nivel de demanda, de mayor a menor, donde se

observó un incremento gradual en el error a medida que se reducía la cantidad de datos utilizados. Esto sugiere que la precisión de las predicciones está directamente influenciada por la cantidad de información disponible. Como punto de consideración para futuras investigaciones, se recomienda replicar estos experimentos con un mayor volumen de datos para validar y fortalecer estos hallazgos.

Asimismo, como parte del trabajo futuro, sería ideal incorporar métricas que permitan evaluar cómo podría mejorar la rotación del inventario a lo largo del tiempo haciendo la planeación de la demanda considerando las predicciones de los modelos propuestos. Entre estas métricas, se proponen aquellas que brinden información relevante sobre el manejo del inventario, específicamente en lo que respecta al sobrestock y la falta de stock. Algunas de las métricas clave a considerar, junto con sus fórmulas, son:

### *Métricas Propuestas*

**1. Rotación de Inventario (Inventory Turnover)[26]:**

$$\text{Rotación de Inventario} = \frac{\text{Costo de los Bienes Vendidos (COGS)}}{\text{Inventario Promedio}}$$

**2. Costo de Mantenimiento de Inventario (Holding Cost)[27]:**

$$\text{Costo de Mantenimiento} = \text{Inventario Promedio} \times \text{Costo de Mantenimiento por Unidad}$$

**3. Tasa de Agotamiento de Stock (Stockout Rate)[28]:**

$$\text{Tasa de Agotamiento} = \left( \frac{\text{Número de Veces que se Agotó el Stock}}{\text{Número Total de Períodos}} \right) \times 100$$

Estas métricas no solo permitirían monitorear la eficiencia del inventario, sino también identificar áreas de oportunidad para optimizar su gestión.

## Bibliografía

- [1] R. Hyndman and G. Athanasopoulos, *Forecasting: Principles and Practice*. OTexts, 2018. Accessed: September 6, 2024.
- [2] S. K. Bernd Heid, Christian Huth and G. Wu, "Ready for inspection: The automotive aftermarket in 2030," 2018.
- [3] R. H. Ballou, *Logística. Administración de la cadena de suministros*. México: Pearson Educación, 5th ed., 2004.
- [4] Lean Manufacturing 10, "Previsión de la demanda: Importancia y métodos para realizarla," 2024.
- [5] M. Javaregowda, P. B. P. Kumar, P. J, and P. G, "Inventory management using machine learning," *International Journal of Engineering and Technical Research*, vol. 9, 2020.
- [6] K. Cai and M. R. Rodavia, "Xgboost analysis based on consumer behavior," 2023.
- [7] B. Ma and X. Shi, "Application of machine learning-based xgboost model for value assessment in the used sailboat trading market," 2023.
- [8] S. B. Tarle, P. P. Paulkar, K. G. Phad, S. V. Jadhav, A. Y. Chaudhari, P. D. Patil, and P. Ahire, "Machine learning approach to predict trends in future for options," in *Machine Learning Approach to Predict Trends in Future for Options*, pp. 1–7, 2023.
- [9] M. Elorza, E. E. Castellano, and S. S. Querol, "Prediction of customer demand for perishable products in retail inventory management, using the hybrid prophet-xgboost model during the post-covid-19 period," *Applied Economics Letters*, pp. 1–7, 2024.
- [10] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. Pearson, 4th ed., 2021. In English.
- [11] O. D. Anderson, *Time Series Analysis and Forecasting*. Butterworth-Heinemann, 1977.
- [12] E. Hao, J. Zhou, and H. Zhang, "An example study of applications based on multiple machine learning algorithms," 2023.
- [13] C. Chen, "A method for sales prediction based on time series," 2022.
- [14] . A. G. Hyndman, R. J., "Forecasting: Principles and practice," 2018.
- [15] P. D. F. . T. D. J. Silver, E. A., "Inventory and production management in supply chains," 2016.
- [16] W. S. C. . H. R. J. Makridakis, S., "Forecasting: Methods and applications.," 1998.

- [17] S. J. Taylor and B. Letham, "Forecasting at scale," *The American Statistician*, vol. 72, pp. 37–45, feb 2018. Open-source time series forecasting tool.
- [18] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, 2016. Efficient and scalable implementation of gradient boosting.
- [19] J. Brownlee, "A gentle introduction to xgboost for applied machine learning," 2016. Comprehensive introduction to XGBoost for practical applications.
- [20] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.
- [21] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.
- [22] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2009.
- [23] C. J. Willmott and K. Matsuura, "Advantages of the mean absolute error (mae) over the root mean square error (rmse) in assessing average model performance," *Climate Research*, vol. 30, no. 1, pp. 79–82, 2005.
- [24] R. J. Hyndman and A. B. Koehler, "Another look at measures of forecast accuracy," *International Journal of Forecasting*, vol. 22, no. 4, pp. 679–688, 2006.
- [25] A. de Myttenaere, B. Golden, B. Le Grand, and F. Rossi, "Mean absolute percentage error for regression models," *Neurocomputing*, vol. 192, pp. 38–48, 2016.
- [26] P. D. F. . P. R. Silver, E. A., "Inventory management and production planning and scheduling.," 1998.
- [27] . M. P. Chopra, S., "Supply chain management: Strategy, planning, and operation.," 2016.
- [28] S. Nahmias, "Production and operations analysis.," 2009.