Escuela de Negocios
Tipo de documento: Tesis de maestría

Master in Management + Analytics

Forecast de demanda y optimización de la distribución de productos lácteos

Autoría: Quintana, Agustín Matías
Año: 2025

UNIVERSIDAD TORCUATO DI TELLA

MASTER IN MANAGEMENT + ANALYTICS

FORECAST DE DEMANDA Y OPTIMIZACIÓN DE LA DISTRIBUCIÓN DE PRODUCTOS LÁCTEOS

TESIS

Agustín Matías Quintana

Mayo 2025

Tutor: Magdalena Cornejo

Resumen

El mercado de bienes de consumo masivo es altamente competitivo, lo que obliga a todas las partes involucradas en la cadena de suministro a trabajar en conjunto para sostener las ventas y garantizar la posición de liderazgo de la empresa. En este contexto, el rol de los contratistas de distribución va más allá del simple transporte de mercadería; también implica asegurar que la demanda de los clientes se satisfaga de manera eficiente para evitar la pérdida de *market share*.

Esta tesis se centra en la previsión de la demanda de productos lácteos a nivel de cliente y semana, utilizando datos históricos de ventas de 2023 y 2024. La empresa en estudio, una compañía contratista de distribución, enfrenta desafíos relacionados con la volatilidad en los pedidos, lo que puede generar pérdidas de ventas y costos innecesarios en la logística. A través de modelos de *machine learning*, en particular *Random Forest* y *XGBoost*, se busca mejorar la precisión del pronóstico de demanda.

El trabajo se desarrolla en dos etapas. En la primera, se entrenan modelos para predecir la cantidad de productos solicitados por cada cliente de manera semanal, permitiendo una mejor gestión del inventario y mitigando el riesgo de pérdida de clientes frente a la competencia. En la segunda etapa, se analiza el impacto de la variabilidad en la demanda sobre los costos de distribución. La optimización del pronóstico permitirá mejorar la planificación de los envíos en camión, reduciendo la necesidad de viajes adicionales y evitando días con carga insuficiente.

Al final del estudio, se evaluará el impacto de la gestión optimizada en la distribución mediante métricas comparativas que medirán los beneficios económicos y la reducción en la variabilidad de los pedidos. Se analizarán escenarios hipotéticos en los que la planificación propuesta se hubiera aplicado, contrastándolos con la gestión real. De esta manera, los resultados de la investigación contribuirán a una gestión más eficiente de la distribución de productos lácteos, proporcionando herramientas para la toma de decisiones basadas en datos y mejorando la rentabilidad operativa de la empresa.

Abstract

The fast-moving consumer goods market is highly competitive, requiring all stakeholders in the supply chain to collaborate in maintaining sales and ensuring the company's market leadership. In this context, the role of distribution contractors extends beyond merely transporting goods; it also involves ensuring that customer demand is met efficiently to prevent market share losses.

This thesis focuses on forecasting dairy product demand at the customer and weekly levels using historical sales data from 2023 and 2024. The studied company, a distribution contractor, faces challenges due to order volatility, leading to potential sales losses and unnecessary logistics costs. By leveraging machine learning models, particularly Random Forest and XGBoost, this research aims to improve demand forecasting accuracy.

The study is divided into two phases. The first phase involves training models to predict the quantity of products requested by each customer on a weekly basis, improving inventory management and mitigating the risk of losing customers to competitors. The second phase analyzes the impact of demand variability on distribution costs. Enhanced forecasting accuracy will allow for better truck dispatch planning, reducing the need for extra trips and avoiding underutilized capacity on certain days.

At the end of the study, the impact of the optimized distribution management will be evaluated using comparative metrics that measure the economic benefits and the reduction in order variability. Hypothetical scenarios where the proposed planning would have been implemented will be contrasted with actual management practices. In this way, the findings of this research will contribute to more efficient dairy product distribution, providing data-driven decision-making tools and improving the company's operational profitability.

A mis padres, Rodrigo y Laura,

y a mis abuelos, Norberto y María,

por enseñarme, con su ejemplo, que el trabajo y el esfuerzo son el camino al progreso.

Índice de Contenidos

Índice de Tablas

Índice de Figuras

1. Introducción

1.1. Contexto

En la industria de consumo masivo, la distribución eficiente de productos es esencial para mantener la competitividad. En este escenario, la logística desempeña un papel clave no solo en la entrega de mercadería, sino también en la sostenibilidad del negocio. Los fleteros, actores fundamentales dentro de este ecosistema, se encargan del transporte y distribución de productos, percibiendo comisiones en función de la facturación neta.

En particular, la logística de productos lácteos presenta desafíos adicionales relacionados con la conservación de la cadena de frío, la alta rotación y la fuerte dependencia de clientes mayoristas (Ministerio de Agricultura, Ganadería y Pesca, 2001; EmergentCold LatAm, 2025). Este trabajo se enfoca en la zona de reparto asignada a Villa Fiorito (partido de Lomas de Zamora, Buenos Aires), donde se desempeña el fletero analizado.

La operatoria incluye el retiro diario de mercadería desde la planta fabril (ubicada en Longchamps) y su posterior distribución entre los clientes de la zona. Según la demanda, pueden despacharse uno o más camiones: en jornadas de baja demanda, los vehículos circulan parcialmente cargados, mientras que, en días de alta demanda, puede requerirse hasta cuadruplicar los envíos. Esta variabilidad provoca ineficiencias logísticas y costos operativos elevados.

En este contexto, prever la demanda no solo permite planificar las entregas de manera más eficiente, sino también optimizar el uso de los camiones, reducir gastos innecesarios y disminuir el desgaste de la operación. Distintos estudios recientes han abordado la problemática del pronóstico de demanda de productos lácteos y perecederos con técnicas de *machine learning*, destacando su capacidad para capturar relaciones no lineales, integrar múltiples fuentes de información y adaptarse a patrones de alta variabilidad (Chongstitvatana & Vithitsoontorn, 2022; Goli et al., 2018).

Para abordar esta problemática, se propone el desarrollo de un sistema integral que combine un modelo de pronóstico de demanda semanal por cliente y producto con una estrategia posterior de asignación óptima de mercadería en camiones. El modelo predictivo se construye mediante técnicas de aprendizaje automático, aprovechando el análisis de patrones históricos de compra. La literatura especializada propone diversas metodologías, como regresión, árboles de decisión y redes neuronales, siendo *Random Forest* y *XGBoost* dos de los enfoques más robustos frente a datos complejos y con *outliers* (James et al., 2013; Chen & Guestrin, 2016). En particular, trabajos aplicados al sector minorista muestran que modelos basados en árboles como *XGBoost* no solo superan en precisión a los métodos estadísticos clásicos, sino que además son más eficientes computacionalmente (Nguyen, 2023).

Asimismo, investigaciones recientes han demostrado que la incorporación de variables externas —económicas, climáticas y de contexto— puede incrementar significativamente el poder explicativo de los modelos predictivos en el sector lácteo, particularmente cuando se emplean enfoques multivariados como SARIMAX o se utilizan estructuras híbridas optimizadas mediante algoritmos metaheurísticos (Dineva & Atanasova, 2023; Goli et al., 2018). Estas estrategias permiten adaptar el modelo a condiciones dinámicas del entorno, mejorando la planificación logística y productiva.

Cabe señalar que, por un lado, SARIMAX (*Seasonal AutoRegressive Integrated Moving Average with eXogenous regressors*) es un modelo estadístico que permite incorporar variables exógenas en series temporales, siendo especialmente útil en contextos con fuerte estacionalidad y relaciones lineales. En contraste, modelos como *Random Forest* y *XGBoost*, basados en árboles de decisión, ofrecen mayor flexibilidad para capturar patrones no lineales sin necesidad de asumir una estructura funcional fija. Por su parte, las "estructuras híbridas" combinan diferentes técnicas predictivas o de optimización —por ejemplo, modelos de regresión ajustados con algoritmos genéticos— para potenciar el rendimiento. Por otro lado, aunque en esta tesis se utiliza búsqueda por grilla y aleatoria para optimizar hiperparámetros, dichas estrategias no se consideran metaheurísticas en sentido estricto, sino enfoques clásicos de optimización en espacios acotados, como lo son los parámetros de los modelos de árboles de decisión.

A diferencia de otros trabajos que se enfocan en la cadena de producción láctea, esta investigación se centra en la etapa de distribución, atendiendo los desafíos propios del reparto minorista y mayorista en zonas urbanas vulnerables. De este modo, se busca contribuir no solo desde una perspectiva técnica, sino también desde un enfoque operativo que busque potenciar la eficiencia logística local.

1.2. Problema

La distribución de productos lácteos está tercerizada y organizada por zonas exclusivas para evitar la competencia entre fleteros. Este estudio se enfoca exclusivamente en la zona asignada a Villa Fiorito.

En esta ubicación, diversos factores dificultan la planificación de la demanda:

- Elasticidad precio: En zonas de bajos ingresos, pequeñas variaciones de precio provocan cambios importantes en el volumen de compras.
- Competencia informal: Algunos fleteros comercializan productos fuera de su zona asignada, generando presión indirecta.
- Mayoristas con sucursales múltiples: Centralizan sus pedidos en un único punto, dificultando el análisis por sucursal.
- Comercialización informal: Algunos minoristas evitan el circuito formal, dificultando el seguimiento de la demanda.

A estas complejidades se suma un segundo desafío logístico: la asignación de camiones. Esta problemática se origina principalmente en la alta variabilidad de los pedidos de las grandes cuentas, quienes solicitan mercadería paletizada que ocupa gran parte del volumen de carga (Observatorio de la Cadena Láctea Argentina (OCLA), 2019). Cuando estos pedidos no se realizan, un solo camión es suficiente; cuando sí se efectúan, se requieren dos o más viajes.

La empresa (nombre enmascarado) asigna preventistas que visitan a estos clientes y coordinan los pedidos. Sin embargo, el fletero también mantiene contacto directo con los responsables y puede sugerir modificaciones en las cargas. Actualmente, este proceso depende exclusivamente de observaciones subjetivas, sin herramientas que permitan prever el volumen a despachar.

La ausencia de planificación genera ineficiencias económicas y operativas: desgaste de vehículos, necesidad de mayor capacidad de almacenamiento en los clientes, menor rotación de stock y jornadas de trabajo desbalanceadas para los repartidores.

1.3. Objetivo

El objetivo principal de esta tesis es desarrollar un sistema que, a partir del pronóstico de demanda semanal por cliente y producto, permita optimizar la planificación logística y maximizar la eficiencia operativa del fletero. Esta integración entre modelado predictivo y toma de decisiones busca brindar una herramienta concreta que acompañe el trabajo cotidiano del distribuidor, reduciendo la dependencia de estimaciones subjetivas y mejorando la capacidad de anticipación frente a escenarios de alta variabilidad.

En este marco, se plantean tres subobjetivos específicos:

1- Identificar patrones de demanda:

Construir un modelo de predicción semanal por cliente y producto basado en técnicas de aprendizaje automático, utilizando datos históricos de ventas y variables externas relevantes (económicas, climáticas y productivas). El objetivo es capturar relaciones complejas y no lineales que expliquen las fluctuaciones en la demanda, contemplando tanto factores internos como condiciones contextuales.

2- Optimizar la estrategia de distribución:

Diseñar una estrategia de asignación de mercadería que utilice como insumo las predicciones del modelo para organizar las cargas de camiones de manera más eficiente. Esta etapa busca reducir la cantidad de viajes necesarios, minimizar el uso subóptimo de la capacidad de carga, y garantizar que los envíos se ajusten a la demanda real proyectada sin sobredimensionamiento innecesario.

3- Evaluar el impacto económico y operativo:

Analizar el efecto de las predicciones y de la optimización sobre distintos indicadores clave, como la cantidad de viajes requeridos y los costos variables logísticos. Se compararán escenarios alternativos para cuantificar los beneficios del enfoque propuesto frente al esquema actual de planificación manual.

Este enfoque integral busca mejorar la toma de decisiones tanto en términos comerciales como operativos. Al combinar predicción y logística, se espera generar un sistema replicable que mejore la planificación del fletero, aumente la eficiencia en la utilización de recursos, y aporte herramientas objetivas que ayuden a navegar escenarios inciertos de demanda.

2. Datos

2.1. Fuente y extracción de datos

Los datos utilizados en este estudio han sido extraídos del sitio web para fleteros de la empresa láctea. Cada fletero accede con sus credenciales y dispone de información relevante sobre sus clientes y operaciones de distribución. Si bien la plataforma permite la descarga de diversos conjuntos de datos, solo algunos resultan pertinentes para la elaboración del modelo de *forecast* de demanda. En particular, se excluyen datos relacionados con rendiciones por faltantes de mercadería, devoluciones y gestión de envases, dado que no aportan información directamente útil para el análisis de la demanda de productos.

La extracción de datos presenta ciertas limitaciones, ya que el sistema permite únicamente descargas diarias, lo que dificulta la obtención de información en rangos de fechas más amplios. Para este estudio, se han recopilado datos históricos desde enero de 2023 hasta diciembre de 2024, aunque los registros están disponibles en el sistema desde octubre de 2021.

2.2. Estructura de los datos

Para la elaboración del modelo de *forecast* de demanda, se han identificado cuatro fuentes de datos relevantes. Las primeras dos son tablas principales del modelo, mientras que las siguientes son complementarias y sirven para validar la integridad y completar posibles datos faltantes.

2.2.1. Tabla de rendiciones de productos

Esta tabla contiene información detallada sobre la mercadería transportada y vendida. Sus principales campos relevantes incluyen:

- Tipo de documento y número legal: Identificador único del documento asociado a cada transacción.
- Fecha: Día en que se realizó la operación.
- Producto: Código interno asignado a cada producto.
- Cantidad: Número de unidades entregadas o facturadas. Los valores positivos corresponden a remitos de entrega de mercadería al fletero, mientras que los valores negativos representan facturas emitidas a clientes.

2.2.2. Base de cálculo de flete por producto

Esta tabla proporciona información económica relevante sobre las ventas y las comisiones del fletero. Los campos principales incluyen:

- Cliente y razón social: Identificación de los clientes atendidos.
- Tipo de documento y número legal: Identificación del documento comercial de referencia.
- Producto y descripción: Código y denominación del producto vendido.
- Base de cálculo: Importe neto total facturado por producto, sin impuestos.
- Flete: Comisión facturada al transportista.
- Porcentaje de flete: Proporción de la base de cálculo destinada a la comisión del fletero, que varía entre el 6% y el 12%, dependiendo del producto y del cliente.

La integración de ambas tablas, a través del campo de documento legal, permite calcular el valor unitario de los productos y detectar posibles promociones por volumen.

2.2.3. Documentos por planilla

Esta tabla resulta útil para la incorporación de datos faltantes vinculados a clientes, en aquellos casos donde las tablas principales no disponen de dicha información. Asimismo, es empleada para realizar análisis generales, tales como la facturación total por cliente o los importes facturados por día. Es importante destacar que esta tabla no contiene información sobre la cantidad de mercadería ni sobre su valor unitario. El campo de importe representa el valor total del documento comercial, incluyendo tanto el precio de los productos como los impuestos asociados (IVA, percepciones de IVA, ingresos brutos, entre otros).

Los campos relevantes incluyen:

- Código de cliente: Identificador del cliente asociado al documento.
- Descripción del cliente: Nombre o razón social del cliente.
- Tipo de factura: Clasificación del tipo de documento comercial.
- Fecha de documento: Día en que se generó el documento.
- Importe: Valor monetario total correspondiente al documento.

2.2.4. Rendiciones y saldos

Esta tabla cumple un rol importante para el análisis de integridad de los datos provenientes de las otras tablas. Se trata de una tabla diaria que muestra totalizadores generales del fletero, como el total diario de facturación a grandes cuentas, el total diario de facturación a clientes minoristas, el total de deuda acumulada correspondiente a devoluciones de mercadería, entre otros datos agregados por día.

Sin embargo, el dato más relevante para nuestros análisis es el total diario de comisiones por flete. Este valor permite contrastar el total de fletes informados en la tabla "Base de Cálculo de Flete por Producto".

Al unir ambas tablas por fecha, se puede calcular la diferencia entre el total de fletes registrados en la base de flete por producto y el valor consolidado de rendiciones y saldos para ese mismo día. Cuando la diferencia es cero, se valida que se ha incluido la totalidad de las mercaderías despachadas a todos los clientes y que se ha calculado correctamente su correspondiente comisión por flete.

2.2.5. Resumen de tablas utilizadas

Podemos resumir los datos extraídos de cada tabla y su utilización en el modelo con la siguiente tabla.

Tabla	Tipo de Información	Campos Relevantes	Uso Principal en el Modelo
Rendiciones de Productos	Operaciones logísticas y facturación	Fecha, Producto, Cantidad, Documento legal	Cantidades entregadas y facturadas
Base de Cálculo de Flete	Datos económicos y comisiones	Cliente, Producto, Base de cálculo, Flete, % Flete	Cálculo de valor unitario, identificación del cliente
Documentos por Planilla	Soporte administrativo y trazabilidad de clientes	Cliente, Tipo de factura, Fecha, Importe	Identificación de cliente y validación de transacciones
Rendiciones y Saldos	Totales diarios consolidados	Fecha, Total comisiones, Total grandes cuentas, Total minoristas	Validación de integridad y consistencia de comisiones

Tabla 1: Descripción de las tablas de datos utilizadas

2.3. Preparación y homogenización de los datos

Una vez identificadas las tablas relevantes, se procederá a la homogenización, limpieza e integración de los datos recopilados. Este proceso incluye la gestión de cambios en códigos y nombres de productos, la identificación y tratamiento de valores faltantes, y la incorporación coherente de variables internas relevantes. Todas estas tareas serán llevadas a cabo utilizando la plataforma *Knime*, herramienta seleccionada por su simplicidad en la gestión y preparación de grandes volúmenes de datos.

2.3.1. Unificación de productos con cambios de código

Con el fin de asegurar la continuidad temporal en el análisis de productos, se construyó una tabla denominada "cambios de código", compuesta por dos campos: código anterior y código nuevo. Esta tabla permitió identificar aquellos productos que, a lo largo del período analizado, cambiaron de código interno debido a modificaciones menores, como ajustes en el etiquetado, cambios mínimos en la composición del producto o actualizaciones en el *packaging*.

Estos cambios, si bien no alteran la esencia del producto, generan un nuevo código desde el punto de vista de la fábrica, provocando la discontinuación del código anterior. Para evitar discontinuidades en las series temporales, se consideraron equivalentes los productos relacionados por esta tabla.

En total, se identificaron 19 casos de productos clave que sufrieron este tipo de migraciones. Entre ellos se destacan leches en sachet, dulces de leche en sus distintas presentaciones, yogures de litro y mantecas.

2.3.2. Depuración de la tabla de rendiciones de productos

El objetivo principal de esta tabla es reflejar la diferencia entre la mercadería solicitada a fábrica y la mercadería finalmente facturada a los clientes. Idealmente, la suma de cantidades debería ser cero, lo cual indicaría un balance perfecto entre entregas y facturación. Sin embargo, en la práctica, esto no siempre ocurre.

Cuando la suma resulta positiva, significa que el fletero recibió más mercadería de la que logró facturar. En esos casos, normalmente al cabo de dos meses, dicha diferencia se le factura al

fletero como un cargo. Si ocurriera el caso contrario —que se haya facturado más de lo recibido— se le emite una nota de crédito al fletero.

Estas diferencias pueden deberse a múltiples causas: por ejemplo, ajustes en remitos que no se ven reflejados en la facturación, devoluciones no acreditadas por fábrica, o notas de crédito que se emiten, pero no se corresponden con devoluciones reales. Todo ese detalle documental queda registrado en esta tabla.

Dado que para nuestro análisis sólo interesa vincular facturas con cantidades entregadas y precios correspondientes, se depuró esta tabla para conservar únicamente los registros que hacen referencia directa a la mercadería efectivamente facturada a los clientes.

2.3.3. Consolidación de cantidades y valores netos por cliente y producto

Una vez depuradas las tablas principales, se procedió a su integración para obtener una base consolidada con información clave: las cantidades compradas y el valor neto facturado, discriminados por cliente, producto y día.

Este proceso se basó en la combinación entre la tabla de Rendiciones de Productos y la tabla de Base de Cálculo de Flete por Producto, permitiendo obtener, para cada transacción, tanto la cantidad como el importe neto correspondiente. Es importante aclarar que este importe está libre de impuestos, percepciones y cualquier otro tipo de recargo, representando únicamente el valor neto facturado del producto.

En esta instancia también fue necesario recurrir a la tabla Documentos por Planilla. En particular, esta tabla resulta útil cuando se producen devoluciones acreditadas a clientes minoristas: en esos casos, la fábrica, por política, no deduce del fletero el valor del flete asociado a dicha devolución. Como resultado, en la tabla de Base de Flete por Producto puede no figurar esa transacción, dado que se le asigna una base de cálculo igual a cero.

Este mecanismo se aplica principalmente a cuentas pequeñas, como una forma de evitar penalizar su comportamiento comercial. Sin embargo, para mantener la coherencia y trazabilidad en nuestro análisis, fue necesario complementar los datos faltantes mediante la tabla Documentos por Planilla, que permitió identificar el cliente asociado a cada documento aun cuando no existiera un registro correspondiente en la tabla de flete.

2.3.4. Cálculo de precios promedio semanales por producto

Adelantándonos al proceso de agregación semanal que será detallado en el punto 2.4, en esta instancia se calculó un precio promedio semanal por producto. Este valor no se encuentra disponible de forma explícita en las bases de datos originales, sino que fue derivado a partir de los datos integrados en los pasos anteriores.

Gracias al *join* entre las tablas de Rendiciones de Productos y Base de Cálculo de Flete por Producto, se dispone de la cantidad facturada y del total neto facturado para cada combinación de cliente, producto y día. A partir de allí, se agruparon los datos por semana y por producto, sin distinguir entre clientes, y se calculó el precio promedio semanal dividiendo el valor neto total por la cantidad total facturada en cada caso.

$$PrecioPromedio_{p,s} = \frac{\sum_{i=1}^{n} Importe_i}{\sum_{i=1}^{n} Cantidad_i}$$

Donde:

- $p =$ producto
- $s =$ semana
- $n =$ total de registros para el producto en la semana
- $Importe_i =$ importe neto facturado del registro i
- $Cantidad_i =$ cantidad facturada del registro i

Para asegurar la cobertura total de semanas y productos, incluso en los casos en los que no hubo ventas, se realizó un *crossjoin* entre un maestro de productos y un maestro de combinaciones de año y semana, construidos a partir de todo el universo de datos disponibles. Esto permitió generar una estructura completa sobre la cual se calcularon los precios, sin omitir combinaciones relevantes por ausencia de datos puntuales.

Además del cálculo del precio promedio semanal por producto, se incorporó una columna adicional que representa el valor relativo de cada producto respecto al más vendido del conjunto de datos: el sachet de leche entera de primera marca. Para ello, se calculó también el precio promedio semanal de este producto específico y se construyó una variable denominada "valor sachet leche". Esta variable se define como el cociente entre el precio promedio del producto correspondiente y el precio promedio del sachet de leche entera en esa misma semana.

$$ValorSachetLeche_{p,s} = \frac{PrecioPromedio_{p,s}}{PrecioPromedio_{sachet,s}}$$

Donde $PrecioPromedio_{sachet,s}$ es el precio promedio semanal del sachet de leche entera de primera marca.

Esta transformación permite evaluar, de forma relativa, cuán caro o barato resulta un producto respecto al principal producto de referencia del sistema de distribución, proporcionando una herramienta de normalización ante el efecto inflacionario.

2.3.5. Aproximación de precios faltantes por semana

Como resultado del *crossjoin* entre todos los productos y todas las semanas del período analizado, se generaron combinaciones en las que ciertos productos no registraban ventas en determinadas semanas. Esto resultó en valores faltantes para el precio promedio en esas combinaciones específicas.

Para resolver este problema, se implementó una estrategia de imputación basada en la información de semanas cercanas. En primer lugar, si existía un valor de precio en una semana anterior (s'), se utilizó ese valor ajustado por el cambio relativo en el precio promedio del sachet de leche entera de primera marca entre ambas semanas. De esta manera, se mantuvo la proporcionalidad entre productos en función del principal producto de referencia.

$$PrecioEstimado_{p,s} = PrecioPromedio_{p,s'} \frac{PrecioPromedio_{sachet,s}}{PrecioPromedio_{sachet,s'}}$$

En los casos en los que no había precios registrados en semanas anteriores, se aplicó la misma lógica utilizando semanas posteriores (s''), hasta encontrar un valor disponible. Este enfoque también permitió cubrir productos que comenzaron a aparecer más adelante en el histórico.

$$PrecioEstimado_{p,s} = PrecioPromedio_{p,s''} \frac{PrecioPromedio_{sachet,s}}{PrecioPromedio_{sachet,s''}}$$

Gracias a este procedimiento, se logró construir una tabla completa con un precio estimado por semana para cada producto. Aunque estos precios estimados no siempre son determinantes para el modelo —dado que los productos sin ventas suelen tener menor relevancia—, reemplazar los valores nulos por estimaciones razonables aporta consistencia general al conjunto de datos y puede contribuir positivamente al desempeño del modelo.

2.3.6. Variables adicionales generadas a partir de los datos internos

Con el objetivo de optimizar el rendimiento del modelo y mejorar la eficiencia computacional, se incorporaron algunas transformaciones adicionales al conjunto de datos final. Por un lado, se unificaron los campos de código y descripción tanto para productos como para clientes, generando nuevas variables que combinan ambos elementos en una sola cadena. Esto permitió reducir la cantidad de columnas generadas en procesos posteriores como el *one-hot encoding*, evitando la creación innecesaria de múltiples variables redundantes.

Por otro lado, se incorporaron versiones logarítmicas de tres variables clave: el precio promedio del producto, el valor relativo del producto respecto al sachet de leche entera (valor sachet leche), y la cantidad facturada. Estas transformaciones se realizaron utilizando el logaritmo natural, añadiendo 1 a cada valor para evitar indefiniciones con ceros. En los casos en que las variables contenían valores negativos, se sumó además el valor absoluto del mínimo, de modo que todos los registros fueran mayores que cero.

$$\text{LogVariable} = \log\left(x + |\min(x)| + 1\right)$$

Estas variables logarítmicas permiten capturar relaciones no lineales y atenuar el impacto de valores extremos, favoreciendo un comportamiento más estable del modelo predictivo.

Variable	Descripción	Tipo	Transformaciones Aplicadas
PrecioUnitario	Valor neto facturado dividido por cantidad facturada	Continua	Derivada del join entre tablas y agregación semanal
ValorSachetLeche	Precio relativo al sachet de leche entera de primera marca	Continua	Cociente de precios promedio
LogPrecioUnitario	Logaritmo natural del precio ajustado	Continua	Logaritmo para estabilizar varianza
LogCantidad	Logaritmo natural de cantidad ajustado	Continua	Logaritmo para reducir impacto de outliers
LogValorSachetLeche	Logaritmo natural del valor relativo al sachet	Continua	Logaritmo para reducir impacto de outliers
CounterSemana	Numeración secuencial semanal	Entera	Incremental en el tiempo
DiasSemana	Cantidad de días trabajados efectivamente	Entera	Calculado por semana con calendario laboral

Tabla 2: Variables generadas a partir de los datos originales

2.4. Agregación de los datos para el análisis

Para cumplir con el objetivo del estudio, es fundamental encontrar un equilibrio entre la oportunidad de los datos y la precisión de las predicciones. Dado que la empresa provee datos diarios, se ha optado por agregarlos a una escala semanal. Esto permite capturar patrones de demanda sin perder granularidad excesiva ni introducir ruido innecesario en el modelo. De esta manera, los datos serán trabajados de manera acumulada por semana, por producto y por cliente. Este enfoque resulta adecuado, ya que el proceso de *forecasting* se ejecutará semanalmente con el objetivo de predecir las ventas de la semana siguiente.

Esta decisión metodológica se alinea con hallazgos de (Chongstitvatana & Vithitsoontorn, 2022), quienes concluyen que la agregación semanal mejora la estabilidad de los modelos de predicción en industrias con alta rotación como la láctea.

Para implementar la agregación semanal, se generó una base completa mediante un *crossjoin* entre tres tablas maestras: una de productos, una de clientes y una de combinaciones de año y semana extraídas del universo total de los datos. Esto permitió estructurar un conjunto de datos con todas las combinaciones posibles de cliente-producto-semana, asegurando que el modelo reciba una base uniforme y consistente para generar predicciones, incluso cuando no haya existido actividad en ciertas combinaciones.

Adicionalmente, se incorporó una variable que representa la cantidad de días hábiles efectivamente trabajados en cada semana. Esta información permite identificar semanas afectadas por feriados, paros o medidas gremiales. Es importante aclarar que esta variable no genera problemas de *data leakage*, ya que puede conocerse de antemano antes de ejecutar cualquier predicción semanal.

Por último, se incorporó una variable secuencial que numera de forma creciente cada semana en el tiempo. Esta variable temporal resulta esencial para capturar la dinámica cronológica del proceso de ventas, evitando ambigüedades derivadas del reinicio anual del número de semana (por ejemplo, la semana 1 de 2023 y la semana 1 de 2024).

$$CounterSemana_s = CounterSemana_{s-1} + 1$$

2.5. Incorporación de fuentes externas

Con el objetivo de enriquecer el modelo y capturar mejor el contexto en el que se desarrolla la demanda de productos, se incorporaron al set de datos variables provenientes de fuentes externas. Estas variables permiten contemplar factores económicos, productivos y climáticos que podrían influir en el comportamiento de compra de los clientes.

Estas fuentes externas se organizan en cuatro grandes grupos: datos económicos del INDEC (INDEC, s.f.), la cotización del dólar paralelo (Ámbito Financiero, s.f.), los indicadores productivos de la Dirección Nacional de Lechería (DNL, s.f.), y las condiciones climáticas provistas por el Servicio Meteorológico Nacional (Meteostat, s.f.). A continuación, se detalla el tratamiento específico y las variables incorporadas desde cada fuente.

Estas variables externas fueron vinculadas temporalmente al set de datos principal, de modo que cada registro semanal por producto y cliente pudiera contar con la información económica y climática correspondiente a esa semana.

El enriquecimiento del set de datos con estas fuentes externas apunta a mejorar la capacidad predictiva del modelo, dotándolo de una mayor comprensión del entorno en el que se inserta la demanda.

2.5.1. Datos del INDEC

Se incorporaron variables de corte mensual publicadas por el Instituto Nacional de Estadística y Censos (INDEC). Estas incluyen:

- Índice de precios al consumidor (IPC) de Gran Buenos Aires para leche, productos lácteos y huevos.
- IPC de Gran Buenos Aires para alimentos (que incluye el anterior y otros productos).
- Precio promedio de la leche.
- Estimador mensual de actividad económica (EMAE).
- Índice de salarios.

Dado que estos datos se publican mensualmente y suelen estar disponibles con una demora, se asumió que el valor del mes actual solo es conocido a partir del mes siguiente, evitando así incurrir en *data leakage*. Para compatibilizarlos con la estructura semanal del análisis, se asignó a cada semana el valor correspondiente al mes en curso. En los casos en que una semana abarca dos meses distintos, se utilizó el promedio simple de los valores de ambos meses.

2.5.2. Cotización del dólar paralelo

Se incorporó la cotización diaria del dólar paralelo (también conocido como dólar *blue*), correspondiente al tipo de cambio vendedor, para los años 2023 y 2024. Estos datos fueron agrupados por semana, calculando el promedio semanal. Esta variable busca capturar posibles cambios de comportamiento de compra asociados a la evolución del tipo de cambio informal, en un contexto económico con alta dolarización de expectativas.

2.5.3. Datos de la dirección nacional de lechería

Desde la DNL se incorporaron indicadores productivos mensuales vinculados al sector lácteo. Al igual que con los datos del INDEC, se asumió que estos valores están disponibles con un mes de retraso y se aplicó la misma lógica de asignación a las semanas. Las variables utilizadas son:

- Producción nacional de litros de leche.
- Variación mensual de la producción en Buenos Aires.
- Utilización de la capacidad instalada.
- Precio al productor por litro de leche en Buenos Aires.
- Ventas internas de leche fluida.

Estas variables permiten contextualizar la oferta y la dinámica de producción del sector, que pueden incidir en la disponibilidad de productos y en las estrategias comerciales adoptadas por la empresa.

2.5.4. Datos del servicio meteorológico nacional

Se utilizaron datos diarios del Servicio Meteorológico Nacional (SMN) correspondientes a la zona de reparto, que fueron agregados a nivel semanal mediante el cálculo del promedio. Las variables incorporadas son:

- Temperatura promedio diaria.
- Temperatura mínima diaria.
- Temperatura máxima diaria.
- Milímetros de precipitación diaria.

Se asume que estas variables no incurren en *data leakage*, dado que en un escenario operativo real se cuenta con pronósticos semanales confiables. Estas variables buscan capturar efectos climáticos sobre el consumo, en especial en productos sensibles a la temperatura como yogures, postres fríos o mantecas.

2.5.5. Resumen de las fuentes externas incorporadas

Podemos resumir los datos incorporados y su naturaleza con la siguiente tabla.

Fuente	Variable	Frecuencia	Unidad
INDEC	IPC alimentos, IPC lácteos, salario, EMAE	Mensual	Índice
SMN	Temperatura promedio/mín./máx., precipitaciones	Diaria	°C / mm
DNL	Producción leche, precio productor, ventas internas	Mensual	Litros / $
Cotización paralela	Tipo de cambio vendedor (blue)	Diaria	$ / USD

Tabla 3: Variables externas incorporadas

2.6. Análisis descriptivo

Inicialmente se llevará a cabo un análisis exploratorio de los datos históricos de ventas correspondientes a los últimos dos años. En esta etapa se identificarán los clientes más relevantes, los productos con mayor volumen de ventas, y se evaluará la variabilidad y dispersión presentes en las ventas históricas. No se eliminarán productos del análisis, aunque se considerará agrupar clientes según su dimensión e importancia relativa, para facilitar el manejo y análisis de los datos.

2.6.1. Resumen general

Para contextualizar la magnitud del conjunto de datos, se presentan los siguientes valores totales para los años 2023 y 2024:

- ✓ 23.499 facturas/notas de crédito procesadas
- ✓ 53 clientes activos
- ✓ 674 tipos de productos comercializados
- ✓ 5.221.875 unidades vendidas en total

2.6.2. Distribución del valor de los tickets

Los valores de los *tickets* presentan una alta dispersión, influenciada principalmente por la heterogeneidad de los clientes. Existen tanto pequeños comercios minoristas como grandes cuentas mayoristas, lo que genera una amplitud significativa en los importes facturados. Para ilustrar esta variabilidad se presenta un análisis visual mediante un *boxplot*.

El análisis revela que el valor máximo de facturación alcanza los $32.069.375, mientras que el mínimo es de −$26.568.889. Ambos valores extremos corresponden a operaciones con clientes mayoristas. En el caso del valor máximo, se trata de una venta de gran volumen a una cuenta clave. Por otro lado, el valor negativo más extremo refleja una nota de crédito emitida para anular una factura con errores, seguida de una refacturación posterior. Este procedimiento es habitual cuando una factura original contiene inconsistencias que deben ser corregidas.

En el resto de los casos, las facturaciones negativas suelen estar asociadas a devoluciones de mercadería, especialmente cuando el valor devuelto supera el importe original del pedido. Este comportamiento, aunque infrecuente, genera distorsiones significativas en el análisis de valores unitarios y demanda una interpretación cuidadosa de los datos.

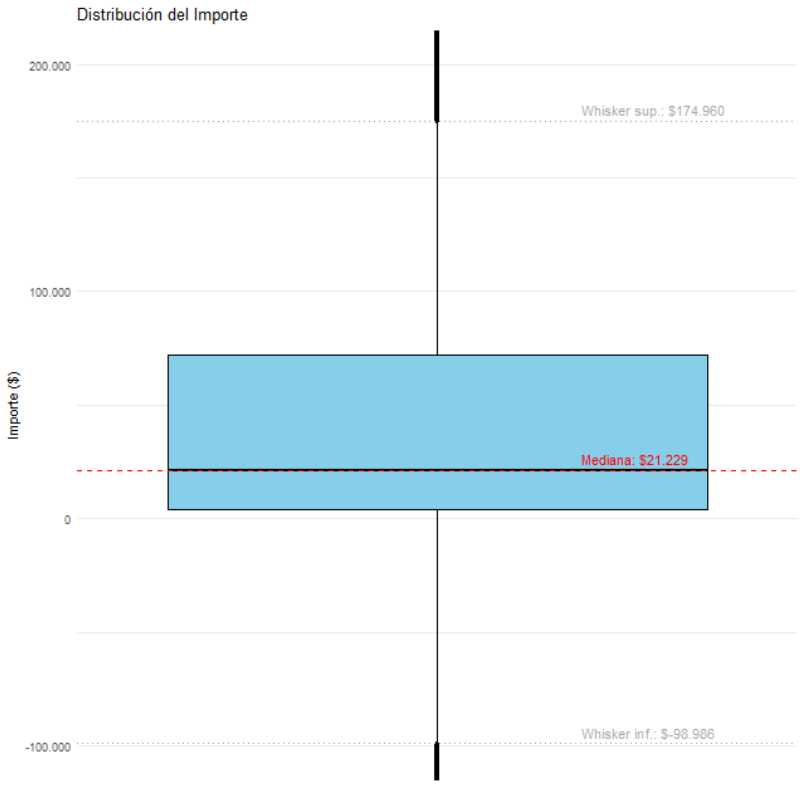

Figura 1: Distribución del valor de los tickets

En el gráfico se observa que la mediana del valor de los *tickets* se encuentra en torno a $21.229, lo que representa el valor central de la distribución. El *whisker* inferior alcanza los $−98.986, mientras que el *whisker* superior llega a $174.960, indicando un rango intercuartílico amplio, típico de una distribución con fuerte asimetría.

Más allá de esos valores, se visualiza una gran cantidad de puntos individuales considerados *outliers*, tanto por debajo como por encima del rango esperado. Estos puntos se concentran en líneas visuales debido a la densidad de observaciones fuera del rango típico, particularmente por las operaciones de los clientes mayoristas. Cabe aclarar que el gráfico fue recortado visualmente para mostrar un rango de valores entre −$100.000 y $200.000, sin eliminar datos, a fin de mejorar la visibilidad del *boxplot* y su estructura central.

2.6.3. Concentración de la facturación por cliente

Dado que los clientes mayoristas (cuyo nombre fue enmascarado) representan una porción significativa de las ventas del reparto, se elaboró un gráfico de torta para visualizar la distribución del neto facturado entre los distintos clientes, identificando a aquellos que concentran la mayor parte de la facturación mensual. En total, se identificaron 53 clientes únicos entre 2023 y 2024. Para facilitar el análisis visual y evitar una sobrecarga en el gráfico, se decidió agrupar a los clientes que no se encuentran entre los cinco con mayor facturación bajo la categoría "Otros clientes".

Esta agrupación se justifica tanto por motivos de visualización como por la distribución real de los importes facturados. A partir del sexto cliente en adelante, la facturación comienza a disminuir progresivamente: el sexto cliente acumuló alrededor de $64 millones en el período, mientras que el de menor facturación apenas alcanzó los $200.000. Solo 15 clientes superaron los $10 millones. En general, los clientes agrupados en "Otros" corresponden a autoservicios medianos, pequeños almacenes, despensas, estaciones de servicio o kioscos, con niveles de compra más bajos y esporádicos.

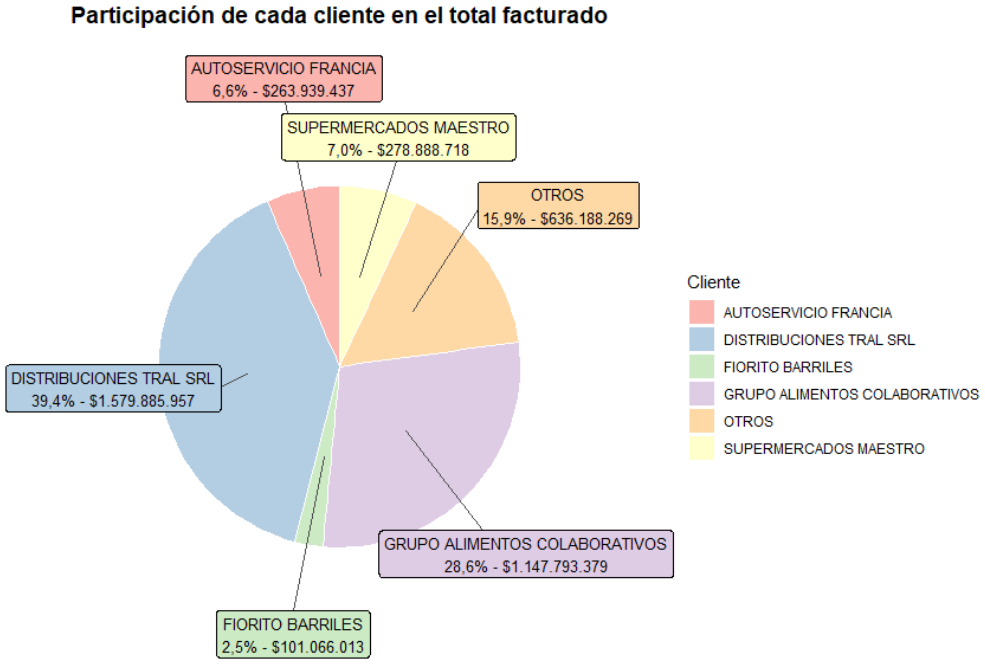

Figura 2: Distribución del importe total neto facturado por cliente

El gráfico revela una fuerte concentración de la facturación en un pequeño grupo de clientes. En primer lugar, se encuentra DISTRIBUCIONES TRAL SRL, con una participación del 39,4% del total facturado, lo que equivale a $1.579.885.957 en el período analizado. Este cliente es un supermercado mayorista que opera como proveedor de otros comercios, abasteciendo a almacenes y autoservicios más pequeños. Su estructura cuenta con depósito, auto elevadores y capacidad para recibir grandes volúmenes de mercadería, comprando generalmente por pallet cerrado.

Le sigue GRUPO ALIMENTOS COLABORATIVOS, con el 28,6% del total facturado, equivalente a $1.147.793.379. Se trata de un cliente con características muy similares a DISTRIBUCIONES TRAL: también es un supermercado mayorista que actúa como central de abastecimiento para otros

comercios. Al igual que el primero, cuenta con infraestructura adecuada para la recepción de pedidos de gran volumen, lo que genera una demanda con picos significativos que impactan en la planificación logística.

En tercer lugar, con una participación considerablemente menor, aparece SUPERMERCADOS MAESTRO, que representa el 7% del total facturado, con un acumulado de $278.888.718. A diferencia de los anteriores, este cliente no se orienta tanto a abastecer a otros comercios, sino que funciona como un autoservicio para venta al público general. Posee auto elevador, aunque su frecuencia de compras por pallet es menor.

El cuarto cliente en facturación es AUTOSERVICIO FRANCIA, con un 6,6% del total y un acumulado de $263.939.437. Se trata de una importante cadena de supermercados con múltiples sucursales en el país. Sus pedidos están automatizados desde su casa matriz, y presenta una demanda homogénea y constante a lo largo del tiempo. Por su estructura organizativa, no utiliza pallets cerrados ni cuenta con auto elevador. A diferencia de los demás clientes, AUTOSERVICIO FRANCIA paga un porcentaje de comisión por flete más bajo: 6%, frente al 10% al 12% que abonan los demás clientes.

Finalmente, FIORITO BARRILES ocupa el quinto lugar, con el 2,5% del total facturado, acumulando $101.066.013. Es un autoservicio de escala mediana, sin auto elevador, y sin operaciones por pallet. A pesar de su menor tamaño, se destaca sobre el resto de los autoservicios incluidos dentro de la categoría "OTROS", lo que justifica su individualización en el gráfico.

En resumen, los dos primeros clientes concentran en conjunto el 68% del total facturado, lo cual implica que cualquier desvío en su comportamiento de compra representa un gran impacto en la operación logística y en el ingreso del fletero. Este nivel de concentración resalta la importancia de monitorear su demanda de forma particular.

2.6.4. Productos más vendidos

Con el objetivo de comprender la composición de la demanda y la importancia relativa de cada producto dentro del reparto, se realizó un análisis de los productos más vendidos en función de dos dimensiones clave: cantidades vendidas y facturación neta acumulada.

En el primer gráfico, se visualiza el top 20 de productos por cantidades vendidas. Se observa que la leche entera de primera marca en sachet se posiciona ampliamente como el producto más demandado, con más de un millón de unidades vendidas, representando por sí sola el 20% del total de unidades entregadas en todo el periodo analizado. Le siguen el queso rallado en sobre de 35 gramos y la leche entera de segunda marca, ambos con participaciones del 10% y 6% respectivamente.

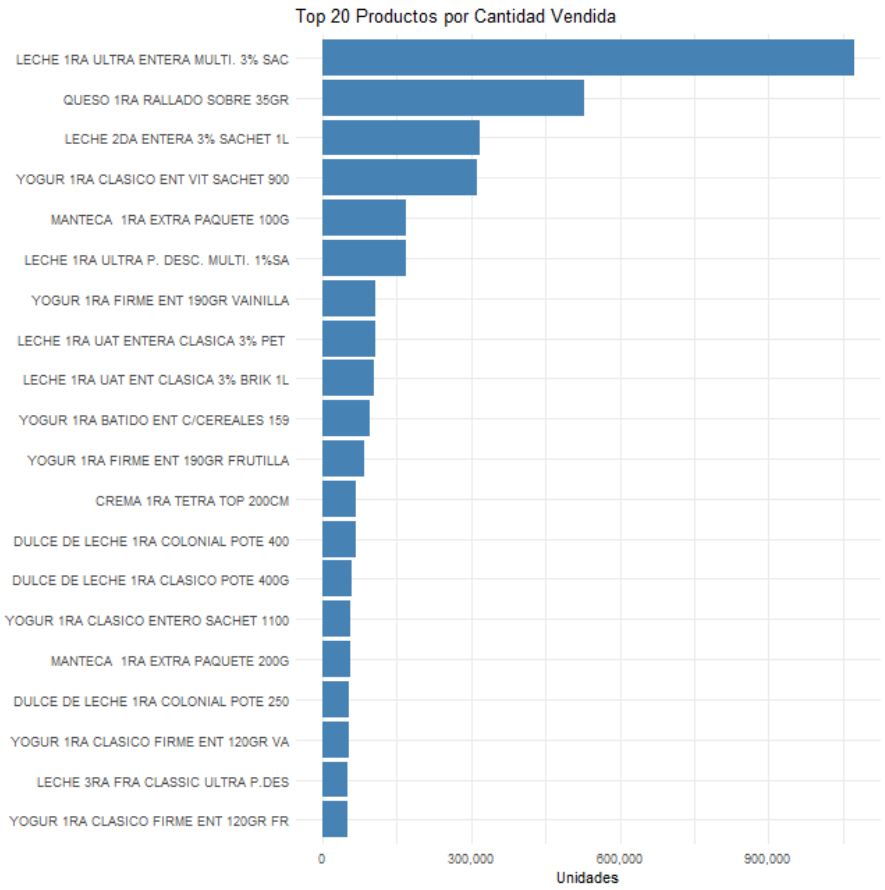

Figura 3: Top 20 de productos por cantidad de unidades vendidas

Se destacan también productos como el yogur clásico en sachet, las mantecas en paquete de 100g, y los yogures firmes en pote, que, aunque no alcanzan los primeros puestos, tienen un volumen considerable de ventas. En el extremo inferior del ranking, encontramos productos con participaciones individuales del orden del 1% o menos, lo que refleja una larga cola de productos de menor rotación pero que enriquecen el surtido ofrecido.

En el segundo gráfico, correspondiente al top 20 de productos por facturación neta, se mantiene la preponderancia de la leche entera en sachet, con un acumulado de más de 638 millones de pesos, lo que representa casi el 20% de la facturación total del reparto. Le siguen el queso rallado y la leche de segunda marca, con aportes del 7% y 6% respectivamente. Aquí también ganan relevancia productos de mayor valor unitario, como los dulces de leche en pote de 400g, el queso cremoso en horma de 3kg o los postres refrigerados, que, si bien no figuran entre los más vendidos por volumen, tienen una incidencia destacada en términos de ingresos.

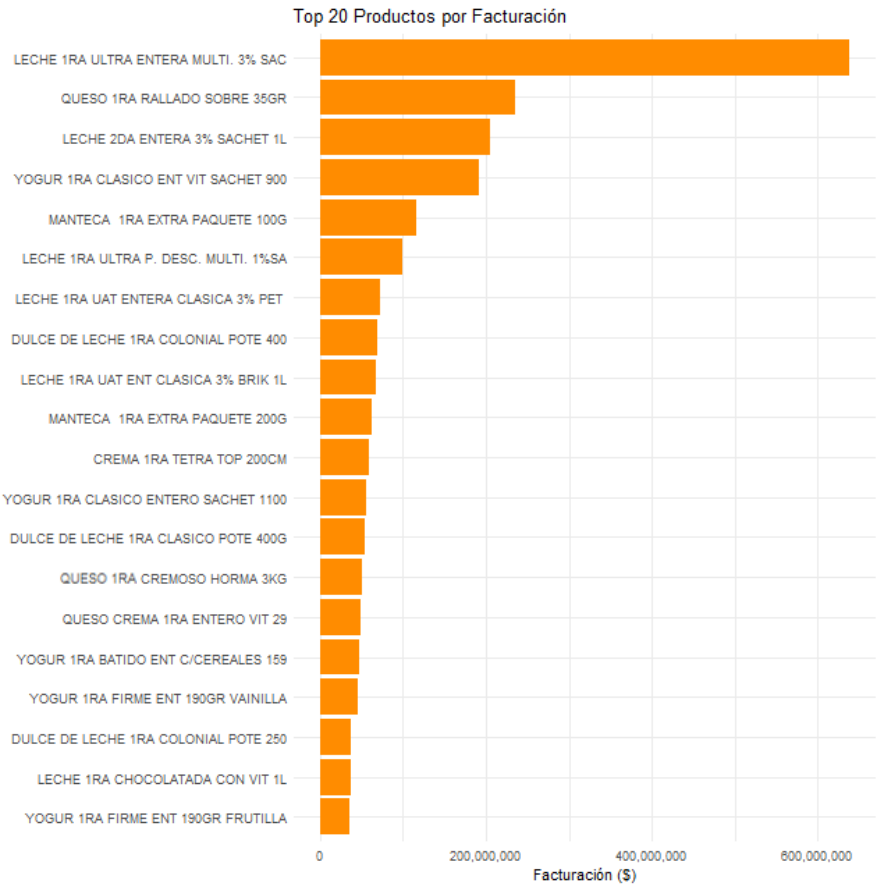

Figura 4: Top 20 de productos por facturación neta acumulada

Esta diferencia entre cantidad y valor resalta la importancia de considerar ambos enfoques al momento de priorizar la atención comercial o diseñar estrategias de abastecimiento. Algunos productos —aunque con bajo volumen— tienen un impacto relevante en la comisión del fletero, dado que la misma se calcula como un porcentaje de la facturación.

En conclusión, si bien el reparto está dominado por productos de consumo masivo, existe una diversificación significativa que incluye productos de alto valor agregado que, aunque menos demandados, generan una parte no trivial de los ingresos y deben ser gestionados con atención.

2.6.5. Variabilidad en la demanda

Uno de los desafíos clave en la planificación logística es la variabilidad de la demanda a lo largo del tiempo, especialmente en contextos donde los envíos y la producción deben ser anticipados con antelación. Entender cómo fluctúa la facturación a lo largo del calendario permite detectar patrones de comportamiento en los clientes y planificar con mayor precisión la distribución. Para abordar esta cuestión, se desarrollaron dos visualizaciones complementarias a partir de los datos históricos de facturación diaria entre enero de 2023 y diciembre de 2024.

La primera de las visualizaciones se enfoca en la evolución semanal de la facturación, agregando los valores diarios por semana calendario y ajustándolos por inflación para permitir una comparación homogénea entre años. En la Figura 5 se observa cómo se comporta la facturación total semana a semana en valores constantes, separados por año. Aun luego del ajuste por IPC, se mantiene una facturación significativamente mayor en 2024 respecto de 2023, lo que sugiere un aumento real en los volúmenes comercializados o una mayor intensidad en la actividad.

Se evidencian niveles bajos de facturación durante las primeras semanas del año, coincidiendo con el período de vacaciones de verano. Luego se observa un primer pico importante hacia marzo, coincidente con la reactivación posvacacional, y un segundo pico alrededor de la mitad del año. A partir de allí, la facturación se mantiene en niveles más estables, con oscilaciones semanales que podrían deberse a factores operativos o dinámicas comerciales específicas.

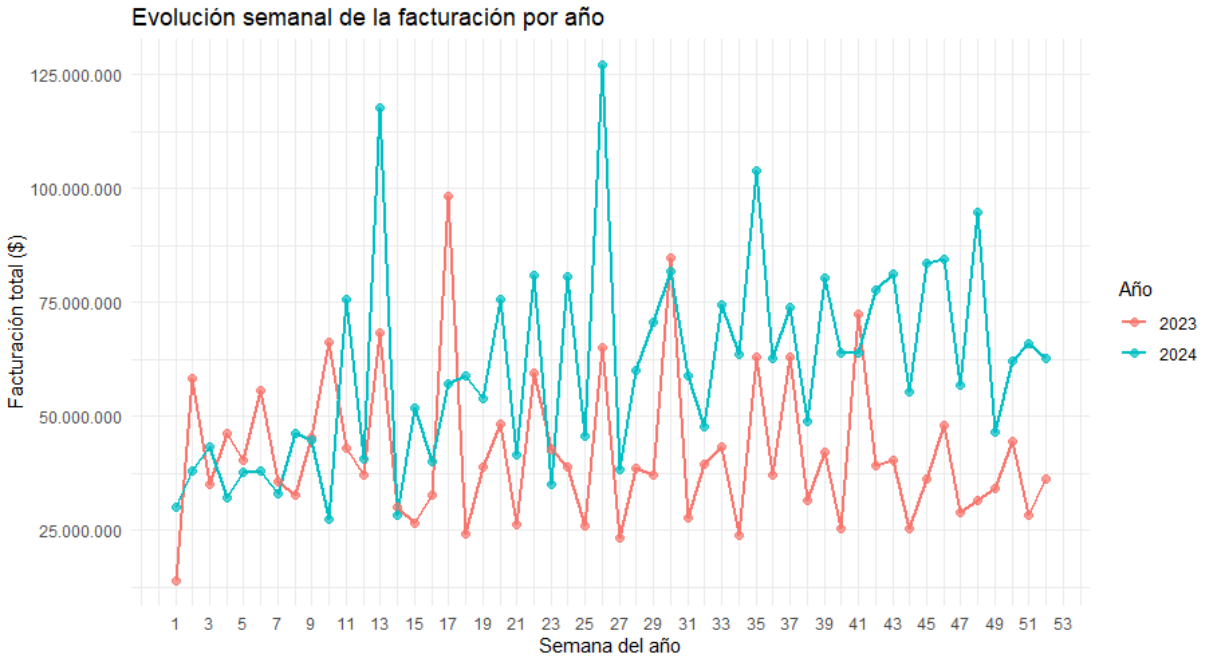

Figura 5: Evolución semanal del total de facturación separada por año

Complementariamente, se desarrolló un *boxplot* que muestra la distribución de la facturación total diaria según el día del mes (Figura 6). En este gráfico, cada caja representa la variabilidad del total facturado en todos los días 1, 2, ..., hasta el 31 del mes, considerando el total del período analizado. El objetivo de esta visualización es detectar si existen patrones sistemáticos relacionados con el calendario mensual, más allá de las semanas o de factores externos.

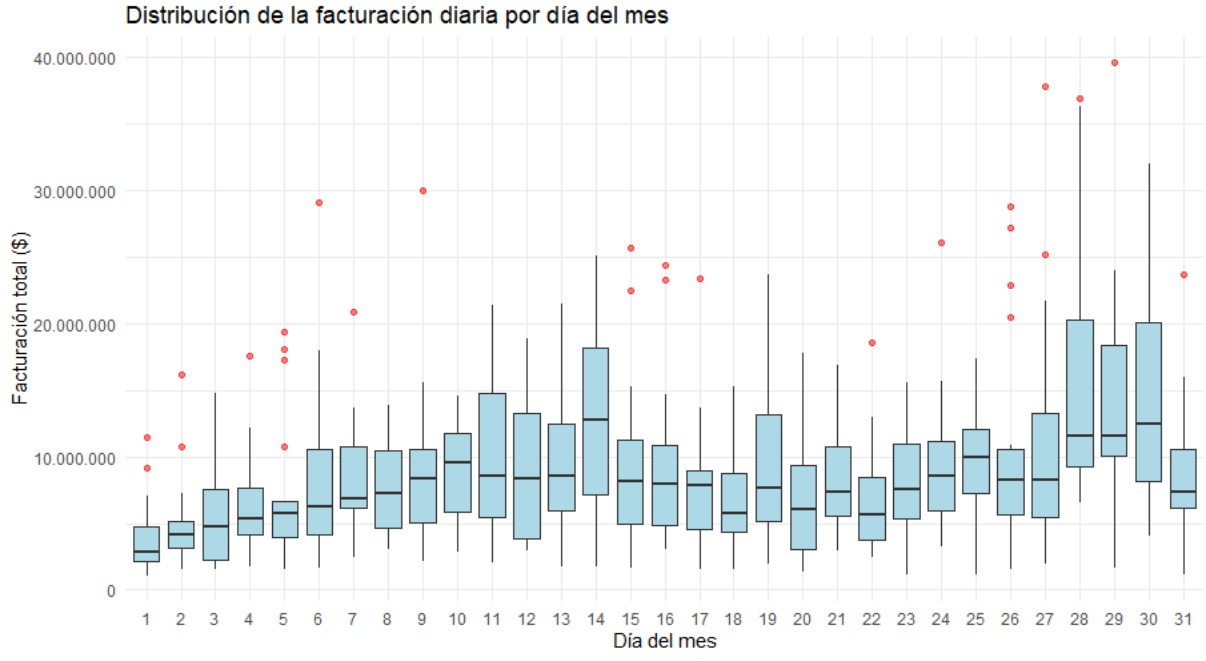

Figura 6: Distribución de la facturación total por cada día del mes

De este análisis se destaca que los días cercanos al final de mes (especialmente del 27 al 31) tienden a concentrar mayores niveles de facturación, tanto en su mediana como en los valores extremos u *outliers*. Este comportamiento puede explicarse por prácticas comerciales habituales, como la intensificación de la actividad por parte de los vendedores para cumplir con metas mensuales de facturación, lo que se traduce en una mayor colocación de pedidos en los últimos días del mes. En contraposición, los primeros días del mes suelen mostrar menores niveles de actividad, posiblemente como consecuencia del arrastre de esfuerzos comerciales del cierre anterior.

Los valores estadísticos refuerzan las observaciones gráficas de la variabilidad en la facturación:

- ✓ Valor promedio de facturación diaria: $5.449.151
- ✓ Desviación absoluta promedio: $4.405.847
- ✓ Mínimo diario: $238.569
- ✓ Máximo diario: $31.195.412

La diferencia entre estos extremos representa un desafío considerable desde el punto de vista logístico, ya que obliga a dimensionar la operación para días de alta demanda, generando al mismo tiempo capacidad ociosa en los días con menor actividad.

Este comportamiento resulta crítico en el diseño de la estrategia de distribución. Cuando la demanda se concentra en pocos días, es necesario reforzar la flota de camiones, incluir personal adicional y ejecutar maniobras logísticas complejas, mientras que en días de baja demanda la capacidad queda subutilizada. Esta ineficiencia no solo incrementa los costos operativos, sino que también dificulta la planificación de recursos y la previsión de necesidades futuras.

2.6.6. Correlación entre la variable target y los predictores

Con el objetivo de comprender las relaciones lineales existentes entre la variable *target* y los predictores numéricos seleccionados, se construyó un análisis de correlación que permite observar qué variables presentan mayor o menor asociación con la cantidad demandada semanal. Esta exploración es especialmente útil para anticipar posibles redundancias, identificar relaciones esperadas o inesperadas, y guiar la selección de variables para los modelos predictivos.

Para este análisis, se excluyeron las siguientes variables:

- Variables categóricas: Cliente y Producto, por ser identificadores alfanuméricos que no pueden ser interpretados mediante correlación de Pearson, la cual requiere variables numéricas continuas.
- Variables temporales de control: Semana_Actual y Year_Actual fueron descartadas por no ser variables explicativas en sí mismas, sino referencias temporales que no aportan información directa sobre la variación de la demanda. Incluirlas en un análisis de correlación puede inducir interpretaciones incorrectas sobre estacionalidades o tendencias.
- Variables rezagadas: Se excluyeron todas las variables rezagadas (*lags*) para centrarse únicamente en los predictores contemporáneos, y se utilizaron los valores correspondientes al mismo período de demanda, sin desfases. En particular, se eliminaron los indicadores de semanas anteriores y se ajustaron los indicadores económicos y productivos para que correspondieran al mes de la observación, y no al mes previo, como se hace en el modelo para evitar *data leakage*.

Esto se realizó con fines exploratorios, ya que el objetivo de este gráfico es comprender relaciones estructurales y no construir un modelo predictivo.

Como puede observarse, los coeficientes de correlación lineal son en general bajos. Esta situación es esperable por varias razones. En primer lugar, la demanda de productos lácteos está determinada por múltiples factores combinados, lo cual diluye el peso explicativo de cada variable individual. En segundo lugar, muchas de estas relaciones no son lineales, sino que se manifiestan en forma de umbrales, interacciones o comportamientos estacionales, que no pueden ser captados por una correlación simple. A ello se suma el hecho de que muchas variables externas —como el clima o los precios— tienen variaciones más suaves (mensuales o estacionales), mientras que la demanda se mide a nivel semanal, lo que introduce ruido y reduce la fuerza de las asociaciones lineales.

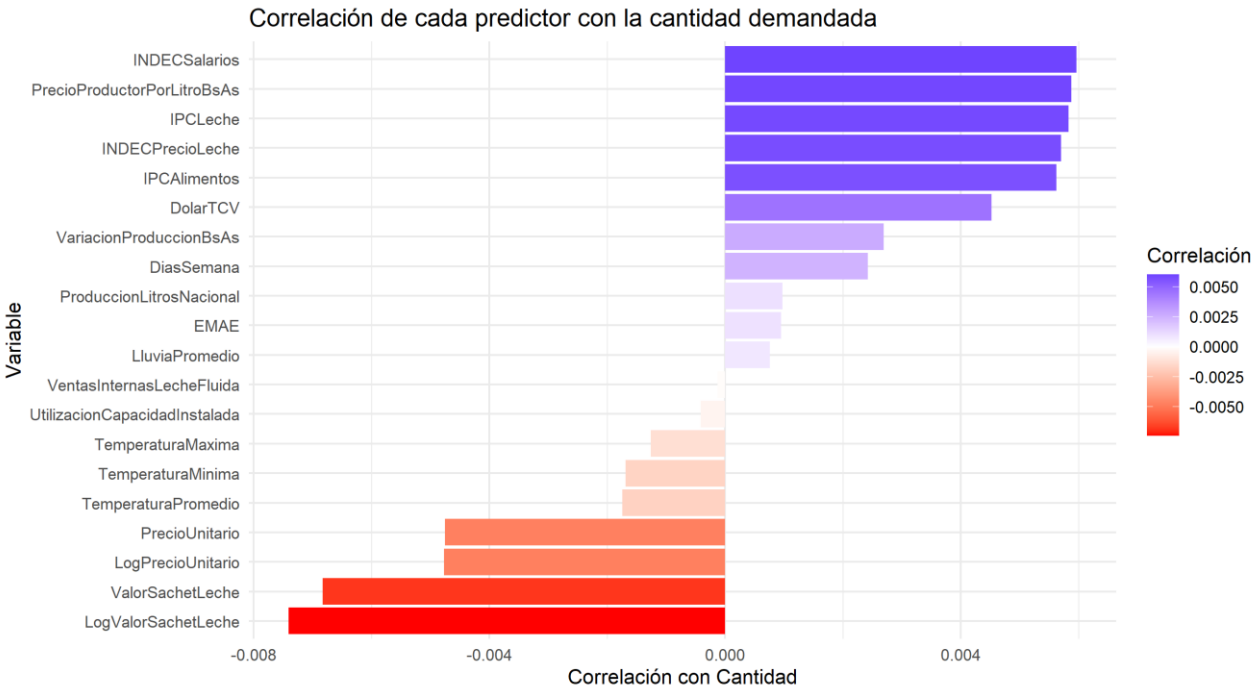

Figura 7: Correlación entre predictores y variable target

A pesar de la baja magnitud de los coeficientes, las direcciones de las correlaciones resultan consistentes con lo esperable desde el punto de vista económico y comercial. Por ejemplo, los precios de los productos muestran una relación negativa con la demanda, tanto en su valor absoluto como relativo (LogValorSachetLeche, PrecioUnitario), lo cual sugiere que, ante aumentos de precio específicos, tiende a caer la cantidad demandada, como sería de esperar en productos con cierta elasticidad.

Por el contrario, los índices de precios generales —como el IPCLeche, el INDECPrecioLeche o el IPCAlimentos— presentan una correlación positiva con la demanda. Esta aparente contradicción puede interpretarse como un reflejo de comportamientos distintos: mientras que un aumento en el precio de un producto puntual desalienta su consumo directo, un aumento generalizado del nivel de precios puede incentivar el consumo en el corto plazo, como mecanismo de protección frente a la pérdida de poder adquisitivo o por la aceleración del gasto ante expectativas de inflación. En contextos de desvalorización de la moneda, este tipo de comportamiento es común, especialmente en productos de consumo habitual.

Por otro lado, variables climáticas como la temperatura promedio o la temperatura máxima también exhiben correlaciones negativas con la demanda. Sin embargo, esta relación podría no deberse a la temperatura en sí, sino a un efecto de calendario: las semanas más cálidas coinciden con el período vacacional del verano, durante el cual disminuye la actividad comercial en zonas no turísticas como la que abarca el presente análisis. Esto fue corroborado previamente en la evolución semanal de la facturación, donde se observó una caída pronunciada en las primeras semanas del año.

A su vez, la variable DíasSemana, que representa la cantidad de días hábiles dentro de cada semana de facturación, muestra una correlación positiva con la cantidad demandada. Esto es coherente con la lógica operativa del negocio: a mayor cantidad de días activos en la semana, mayor es la oportunidad de venta y mayor el movimiento económico general.

En resumen, si bien las correlaciones individuales no revelan fuertes asociaciones lineales, su signo y sentido sí aportan evidencia cualitativa valiosa sobre los mecanismos subyacentes que explican la variación en la demanda. Esta información contribuye a contextualizar los resultados del modelo predictivo y refuerza la pertinencia de incluir estas variables como parte del set de predictores, especialmente cuando se los considera en conjunto y bajo técnicas que capturan relaciones no lineales e interacciones complejas.

3. Metodología para el modelo predictivo

La presente sección describe la metodología adoptada para desarrollar y evaluar los modelos de pronóstico de demanda semanal por cliente y producto. Se abordan tanto los fundamentos teóricos de los algoritmos utilizados como su aplicación concreta sobre el conjunto de datos, así como las métricas empleadas para evaluar su rendimiento.

3.1. Fundamentos teóricos de los modelos utilizados

Se emplearán dos algoritmos de aprendizaje automático supervisado: *Random Forest* y *XGBoost*, ambos ampliamente utilizados para tareas de regresión en contextos de alta dimensionalidad.

3.1.1. Random Forest

El algoritmo de *Random Forest* consiste en un ensamblado de múltiples árboles de decisión, construidos sobre subconjuntos aleatorios de los datos de entrenamiento. El proceso se desarrolla en los siguientes pasos (Hastie et al., 2009, pp. 587–604):

1. Se generan múltiples subconjuntos de entrenamiento mediante *bootstrap sampling*, es decir, seleccionando aleatoriamente muestras con reemplazo del *dataset* original.
2. Para cada uno de estos subconjuntos, se entrena un árbol de decisión utilizando una versión modificada del algoritmo CART:
   - En cada nodo, se selecciona aleatoriamente un subconjunto de variables predictoras (de tamaño *mtry*).
   - Se elige la mejor variable de división dentro de ese subconjunto.
   - Se divide el nodo y el proceso se repite recursivamente hasta alcanzar un tamaño mínimo de nodo (*min.node.size*).
3. Una vez entrenados todos los árboles (*num.trees*), el modelo final consiste en la agregación de las predicciones individuales:
   - En regresión, se promedia la salida de todos los árboles.
   - En clasificación, se toma la clase con mayor cantidad de votos.

Algorithm 15.1 *Random Forest for Regression or Classification.*

1. For $b = 1$ to B:
   (a) Draw a bootstrap sample $\mathbf{Z}^*$ of size N from the training data.
   (b) Grow a random-forest tree T_b to the bootstrapped data, by recursively repeating the following steps for each terminal node of the tree, until the minimum node size n_{min} is reached.
      i. Select m variables at random from the p variables.
      ii. Pick the best variable/split-point among the m.
      iii. Split the node into two daughter nodes.
2. Output the ensemble of trees $\{T_b\}_1^B$.

To make a prediction at a new point x:

Regression: $\hat{f}_{\text{rf}}^B(x) = \frac{1}{B}\sum_{b=1}^{B} T_b(x)$.

Classification: Let $\hat{C}_b(x)$ be the class prediction of the bth random-forest tree. Then $\hat{C}_{\text{rf}}^B(x) = \textit{majority vote}\ \{\hat{C}_b(x)\}_1^B$.

Figura 8: Algoritmo de Random Forest en "The Elements of Statistical Learning"

Este enfoque reduce la varianza del modelo y mitiga el riesgo de sobreajuste al incorporar aleatoriedad tanto en los datos como en la selección de variables. La combinación de múltiples árboles débiles produce un modelo robusto con buen desempeño predictivo incluso en contextos de alta dimensionalidad.

3.1.2. XGBoost

XGBoost (*Extreme Gradient Boosting*) se basa en la técnica de *boosting*, donde múltiples árboles se construyen de manera secuencial, cada uno corrigiendo los errores del anterior. A diferencia del *Random Forest*, que entrena todos los árboles en paralelo, *XGBoost* optimiza cada árbol teniendo en cuenta los errores residuales acumulados de los anteriores (James et al., 2013, pp. 320–327).

El procedimiento se desarrolla de la siguiente forma:

1. Se inicia con una predicción constante para todos los valores ($\hat{f}(x) = 0$) y se calculan los residuos como la diferencia entre los valores reales y los predichos.
2. En cada iteración b, se ajusta un árbol de regresión a los residuos (r_i), generalmente con una profundidad baja para evitar sobreajuste.
3. La predicción del modelo se actualiza sumando una versión escalada del nuevo árbol, con un parámetro de aprendizaje (λ, o eta) que controla la magnitud de la actualización.
4. Se recalculan los residuos y se repite el proceso.

Algorithm 8.2 *Boosting for Regression Trees*

1. Set $\hat{f}(x) = 0$ and $r_i = y_i$ for all i in the training set.
2. For $b = 1, 2, \ldots, B$, repeat:
   (a) Fit a tree $\hat{f}^b$ with d splits ($d+1$ terminal nodes) to the training data (X, r).
   (b) Update $\hat{f}$ by adding in a shrunken version of the new tree:
   $$\hat{f}(x) \leftarrow \hat{f}(x) + \lambda \hat{f}^b(x). \quad (8.10)$$
   (c) Update the residuals,
   $$r_i \leftarrow r_i - \lambda \hat{f}^b(x_i). \quad (8.11)$$
3. Output the boosted model,
   $$\hat{f}(x) = \sum_{b=1}^{B} \lambda \hat{f}^b(x). \quad (8.12)$$

Figura 9: Algoritmo de Boosting en "An Introduction to Statistical Learning"

El modelo final es la suma ponderada de todos los árboles generados. Esta técnica permite construir modelos altamente expresivos, con un control refinado sobre el sobreajuste gracias a la regularización incorporada y al uso de árboles débiles. Además, *XGBoost* ofrece mejoras adicionales como manejo automático de valores faltantes, paralelización y poda de árboles en función de la ganancia esperada.

3.2. Partición del conjunto de datos

Los modelos fueron entrenados sobre un conjunto de datos compuesto por más de 3 millones de observaciones. Cada fila representa la combinación única de semana, cliente y producto, con variables asociadas que permiten caracterizar la operación logística y comercial. Se dispone de más de 400 variables, entre las cuales se encuentran:

- Variables numéricas: cantidades, precios, indicadores económicos, meteorológicos, etc.
- Variables categóricas: cliente (53 niveles) y producto (más de 600 niveles).

En la práctica de modelado predictivo, es común dividir los datos disponibles en dos conjuntos: uno de entrenamiento y otro de prueba. El conjunto de entrenamiento se utiliza para ajustar el modelo, mientras que el conjunto de prueba se reserva para evaluar su desempeño en datos no vistos. Esta separación permite obtener una estimación realista de la capacidad de generalización del modelo.

Una recomendación habitual, especialmente con grandes sets de datos como el presente, es asignar aproximadamente el 80% de los datos al conjunto de entrenamiento y el 20% restante al de prueba. Esta proporción busca equilibrar la necesidad de proporcionar suficiente información al modelo para su ajuste, sin comprometer la evaluación objetiva de su rendimiento.

En nuestro caso, debemos dividir el período que abarca desde enero de 2023 hasta diciembre de 2024. El 80% de las semanas iniciales fue destinado al entrenamiento de los modelos, y el 20% restante (semanas finales) al testeo. Esta partición permite preservar la secuencia temporal y simular un entorno de predicción realista. (Hyndman & Athanasopoulos, 2020)

Figura 10: División del set de datos en entrenamiento y testeo

3.3. Equilibrio entre sesgo y varianza

Una vez definido cuál será el conjunto de entrenamiento, debemos tener especial atención a elegir aquel modelo que optimice el punto de equilibrio entre el sesgo (*bias*) y la varianza (*variance*). Este compromiso, conocido como *bias-variance tradeoff*, es fundamental para maximizar la capacidad de generalización del modelo y minimizar el error total sobre datos no vistos.

El sesgo se refiere a la diferencia sistemática entre las predicciones del modelo y los valores reales. Modelos con alto sesgo tienden a ser demasiado simples y a no capturar correctamente las relaciones presentes en los datos (*underfitting*). Por otro lado, la varianza se refiere a la sensibilidad del modelo a las fluctuaciones del conjunto de entrenamiento. Modelos con alta varianza tienden a ajustarse demasiado a los datos disponibles, incluyendo su ruido, lo cual los vuelve poco robustos ante nuevas observaciones (*overfitting*).

Esta dinámica se representa gráficamente en la primera figura a continuación, donde se observa cómo el error cuadrático medio (MSE) sobre datos de testeo (línea roja) presenta una forma en "U" al aumentar la flexibilidad del modelo. En los extremos, modelos con poca flexibilidad muestran alto sesgo y bajo error de entrenamiento, pero mal desempeño en testeo. A medida

que se incrementa la complejidad, el sesgo disminuye, pero la varianza crece. El punto de equilibrio se alcanza donde el MSE sobre test es mínimo: allí el modelo logra capturar patrones relevantes sin sobreajustarse al ruido de los datos.

Este fenómeno también puede visualizarse en la segunda figura, donde se compara la tasa de error en entrenamiento y testeo para distintas configuraciones de un modelo de clasificación. Mientras que el error de entrenamiento disminuye continuamente al aumentar la complejidad, el error en testeo alcanza un mínimo y luego comienza a crecer, indicando un punto óptimo de complejidad a partir del cual el modelo empieza a sobreajustarse. (James et al., 2013, pp. 36–42)

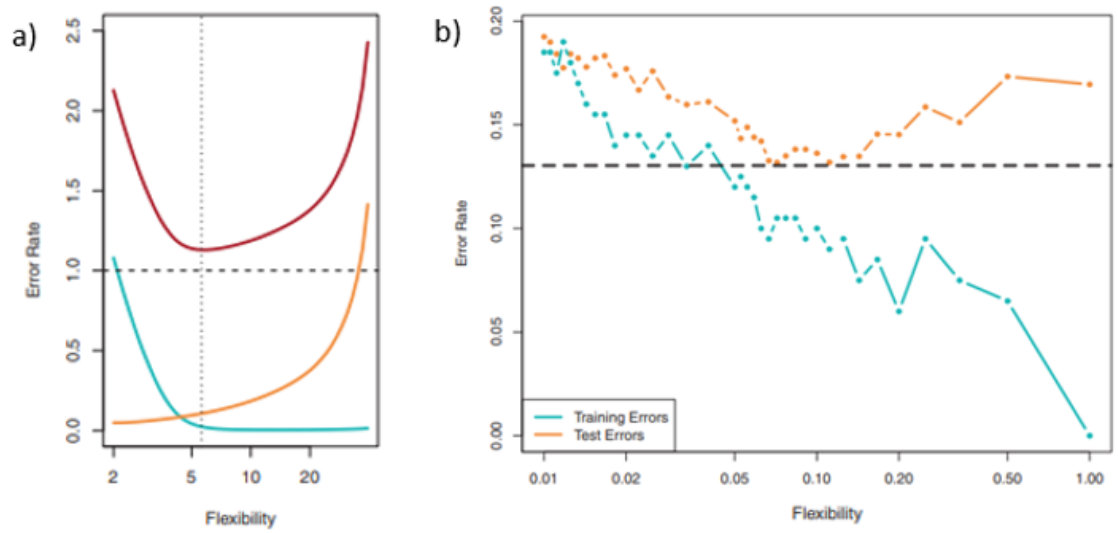

Figura 11: a) Error por Sesgo-Varianza
b) Error de train y test ajustado por flexibilidad

En resumen, un buen modelo no es aquel que logra el menor error sobre el conjunto de entrenamiento, sino aquel que mantiene un balance adecuado entre sesgo y varianza, optimizando su desempeño en nuevos datos. Este equilibrio es especialmente crítico en contextos de alta dimensionalidad o cuando se dispone de una gran cantidad de variables, como ocurre en el presente trabajo.

3.4. Uso de validación cruzada

En esta tesis se adoptó la validación cruzada como método para estimar la capacidad de generalización de los modelos y seleccionar los mejores hiperparámetros. Particularmente, se implementó *k-fold cross-validation* con k = 3 y k = 5, aplicada exclusivamente sobre el conjunto de entrenamiento. Esta decisión tiene un fuerte respaldo teórico, tanto desde la perspectiva estadística como desde la experiencia empírica en problemas reales.

La validación cruzada *k-fold* consiste en dividir el conjunto de entrenamiento en k subconjuntos del mismo tamaño. En cada iteración, uno de ellos se reserva como conjunto de validación y el modelo se entrena sobre los k − 1 restantes. Este procedimiento se repite k veces, asegurando que cada observación se utilice una vez para validación. Finalmente, se promedian los errores de cada iteración para obtener una estimación del error de generalización.

Tal como se menciona en *Introduction to Statistical Learning* (James et al., 2013, pp. 181–183), este enfoque ofrece importantes ventajas respecto a otros métodos como el *Leave-One-Out* CV (LOOCV) o el *validation set approach*. En particular:

- Eficiencia computacional: LOOCV requiere entrenar el modelo n veces (una por cada observación), lo cual resulta prohibitivo en *datasets* grandes. En contraste, *k-fold* CV con k bajo (como 5 o 10) reduce el número de modelos a entrenar, manteniendo una buena aproximación del error.
- Menor varianza que el *validation set approach*: al utilizar múltiples particiones, se reduce la sensibilidad a cómo se dividen los datos, logrando una estimación del error más estable.
- Buen compromiso entre sesgo y varianza: si bien LOOCV tiene bajo sesgo, su varianza es elevada. En cambio, *k-fold* CV con k = 5 o k = 10 ofrece un equilibrio más favorable entre ambos, lo que lo convierte en una opción preferida en la práctica.

Otro aspecto fundamental es que, en muchos casos, el objetivo de la validación cruzada no es obtener una estimación precisa del error absoluto de testeo, sino identificar qué modelo (o conjunto de hiperparámetros) minimiza dicho error. En ese sentido, *k-fold* CV permite encontrar el punto de mínima pérdida estimada, incluso cuando la estimación puntual del error pueda estar sesgada. La siguiente fórmula representa la estimación promedio del error de predicción obtenida por validación cruzada:

$$CV_{(k)} = \frac{1}{k}\sum_{i=1}^{k} MSE_i$$

Donde:

- k es el número de pliegues en los que se divide el conjunto de entrenamiento
- MSE_i es el error cuadrático medio calculado en el pliegue número i

Lo importante es que la forma de la curva de error estimado mediante CV suele reflejar correctamente qué tan flexible debe ser el modelo para alcanzar su mejor rendimiento. Este comportamiento se ilustra claramente en la siguiente figura, donde, si bien los valores absolutos estimados del error (en naranja) pueden no coincidir con el verdadero error de testeo (en azul), los mínimos de las curvas se alinean correctamente con el nivel de flexibilidad óptimo.

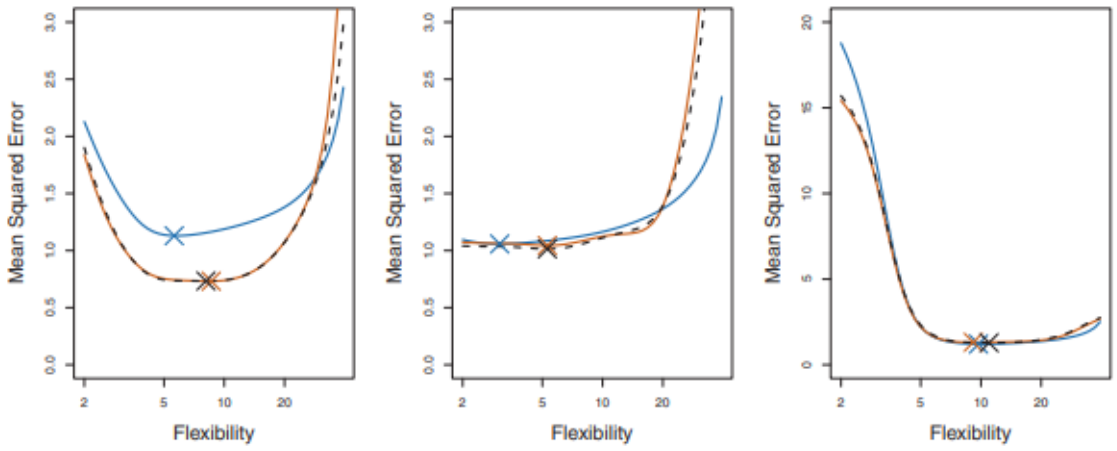

Figura 12: Errores de train y test usando k-folds CV para 3 sets de datos diferentes

La validación cruzada también guarda una relación directa con el equilibrio entre *bias* y varianza, siendo el valor de k en *k-fold* CV quien influye en este equilibrio. Por ejemplo:

- → LOOCV (con k = n) tiende a tener bajo sesgo, ya que los modelos son entrenados con casi toda la muestra disponible, pero su varianza es alta, debido a que cada modelo se entrena con *datasets* muy similares, generando estimaciones altamente correlacionadas entre sí.
- → En cambio, *k-fold* CV con k bajo (como 5 o 10) presenta un sesgo ligeramente mayor, pero una varianza significativamente menor, ya que las particiones de entrenamiento varían más entre sí.

Este balance resulta fundamental: si se prioriza únicamente el sesgo, se corre el riesgo de tener estimaciones inestables; si se reduce demasiado la varianza, se pueden introducir errores sistemáticos. Elegir un valor intermedio de k (como 5 o 10) ha demostrado ser eficaz para capturar este equilibrio y obtener estimaciones confiables del error de generalización, tanto en términos absolutos como para la comparación relativa entre modelos o configuraciones. (James et al., 2013, pp. 183–184)

3.5. Implementación en R y configuración de entrenamiento

Los dos algoritmos elegidos —*Random Forest* y *XGBoost*— fueron entrenados utilizando la función *train*() del paquete *caret*, con la misma estructura de partición y validación. En particular:

- ❖ *Random Forest*: se aplicó validación cruzada inicial de 3 pliegues y posterior de 5 pliegues sobre el 80% del *dataset*. Se desestimó el uso de validación *out-of-bag* (OOB) para mantener la homogeneidad en la comparación.
- ❖ *XGBoost*: se utilizó el mismo esquema de partición y validación cruzada, asegurando que las métricas de evaluación fueran comparables en igualdad de condiciones experimentales.

Este enfoque, fundamentado en la literatura especializada (Probst et al., 2018; Bischl et al., 2021), refuerza la validez de las comparaciones realizadas entre algoritmos y asegura que las métricas reportadas reflejen con mayor precisión la capacidad de generalización de cada modelo.

Como fue anticipado, para la implementación de los modelos y la optimización de sus hiperparámetros, se utilizó el paquete *caret* en R, que permite una interfaz unificada para el entrenamiento, evaluación y comparación de distintos algoritmos de aprendizaje automático. A través de la función *train*(), *caret* facilita la integración de múltiples técnicas de validación y ajuste de hiperparámetros.

A continuación, se detallan los principales hiperparámetros utilizados en cada modelo:

Random Forest (method = "ranger")

- → *num.trees*: número de árboles del bosque.
- → *mtry*: número de variables seleccionadas aleatoriamente para cada división.
- → *min.node.size*: número mínimo de observaciones que debe tener una hoja terminal.
- → *splitrule*: criterio para la división de nodos (en este caso, "*variance*").

XGBoost (method = "xgbTree")

- → *nrounds*: número total de iteraciones de *boosting*.
- → *eta*: tasa de aprendizaje que controla el impacto de cada árbol.
- → *max_depth*: profundidad máxima de los árboles.
- → *gamma*: penalización por complejidad en nodos terminales.
- → *colsample_bytree*: fracción de columnas a muestrear por árbol.
- → *min_child_weight*: peso mínimo de una hoja.
- → *subsample*: fracción de datos a utilizar en cada iteración.

Para garantizar la replicabilidad de los resultados, se utilizó la función *set.seed*() con una semilla fija, y se indexaron explícitamente los folds de validación cruzada a través del parámetro *index* dentro del objeto *trainControl*(). Esto permite que tanto *Random Forest* como *XGBoost* compartan exactamente los mismos subconjuntos de entrenamiento y validación en cada iteración, asegurando una comparación justa.

Como fue mencionado, para esta validación cruzara se eligió un esquema inicial de k = 3 pliegues, y luego para la optimización de hiperparámetros se utilizaron 5 pliegues o *folds*, en línea con las recomendaciones teóricas del libro *Introduction to Statistical Learning* (James et al., 2013, pp. 181–183). Cabe resaltar que la validación cruzada con k < n permite alcanzar un equilibrio adecuado entre el sesgo y la varianza de la estimación del error, con una carga computacional moderada.

3.6. Métricas de evaluación

Para comparar el rendimiento de los modelos, se utilizaron cuatro métricas complementarias:

1. Raíz del error cuadrático medio (RMSE):

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

Donde:

- y_i es el valor real observado para el caso i
- $\hat{y}_i$ es el valor predicho por el modelo
- n es el número total de observaciones

La métrica RMSE cuantifica el error promedio entre los valores reales y los valores predichos por el modelo, elevando al cuadrado las diferencias individuales antes de promediar. Esta característica tiene dos implicancias clave:

- → Penaliza errores grandes con mayor severidad, ya que los errores se elevan al cuadrado.
- → Es sensible al rango y la escala de la variable objetivo: si la variable tiene valores bajos o muchos ceros, un pequeño error puede parecer grande en proporción; y viceversa, errores relativamente grandes pueden quedar "ocultos" si el *dataset* contiene muchas observaciones con valores cercanos a cero.

En *datasets* desbalanceados, como es el caso de esta tesis con sus datos desagregados —donde muchas combinaciones cliente-producto-semana tienen demanda cero—, el RMSE puede no

reflejar adecuadamente el rendimiento del modelo sobre las observaciones más relevantes en términos logísticos o económicos.

2. RMSE relativo al promedio de la variable observada:

$$RMSE\ relativo = \frac{RMSE}{\bar{y}}$$

Donde:

- $\bar{y}$ es el valor promedio de la variable observada

Esta variante normaliza el RMSE en relación con el valor promedio observado de la variable objetivo. Tiene como propósito ajustar la magnitud del error según el contexto de los valores reales. Es particularmente útil cuando:

→ La variable tiene una distribución asimétrica o muchos ceros.
→ Se requiere comparar modelos sobre distintos subconjuntos de datos donde las escalas cambian (como entre clientes grandes y chicos).
→ Se busca evaluar no solo la precisión absoluta, sino también su importancia relativa.

El RMSE relativo permite interpretar el error como un porcentaje del valor medio observado, lo cual es muy útil en contextos comerciales donde importa saber "cuánto se equivocó el modelo en proporción a lo que realmente se vende".

3. Coeficiente de determinación (R^2):

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2}$$

Donde:

- y_i es el valor real observado para el caso i
- $\hat{y}_i$ es el valor predicho por el modelo
- $\bar{y}$ es el valor promedio de la variable objetivo
- n es el número total de observaciones

El coeficiente de determinación R^2 expresa qué proporción de la varianza total de la variable objetivo es explicada por el modelo. Sus valores oscilan entre 0 y 1, donde:

→ Un R^2 cercano a 1 indica que el modelo explica una gran parte de la variabilidad observada.
→ Un R^2 cercano a 0 implica que el modelo no logra capturar la variabilidad de los datos.

Esta métrica tiene particular relevancia cuando se desea evaluar la capacidad del modelo para adaptarse a patrones generales del *dataset*, y no solo su precisión sobre valores puntuales.

En *datasets* con alto grado de dispersión, como ocurre con series de demanda por cliente-producto, un valor de R^2 moderado no necesariamente indica un mal desempeño. Al contrario, puede reflejar una adecuada generalización si el modelo logra capturar las principales tendencias sin sobreajuste. No obstante, R^2 puede verse afectado por la presencia de valores extremos o distribuciones muy asimétricas, y no proporciona información sobre el tamaño absoluto del error.

4. Error absoluto medio (MAE):

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

Donde:

- y_i es el valor real observado para el caso i
- $\hat{y}_i$ es el valor predicho por el modelo
- n es el número total de observaciones

El MAE mide el promedio de las diferencias absolutas entre los valores reales y los valores predichos. A diferencia del RMSE, no penaliza los errores grandes de forma desproporcionada, por lo que ofrece una medida más robusta en presencia de *outliers*.

Algunas características clave del MAE:

→ Tiene una interpretación directa en las mismas unidades de la variable objetivo.
→ Es menos sensible a valores extremos que el RMSE, y por eso complementa su análisis.
→ Resulta útil en contextos donde todos los errores son igualmente relevantes, sin importar su magnitud.

En esta tesis, el MAE permite comprender cuánto se desvía el modelo en promedio al estimar la cantidad demandada, lo cual es especialmente útil en la planificación logística: un error promedio de 20 unidades puede tener implicancias distintas según el volumen habitual del cliente o producto en cuestión. En combinación con otras métricas, el MAE ayuda a construir una evaluación más equilibrada del desempeño del modelo.

3.7. Entrenamiento sobre set simplificado y optimización de hiperparámetros

Con el objetivo de alimentar un modelo más robusto para el segundo problema (asignación de camiones), se definirá un conjunto de datos simplificado en el que se conservarán únicamente los tres principales clientes individualizados, agrupando todos los demás bajo la categoría "Otros clientes". Esta transformación no solo responde a una conveniencia logística —al reducir la complejidad del modelo de distribución—, sino que también implicará una disminución significativa en la cantidad de observaciones. Esta reducción permitirá intensificar el esfuerzo computacional dedicado a la calibración de los modelos predictivos, abriendo la posibilidad de realizar una búsqueda más exhaustiva de combinaciones de hiperparámetros.

A partir de este set simplificado, se volverán a entrenar los modelos *Random Forest* y *XGBoost*. En una primera instancia, se utilizará la misma configuración de hiperparámetros aplicada sobre el set completo, a fin de asegurar una base comparable. Sin embargo, con el objetivo de aprovechar la reducción en el tamaño del *dataset*, se avanzará con un proceso sistemático de optimización de hiperparámetros. En primer lugar, se aplicará una búsqueda por grilla (*grid search*), definida sobre un conjunto de valores específicos para cada modelo: *mtry* y *min.node.size* en el caso de *Random Forest*; y *eta*, *max_depth*, *gamma*, *colsample_bytree*, *min_child_weight* y *subsample* para *XGBoost*. Luego, se implementará una búsqueda aleatoria (*random search*) para explorar regiones adicionales del espacio de hiperparámetros y verificar si es posible obtener combinaciones aún más eficientes que las halladas mediante búsqueda por grilla.

Ambos enfoques de optimización se fundamentan en la literatura especializada, que destaca el impacto del ajuste fino de hiperparámetros en la mejora del desempeño predictivo (Probst et al., 2018; Bischl et al., 2021).

La figura a continuación ilustra de manera conceptual la diferencia entre *Grid Search* y *Random Search* en la exploración del espacio de hiperparámetros. En el panel izquierdo (*Grid Layout*), los puntos representan combinaciones evaluadas mediante una grilla uniforme. Este enfoque puede resultar ineficiente cuando algunos parámetros no influyen significativamente en el rendimiento del modelo (como el eje vertical en la figura), ya que muchas combinaciones quedan desaprovechadas. En cambio, el panel derecho (*Random Layout*) muestra cómo el muestreo aleatorio permite cubrir el espacio de manera más diversa, aumentando las chances de encontrar combinaciones óptimas, incluso si están fuera de los puntos fijos definidos por una grilla. Esta propiedad del Random Search resulta especialmente útil cuando solo unos pocos hiperparámetros tienen un impacto fuerte en la performance del modelo (como el eje horizontal, "*Important parameter*"). (Bergstra & Bengio, 2012, p. 284)

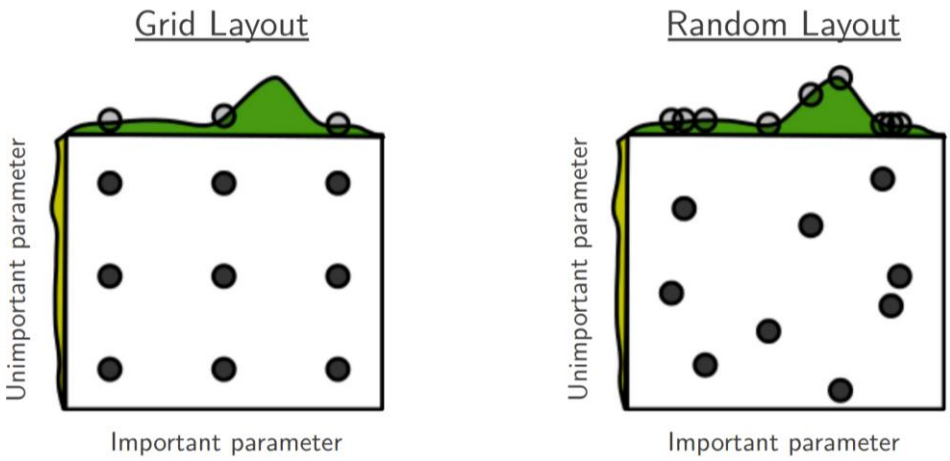

Figura 13: Exploración de hiperparámetros mediante Grid Search y Random Search

Finalmente, se seleccionará para cada algoritmo la combinación de hiperparámetros que arroje el mejor resultado según la métrica de validación cruzada, y se utilizará dicho modelo para la comparación definitiva entre *Random Forest* y *XGBoost*.

4. Resultados del modelo predictivo

4.1. Modelado sobre el conjunto completo de datos

4.1.1. Resultados con Random Forest

Hiperparámetros utilizados

Para entrenar el modelo de *Random Forest* sobre el conjunto completo de datos (más de 3 millones de registros), se utilizó la función *train()* del paquete caret con el motor ranger. Se empleó una muestra aleatoria del 20% del total debido a restricciones computacionales, lo que permitió una mayor velocidad de entrenamiento sin comprometer la validez del experimento.

Los hiperparámetros seleccionados fueron:

- *mtry* = 21: cantidad de variables consideradas en cada *split*, equivalente a la raíz cuadrada del total de variables predictoras.
- *min.node.size* = 10: cantidad mínima de observaciones por nodo terminal, para controlar la profundidad del árbol.
- *num.trees* = 50: cantidad total de árboles del bosque. Se utilizó un valor moderado para acelerar el entrenamiento.

Dado el tamaño del set de datos desagregado, el entrenamiento se realizó con validación cruzada de 3 *folds*. El mismo criterio se utilizará más adelante para una primera comparación justa tanto con el modelo de *XGBoost* como con los modelos trabajados sobre los datos agrupados, sin embargo, con éstos últimos luego se podrá abordar una validación cruzada de 5 *folds* tal como lo recomienda la teoría.

Importancia de las variables

Una vez entrenado el modelo, se extrajo la importancia relativa de cada variable predictora. Las variables con mayor aporte al modelo fueron las relativas a las cantidades en semanas anteriores, precios ponderados, y sus transformaciones logarítmicas:

- Cantidad: 36,02%
- LogCantidad: 35,89%
- ValorSachetLeche: 3,78%
- LogValorSachetLeche: 2,98%
- LluviaPromedio: 2,08%

Estas cinco variables explican juntas más del 80% del peso total del modelo. Esto revela que el histórico de cantidades compradas y los precios relativos son los mayores determinantes de la demanda futura.

Es importante destacar que muchas de las variables más importantes corresponden a cantidades compradas en semanas anteriores (Cantidad_01 a Cantidad_19, y sus versiones logarítmicas). Esto sugiere que el modelo está captando de manera efectiva la estructura temporal del comportamiento de compra, aun sin recurrir explícitamente a variables cronológicas como Semana_Actual o Counter_Semana. En otras palabras, la temporalidad está representada indirectamente a través del histórico de compras, lo que refuerza la importancia de contar con un buen histórico consolidado para alimentar el modelo predictivo.

Las variables utilizadas también fueron clasificadas según su fuente de origen: datos internos, climáticos, económicos o contextuales. A partir de esta clasificación, se calculó el aporte acumulado por cada grupo de variables al modelo:

Fuente	Importancia Relativa
Base interna	84,08%
Servicio Meteorológico Nacional	6,59%
Dirección Nacional de Lechería	4,26%
INDEC	4,03%
Dólar Blue	1,05%

Tabla 4: Importancias relativas por fuente para Random Forest con datos completos

Esto confirma que el modelo se apoya fuertemente en el comportamiento histórico del cliente y los productos, mientras que las variables externas aportan de forma marginal en esta etapa.

Evaluación de performance

Métrica	Train	Test
RMSE	37,91	49,92
RMSE relativo (%)	29,92	25,20
R^2	0,44	0,61
MAE	1,40	4,07

Tabla 5: Performance de Random Forest con datos completos

Estos resultados muestran una diferencia moderada entre el error en entrenamiento y testeo. La diferencia es esperable en un modelo sin sobreajuste, y permite validar que el modelo mantiene un rendimiento razonable fuera de muestra.

La métrica de RMSE relativo es particularmente útil en este caso debido a la presencia de una gran cantidad de ceros en la variable objetivo. Así, la métrica contextualiza el error medio en función del valor promedio real, y ayuda a valorar el modelo incluso cuando el RMSE absoluto es relativamente bajo por influencia de valores pequeños.

El valor de R^2 también aporta información relevante: si bien el modelo alcanza un valor de 0,44 en entrenamiento y mejora a 0,61 en test, estos niveles pueden considerarse moderados. Un R^2 mayor a 0,60 en test sugiere que el modelo logra capturar buena parte de la variabilidad de la demanda, aun con las limitaciones impuestas por la dispersión de los datos y la alta proporción de ceros. No obstante, el valor más bajo en entrenamiento puede atribuirse a la dispersión natural del comportamiento individual por cliente-producto, especialmente en contextos con alta estacionalidad o compras esporádicas.

Por otro lado, el MAE, que representa el error absoluto promedio sin penalizar fuertemente los errores grandes como en el RMSE, confirma la diferencia entre ambos conjuntos: 1,40 en entrenamiento y 4,07 en test. Esta diferencia indica que el modelo tiende a ser más preciso sobre los datos conocidos, aunque mantiene un margen de error aceptable cuando se enfrenta a datos nuevos. La combinación de RMSE y MAE ofrece una visión complementaria del error:

mientras el primero penaliza fuertemente los desvíos grandes, el segundo ofrece una mirada más robusta a la media de las desviaciones absolutas.

El promedio de la variable objetivo Cantidad_00 es bajo en ambos conjuntos, aunque relativamente mayor en el segundo: 1,26 en entrenamiento y 1,98 en testeo. Esto refuerza la idea de que una proporción significativa del *dataset* presenta valores bajos o nulos, lo que influye directamente en la interpretación de todas las métricas. En este contexto, tanto la incorporación del RMSE relativo como del MAE resultan fundamentales para realizar una evaluación precisa del rendimiento predictivo.

Como conclusión, el modelo *Random Forest* entrenado sobre el conjunto completo muestra una fuerte dependencia en variables internas y alcanza un buen nivel de precisión. Esto justifica continuar con el desarrollo del enfoque de segmentación por clientes para mejorar la granularidad del análisis y la optimización logística posterior.

4.1.2. Resultados con XGBoost

Hiperparámetros utilizados

Para el modelo *XGBoost* se utilizó nuevamente el paquete caret con validación cruzada de 3 *folds*. Se aplicó el mismo conjunto reducido de datos utilizado en *Random Forest* (20% del total) para facilitar la comparación de resultados.

Los hiperparámetros utilizados fueron:

- *nrounds* = 100: número total de iteraciones (árboles secuenciales).
- *eta* = 0.1: tasa de aprendizaje.
- *max_depth* = 6: profundidad máxima de los árboles.
- *gamma* = 0: sin penalización adicional por complejidad.
- *colsample_bytree* = 1: se usan todas las variables por árbol.
- *min_child_weight* = 1: mínimo peso de muestra por nodo.
- *subsample* = 1: todos los datos se usan por iteración.

Importancia de las variables

Al igual que en el modelo anterior, se calculó la importancia relativa de cada variable. Las variables más relevantes fueron nuevamente las cantidades históricas de compra por cliente-producto:

- Cantidad: 79,63%
- LluviaPromedio: 4,01%
- LogValorSachetLeche: 3,51%
- Cliente: 2,91%
- Producto: 2,67%

En menor medida también destacaron otras variables climáticas, económicas y de precios. Sin embargo, el fuerte predominio de las cantidades históricas refuerza la idea de que el modelo captura muy bien la tendencia de consumo a partir del historial registrado, sin necesidad explícita de variables temporales como *Semana_Actual* o *Counter_Semana*.

Nuevamente, las variables utilizadas fueron clasificadas según su fuente de origen: datos internos, climáticos, económicos o contextuales. A partir de esta clasificación, se calculó el aporte acumulado por cada grupo de variables al modelo:

Fuente	Importancia Total
Base interna	92,94%
Dirección Nacional de Lechería	4,40%
Servicio Meteorológico Nacional	1,90%
INDEC	0,54%
Dólar Blue	0,23%

Tabla 6: Importancias relativas por fuente para XGBoost con datos completos

Esto confirma, una vez más, que el mayor peso del modelo se encuentra en los datos históricos del cliente, siendo marginal el aporte de fuentes externas.

Evaluación de performance

Métrica	Train	Test
RMSE	38,82	50,33
RMSE relativo (%)	30,64	25,41
R^2	0,42	0,60
MAE	1,33	1,72

Tabla 7: Performance de XGBoost con datos completos

El error absoluto de entrenamiento fue apenas superior al observado en *Random Forest*, lo que indica un ajuste similar del modelo sobre los datos. En testeo, el RMSE también se mantiene en niveles aceptables, lo que sugiere una buena capacidad de generalización.

La métrica de RMSE relativo permite contextualizar los valores observados en función del bajo promedio de la variable objetivo, mostrando que los errores, si bien no despreciables, son proporcionales a los volúmenes esperados.

XGBoost demostró un desempeño consistente tanto en entrenamiento como en testeo, logrando un buen equilibrio entre precisión y estabilidad del modelo, incluso frente a una variable objetivo con alta proporción de ceros.

4.1.3. Comparación entre modelos

A continuación, se presenta una comparación directa entre ambos modelos sobre los mismos datos, estructura de validación y métricas:

Métrica	Random Forest	XGBoost
RMSE Train	37,91	38,82
RMSE Test	49,92	50,33
RMSE Relativo Train (%)	29,92	30,64
RMSE Relativo Test (%)	25,20	25,41
R^2 Train	0,44	0,42
R^2 Test	0,61	0,60
MAE Train	1,40	1,33
MAE Test	4,07	1,72

Tabla 8: Comparación de la performance de Random Forest y XGBoost con datos completos

Como se observa, ambos modelos presentan desempeños muy similares en términos generales. *Random Forest* obtiene una ligera ventaja en la mayoría de las métricas tanto en entrenamiento como en testeo, incluyendo un menor RMSE, un RMSE relativo inferior y un R^2 levemente superior. Sin embargo, *XGBoost* muestra una mejora considerable en el MAE sobre el conjunto de *test*, lo que indica una mejor capacidad para reducir errores promedio.

Esta diferencia sugiere que *XGBoost* podría tener una mejor precisión en la predicción de valores pequeños o moderados. Esto puede explicarse por la naturaleza de su algoritmo de *boosting*, que ajusta sucesivamente los errores cometidos en iteraciones anteriores, permitiéndole capturar patrones más finos en los datos. No obstante, en términos de errores más grandes (capturados por el RMSE), ambos modelos muestran una capacidad muy similar.

Como conclusión, los dos modelos se muestran como herramientas válidas y sólidas para el problema de predicción de demanda. *Random Forest* destaca por su consistencia y robustez en todas las métricas, mientras que *XGBoost* ofrece una ventaja clara al minimizar el error absoluto promedio en *test*.

Esta observación motiva la próxima etapa del trabajo, donde se utilizará un conjunto de datos reducido, lo cual permitirá reducir la dimensionalidad y aplicar una búsqueda más exhaustiva de hiperparámetros. A partir de esta nueva exploración, se buscará optimizar el desempeño y seleccionar el modelo más adecuado para alimentar la estrategia de asignación logística posterior.

4.2. Modelado sobre datos agrupados

Para evaluar el desempeño de los modelos sobre un *dataset* más reducido y estratégico, se realizó una agrupación de clientes en dos grupos: los tres principales clientes (por volumen de compra) y un grupo agregado que concentra al resto. Esta simplificación permite explorar el comportamiento predictivo en un contexto más orientado a la logística y la planificación.

Al momento de evaluar las importancias de las variables, obtenemos resultados similares para todos los modelos, con una predominancia de entre el 80-90% de los datos de la base interna, con aportes marginales de datos climáticos y de la producción general de lácteos. No entraremos en detalle de sus valores para cada uno de los siguientes modelos para evitar repeticiones, sin

embargo, cuando obtengamos nuestro modelo definitivo, repasaremos las importancias relativas para verificar si se ha mantenido la tendencia.

4.2.1. Resultados con Random Forest

Hiperparámetros utilizados

Se utilizó el mismo procedimiento de modelado que en la sección anterior, manteniendo los mismos hiperparámetros del modelo *Random Forest*:

- *mtry* = 21: cantidad de variables consideradas en cada *split*, equivalente a la raíz cuadrada del total de variables predictoras.
- *min.node.size* = 10: cantidad mínima de observaciones por nodo terminal, para controlar la profundidad del árbol.
- *num.trees* = 50: cantidad total de árboles del bosque. Se utilizó un valor moderado para acelerar el entrenamiento.

Se trabajó nuevamente con un 20% aleatorio del *dataset* reducido, utilizando validación cruzada de 3 *folds*. Si bien el tamaño del set de datos en este escenario es significativamente más liviano, tanto el porcentaje de muestreo como la cantidad de pliegues no fueron alterados para luego comparar los modelos sobre datos completos y los modelos sobre datos agrupados bajo parámetros equivalentes y justos.

Evaluación de performance

Los resultados del modelo fueron los siguientes:

Métrica	**Train (CV)**	**Test**
RMSE	141,18	176,49
RMSE relativo (%)	7,67	6,60
R^2	0,56	0,65
MAE	16,51	31,00

Tabla 9: Performance de Random Forest con datos agrupados

A pesar de que el RMSE absoluto es considerablemente más alto que en el modelo anterior, esto se debe a que las cantidades promedio son mucho mayores: 18,40 en entrenamiento y 26,74 en testeo. El RMSE relativo, en cambio, es notablemente más bajo, lo cual indica una mejora significativa en la capacidad del modelo de capturar las variaciones reales de la demanda en este nuevo escenario.

El modelo entrenado sobre los datos agrupados logra una mejora clara en términos de error relativo. Esta diferencia se explica por la menor cantidad de valores nulos y por la mayor homogeneidad del comportamiento de los grupos considerados. Esto posiciona al enfoque de segmentación como una estrategia efectiva para mejorar la precisión y robustez del modelo predictivo en contextos logísticos.

4.2.2. Resultados con XGBoost

Hiperparámetros utilizados

Al igual que en el caso anterior, se entrenó un modelo *XGBoost* sobre el *dataset* simplificado (Top 3 clientes + resto), utilizando la misma muestra aleatoria del 20% para garantizar la

comparabilidad de los resultados. Se utilizaron los mismos hiperparámetros aplicados previamente:

- *nrounds* = 100: número total de iteraciones (árboles secuenciales).
- *eta* = 0.1: tasa de aprendizaje.
- *max_depth* = 6: profundidad máxima de los árboles.
- *gamma* = 0: sin penalización adicional por complejidad.
- *colsample_bytree* = 1: se usan todas las variables por árbol.
- *min_child_weight* = 1: mínimo peso de muestra por nodo.
- *subsample* = 1: todos los datos se usan por iteración.

Evaluación de performance

Métrica	Train (CV)	Test
RMSE	162,80	168,98
RMSE relativo (%)	8,85	6,32
R^2	0,45	0,68
MAE	16,14	20,28

Tabla 10: Performance de XGBoost con datos agrupados

En términos de error absoluto, el modelo muestra valores mayores a los de *Random Forest*, aunque con un descenso en el error relativo en el conjunto de testeo. Esto refleja una mejora en la capacidad de generalización en este contexto particular.

XGBoost se adapta de manera adecuada al nuevo *dataset* simplificado. Su capacidad de modelar relaciones no lineales y complejas le permite capturar patrones relevantes, especialmente cuando las cantidades son mayores y los ceros menos frecuentes. Su rendimiento es competitivo respecto a *Random Forest*, lo que justifica su consideración para la optimización de hiperparámetros.

4.2.3. Comparación entre Random Forest y XGBoost

A continuación, se presenta una comparación directa entre ambos modelos utilizando los mismos parámetros de validación y aplicados sobre el 20% de los datos agrupados:

Métrica	Random Forest	XGBoost
RMSE (Train CV)	141,18	162,80
RMSE (Test)	176,49	168,98
RMSE relativo (Train CV) (%)	7,67	8,85
RMSE relativo (Test) (%)	6,60	6,32
R^2 (Train CV)	0,56	0,45
R^2 (Test)	0,65	0,68
MAE (Train CV)	16,51	16,14
MAE (Test)	31,00	20,28

Tabla 11: Performance de Random Forest y XGBoost con datos agrupados

Como se observa en los resultados, *Random Forest* presenta un mejor rendimiento durante el entrenamiento, evidenciado por un menor RMSE, RMSE relativo, MAE y un mayor R^2 en validación cruzada. No obstante, cuando se evalúan los modelos sobre el conjunto de *test*, XGBoost muestra una mejor capacidad de generalización. En particular, obtiene menores valores de RMSE, RMSE relativo y MAE, junto con un R^2 más alto que el de *Random Forest*.

Este comportamiento puede interpretarse como un indicio de sobreajuste por parte del modelo de *Random Forest*: aunque ajusta mejor a los datos de entrenamiento, su rendimiento se degrada más en el conjunto de prueba. En contraste, *XGBoost*, a pesar de tener un RMSE más alto en entrenamiento, mantiene una mayor consistencia al evaluar datos no vistos. Esta diferencia puede atribuirse a la arquitectura de *boosting*, que adapta los árboles en función del error acumulado, mejorando la capacidad del modelo para capturar patrones que generalizan mejor.

Por último, cabe señalar que esta comparación se realizó utilizando los mismos hiperparámetros para ambos modelos. La superioridad de *XGBoost* en testeo sugiere que, aun sin una búsqueda exhaustiva de parámetros, logra adaptarse mejor a las características de los datos agrupados, lo que refuerza su robustez como herramienta predictiva en escenarios de alta variabilidad y menor granularidad. A pesar de ello, la diferencia entre los modelos no es drástica, y ambos muestran solidez predictiva. En las próximas secciones se procederá a realizar una optimización más profunda de hiperparámetros sobre ambos algoritmos, utilizando búsqueda en grilla y aleatoria para maximizar su rendimiento.

4.2.4. Comparación entre modelados completos y agrupados

Modelo	Dataset	RMSE Train	RMSE Rel. Train (%)	R^2 Train	MAE Train
Random Forest	Completo	37,91	29,92	0,44	1,40
XGBoost	Completo	38,82	30,64	0,42	1,33
Random Forest	Agrupado	141,18	7,67	0,56	16,51
XGBoost	Agrupado	162,80	8,85	0,45	16,14

Tabla 12: Performance de Random Forest y XGBoost para datos completos y agrupados

Como se observa, los modelos entrenados sobre el *dataset* agrupado muestran un RMSE absoluto más alto, pero un RMSE relativo significativamente menor. Esto es coherente con el aumento del promedio de la variable objetivo al trabajar con clientes más grandes y menos ceros, lo que reduce la distorsión generada por valores cercanos a cero en las métricas.

El modelo *Random Forest* con datos agrupados presenta la mejor combinación de R^2 y RMSE relativo, mientras que *XGBoost* con datos agrupados obtuvo el menor MAE, lo que lo posiciona como una opción más robusta frente a valores promedio.

Debemos considerar que la elección del mejor modelo se basará su rendimiento en el conjunto de entrenamiento, y como observamos, ambos algoritmos muestran un rendimiento aceptable. Dado que los resultados no son concluyentes y ambos modelos presentan fortalezas específicas —como un mejor R^2 en *Random Forest* y un menor MAE en *XGBoost*—, se trabajará con ambos enfoques para la próxima etapa de optimización avanzada de hiperparámetros y planificación logística.

Como conclusión general, obtenemos que el agrupamiento de los datos según tipo de cliente permitió mejorar la interpretación y el rendimiento relativo de los modelos. Dado que en contextos logísticos interesa prever mejor los grandes movimientos de carga y no tanto los pedidos pequeños, se concluye que el *dataset* reducido (agrupado) es más apropiado para continuar con la optimización de hiperparámetros y con la planificación logística posterior.

4.3. Modelado sobre datos agrupados – Optimización de hiperparámetros

4.3.1. Random Forest – Búsqueda por grilla

Parámetros de búsqueda

Con el objetivo de mejorar la performance del modelo, se entrenó un *Random Forest* sobre el *dataset* agrupado utilizando búsqueda en grilla para optimizar los hiperparámetros más influyentes.

Dado a que ya establecimos que la base agrupada de datos es la más apropiada para la posterior planificación logística, y siendo este set de datos bastante menor al original, es factible avanzar sin realizar submuestreo alguno, y con una validación cruzada de *5-folds* tal como lo recomienda la teoría.

Se exploraron distintas combinaciones de los siguientes hiperparámetros:

- *mtry*: número de variables seleccionadas aleatoriamente en cada *split*. Se evaluaron valores de $\sqrt{p}$, $log_2(p)$ y $0{,}2p$, donde p es el total de variables predictoras. Esta selección responde a recomendaciones ampliamente aceptadas en la literatura (Kuhn & Johnson, 2013, p. 387), ya que permiten equilibrar el sesgo y la varianza. Valores bajos de *mtry* promueven árboles más diversos (reduciendo la correlación entre ellos), mientras que valores altos tienden a reducir el error de cada árbol individual, aunque con menor diversidad. Dado el tamaño y complejidad del *dataset*, se incluyó un valor alto como $0{,}2p$ para explorar el beneficio de ampliar la cantidad de variables consideradas en cada *split*.
- *min.node.size*: tamaño mínimo de observaciones por nodo terminal. Se probaron valores de 5, 10 y 20. Esta selección busca cubrir un rango amplio de profundidad de los árboles. Valores más bajos permiten árboles más profundos y flexibles, que pueden adaptarse más finamente a los datos, aunque con mayor riesgo de sobreajuste. En cambio, valores más altos promueven árboles más simples que favorecen la generalización. Dado que el *dataset* agrupado tiene menos combinaciones posibles de cliente-producto, esta poda resulta clave para evitar complejidades innecesarias.
- *splitrule*: se mantuvo fijo en "*variance*" por tratarse de un problema de regresión.
- *num.trees*: 300 árboles, buscando una mayor robustez sin comprometer tiempos excesivos.

Resultados de la búsqueda en grilla

La combinación de hiperparámetros que menor RMSE obtuvo fue:

- *mtry* = 88
- *min.node.size* = 10

La combinación ganadora sugiere que un valor relativamente alto de *mtry* y un *min.node.size* intermedio resultan efectivos para capturar patrones finos en el *dataset*. Esto indica que, para

este problema, es preferible generar árboles más profundos y con mayor variedad de variables en cada división, maximizando la flexibilidad del modelo.

A continuación, se muestran las 5 mejores combinaciones encontradas, ordenadas por performance en RMSE:

mtry	min.node.size	RMSE	R^2	MAE
88	10	142,46	0,54	14,33
88	5	142,83	0,53	14,26
88	20	143,02	0,53	14,46
21	10	143,98	0,53	14,88
21	5	144,12	0,53	14,72

Tabla 13: Mejores 5 combinaciones de hiperparámetros en Random Forest (Grid Search)

Como se observa, el valor óptimo de *mtry* fue elevado, lo que resulta consistente con el alto número de variables predictoras del *dataset* agrupado. A mayor *mtry*, el modelo accede a más información en cada *split*, lo que puede favorecer la detección de relaciones relevantes cuando no existe tanto ruido o multicolinealidad severa. Esta estrategia es particularmente útil en *datasets* estructurados, donde muchas variables aportan información complementaria.

En cuanto a *min.node.size*, el mejor resultado se obtuvo con el valor 10. Este valor implica una poda algo más agresiva respecto a configuraciones más permisivas como 5, promoviendo árboles más simples y generalizables. En este caso, una estructura menos profunda parece haber sido beneficiosa para evitar el sobreajuste, posiblemente debido a que los patrones relevantes pueden capturarse adecuadamente sin necesidad de árboles excesivamente complejos.

En el siguiente gráfico podemos observar la performance de las diferentes combinaciones de hiperparámetros en función de su RMSE.

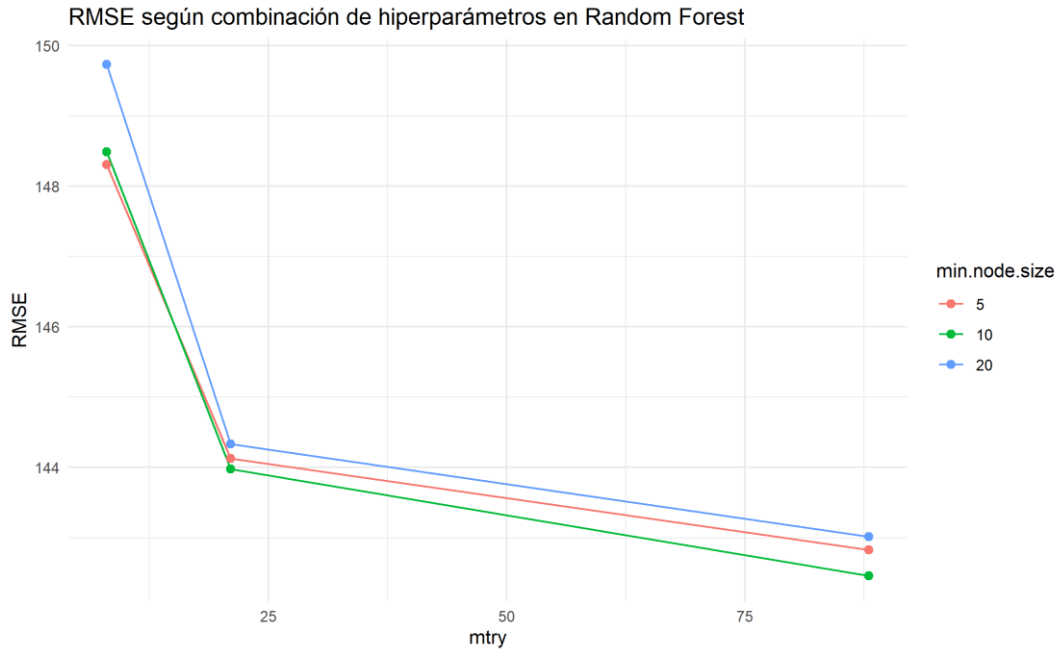

Figura 14: Performance por combinación de hiperp. en Random Forest (Grid Search)

El gráfico evidencia que el hiperparámetro *mtry* tiene una influencia considerable sobre el desempeño del modelo, con una clara reducción en el RMSE a medida que se incrementa su valor. En cambio, el parámetro *min.node.size* muestra un impacto mucho menor, ya que las curvas asociadas a sus distintos valores prácticamente se superponen. Esto sugiere que, para este problema, incrementar la cantidad de variables consideradas en cada división es clave para mejorar la precisión del modelo.

A continuación, se resumen las métricas de desempeño obtenidas para los datos de entrenamiento y prueba:

Conjunto	RMSE	R^2	MAE
Train	142,46	0,54	14,33
Test	158,54	0,71	25,99

Tabla 14: Performance de Random Forest con hiperparámetros optimizados (Grid Search)

Aunque el RMSE absoluto en *train* puede parecer elevado, es importante destacar que no debe compararse directamente con los valores obtenidos en pruebas anteriores sobre un subconjunto menor de datos del 20%, dado que las configuraciones de entrenamiento y validación son distintas. En este caso, el modelo fue entrenado con la totalidad del *dataset* agrupado y evaluado mediante validación cruzada de 5 pliegues. La consistencia entre las métricas obtenidas en entrenamiento y testeo sugiere que el modelo mantiene una buena capacidad de generalización, siendo notablemente superior a la capacidad de los modelos entrenados con un submuestreo del 20% y 3 pliegues. Este bajo nivel de sobreajuste observado —medido por la cercanía entre las métricas en conjuntos de *train* y *test*— refuerza la estabilidad y robustez del modelo seleccionado.

4.3.2. Random Forest – Búsqueda aleatoria

Parámetros de búsqueda

Con el objetivo de validar la robustez del modelo de *Random Forest* y explorar combinaciones adicionales de hiperparámetros más allá de las definidas manualmente en una grilla, se implementó una estrategia de optimización basada en búsqueda aleatoria (*Random Search*). Esta técnica permite cubrir de forma más amplia el espacio de combinaciones posibles con un menor costo computacional en comparación con una grilla exhaustiva, resultando especialmente útil cuando se trabaja con *datasets* de gran dimensión como en este caso.

Se definieron rangos razonables para los parámetros más sensibles del algoritmo:

- *mtry*: se sorteó entre 50 y 150 sin reemplazo, lo que garantiza un rango amplio sin correr riesgo de generar combinaciones excesivamente costosas en tiempo de cómputo.
- *min.node.size*: se seleccionó aleatoriamente entre 5, 10 y 15 con reemplazo, permitiendo así evaluar distintos niveles de poda del árbol.

En total se generaron 15 combinaciones aleatorias y se evaluaron mediante validación cruzada de 5 pliegues.

Resultados de la búsqueda aleatoria

La combinación de hiperparámetros que menor RMSE obtuvo fue:

- *mtry* = 123
- *min.node.size* = 10

La mejor combinación obtenida refuerza la tendencia observada en la grilla: valores altos de *mtry* conducen a un mejor rendimiento del modelo. Esto sugiere que, en este problema, permitir que cada árbol considere una mayor cantidad de variables en cada división resulta beneficioso para capturar relaciones relevantes en los datos.

A continuación, se muestran las 5 mejores combinaciones encontradas, ordenadas por performance en RMSE:

mtry	min.node.size	RMSE	R^2	MAE
123	10	141,95	0,54	14,22
84	5	141,99	0,54	14,21
86	15	142,18	0,54	14,35
135	15	142,28	0,54	14,27
146	5	142,34	0,54	14,15

Tabla 15: Mejores 5 combinaciones de hiperparámetros en Random Forest (Random Search)

Como se observa, el valor óptimo de *mtry* volvió a encontrarse en un rango elevado (123). Este resultado está en línea con la gran cantidad de variables disponibles en el *dataset* agrupado. Al permitir que cada árbol del bosque considere un mayor número de predictores en cada división (*split*), se incrementa la probabilidad de capturar relaciones significativas entre variables sin necesidad de estructuras extremadamente profundas. En *datasets* estructurados y con fuerte correlación entre variables, este comportamiento resulta esperable y deseable para mejorar la precisión del modelo.

En cuanto a *min.node.size*, nuevamente el mejor resultado se obtuvo con el valor intermedio de 10. Este valor representa un equilibrio entre profundidad y generalización: evita árboles excesivamente complejos (como podría suceder con nodos muy pequeños), pero tampoco realiza podas tan agresivas como *min.node.size* = 15. Esta selección sugiere que, en este contexto, es beneficioso construir árboles con una estructura de complejidad moderada, lo que permite capturar patrones complejos sin sobreajustarse a subconjuntos poco representativos.

En el siguiente gráfico podemos observar la performance de las diferentes combinaciones de hiperparámetros en función de su RMSE.

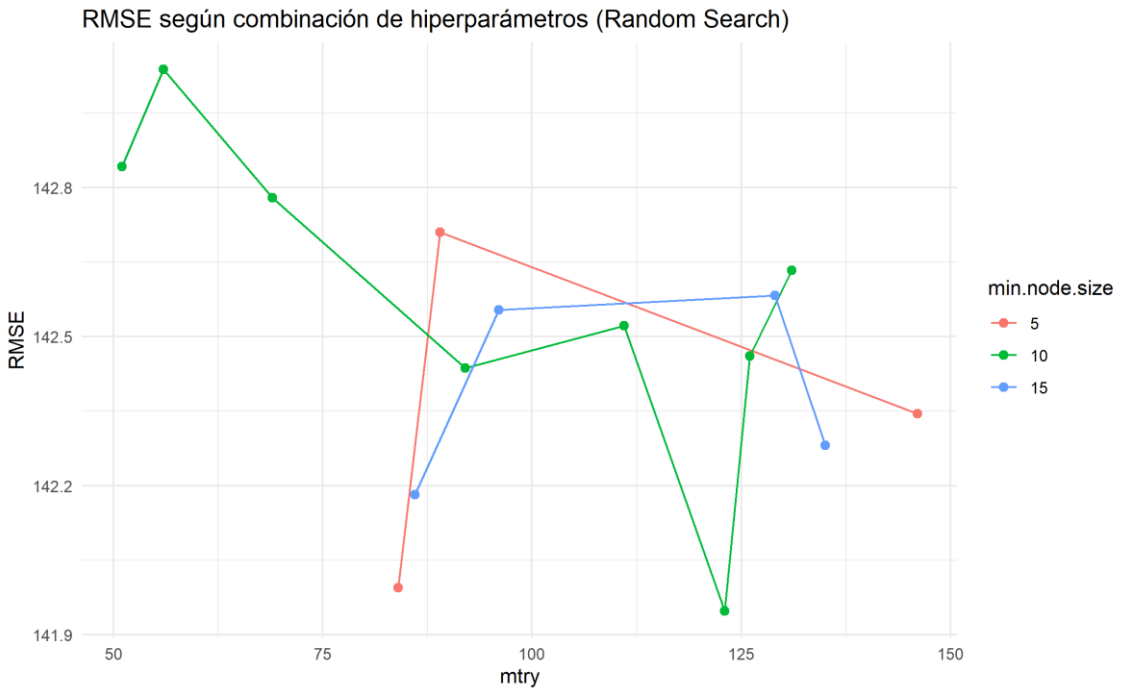

Figura 15: Performance por combinación de hiperp. en Random Forest (Random Search)

El gráfico sugiere nuevamente que valores más altos de *mtry* tienden a mejorar el desempeño del modelo, especialmente alrededor de 123, donde se alcanzó el menor RMSE. Sin embargo, también se destaca un rendimiento competitivo en torno a *mtry* = 80, que presenta un mínimo local interesante antes de ser superado por valores mayores. En cuanto a *min.node.size*, el mejor resultado se obtuvo con el valor 10, mientras que configuraciones más extremas (como 5 o 15) condujeron a un mayor error. Esto sugiere que una poda intermedia logra un equilibrio adecuado entre profundidad del árbol y capacidad de generalización.

A continuación, se resumen las métricas de desempeño obtenidas para los datos de entrenamiento y prueba:

Conjunto	RMSE	R^2	MAE
Train	141,95	0,54	14,22
Test	159,40	0,71	25,30

Tabla 16: Performance de Random Forest con hiperp. optimizados (Random Search)

Los resultados en entrenamiento muestran una mejora respecto al modelo ajustado mediante búsqueda por grilla. El RMSE en *train* disminuyó levemente (de 142,46 a 141,95), sin embargo, para los datos de prueba, el rendimiento fue ligeramente menor pasando de 158,54 a 159,40. Esto sugiere que el nuevo modelo encontrado tiene un ajuste más fino sobre el conjunto de entrenamiento, aunque podría estar comenzando a sobreajustarse, lo cual se refleja en una ligera pérdida de generalización.

No obstante, es importante aclarar que una mayor diferencia entre los errores de entrenamiento y prueba no implica necesariamente una peor calidad del modelo. La performance en el set de prueba no se utiliza como criterio de selección final, sino como una estimación del nivel de generalización sobre nuevas observaciones. Por lo tanto, esta métrica

está sujeta a la variabilidad inherente al conjunto de *test* utilizado, y debe interpretarse con cautela dentro del proceso de validación cruzada y experimentación.

4.3.3. XGBoost – Búsqueda por grilla

Parámetros de búsqueda

Para optimizar el rendimiento del modelo *XGBoost* se llevó a cabo una búsqueda en grilla (*grid search*) de combinaciones de hiperparámetros. La exploración se centró en seis parámetros clave que influyen directamente en la profundidad, la tasa de aprendizaje, la regularización y la diversidad de árboles en el modelo (Jain, 2016).

- *eta*: Tasa de aprendizaje. Un valor más bajo permite al modelo aprender de manera más conservadora, reduciendo el riesgo de sobreajuste, aunque requiere más iteraciones. Se seleccionaron valores de 0,01, 0,1 y 0,3.
- *max_depth*: Se refiere a la profundidad máxima del árbol, siendo que árboles más profundos pueden capturar relaciones complejas, pero también aumentan el riesgo de sobreajuste. Se seleccionaron valores de 4, 6 y 8.
- *gamma:* Reducción mínima de pérdida para una partición adicional. Se utilizaron valores de 0 y de 1.
- colsample_bytree: Fracción de columnas usadas por árbol. Se seleccionaron valores de 0,7 y de 1, es decir que se alternó entre el uso del 70% de las columnas y del total de ellas.
- min_child_weight: Peso mínimo de hijos para una partición. Se eligieron valores de 1 y 10.
- subsample: Fracción de muestras por árbol, la importancia de este parámetro radica en que submuestrear las observaciones ayuda a prevenir el sobreajuste y mejora la generalización del modelo. Se usaron valores de 0,7 y 1.

Este enfoque dio lugar a un total de 72 combinaciones posibles entrenadas sin hacer submuestreo, y evaluadas mediante validación cruzada de 5 *folds*, tal como se aplicó con los modelos de *Random Forest*.

Resultados de la búsqueda en grilla

La combinación de hiperparámetros que menor RMSE obtuvo fue:

- *eta* = 0,1
- *max_depth* = 8
- *gamma* = 0
- *colsample_bytree* = 0,7
- *min_child_weight* = 10
- *subsample* = 0,7

Esta combinación óptima presenta un equilibrio entre profundidad, regularización y diversidad. Un *max_depth* relativamente alto y un *min_child_weight* elevado indican un modelo que evita divisiones innecesarias, exigiendo una mayor cantidad de datos para generar nuevas particiones. Esto reduce el riesgo de sobreajuste y promueve árboles robustos, capaces de capturar relaciones sólidas sin caer en excesiva complejidad. Además, una tasa de aprendizaje baja (*eta* = 0,1) permite un ajuste progresivo y estable, mientras que los valores intermedios de *colsample_bytree* y *subsample* aportan diversidad estructural, mejorando la capacidad del modelo para generalizar frente a datos nuevos.

A continuación, se presentan los mejores resultados obtenidos ordenados por RMSE:

eta	max_ depth	gamma	colsample_ bytree	min_child_ weight	subsample	RMSE	R^2	MAE
0,1	8	0	0,7	10	0,7	140,16	0,55	15,11
0,1	6	1	0,7	10	1	140,48	0,55	14,89
0,1	6	0	1	10	1	140,62	0,55	14,97
0,1	6	1	1	10	1	140,62	0,55	14,97
0,1	8	0	0,7	10	1	141,07	0,54	14,72

Tabla 17: Mejores 5 combinaciones de hiperparámetros en XGBoost (Grid Search)

La observación de las mejores combinaciones permite identificar ciertos patrones consistentes. En primer lugar, el valor óptimo de *eta* = 0,1 confirma que una tasa de aprendizaje moderada permite lograr un equilibrio adecuado entre velocidad de convergencia y capacidad de generalización, sin caer en sobreajuste.

Además, el parámetro *max_depth* entre 6 y 8 sugiere que una estructura de árboles de profundidad intermedia es suficiente para capturar las relaciones significativas del *dataset* agrupado, evitando una complejidad excesiva. Lo mismo ocurre con *min_child_weight* = 10, que impone un umbral alto de representatividad para realizar divisiones, promoviendo modelos más conservadores frente a ruidos o valores atípicos.

La combinación de *colsample_bytree* = 0,7 y *subsample* = 0,7 indica que el modelo se beneficia de introducir cierto grado de aleatoriedad en la selección de predictores y observaciones para cada árbol. Esta estrategia de muestreo parcial permite reducir la varianza del modelo, mejorar la generalización y prevenir el sobreajuste, especialmente en contextos con alta dimensionalidad y posibles redundancias entre variables predictoras. No obstante, dado que en el top 5 de combinaciones estos valores oscilan entre 0,7 y 1, su efecto no parece ser determinante por sí solo, lo que sugiere que el desempeño del modelo está más influido por otros hiperparámetros como *eta*, *max_depth* y *min_child_weight*.

En conjunto, estos resultados reflejan un modelo robusto y controlado, que prioriza relaciones fuertes y evita ramificaciones innecesarias, lo cual es especialmente valioso cuando se trabaja con datos agrupados de alta dimensionalidad.

A fin de comprender la interacción y relevancia entre parámetros y su efecto sobre el RMSE, se utilizó un *heatmap* facetado que visualiza las combinaciones de *eta* y *max_depth*, diferenciadas por valores de *min_child_weight*, *colsample_bytree*, *gamma* y *subsample*.

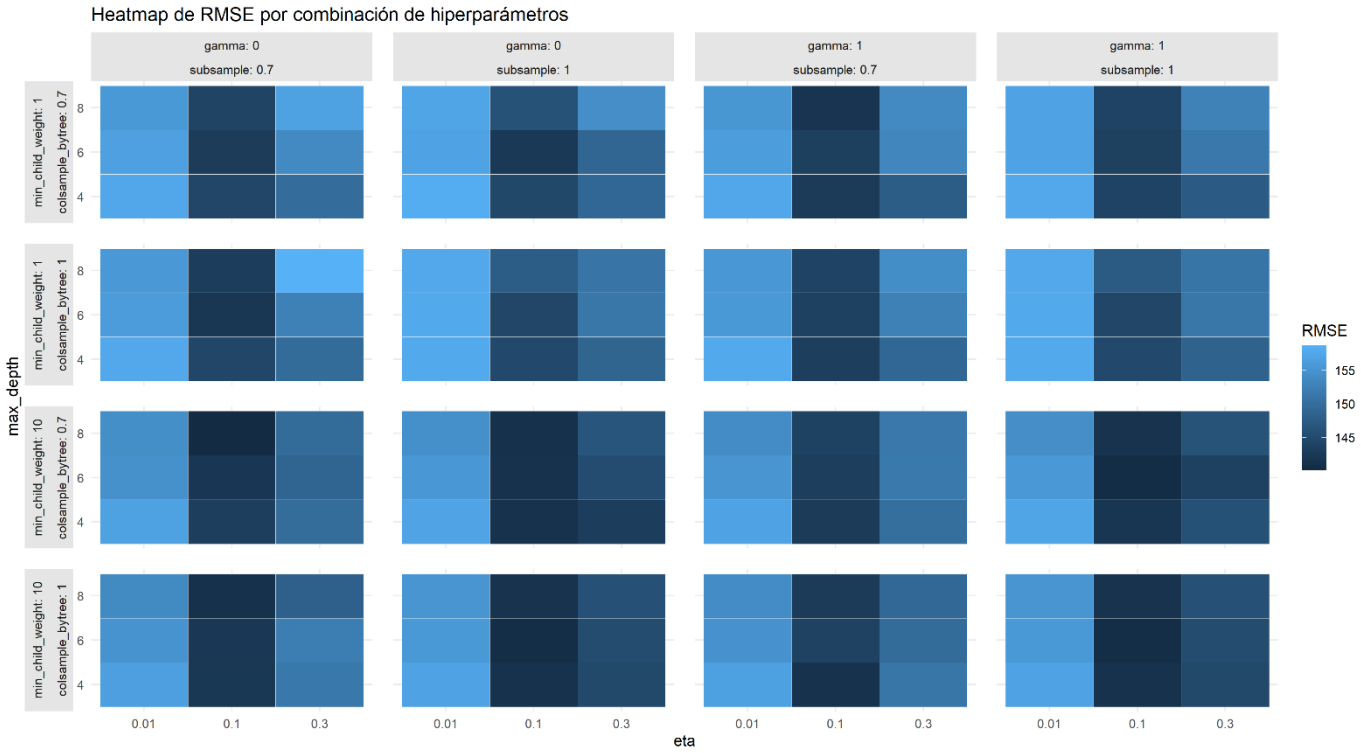

Figura 16: Performance por combinación de hiperparámetros en XGBoost (Grid Search)

A partir del gráfico, podemos destacar algunas tendencias claras. El valor de *eta* = 0,1 se posiciona como el más efectivo en prácticamente todos los escenarios evaluados, mostrando consistentemente los menores valores de RMSE. Esto indica que una tasa de aprendizaje moderada ofrece un buen equilibrio entre velocidad de convergencia y generalización, superando tanto a valores bajos (*eta* = 0,01) como altos (*eta* = 0,3). Además, se observa que las mejores combinaciones se concentran en valores intermedios de profundidad (*max_depth*) lo cual sugiere que una profundidad moderada permite capturar relaciones complejas sin incurrir en sobreajuste. En lo que respecta a los demás parámetros, si bien fueron explorados sistemáticamente en la búsqueda por grilla, sus efectos sobre el rendimiento del modelo resultaron ser secundarios. Dentro de cada panel facetado, los cambios en RMSE son poco pronunciados, lo cual indica que estas variables no influyeron de forma determinante en la capacidad predictiva del modelo, al menos dentro del rango de valores evaluado.

A continuación, se resumen las métricas de desempeño obtenidas para los datos de entrenamiento y prueba:

Conjunto	RMSE	R^2	MAE
Train	140,16	0,55	15,11
Test	160,63	0,71	20,66

Tabla 18: Performance de XGBoost con hiperparámetros optimizados (Grid Search)

Los resultados obtenidos muestran un desempeño sólido por parte del modelo XGBoost optimizado mediante búsqueda por grilla. Con un RMSE de 140,16 en entrenamiento, logró superar incluso al mejor modelo de *Random Forest*, lo que evidencia una gran capacidad de ajuste al conjunto de entrenamiento. Por su parte, el RMSE en test fue de 160,63, lo que representa una diferencia moderada respecto al entrenamiento, sin evidencias claras de sobreajuste y con buenos niveles de generalización.

No obstante, al comparar estos resultados con el modelo *Random Forest* ajustado mediante búsqueda aleatoria —que obtuvo un RMSE de 141,95 en *train* y 159,40 en *test*—, se observa que, si bien *XGBoost* logra un mejor ajuste en entrenamiento, presenta una leve desmejora en el conjunto de prueba. Esta diferencia es reducida y no concluyente, pero refuerza la idea de que las métricas en test deben interpretarse con cautela: no se utilizan para seleccionar el modelo, sino como una estimación aproximada de su capacidad de generalización, sujeta a la variabilidad del conjunto de evaluación.

Este resultado también abre la posibilidad de que la configuración óptima de XGBoost no haya sido alcanzada en la búsqueda por grilla, y que una exploración más amplia mediante búsquedas aleatorias pueda dar lugar a combinaciones de hiperparámetros más eficaces, con un mejor equilibrio entre ajuste y generalización.

4.3.4. XGBoost – Búsqueda aleatoria

Parámetros de búsqueda

Para complementar el enfoque anterior, se realizó una búsqueda aleatoria (*random search*) sobre el espacio de hiperparámetros de *XGBoost*, con el objetivo de explorar combinaciones alternativas que pudieran mejorar la performance del modelo. Esta estrategia resulta útil cuando el espacio de búsqueda es extenso y no se cuenta con indicios suficientes para definir una grilla eficiente.

Se definieron 30 combinaciones aleatorias que fueron evaluadas mediante validación cruzada de 5 *folds* sobre el *dataset* agrupado completo. Tales combinaciones fueron generadas considerando los siguientes rangos para cada hiperparámetro:

- *eta*: Tasa de aprendizaje. Un valor más bajo permite al modelo aprender de manera más conservadora, reduciendo el riesgo de sobreajuste, aunque requiere más iteraciones. Se usaron valores aleatorios entre 0,01 y 0,3, redondeados a 3 espacios decimales.
- *max_depth*: Se refiere a la profundidad máxima del árbol, siendo que árboles más profundos pueden capturar relaciones complejas, pero también aumentan el riesgo de sobreajuste. Se usaron valores enteros aleatorios entre 3 y 9.
- *gamma:* Reducción mínima de pérdida para una partición adicional. Se utilizaron valores aleatorios entre 0 y 1, redondeados a 2 espacios decimales.
- colsample_bytree: Fracción de columnas usadas por árbol. Se usaron valores aleatorios entre 0,5 y 1, redondeados a 2 espacios decimales.
- min_child_weight: Peso mínimo de hijos para una partición. Se eligió aleatoriamente entre valores fijos de 1, 5 y 10.
- subsample: Fracción de muestras por árbol, la importancia de este parámetro radica en que submuestrear las observaciones ayuda a prevenir el sobreajuste y mejora la generalización del modelo. Se usaron valores aleatorios entre 0,5 y 1, redondeados a 2 espacios decimales.

Estos rangos fueron definidos en base a valores recomendados en la literatura y al comportamiento observado durante la búsqueda por grilla.

Resultados de la búsqueda aleatoria

La combinación de hiperparámetros que menor RMSE obtuvo fue:

- *eta* = 0,084
- *max_depth* = 4
- *gamma* = 2,56
- *colsample_bytree* = 0,53
- *min_child_weight* = 10
- *subsample* = 0,97

Esta combinación no logró superar el rendimiento del mejor modelo encontrado mediante búsqueda por grilla, especialmente en el conjunto de entrenamiento. Este resultado demuestra que, si bien la búsqueda aleatoria permite explorar regiones más amplias del espacio de hiperparámetros, no siempre garantiza mejoras sustanciales en el desempeño. A continuación, se presentan los mejores resultados obtenidos ordenados por RMSE:

eta	max_ depth	gamma	colsample_ bytree	min_child_ weight	subsample	RMSE	R^2	MAE
0,084	4	2,56	0,53	10	0,97	142,21	0,53	15,50
0,038	9	2,47	0,51	1	0,99	142,37	0,53	14,57
0,105	8	4,91	0,80	5	0,58	143,16	0,53	15,14
0,088	5	4,96	0,96	5	0,60	143,53	0,53	15,61
0,117	3	3,93	0,87	5	0,68	143,79	0,53	16,05

Tabla 19: Mejores 5 combinaciones de hiperparámetros en XGBoost (Random Search)

A partir de los modelos con menor RMSE, se observó nuevamente que los parámetros *eta* (tasa de aprendizaje) y *max_depth* (profundidad máxima de los árboles) mostraban una clara relación con el rendimiento. En particular, valores bajos de *eta* y profundidades intermedias se asociaron con los mejores resultados. Esta regularidad sugiere una alta sensibilidad del modelo a estos dos hiperparámetros, convirtiéndolos en buenos candidatos para ser representados gráficamente.

En cambio, otros parámetros como *gamma*, *subsample*, *colsample_bytree* y *min_child_weight* no exhibieron un patrón claro entre sus valores y el RMSE. Su distribución dentro del top de modelos fue heterogénea y no permitió inferir una tendencia sistemática. Por este motivo, se decidió mantener fijos estos hiperparámetros y concentrar el análisis visual exclusivamente en *eta* y *max_depth*, para así facilitar una representación más clara y enfocada del espacio de búsqueda más prometedor.

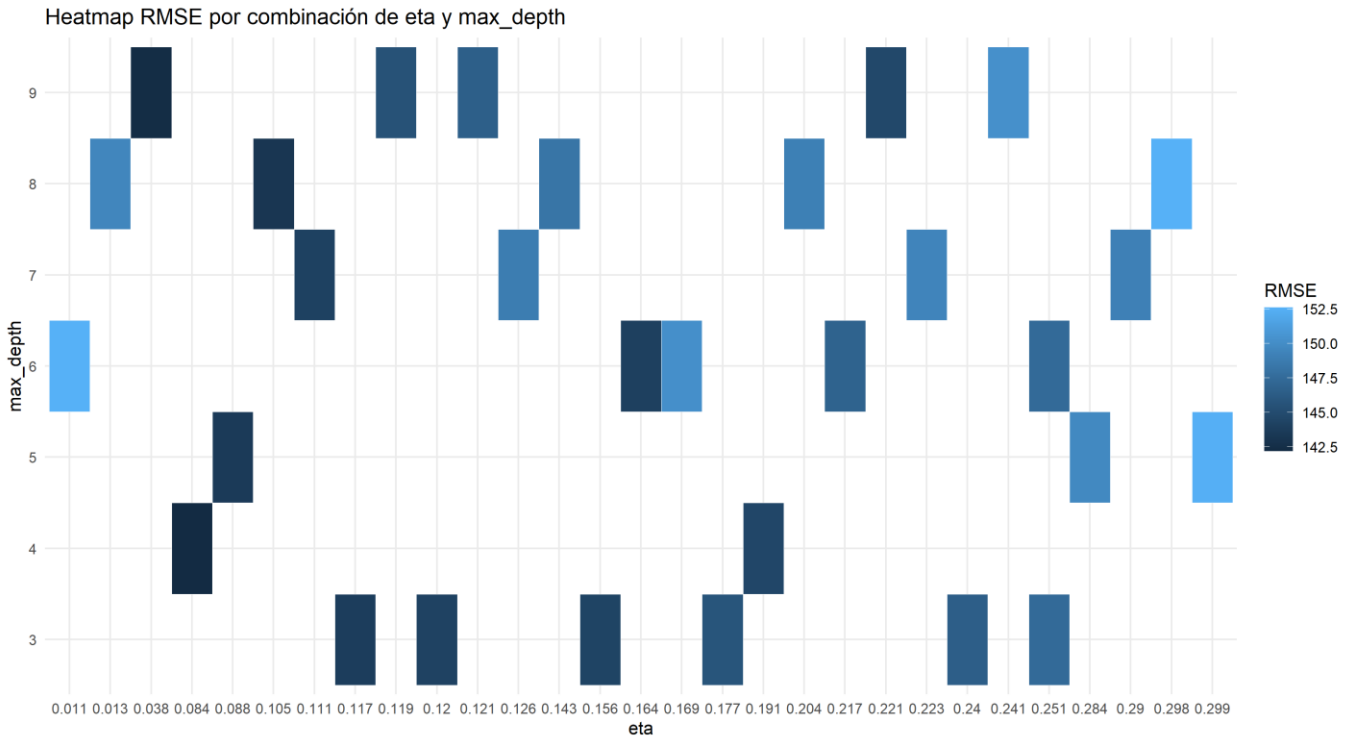

Figura 17: Performance por combinación de hiperparámetros en XGBoost (Random Search)

A partir del gráfico anterior, se observa que las combinaciones que arrojaron los menores valores de RMSE se concentran en la región donde *eta* es bajo (entre 0,013 y 0,12). Esto sugiere que el modelo se beneficia de tasas de aprendizaje pequeñas, lo cual favorece un proceso de entrenamiento más gradual y menos propenso al sobreajuste.

Asimismo, se evidencia que profundidades moderadas (*max_depth* entre 3 y 6) tienden a lograr mejor rendimiento que profundidades más elevadas, lo cual puede explicarse por la capacidad de estos árboles para capturar interacciones relevantes sin sobreajustarse al ruido del *dataset*. Por el contrario, combinaciones con *eta* más altas (mayores a 0.2) o *max_depth* extremos exhiben consistentemente un RMSE superior, lo cual refuerza la importancia de mantener un modelo balanceado en términos de complejidad y tasa de aprendizaje.

A continuación, se resumen las métricas de desempeño obtenidas para los datos de entrenamiento y prueba:

Conjunto	RMSE	R^2	MAE
Train	142,21	0,53	15,50
Test	159,23	0,71	20,49

Tabla 20: Performance de XGBoost con hiperparámetros optimizados (Random Search)

Al comparar los resultados del modelo obtenido mediante búsqueda aleatoria con los alcanzados por la búsqueda por grilla, se observa que el error en entrenamiento (RMSE = 142,21) es ligeramente superior al del mejor modelo anterior (RMSE = 140,16). Sin embargo, el desempeño en el conjunto de *test* muestra un comportamiento inverso: el modelo por *random search* presenta un RMSE de 159,23, mejor que el 160,63 obtenido por el modelo ajustado por grilla.

Esta diferencia, aunque pequeña, podría deberse a una mayor robustez del modelo generado por búsqueda aleatoria frente a ciertos patrones del conjunto de prueba. No obstante, nuevamente, dado que la selección del modelo se basa exclusivamente en su rendimiento sobre el conjunto de entrenamiento validado por *cross-validation*, la métrica en *test* debe tomarse como una referencia no decisiva.

En este sentido, los resultados refuerzan la validez de los hiperparámetros seleccionados originalmente mediante búsqueda por grilla, ya que incluso al ampliar aleatoriamente el espacio de combinaciones posibles, no se logró una mejora sustancial del modelo en entrenamiento. Esto sugiere que el espacio acotado utilizado en la búsqueda inicial ya capturaba configuraciones efectivas para el problema planteado.

4.4. Selección del mejor modelo y conclusión del modelado predictivo

En esta sección se presenta una síntesis del proceso seguido para construir y evaluar los modelos predictivos, así como la selección final del modelo a utilizar en las etapas siguientes del trabajo.

El análisis comenzó comparando el desempeño del modelo sobre dos versiones del *dataset*: uno con los datos completos y otro con los datos previamente agrupados por los mayores clientes. Dado el tamaño original del *dataset*, se aplicó un muestreo aleatorio del 20% para facilitar el procesamiento inicial, y se usó un esquema de validación cruzada de solo 3 pliegues. Si bien el modelo entrenado sobre los datos agrupados presentó un RMSE absoluto mayor, al comparar el RMSE relativo al promedio de la variable objetivo se observó un mejor rendimiento, lo cual justificó continuar el modelado y ajuste de hiperparámetros utilizando esta versión agrupada de los datos.

En la sección 4.2 se entrenaron ambos modelos (*Random Forest* y *XGBoost*) utilizando ese 20% de los datos agrupados, también con validación cruzada de 3 *folds*. En esta instancia inicial y sin optimización de parámetros, *Random Forest* mostró un RMSE de entrenamiento menor que *XGBoost*, lo que indicaba un mayor poder de ajuste. Sin embargo, al evaluar ambos modelos sobre el conjunto de *test*, *XGBoost* logró mantener un desempeño más parejo, mientras que *Random Forest* tendió al sobreajuste, con un incremento notorio en su error. Esto sugería que, en esa etapa, el modelo de *XGBoost* era más parsimónico y generalizaba mejor.

No obstante, al ampliar el entrenamiento al 100% de los datos agrupados y con validación cruzada de 5 pliegues en la sección 4.3, esta tendencia cambió. Ambos modelos mostraron un comportamiento más estable, y las diferencias en rendimiento entre entrenamiento y test se redujeron. El mayor volumen de datos permitió a *Random Forest* reducir su tendencia al sobreajuste y alcanzar una mejor correspondencia entre los conjuntos.

A partir de allí, se avanzó en la optimización de hiperparámetros, implementando primero una búsqueda por grilla y luego una búsqueda aleatoria con rangos más amplios, definidos a partir de los patrones observados en los resultados de la grilla y del análisis de la sensibilidad de cada hiperparámetro. En el caso de *Random Forest*, la búsqueda aleatoria permitió mejorar las métricas de entrenamiento respecto a las combinaciones encontradas por grilla, validando el enfoque de ampliar el espacio de búsqueda más allá de configuraciones manuales. Sin embargo, en *XGBoost* no se logró una mejora del rendimiento en entrenamiento al aplicar búsqueda aleatoria, lo que sugiere que los hiperparámetros definidos en la búsqueda por grilla ya capturaban combinaciones cercanas al óptimo en el espacio explorado.

A continuación, se resumen las métricas de desempeño en entrenamiento de los mejores modelos obtenidos para cada algoritmo:

Modelo	RMSE	R^2	MAE
Random Forest	141,95	0,54	14,22
XGBoost	140,16	0,55	15,11

Tabla 21: Métricas de entrenamiento del mejor modelo encontrado para cada algoritmo

Como puede observarse, el modelo optimizado de *XGBoost* logró un rendimiento superior al de *Random Forest* en las métricas de RMSE y R^2 sobre el conjunto de entrenamiento, mientras que *Random Forest* presentó un MAE levemente inferior. Esta diferencia sugiere que *XGBoost* es más efectivo para capturar adecuadamente los errores asociados a volúmenes de demanda elevados, dado que el RMSE penaliza más fuertemente las desviaciones grandes. Por ello, y considerando además su comportamiento consistente al utilizar el total del *dataset*, se justifica avanzar con *XGBoost* como el modelo seleccionado para la etapa de pronóstico.

Habiendo establecido el mejor modelo encontrado, tal como se mencionó al principio del título 4.2., se procede a verificar si la tendencia en el uso e importancia de variables predictoras se mantuvo. A continuación, se muestran las principales 5 variables utilizadas por el mejor modelo de XGBoost, ordenadas de mayor a menor por importancia relativa:

- Cantidad: 64,47%
- LogCantidad: 17,93%
- LogValorSachetLeche: 2,59%
- TemperaturaMaxima: 2,44%
- LluviaPromedio: 2,09%

De manera análoga al análisis del título 4.1., las variables utilizadas también fueron clasificadas según su fuente de origen: datos internos, climáticos, económicos o contextuales. A partir de esta clasificación, se calculó el aporte acumulado por cada grupo de variables al modelo:

Fuente	Importancia Relativa
Base interna	89,10%
Servicio Meteorológico Nacional	4,88%
Dirección Nacional de Lechería	4,32%
INDEC	0,97%
Dólar Blue	0,73%

Tabla 22: Importancias relativas por fuente del mejor modelo obtenido

Los resultados presentados reafirman la tendencia observada previamente en la sección 4.1, cuando se evaluaron las importancias relativas del modelo ajustado sobre los datos desagregados. En ambos casos, las variables más relevantes provienen de la base interna, y están asociadas directamente al comportamiento histórico de la demanda. Las cantidades demandadas en semanas anteriores para cada combinación cliente-producto-semana, junto con su versión logarítmica, continúan siendo los predictores con mayor aporte explicativo. A estos se suma el logaritmo del valor relativo del producto respecto a la leche (LogValorSachetLeche),

lo que refuerza la idea de que el modelo detecta patrones vinculados tanto al comportamiento pasado como al posicionamiento relativo del precio de cada producto.

En términos de agrupación por fuente, se confirma nuevamente la fuerte predominancia de las variables internas, que concentran más del 89% de la importancia explicativa total. Entre las fuentes externas, el Servicio Meteorológico Nacional y la Dirección Nacional de Lechería presentan una participación moderada, mientras que los datos del INDEC y el dólar blue muestran una contribución residual.

Este patrón sugiere que, por un lado, las variables internas capturan adecuadamente la lógica de reposición y estacionalidad implícita en los hábitos de compra. A la vez, el aporte del clima parece tener cierta relevancia, posiblemente asociada a fluctuaciones de consumo estacional o al impacto logístico en determinadas semanas. Por su parte, la influencia de la oferta, reflejada en los indicadores productivos de la Dirección Nacional de Lechería, podría estar actuando como un factor indirecto: es razonable suponer que en semanas de mayor producción haya un esfuerzo comercial adicional que potencie la demanda.

Este análisis final no solo permite validar la calidad del modelo desde el punto de vista predictivo, sino también comprender mejor los factores que inciden en la demanda y cómo estos pueden ser aprovechados para tomar decisiones operativas más informadas. Con base en este entendimiento, y habiendo definido al modelo de XGBoost como la mejor alternativa para anticipar la demanda semanal a nivel cliente-producto, en la siguiente sección se explora su aplicación práctica dentro de una estrategia de optimización logística.

En particular, se utilizarán las predicciones generadas por el modelo de XGBoost como insumo para el desarrollo de una estrategia de optimización en los envíos de mercadería, que permite mejorar su planificación y reducir costos asociados a la variabilidad de la demanda. Esta integración entre predicción y decisión constituye el segundo eje principal del trabajo, donde se busca traducir el valor del modelo predictivo en mejoras operativas concretas.

5. Metodología de optimización logística

5.1. Fundamento del enfoque basado en datos

Esta sección describe en detalle el proceso desarrollado para simular una mejora en la eficiencia logística a partir de las predicciones del modelo de demanda. El objetivo es reducir la cantidad total de camiones necesarios para la distribución sin alterar el *mix* de productos entregados ni superar las cantidades efectivamente demandadas por los principales clientes.

En lugar de implementar un modelo de optimización matemática formal, se optó por una metodología heurística basada en manipulación de datos, que permitió simular decisiones operativas dinámicas mediante un procedimiento secuencial. Esta lógica se apoya en reglas operativas simples pero claves, que permiten capturar de forma realista las decisiones que podrían tomarse en un entorno logístico real ante la disponibilidad de predicciones semanales confiables.

Este tipo de enfoques heurísticos, aplicados sobre estructuras de datos enriquecidas, ha demostrado ser eficaz en problemas complejos de logística y transporte, especialmente cuando se busca flexibilidad, simplicidad de implementación y resultados interpretables (Deniz & Ozceylan, 2023). A su vez, el uso de enfoques heurísticos combinados con simulaciones ha sido explorado en trabajos como "*Simulation-based optimization in transportations and logistics"* (Juan et al., 2019), donde se ha desarrollado el concepto de *simheuristics*: una metodología que permite tomar decisiones operativas mediante reglas simples, incorporando la incertidumbre de la demanda a través de escenarios simulados. Este enfoque resulta especialmente útil en problemas logísticos dinámicos, similares al contexto de distribución semanal abordado en este trabajo, donde las decisiones deben adaptarse en función de predicciones sujetas a variabilidad.

Así, la metodología adoptada en este estudio se alinea con las prácticas contemporáneas de optimización logística impulsadas por datos, que buscan mejoras medibles en el desempeño sin requerir necesariamente la formulación y resolución de modelos matemáticos rígidos (Advanced Logistics, 2025). Esta decisión metodológica permite una implementación replicable y flexible, adaptada a la lógica de operación real del reparto, sin recurrir a software especializado ni a supuestos fuertes sobre los costos de transporte o penalizaciones por asignación.

5.2. Enfoque general y objetivos

El análisis se centró exclusivamente en los tres clientes más relevantes del canal mayorista, que realizan pedidos por unidad de pallet. Esto permitió simplificar la simulación considerando solamente la parte del camión principal destinada a este tipo de entregas (10 pallets), excluyendo el segmento minorista (2 pallets) que no condiciona la optimización, ya que sus volúmenes son bajos y estables.

La simulación se realizó sobre el conjunto de *test* utilizado en la evaluación del modelo predictivo, es decir, las últimas semanas del año 2024, con el fin de evitar cualquier solapamiento entre datos de entrenamiento (80%) y evaluación (20%).

El objetivo fue simular qué decisiones podrían haberse tomado si se hubiera contado con las predicciones semanales del modelo y se hubieran adelantado pedidos dentro de la misma semana para aprovechar al máximo la capacidad de carga diaria, reduciendo así la cantidad total de camiones requeridos.

5.3. Fuentes de datos utilizadas

Se emplearon tres fuentes principales de información:

- Pedidos reales diarios: registros por cliente, producto y día.
- Predicciones semanales: resultado del mejor modelo de predicción, por cliente y producto.
- Tabla de conversión a pallets: relevamiento propio que indica cuántas unidades conforman un pallet para cada producto relevante.

Ambos conjuntos de datos (reales y predichos) fueron transformados a unidades de pallet, ya que es la forma en que se gestiona la operación logística con estos clientes. A su vez, los datos diarios fueron enriquecidos con la identificación del año, semana del año y número de día dentro de la semana (1 a 6), para permitir su uso en una lógica iterativa.

5.4. Relevamiento de cantidades por pallet

Como fue adelantado en el subtítulo que antecede, para poder simular de forma realista el comportamiento logístico de los pedidos, fue necesario contar con información sobre cuántas unidades componen un pallet de cada producto. Esta información no se encontraba directamente disponible en el sistema, por lo que fue relevada manualmente a partir de pedidos históricos registrados en el sitio interno del fletero. Se analizaron más de dos años de datos, identificando aquellos productos que fueron solicitados recurrentemente en formato palletizado por los principales clientes mayoristas.

En total, se identificaron 60 productos distintos que fueron solicitados al menos una vez por pallet en los últimos dos años. Estos productos comprenden principalmente lácteos como leches, yogures, mantecas, quesos rallados y dulces de leche. El volumen de pallets pedidos por producto es heterogéneo, pero se observa una clara concentración en un subconjunto reducido de productos, siendo los primeros en el ranking responsables de una proporción significativa del volumen total.

A continuación, se presenta una tabla con los 20 productos más pedidos por pallet, ordenados por volumen total de pallets solicitados. Esta información fue esencial para convertir tanto las cantidades predichas como las observadas a unidades de pallet, lo que permitió comparar, reasignar y evaluar la ocupación de camiones de forma coherente con la lógica operativa de la distribución.

Descripción del producto	Cantidad por pallet	Pallets pedidos totales
Leche 1ra Entera 3% Sachet	918	957
Queso 1ra Rallado Caja Sobre 35gr	720	622
Leche 2da Entera 3% Sachet	918	213
Leche 1ra Uat Entera 3% Pet	900	134
Leche 1ra Uat Entera 3% Brik	900	133
Yogur 1ra Clásico Sachet 900gr Frutilla	1026	117
Manteca 1ra Paquete 100gr	1000	112
Yogur 1ra Clásico Sachet 900gr Vainilla	1026	83
Dulce De Leche 1ra Colonial Pote 400gr	1008	58
Yogur 1ra Firme Entero 190gr Vainilla	880	52
Leche 1ra P. Desc. 1% Sachet	918	50
Dulce De Leche 1ra Colonial Pote 250gr	1008	47

Yogur 1ra Batido Entero C/Cereales 159gr	880	52
Leche 1ra Uat P. Desc. 1% Brik	900	51
Leche 1ra Uat P. Desc. 1% Pet	918	45
Dulce De Leche 1ra Clásico Pote 400gr	1008	45
Yogur 1ra Firme Entero 190gr Frutilla	880	43
Leche 1ra Uat P. Desc. Zero Lactosa	816	38
Leche 1ra Chocolatada 900gr	900	33
Dulce De Leche 1ra Clásico Pote 250gr	1008	29

Tabla 23: Top 20 productos por cantidad de pallets pedidos

5.5. Preprocesamiento de pedidos reales

Los pasos aplicados sobre los datos de pedidos reales fueron los siguientes:

- → Se filtraron los registros correspondientes a los tres clientes principales.
- → Se unieron con la tabla de equivalencias para obtener los pedidos expresados en pallets.
- → Se asignaron columnas auxiliares de año, número de semana y número de día dentro de cada semana.
- → Se identificó la cantidad de días hábiles en cada semana y se creó un identificador de día relativo (de 1 a 6).
- → Finalmente, se calculó la cantidad total de pallets por día para cada semana y se derivó la capacidad ociosa diaria, utilizando reglas operativas según el volumen diario total:

$$CO_d = \begin{cases} 40 - \lceil P_d \rceil \; si \; P_d > 30 \\ 30 - \lceil P_d \rceil \; si \; 20 < P_d \leq 30 \\ 20 - \lceil P_d \rceil \; si \; 10 < P_d \leq 20 \\ 10 - \lceil P_d \rceil \; si \; P_d \leq 10 \end{cases}$$

Donde:

- CO_d es la capacidad ociosa del día d (en pallets).
- P_d es la cantidad de pallets reales enviados ese día.
- $\lceil . \rceil$ representa la función de redondeo hacia arriba.

Estas reglas reflejan el uso de uno, dos, tres o hasta cuatro camiones en simultáneo, considerando la capacidad máxima de 10 pallets por camión.

5.6. Preprocesamiento de predicciones

Para las predicciones semanales, se aplicaron los siguientes pasos:

- → Se transformaron las cantidades predichas a unidades de pallet mediante la misma tabla de equivalencias.
- → Las predicciones se redondearon al entero más cercano (por ejemplo, 7,9 → 8; 7,1 → 7), replicando el comportamiento operativo ante pedidos reales.
- → Se unieron con la variable target real (también en pallets), lo que permitió derivar una nueva variable clave:

$$P_{cps}^{reas} = \min\left(\hat{P}_{cps}, P_{cps}^{real}\right)$$

Donde:

- P_{cps}^{reas} es la cantidad de pallets disponibles para reasignación del cliente c, producto p, semana s.
- $\hat{P}_{cps}$ es la cantidad de pallets predicha por del modelo.
- P_{cps}^{real} es la cantidad de pallets efectivamente pedida para esa semana.

Este procedimiento de preprocesamiento permitió construir, para cada combinación de semana, cliente y producto, la variable clave que representa la cantidad de pallets que podrían haberse ofrecido anticipadamente, dentro de los márgenes ya comprometidos por la demanda real. A continuación, se presenta un diagrama de flujo que resume gráficamente la lógica utilizada para calcular dicha variable de disponibilidad. Este diagrama refleja el recorrido lógico aplicado sobre los datos predichos y reales, considerando los diferentes escenarios posibles y estableciendo con precisión los casos en los que el cliente podría aceptar una entrega anticipada sin incurrir en sobreentregas.

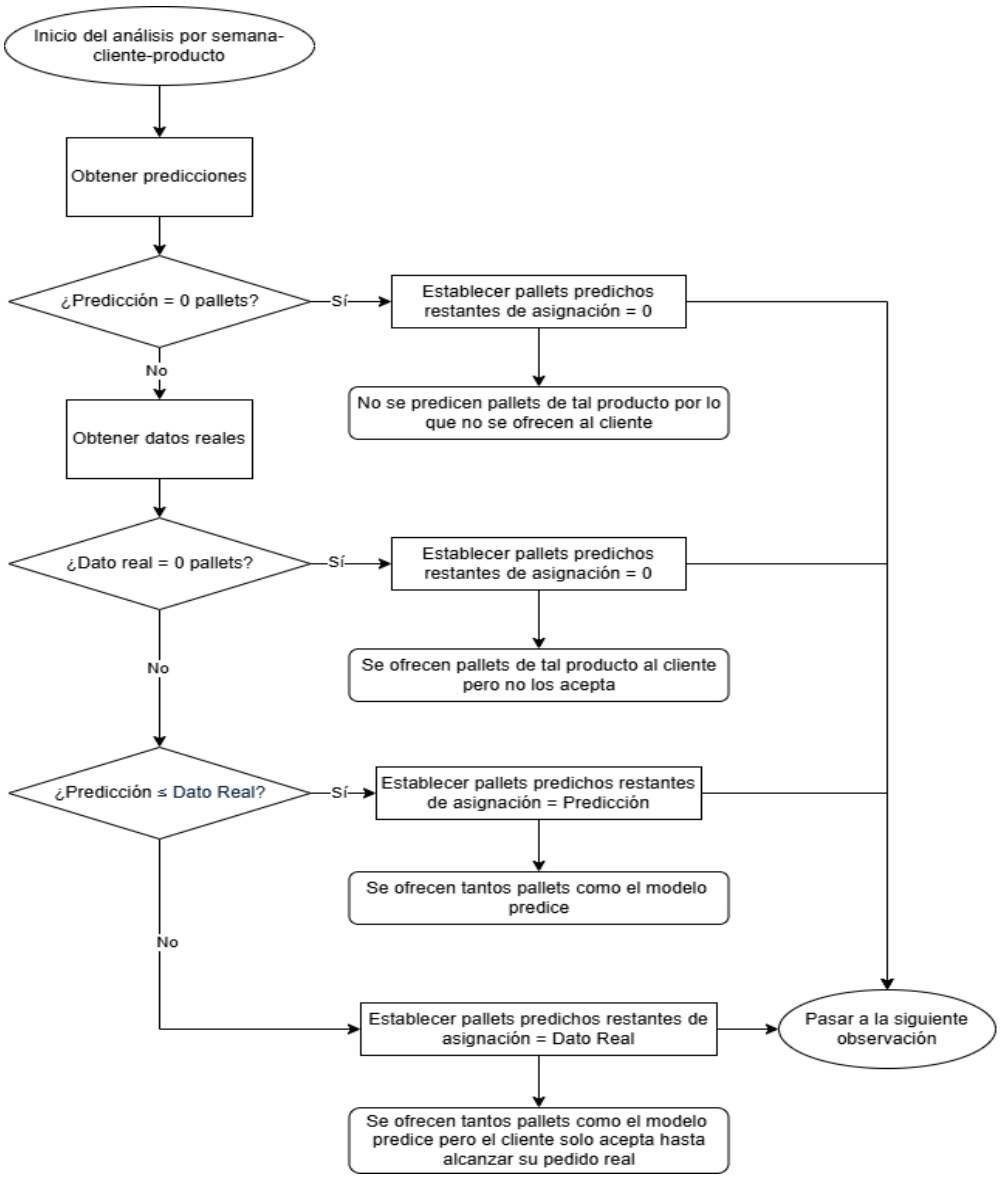

Figura 18: Diagrama de flujo para los pallets a reasignar

5.7. Estructura del algoritmo secuencial

Una vez construidas ambas bases de datos sobre cantidades palletizadas, se integraron mediante un *pivot* y uniones por cliente-producto-semana. Se obtuvo una tabla con:

- Una columna para cada día de la semana (de 1 a 6), con la cantidad de pallets realmente pedidos.
- Una columna con la variable de "Pallets restantes de reasignación".
- Una fila por cliente y producto.

A partir de esta estructura se ejecutó el procedimiento secuencial semana por semana.

A fin de facilitar la comprensión del procedimiento implementado para la reasignación de pallets, a continuación, se presenta un diagrama de flujo que resume gráficamente la lógica del algoritmo desarrollado. En este se pueden observar los ciclos semanales y diarios, las condiciones que guían el proceso (como la existencia de pallets restantes o capacidad ociosa disponible), y las acciones que se ejecutan en cada etapa, desde la actualización de variables hasta la reasignación anticipada de pedidos.

El diagrama refleja el carácter iterativo de la metodología, en el cual las decisiones tomadas en un día afectan las condiciones operativas de los días siguientes, reforzando así la naturaleza dinámica del enfoque adoptado.

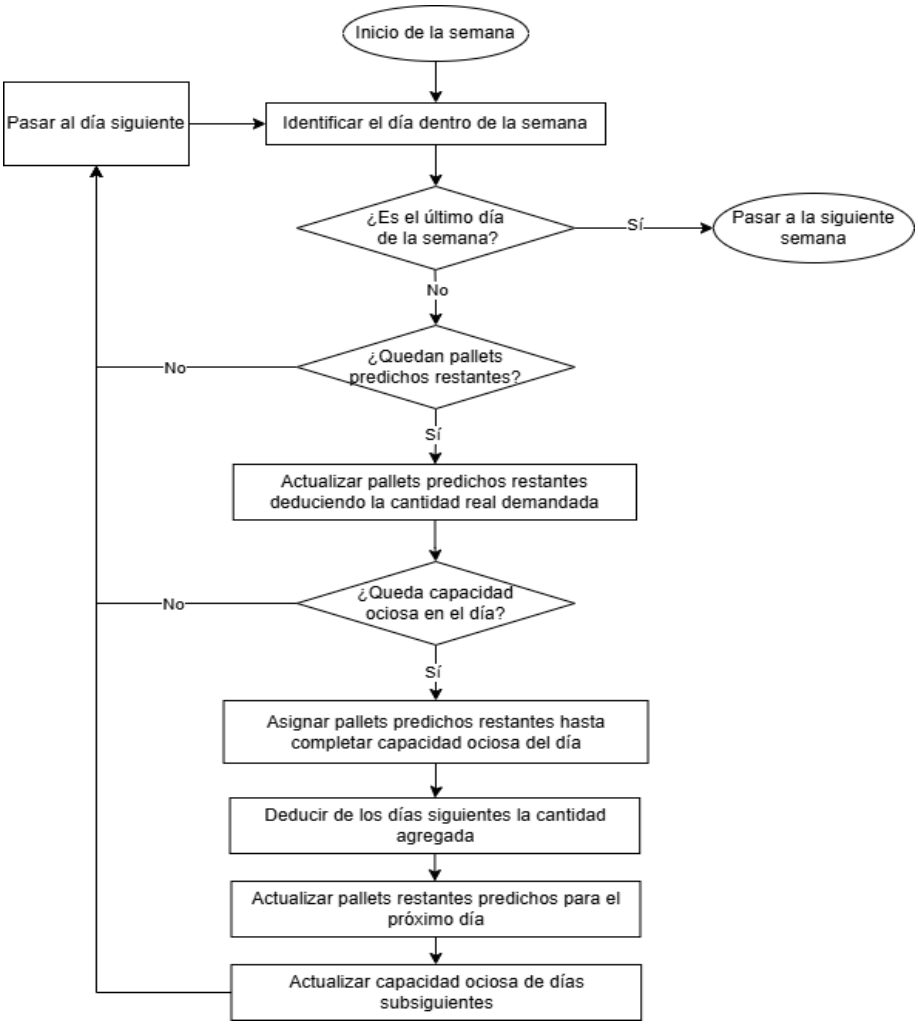

Figura 19: Diagrama de flujo para la reasignación de pallets

De esta manera, para cada día de la semana:

1. Se evalúa la capacidad ociosa de ese día.
    a. Se actualiza la variable de pallets restantes de reasignación descontando lo que ya fue efectivamente pedido ese día.
    $$P_{cps}^{reas} \coloneqq P_{cps}^{reas} - P_{cpsd}^{real}$$
    Donde:
        - P_{cpsd}^{real} es la cantidad de pallets efectivamente pedidos del cliente c, producto p, semana s, y día d.
    b. Se ordenan las combinaciones cliente-producto por cantidad de pallets restantes de reasignación.
    c. Se seleccionan pedidos potencialmente adelantables hasta completar la capacidad ociosa disponible CO_d.
    d. Se agregan estos pallets al día actual.
    $$P_{cpsd}^{real} \coloneqq P_{cpsd}^{real} + \sigma$$
    Donde σ es la cantidad de pallets reasignados al día actual
    e. Se descuentan de los días siguientes (de atrás hacia adelante) las cantidades equivalentes, simulando una redistribución realista.
    $P_{cpsd'}^{real} \coloneqq P_{cpsd'}^{real} - \sigma$ con $d' > d$
2. Al finalizar el día, se recalcula la capacidad ociosa para los días posteriores.
3. El procedimiento se repite para todos los días de la semana.

Esta lógica dinámica es clave: las decisiones tomadas en los primeros días alteran la carga prevista para los días posteriores, lo que puede evitar que se requiera un camión adicional.

5.8. Validación e integridad del proceso

Finalizado el procedimiento para todas las semanas, se realizaron controles para asegurar que las reasignaciones implementadas no afectaran la integridad de los pedidos originales. En primer lugar, se verificó que la cantidad total de pallets entregados por cliente y producto se mantuviera constante, lo que implica que no se introdujeron cantidades adicionales ni se redujo el volumen total demandado originalmente. En segundo lugar, se comprobó que el *mix* de productos entregado a cada cliente no fuera alterado, es decir, que no se sustituyeran productos ni se modificaran proporciones.

Esto se expresa formalmente como:

$$P_{cps}^{real} = P_{cps}^{mod}$$

Donde P_{cps}^{mod} es la cantidad de pallets reasignados al cliente c, producto p, en la semana s.

El procedimiento desarrollado persigue como objetivo principal la reasignación temporal de pallets dentro de la misma semana, adelantando entregas en función de las predicciones disponibles y de la capacidad ociosa presente en los primeros días de reparto. Esta redistribución se realiza únicamente dentro de los márgenes ya comprometidos de productos, sin sobreentregar ni sustituir ítems, y sin exceder la cantidad efectivamente demandada por el cliente.

De este modo, el modelo preserva la coherencia comercial y operativa de los pedidos, limitándose exclusivamente a modificar el momento de entrega dentro de la semana. Esta

condición fue verificada en todos los casos del conjunto de *test*, validando así la integridad del procedimiento de simulación.

5.9. Medición del impacto económico

Para cuantificar los beneficios operativos derivados de la reasignación logística, se definieron los siguientes ocho indicadores clave, calculados a partir de las semanas evaluadas en el set de *test*. Los primeros cuatro indicadores se calculan por semana; los últimos cuatro sintetizan los resultados en forma agregada, permitiendo evaluar la tendencia general del ahorro más allá de particularidades semanales.

1. Reducción absoluta de camiones utilizados: Representa la cantidad total de camiones que se lograron evitar tras la optimización.

$$\Delta C_s = C_s^{original} - C_s^{optimizado}$$

Donde:

- ΔC_s es la reducción absoluta de camiones en la semana s.
- $C_s^{original}$ es la cantidad de camiones utilizados originalmente.
- $C_s^{optimizado}$ es la cantidad de camiones utilizados tras la reasignación.

2. Reducción porcentual de camiones utilizados: Indica el porcentaje de camiones que se dejaron de utilizar con respecto al total original.

$$\Delta C_s^{\%} = \frac{\Delta C_s}{C_s^{original}} x100$$

Donde:

- $\Delta C_s^{\%}$ es el porcentaje de reducción en el uso de camiones en la semana s.

3. Ahorro económico absoluto: Representa el valor monetario total ahorrado al evitar el uso de camiones adicionales.

$$A_s = \sum_{i=1}^{\Delta C_s} Costo_{camión,i}$$

Donde:

- A_s es el ahorro económico en pesos para la semana s.
- $Costo_{camión,i}$ es el costo variable del camión i evitado (según el tipo).

4. Ahorro económico porcentual: Indica qué porcentaje del costo variable original por la optimización.

$$A_s^{\%} = \frac{A_s}{Costo\ total\ original_s} x100$$

Donde:

- $A_s^{\%}$ es el porcentaje de ahorro sobre el costo variable original.
- $Costo\ total\ original_s$ es la suma de costos variables de todos los camiones utilizados originalmente en la semana s.

5. Reducción promedio de camiones por semana: Mide la reducción promedio en cantidad de camiones utilizados a lo largo de todas las semanas simuladas.

$$\overline{\Delta C} = \frac{1}{n}\sum_{s=1}^{n} \Delta C_s$$

Donde:

- $\overline{\Delta C}$ es la reducción promedio de camiones por semana.
- n es el número total de semanas analizadas.

6. Promedio semanal de reducción porcentual de camiones: Mide la reducción porcentual promedio en camiones a lo largo de todas las semanas simuladas.

$$\overline{\Delta C^{\%}} = \frac{1}{n}\sum_{s=1}^{n} \Delta C_s^{\%}$$

Donde:

- $\overline{\Delta C^{\%}}$ es el promedio semanal de reducción porcentual de camiones.
- n es el número total de semanas analizadas.

7. Ahorro económico promedio por semana: Mide el ahorro económico promedio en pesos respecto al costo variable original en todas las semanas simuladas.

$$\bar{A} = \frac{1}{n}\sum_{s=1}^{n} A_s$$

Donde:

- $\bar{A}$ es el ahorro económico promedio por semana.
- n es el número total de semanas analizadas.

8. Promedio semanal de ahorro económico porcentual: Mide el porcentaje promedio de ahorro económico respecto del costo variable original en todas las semanas simuladas.

$$\overline{A^{\%}} = \frac{1}{n}\sum_{s=1}^{n} A_s^{\%}$$

Donde:

- $\overline{A^{\%}}$ es el promedio semanal de ahorro económico porcentual.
- n es el número total de semanas analizadas.

6. Resultados de la optimización logística

6.1. Caso ilustrativo de redistribución semanal

Con el objetivo de ilustrar el funcionamiento del modelo de optimización logística desarrollado, se presenta a continuación un caso correspondiente a la semana 36 del año 2024, que marca el inicio del período de *testing*. Esta semana fue tomada a modo de ejemplo, sin haber sido seleccionada por ninguna característica particular, sino por tratarse de la primera del tramo final del año utilizado para validar el modelo.

En este análisis se incluyen los tres principales clientes que realizan sus pedidos en formato pallet, dado que concentran un volumen significativo de la demanda y su logística permite mayor flexibilidad para realizar ajustes de carga sin alterar el *mix* de productos solicitado.

Se comparan los pedidos reales registrados en cada día de la semana con las predicciones generadas por el modelo, y se aplica una estrategia de reasignación que busca maximizar la ocupación de los camiones —con capacidad de hasta 10 pallets— sin superar la demanda total efectivamente observada para cada cliente.

El objetivo de esta simulación es evaluar si, a través de una redistribución más eficiente de los pedidos dentro de la misma semana, es posible reducir la cantidad total de envíos necesarios, minimizando así los costos logísticos sin afectar el nivel de servicio ni el cumplimiento de la demanda.

6.1.1. Capacidad ociosa inicial por día

Antes de analizar la redistribución de pallets, es necesario comprender las restricciones operativas y logísticas que condicionan el uso de los camiones en la semana. En este caso, la semana 36 incluye seis días de operación logística (de lunes a sábado), durante los cuales se debe realizar la entrega de pedidos a los principales clientes mayoristas.

Cada camión posee una capacidad estándar de 10 pallets por viaje. Si bien el primer camión disponible tiene capacidad para 12 pallets por ser de mayor tamaño, recordemos que el espacio equivalente a 2 pallets queda reservado para los pedidos minoristas, por lo que podemos considerar homogéneamente una capacidad de 10 pallets para todos los camiones en lo respectivo a pedidos de clientes mayoristas.

La siguiente tabla muestra, para cada día de la semana, la cantidad total de pallets despachados según la demanda real y la capacidad restante disponible en los camiones para realizar posibles reasignaciones:

Día de la semana	Pallets Reales	Pallets Disponibles
Lunes (Día 1)	3,22	6
Martes (Día 2)	4,40	5
Miércoles (Día 3)	4,83	5
Jueves (Día 4)	22,83	7
Viernes (Día 5)	10,35	9
Sábado (Día 6)	5,98	4

Tabla 24: Espacio inicial disponible (en pallets) para cada día de la semana

Tal como se definió en la metodología, para el cálculo de los pallets disponibles por día, se parte de la fórmula la capacidad total de camiones (10, 20, 30 o 40, según el caso) menos el número entero superior de la demanda real. Este redondeo se aplica para reflejar el uso real del camión en base a fracciones de pallets, ya que en la práctica no es posible aprovechar parcialmente un espacio de carga.

Como se observa en la tabla, algunos días presentan una subutilización significativa de los camiones, mientras que otros concentran una mayor carga operativa. En particular, se destaca que los días jueves y viernes se requirieron tres y dos camiones respectivamente, lo cual representa un pico logístico. A simple vista, se infiere que adelantar parte de los pallets a los primeros días de la semana podría permitir reducir el número total de camiones necesarios.

Sin embargo, esta estrategia sólo será factible en la medida en que el modelo predictivo haya sido capaz de anticipar correctamente los pallets que se demandarán en esos días de alta carga. Por lo tanto, el éxito de la optimización dependerá tanto de la capacidad de redistribución como de la precisión del *forecast* realizado.

6.1.2. Comparación entre pallets reales y predichos

Para continuar con este análisis de la semana 36, se realizó una primera comparación entre los pallets efectivamente pedidos por los tres principales clientes mayoristas y los valores predichos por el modelo para esa misma semana. Esta comparación permite evaluar la precisión del modelo no solo en términos cuantitativos, sino también cualitativos, en el sentido de identificar qué combinaciones cliente-producto debían ser anticipadas.

La siguiente tabla resume los resultados para las combinaciones donde se observaron valores diferentes de cero, tanto en la demanda real como en la predicción del modelo. Nuevamente, los nombres de los clientes y las marcas fueron enmascarados para preservar la confidencialidad.

Cliente	Producto	Pallets Reales	Pallets Predichos	Evaluación
Distribuciones Tral SRL	Leche 1ra Entera 3% Sachet	10	7	Subestimó
Grupo Alimentos Colaborativos	Queso 1ra Rallado Caja Sobre 35gr	10	7	Subestimó
Grupo Alimentos Colaborativos	Leche 2da Entera 3% Sachet	3	5	Sobreestimó
Grupo Alimentos Colaborativos	Leche 1ra Entera 3% Sachet	4	4	Exacto
Distribuciones Tral SRL	Leche 2da Entera 3% Sachet	0	2	Sobreestimó
Distribuciones Tral SRL	Yogur 1ra Firme Entero 190gr Vainilla	0	2	Sobreestimó
Supermercados Maestro	Queso 1ra Rallado Caja Sobre 35gr	0	2	Sobreestimó
Distribuciones Tral SRL	Leche 1ra P. Desc. 1% Sachet	0	1	Sobreestimó
Distribuciones Tral SRL	Queso 1ra Rallado Caja Sobre 35gr	0	1	Sobreestimó
Distribuciones Tral SRL	Yogur 1ra Batido Entero C/Cereales 159gr	0	1	Sobreestimó
Distribuciones Tral SRL	Leche 2ra Chocolatada 1 Litro	0	1	Sobreestimó
Grupo Alimentos Colaborativos	Leche 1ra Uat Entera 3% Brik	0	1	Sobreestimó
Supermercados Maestro	Yogur 1ra Clásico Sachet 1100gr Vainilla	1	0	Subestimó
Supermercados Maestro	Yogur 1ra Clásico Sachet 1100gr Frutilla	1	0	Subestimó

Tabla 25: Comparación entre pallets reales y predichos por cliente-producto

Como puede observarse, el modelo logró anticipar de forma precisa varias de las combinaciones, y cuando hubo diferencias, estas fueron moderadas. Las subestimaciones más marcadas ocurrieron en los casos de la Leche 1ra Entera 3% Sachet para Distribuciones Tral SRL y el Queso Rallado Sobre 35g para Grupo Alimentos Colaborativos, donde las predicciones fueron de 7 pallets frente a una demanda real de 10 cada uno. Sin embargo, observamos que el modelo anticipó una parte considerable de esa demanda.

Otro punto destacado es que el resto de las combinaciones cliente-producto, que no se presentan en esta tabla, registraron tanto predicciones como demandas reales iguales a cero. Si bien en algunos de esos casos se observó la venta de unidades sueltas, el volumen fue mínimo y no justificó la conformación de pallets completos. Esto demuestra que el modelo no generó recomendaciones vacías o erróneas en exceso, lo cual es fundamental para la comunicación con los clientes.

Este paso inicial establece una base sólida para continuar con el análisis de cómo se redistribuyó la carga y qué impacto tuvo la optimización propuesta, lo cual se desarrolla en los apartados siguientes.

6.1.3. Determinación de pallets disponibles para reasignación

Luego de comparar los pedidos reales con las predicciones del modelo, se procedió al siguiente paso del flujo lógico: determinar cuántos pallets efectivamente pueden ser reasignados dentro de la misma semana sin sobrepasar la demanda real observada. Este paso sigue la lógica establecida en el diagrama de flujo descripto anteriormente, en el cual se evita proponer envíos por encima de la demanda efectiva.

Para cada combinación cliente-producto se tomó el mínimo entre los pallets predichos y los pallets efectivamente pedidos, asegurando de esta forma que no se redistribuyan más unidades de las que realmente fueron solicitadas por los clientes. Este recorte define los pallets válidos para reasignación, es decir, aquellos que pueden reubicarse en los distintos días de la semana para mejorar la ocupación de los camiones sin incurrir en sobrestock.

En la tabla que se muestra a continuación se resumen las combinaciones que registraron pallets reasignables para la semana 36. El campo "Pallets Reasignación" indica la cantidad que podrá ser redistribuida internamente durante el proceso de optimización logística.

Cliente	Producto	Pallets Reales	Pallets Predichos	Pallets Reasignación
Distribuciones Tral SRL	Leche 1ra Entera 3% Sachet	10	7	7
Grupo Alimentos Colaborativos	Queso 1ra Rallado Caja Sobre 35gr	10	7	7
Grupo Alimentos Colaborativos	Leche 1da Entera 3% Sachet	4	4	4
Grupo Alimentos Colaborativos	Leche 2ra Entera 3% Sachet	3	5	3

Tabla 26: Determinación de pallets disponibles de reasignación por cliente-producto

Entre las combinaciones analizadas, se identificaron un total de cuatro cliente-producto con pallets válidos para reasignación. Los resultados muestran que, al aplicar la lógica de mínimos entre predicción y demanda real, se obtuvieron 21 pallets en total que pueden ser redistribuidos de manera eficiente a lo largo de la semana, respetando siempre los límites de la demanda observada.

En particular:

- Distribuciones Tral SRL cuenta con 7 pallets de Leche 1ra Entera 3% Sachet disponibles para redistribuir.
- Grupo Alimentos Colaborativos dispone de 7 pallets de Queso 1ra Rallado Caja Sobre 35gr, 4 pallets de Leche 2da Entera 3% Sachet, y 3 pallets de Leche 1ra Entera 3% Sachet.

Este subconjunto constituye la base del siguiente paso del análisis, en el que se busca optimizar la asignación diaria de estos pallets para lograr una mejor ocupación de los camiones y reducir la cantidad total de envíos necesarios. El hecho de contar con estas cantidades confirmadas — limitadas por la propia demanda real— garantiza que la simulación se mantenga dentro de márgenes operativos viables.

6.1.4. Redistribución optimizada por día

En esta sección se describe cómo se procedió con la redistribución diaria de pallets, partiendo de los pallets disponibles predichos por el modelo y las capacidades remanentes observadas en cada día. El objetivo fue anticipar entregas a jornadas anteriores con espacio libre en camión,

minimizando así la cantidad total de camiones necesarios sin superar los pedidos reales en la semana.

La reasignación se realizó respetando los siguientes principios:

- Nunca se superó la demanda real total por producto-cliente.
- Solo se reasignaron pallets que habían sido correctamente anticipados por el modelo.
- Se priorizó llenar los camiones a su capacidad máxima (10 pallets).

A continuación, se detalla cómo se fue utilizando esta lógica en cada día de la semana:

Día 1: lunes

En el primer día de la semana, con una carga real de 3,22 pallets y una capacidad disponible de hasta 6 pallets, se dio inicio al proceso de redistribución optimizada. El primer paso consistió en recalcular los pallets disponibles para reasignar, descontando del total predicho por el modelo la cantidad de pallets efectivamente pedidos por cliente-producto ese mismo día. Este ajuste asegura que no se intente "adelantar" productos que ya fueron entregados en su fecha original.

Cliente	Producto	Día 1 (Reales)	Pallets disponibles de reasignación luego del día 1
Distribuciones Tral SRL	Leche 1ra Entera 3% Sachet	1	6
Grupo Alimentos Colaborativos	Queso 1ra Rallado Caja Sobre 35gr	0	7
Grupo Alimentos Colaborativos	Leche 2da Entera 3% Sachet	1	3
Grupo Alimentos Colaborativos	Leche 1ra Entera 3% Sachet	0,556	2

Tabla 27: Pallets disponibles de reasignación netos de la demanda real del día 1

- Distribuciones Tral SRL – Leche 1ra Entera 3% Sachet: El modelo había predicho 7 pallets para la semana. El cliente realizó un pedido de 1 pallet el día 1, por lo que se restó ese valor del total predicho, resultando en 6 pallets disponibles para redistribución.
- Grupo Alimentos Colaborativos – Queso 1ra Rallado Caja Sobre 35gr: No se registró ningún pedido real este día. Se mantiene el total de 7 pallets disponibles para redistribución.
- Grupo Alimentos Colaborativos – Leche 2da Entera 3% Sachet: Se entregó 1 pallet el día 1. De los 4 pallets predichos, quedaron 3 disponibles.
- Grupo Alimentos Colaborativos – Leche 1ra Entera 3% Sachet: Se entregó 0.556 pallets el día 1 (según la proporción del registro), lo cual fue redondeado a 1 pallet completo asignado. A partir de los 3 pallets disponibles inicialmente, quedaron 2 pallets para redistribuir.

Luego de esta depuración, la selección de pallets a reasignar en el día se realizó aplicando un criterio proporcional basado en el volumen restante de pallets por cada combinación cliente-producto. En particular, se definió un peso relativo como el cociente entre la cantidad de pallets restantes de una combinación determinada y el total de pallets restantes disponibles para reasignación.

Sobre esa base, se calculó cuántos pallets podían ser agregados ese día utilizando la siguiente fórmula:

$$\min\left(\left\lfloor \frac{Pallets\ Restantes}{\sum Pallets\ Restantes}\ x\ Capacidad\ Disponible\ del\ Día \right\rfloor, Pallets\ Restantes\right)$$

Esta fórmula garantiza dos cosas:

1. Que la distribución sea proporcional al volumen pendiente por reasignar.
2. Que nunca se asignen más pallets de los que efectivamente están disponibles para esa combinación.

El resultado de esta lógica se consolidó en una nueva columna denominada “Pallets Agregados”, que indica cuántos pallets de cada combinación serían efectivamente adelantados en ese día. A partir de allí, se fueron sumando combinaciones cliente-producto hasta alcanzar o completar la capacidad total disponible para ese día, sin superarla.

Luego se sumaron directamente los valores de “Pallets Agregados” a la columna correspondiente al Día 1 en la tabla de planificación. Esto dio lugar a una nueva distribución de pallets para ese día, que combinó los pedidos originales con las cantidades adelantadas. Además, adelantándonos a la iteración para el Día 2, se actualizaron los pallets restantes de reasignación deduciéndole también los pallets agregados.

Cliente	Producto	Día 1 (Actualizado)	Pallets disponibles de reasignación luego del día 1 (Actualizado)
Distribuciones Tral SRL	Leche 1ra Entera 3% Sachet	**3**	**4**
Grupo Alimentos Colaborativos	Queso 1ra Rallado Caja Sobre 35gr	**3**	**4**
Grupo Alimentos Colaborativos	Leche 2da Entera 3% Sachet	**2**	**2**
Grupo Alimentos Colaborativos	Leche 1ra Entera 3% Sachet	0,556	2

Tabla 28: Pallets actualizados y disponibles de reasig. netos de la reasignación del día 1

Posteriormente, se recorrieron los días restantes de la semana —desde el Día 2 hacia el Día 6— y se restaron, para cada combinación, las cantidades que fueron agregadas al Día 1. Este paso asegura que el total de pallets entregados por combinación cliente-producto no supere nunca la demanda real semanal, y que efectivamente se mantenga constante la carga total original.

La comparación entre la distribución original y la ajustada puede observarse en las siguientes tablas.

Cargas originales:

Cliente	Producto	Día 1	Día 2	Día 3	Día 4	Día 5	Día 6
Distribuciones Tral SRL	Leche 1ra Entera 3% Sachet	1	1	1	4	2	1
Grupo Alimentos Colaborativos	Queso 1ra Rallado Caja Sobre 35gr	0	0	0	10	0	0
Grupo Alimentos Colaborativos	Leche 2da Entera 3% Sachet	1	0,986	1	1	0,98	1
Grupo Alimentos Colaborativos	Leche 1ra Entera 3% Sachet	0,556	0,973	1	1	1,977	0

Tabla 29: Pallets por cliente-producto por día previos a la reasignación del día 1

Cargas ajustadas luego del primer día:

Cliente	Producto	Día 1	Día 2	Día 3	Día 4	Día 5	Día 6
Distribuciones Tral SRL	Leche 1ra Entera 3% Sachet	3	0	0	4	2	1
Grupo Alimentos Colaborativos	Queso 1ra Rallado Caja Sobre 35gr	3	0	0	7	0	0
Grupo Alimentos Colaborativos	Leche 2da Entera 3% Sachet	2	0,986	0	1	0,98	1
Grupo Alimentos Colaborativos	Leche 1ra Entera 3% Sachet	0,556	0,973	1	1	1,977	0

Tabla 30: Pallets por cliente-producto por día posteriores a la reasignación del día 1

Finalmente, se procedió a recalcular la capacidad logística disponible para cada jornada del resto de la semana. La siguiente tabla muestra cómo quedó conformada la cantidad de pallets a entregar y el espacio restante por día:

Día de la semana	Pallets Totales	Pallets Disponibles
Lunes (Día 1)	9,22	0
Martes (Día 2)	3,40	6
Miércoles (Día 3)	2,83	7
Jueves (Día 4)	19,83	0
Viernes (Día 5)	10,35	9
Sábado (Día 6)	5,98	4

Tabla 31: Espacio disponible (pallets) para cada día de la semana post reasignación del día 1

Se observa que el Día 1 completó su capacidad de camión, alcanzando prácticamente el tope de los 10 pallets. Esta asignación temprana permitió una distribución más equilibrada a lo largo de la semana.

Lo más destacable es lo que ocurre en el Día 4, donde originalmente se requerían tres camiones para atender la demanda (por un total superior a 20 pallets). Gracias a las entregas anticipadas realizadas en los días previos, la demanda efectiva bajó a 19,83 pallets, lo que permitió eliminar por completo la necesidad de uno de esos camiones adicionales, cumpliendo con el principal objetivo de la optimización: reducir el número de envíos sin dejar de cubrir la demanda.

Día 2: martes

A partir del segundo día de la semana, el procedimiento de reasignación sigue la misma lógica que la aplicada en el Día 1, utilizando la información actualizada de pallets disponibles y el resultado de los movimientos anteriores. Por esa razón, a partir de este punto se opta por centrar el análisis exclusivamente en los resultados, sin repetir la descripción del algoritmo en cada caso.

La siguiente secuencia de tablas ilustra los principales resultados del Día 2.

1- Deducción de pallets disponibles de reasignación luego de la consideración del pedido real:

Cliente	Producto	Día 2 (Reales neto)	Pallets disponibles de reasignación luego del día 2
Distribuciones Tral SRL	Leche 1ra Entera 3% Sachet	0	4
Grupo Alimentos Colaborativos	Queso 1ra Rallado Caja Sobre 35gr	0	4
Grupo Alimentos Colaborativos	Leche 2da Entera 3% Sachet	0,986	**1**
Grupo Alimentos Colaborativos	Leche 1ra Entera 3% Sachet	0,973	**1**

Tabla 32: Pallets disponibles de reasignación netos de la demanda real del día 2

2- Adición proporcional de pallets al presente día hasta alcanzar la capacidad máxima:

Cliente	Producto	Día 2 (Actualizado)	Pallets disponibles de reasignación luego del día 2 (Actualizado)
Distribuciones Tral SRL	Leche 1ra Entera 3% Sachet	**3**	**1**
Grupo Alimentos Colaborativos	Queso 1ra Rallado Caja Sobre 35gr	**3**	**1**
Grupo Alimentos Colaborativos	Leche 2da Entera 3% Sachet	0,986	1
Grupo Alimentos Colaborativos	Leche 1ra Entera 3% Sachet	0,973	1

Tabla 33: Pallets actualizados y disponibles de reasig. netos de la reasignación del día 2

3- Distribución de la carga (para los clientes-productos predichos) previo a la reasignación del presente día:

Cliente	Producto	Día 1	Día 2	Día 3	Día 4	Día 5	Día 6
Distribuciones Tral SRL	Leche 1ra Entera 3% Sachet	3	0	0	4	2	1
Grupo Alimentos Colaborativos	Queso 1ra Rallado Caja Sobre 35gr	3	0	0	7	0	0
Grupo Alimentos Colaborativos	Leche 2da Entera 3% Sachet	2	0,986	0	1	0,98	1
Grupo Alimentos Colaborativos	Leche 1ra Entera 3% Sachet	0,556	0,973	1	1	1,977	0

Tabla 34: Pallets por cliente-producto por día previos a la reasignación del día 2

4- Distribución de la carga (para los clientes-productos predichos) luego de la reasignación del presente día:

Cliente	Producto	Día 1	Día 2	Día 3	Día 4	Día 5	Día 6
Distribuciones Tral SRL	Leche 1ra Entera 3% Sachet	3	**3**	0	**1**	2	1
Grupo Alimentos Colaborativos	Queso 1ra Rallado Caja Sobre 35gr	3	**3**	0	**4**	0	0
Grupo Alimentos Colaborativos	Leche 2da Entera 3% Sachet	2	0,986	0	1	0,98	1
Grupo Alimentos Colaborativos	Leche 1ra Entera 3% Sachet	0,556	0,973	1	1	1,977	0

Tabla 35: Pallets por cliente-producto por día posteriores a la reasignación del día 2

5- Pallets disponibles luego de la reasignación:

Día de la semana	Pallets Totales	Pallets Disponibles
Lunes (Día 1)	9,22	0
Martes (Día 2)	**9,40**	**0**
Miércoles (Día 3)	2,83	7
Jueves (Día 4)	**13,83**	**6**
Viernes (Día 5)	10,35	9
Sábado (Día 6)	5,98	4

Tabla 36: Espacio disponible (pallets) para cada día de la semana post reasignación del día 2

Luego de correr la simulación para el segundo día, obtenemos que nuevamente se agotaron los espacios disponibles para pallets en el camión del día, pasando la carga de 3,40 pallets a ser de

9,40 pallets. Como consecuencia, el pedido de tales pallets que habría ocurrido el jueves ya no acontecería. Por lo tanto, la carga del jueves pasó de 19,83 pallets a 13,83.

En esta oportunidad, la simulación no logró reducir la cantidad de camiones, aunque logró reducir la variabilidad en las cargas.

Día 3: miércoles

La siguiente secuencia de tablas ilustra los principales resultados del Día 3.

1- Deducción de pallets disponibles de reasignación luego de la consideración del pedido real:

Cliente	Producto	Día 3 (Reales neto)	Pallets disponibles de reasignación luego del día 3
Distribuciones Tral SRL	Leche 1ra Entera 3% Sachet	0	1
Grupo Alimentos Colaborativos	Queso 1ra Rallado Caja Sobre 35gr	0	1
Grupo Alimentos Colaborativos	Leche 2da Entera 3% Sachet	0	1
Grupo Alimentos Colaborativos	Leche 1ra Entera 3% Sachet	1	**0**

Tabla 37: Pallets disponibles de reasignación netos de la demanda real del día 3

2- Adición proporcional de pallets al presente día hasta alcanzar la capacidad máxima:

Cliente	Producto	Día 3 (Actualizado)	Pallets disponibles de reasignación luego del día 3 (Actualizado)
Distribuciones Tral SRL	Leche 1ra Entera 3% Sachet	**1**	**0**
Grupo Alimentos Colaborativos	Queso 1ra Rallado Caja Sobre 35gr	**1**	**0**
Grupo Alimentos Colaborativos	Leche 2da Entera 3% Sachet	**1**	**0**
Grupo Alimentos Colaborativos	Leche 1ra Entera 3% Sachet	1	0

Tabla 38: Pallets actualizados y disponibles de reasig. netos de la reasignación del día 3

3- Distribución de la carga (para los clientes-productos predichos) previo a la reasignación del presente día:

Cliente	Producto	Día 1	Día 2	Día 3	Día 4	Día 5	Día 6
Distribuciones Tral SRL	Leche 1ra Entera 3% Sachet	3	3	0	1	2	1
Grupo Alimentos Colaborativos	Queso 1ra Rallado Caja Sobre 35gr	3	3	0	4	0	0
Grupo Alimentos Colaborativos	Leche 2da Entera 3% Sachet	2	0,986	0	1	0,98	1
Grupo Alimentos Colaborativos	Leche 1ra Entera 3% Sachet	0,556	0,973	1	1	1,977	0

Tabla 39: Pallets por cliente-producto por día previos a la reasignación del día 3

4- Distribución de la carga (para los clientes-productos predichos) luego de la reasignación del presente día:

Cliente	Producto	Día 1	Día 2	Día 3	Día 4	Día 5	Día 6
Distribuciones Tral SRL	Leche 1ra Entera 3% Sachet	3	3	**1**	**0**	2	1
Grupo Alimentos Colaborativos	Queso 1ra Rallado Caja Sobre 35gr	3	3	**1**	**3**	0	0
Grupo Alimentos Colaborativos	Leche 2da Entera 3% Sachet	2	0,986	**1**	**0**	0,98	1
Grupo Alimentos Colaborativos	Leche 1ra Entera 3% Sachet	0,556	0,973	1	1	1,977	0

Tabla 40: Pallets por cliente-producto por día posteriores a la reasignación del día 3

5- Pallets disponibles luego de la reasignación:

Día de la semana	Pallets Totales	Pallets Disponibles
Lunes (Día 1)	9,22	0
Martes (Día 2)	9,40	0
Miércoles (Día 3)	**5,83**	**4**
Jueves (Día 4)	**10,83**	**9**
Viernes (Día 5)	10,35	9
Sábado (Día 6)	5,98	4

Tabla 41: Espacio disponible (pallets) para cada día de la semana post reasignación del día 3

Como puede observarse, el Día 3 aumentó su carga logística de 2,83 pallets a 5,83, utilizando parte de la capacidad disponible. En consecuencia, el Día 4, que inicialmente representaba uno

de los mayores desafíos logísticos de la semana, redujo su carga de 13,83 a 10,83 pallets, incrementando su capacidad disponible de 6 a 9 pallets.

Sin embargo, a pesar de este reequilibrio, no se logró reducir la cantidad de camiones requeridos para el Día 4 más allá de lo ya conseguido en la redistribución de días anteriores. Es decir, no se produjo un ahorro adicional de envíos.

Además, luego de esta tercera iteración, todos los pallets restantes que el modelo había predicho como reasignables ya fueron distribuidos, por lo que el algoritmo continuó corriendo en las siguientes jornadas (días 4 a 6) pero sin realizar nuevas modificaciones, ya que no quedaban combinaciones producto-cliente disponibles para adelantar.

6.1.5. Conclusión del caso ilustrativo

El caso desarrollado a lo largo de esta sección, correspondiente a la semana 36, permitió mostrar en detalle el funcionamiento de la lógica de redistribución semanal basada en las predicciones del modelo y en la capacidad logística disponible por día.

El principal resultado operativo fue la reducción efectiva de un camión completo en el Día 4, lo que representa un ahorro logístico concreto sin comprometer el nivel de servicio ni exceder los límites de la demanda real observada. Este ahorro se logró mediante el adelantamiento estratégico de pallets a días con menor carga, aprovechando de forma inteligente la capacidad disponible.

Si bien no se consiguió reducir un segundo camión, la redistribución generó una regularización significativa en la carga diaria a lo largo de la semana. Este balance más parejo entre jornadas es clave para mejorar la planificación operativa, ya que contribuye a homogeneizar el tiempo de trabajo del personal involucrado en la logística, reduciendo picos de exigencia y permitiendo un uso más eficiente de los recursos disponibles.

Esta lógica de optimización se aplica sistemáticamente en cada una de las semanas subsiguientes del período de *testing*. De este modo, verificamos que el modelo predictivo no sólo anticipa la demanda semanal con una considerable precisión, sino que también habilita decisiones concretas que mejoran la eficiencia del sistema logístico.

6.2. El rol del modelo predictivo en la relación comercial

Antes de pasar al análisis de las métricas obtenidas por la reasignación de los pedidos, cabe hacer una reflexión al respecto de los ofrecimientos que se le realizan a los clientes mayoristas en nuestra simulación logística.

Más allá de su impacto operativo, el modelo predictivo desarrollado desempeña un papel central en la estrategia comercial semanal con los grandes clientes. En este esquema, no se consulta al cliente todos los días ni se le ofrece todo el catálogo disponible. Por el contrario, se realiza una propuesta semanal de productos basada en las predicciones, que busca optimizar tanto la logística como la interacción comercial, generando ofertas específicas y pertinentes.

Cabe destacar que esta operatoria se focaliza en tres clientes puntuales, los cuales concentran el volumen de pallets cerrados, y que el criterio de si un cliente “acepta” o “no acepta” una propuesta está definido por la naturaleza de la simulación logística desarrollada. En ese marco, se considera aceptado cuando el producto efectivamente aparece luego en la demanda real semanal, dentro de los límites definidos por el mínimo entre lo ofrecido y lo pedido.

Al evaluar el desempeño del modelo en las 17 semanas de *testing*, se obtuvieron los siguientes resultados:

- → 132 combinaciones cliente-producto fueron pedidas realmente por el cliente, pero no anticipadas por el modelo, lo cual representa un promedio de 7,76 combinaciones por semana.
- → 102 combinaciones cliente-producto fueron anticipadas por el modelo pero no se concretaron en pedidos reales, es decir, fueron ofrecimientos sin aceptación de anticipación, con un promedio semanal de 6 combinaciones.
- → 93 combinaciones cliente-producto fueron correctamente anticipadas y luego efectivamente solicitadas por el cliente, alcanzando un promedio de 5,47 aciertos semanales.

Adicionalmente, cuando se produce esta coincidencia (predicho y pedido), el valor medio del mínimo entre ambas cantidades es de 4,25 pallets, lo que indica que no se trata de aciertos triviales o marginales, sino de volúmenes significativos que reflejan una buena calibración del modelo.

Es importante entender que el objetivo del modelo no es acertar todo ni evitar errores aislados, sino encontrar un equilibrio entre anticipación y eficiencia, tanto en lo logístico como en lo comercial. Por eso resulta natural que exista cierto grado de ofrecimientos no aceptados y pedidos no anticipados. Lo relevante es que el modelo reduce significativamente la carga de negociación, mejora la planificación previa, y fortalece el vínculo con el cliente al ofrecerle productos con alta probabilidad de aceptación.

6.3. Métricas de desempeño logístico

6.3.1. Validación de integridad logística

Antes de analizar el impacto operativo del modelo, resulta fundamental validar que el proceso de optimización no haya alterado el volumen total de pallets asignado semanalmente. Tal como se definió en la metodología, la lógica implementada solo permite redistribuciones dentro de la misma semana, sin alterar la cantidad total de pallets entregados por cliente-producto.

La siguiente tabla resume esta comprobación para las 17 semanas del período de *testing*.

Año	Semana	Pallets Antes	Pallets Después	Diferencia
2024	36	51,606	51,606	0
2024	37	65,834	65,834	0
2024	38	36,874	36,874	0
2024	39	73,081	73,081	0
2024	40	59,054	59,054	0
2024	41	59,124	59,124	0
2024	42	69,293	69,293	0
2024	43	75,315	75,315	0
2024	44	43,225	43,225	0

2024	45	79,101	79,101	0
2024	46	83,539	83,539	0
2024	47	54,188	54,188	0
2024	48	91,771	91,771	0
2024	49	34,399	34,399	0
2024	50	58,200	58,200	0
2024	51	61,589	61,589	0
2024	52	62,796	62,796	0

Tabla 42: Pallets totales por semana antes y después de la optimización

Este análisis confirma que el algoritmo mantuvo la integridad logística del sistema: toda mejora obtenida se debió a una redistribución más eficiente de los mismos recursos, respetando la demanda real y sin alterar el volumen semanal total.

6.3.2. Estructura de costos variables asociados a la cantidad de camiones

Para evaluar con precisión el impacto económico de la optimización logística, es esencial comprender la estructura de costos variables diarios según la cantidad de camiones requeridos. A diferencia de una tarifa lineal, el sistema utilizado contempla escalonamientos de costos que dependen tanto del tipo de camión como de su origen (propio o tercerizado).

La estructura diaria de costos es la siguiente:

- Primer camión (camión principal):
  De uso obligatorio y constante. Su costo no se considera variable, ya que forma parte de la estructura fija de operación. Siempre está disponible, independientemente de la cantidad de pallets o del resto de la planificación.
- Segundo camión (camión auxiliar propio):
  Pertenece a la flota propia y su uso representa un costo adicional:
  - Horas extras del personal: $80.000 por día.
  - Combustible: $12.700 por día (trayecto cochera – fábrica – clientes – cochera).

  Costo variable total diario: $92.700.
- Tercer y cuarto camión (camiones de terceros):
  En caso de superar la capacidad de los dos camiones propios, se contratan camiones externos disponibles en planta. Estos servicios incluyen carga y descarga, por lo que no implican horas extra ni gasto interno de combustible.
  Costo diario por cada camión tercerizado: $304.920 (sin IVA).

Este esquema genera una estructura de costos escalonada por día, donde evitar la necesidad de un tercer camión puede significar una reducción abrupta de más de $300.000 diarios. Estas definiciones de costos variables se utilizarán en el siguiente punto para comparar el costo logístico total diario antes y después de la optimización.

6.3.3. Comparación semanal de desempeño logístico

Para evaluar los efectos de la optimización logística semana por semana, se utilizaron cuatro métricas clave definidas en la metodología:

1. Variación absoluta en la cantidad de camiones utilizados.
2. Variación porcentual de la cantidad de camiones.
3. Ahorro económico absoluto en costos logísticos variables.
4. Ahorro económico porcentual respecto al costo original.

A continuación, se presentan dos tablas separadas. La primera refleja el comportamiento logístico en términos de cantidad de camiones, y la segunda muestra el impacto económico asociado.

Esta tabla resume cuántos camiones fueron necesarios antes y después de la optimización en cada semana, así como su reducción absoluta y porcentual. A su vez, podemos ver gráficamente este cambio en el gráfico de barras agrupadas que le sigue.

Año	Semana	Camiones Antes	Camiones Después	Reducción Absoluta (ΔC_s)	Reducción Porcentual ($\Delta C_s^{\%}$)
2024	36	9	8	1	11,11%
2024	37	9	7	2	22,22%
2024	38	6	6	0	0%
2024	39	10	8	2	20%
2024	40	10	8	2	20%
2024	41	7	7	0	0%
2024	42	10	9	1	10%
2024	43	10	8	2	20%
2024	44	7	7	0	0%
2024	45	11	10	1	9,09%
2024	46	11	9	2	18,18%
2024	47	8	7	1	12,5%
2024	48	12	10	2	16,67%
2024	49	6	6	0	0%
2024	50	9	8	1	11,11%
2024	51	9	8	1	11,11%
2024	52	8	8	0	0%

Tabla 43: Variación absoluta y porcentual de camiones por semana tras optimización

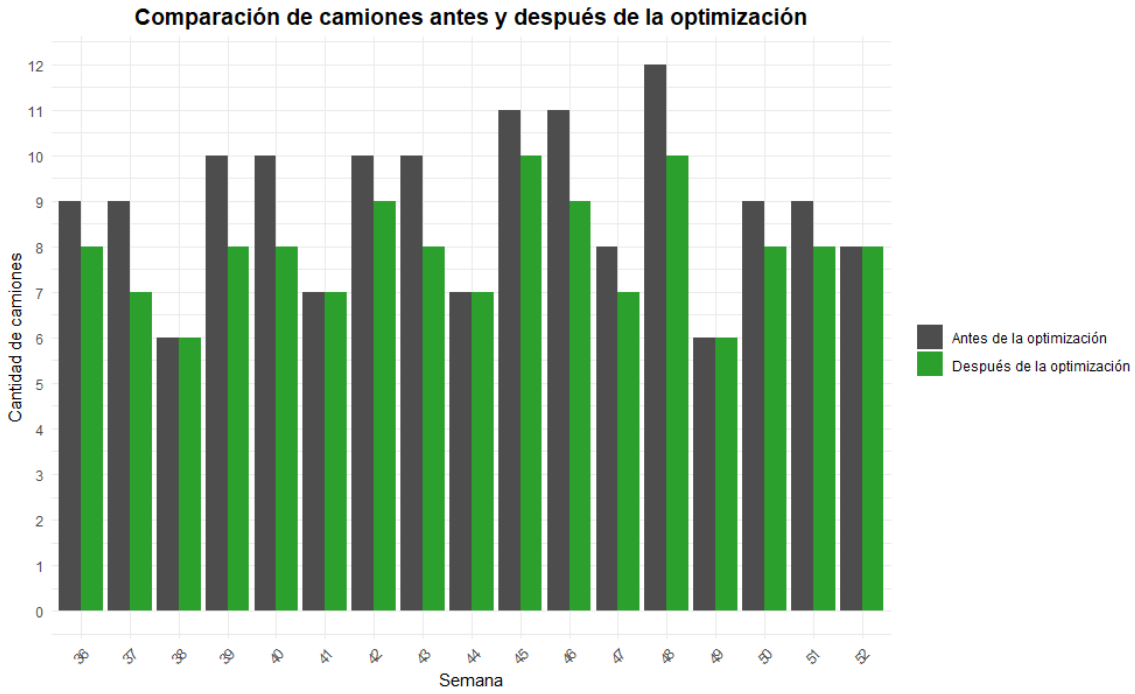

Figura 20: Cantidad de camiones por semana antes y después de la optimización

Como resultado, en la mayoría de los casos la optimización permitió reducir entre 1 y 2 camiones por semana. Esta disminución se traduce directamente en ahorro operativo y en una mejor planificación de recursos humanos y logísticos.

Es importante destacar que al menos un camión por día es obligatorio, ya que existe una unidad fija de reparto que debe salir todos los días hábiles. Por lo tanto, el mínimo teórico de camiones por semana está condicionado por la cantidad de días operativos, que normalmente es de 6 días por semana (salvo cuando hay feriados). Es por eso por lo que, cuando la cantidad de camiones está cercana al mínimo operativo, al algoritmo le resulta más difícil reducir camiones, dado a que se requirieron pocos/nulos auxilios en tal semana.

Se verifica que el peso relativo de ahorrar 1 o 2 envíos por semana es considerable, ya que el número total de camiones semanales apenas si alcanza la decena. Esta variación relativa se acrecienta aún más al considerar costos variables al considerar que el camión principal es siempre un costo fijo.

La siguiente tabla detalla los costos logísticos variables asociados al uso de camiones antes y después de aplicar la redistribución, junto con los ahorros absolutos y porcentuales obtenidos. Análogamente al análisis anterior, también podemos visualizar este cambio en el gráfico de barras agrupadas que le sigue.

Año	Semana	Costo Variable Antes	Costo Variable Después	Ahorro Absoluto (A_s)	Ahorro Porcentual $(A_s^{\%})$
2024	36	490.320	185.400	304.920	62,19%
2024	37	370.800	185.400	185.400	50%
2024	38	0	0	0	0%
2024	39	583.020	185.400	397.620	68,2%
2024	40	583.020	185.400	397.620	68,2%
2024	41	397.620	397.620	0	0%
2024	42	583.020	278.100	304.920	52,3%
2024	43	583.020	185.400	397.620	68,2%
2024	44	92.700	92.700	0	0%
2024	45	887.940	795.240	92.700	10,44%
2024	46	675.720	278.100	397.620	58,84%
2024	47	490.320	185.400	304.920	62,19%
2024	48	1.192.860	583.020	609.840	51,12%
2024	49	0	0	0	0%
2024	50	490.320	185.400	304.920	62,19%
2024	51	490.320	185.400	304.920	62,19%
2024	52	278.100	278.100	0	0%

Tabla 44: Variación absoluta y porcentual de costo variable por semana tras optimización

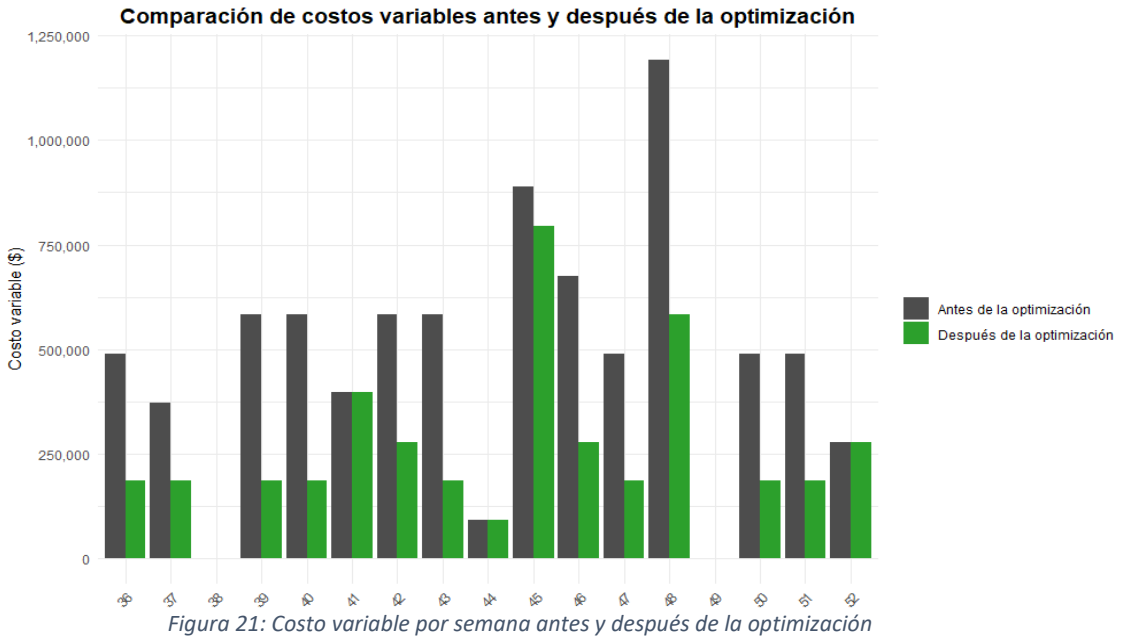

Figura 21: Costo variable por semana antes y después de la optimización

Los resultados presentados reflejan con claridad la eficiencia económica alcanzada gracias a la integración del modelo predictivo con el algoritmo de redistribución logística. La reducción de costos obtenida en la mayoría de las semanas es significativa, con valores que en varios casos superan los $300.000 diarios y, en el caso de la semana 48, alcanzan más de $600.000 de ahorro.

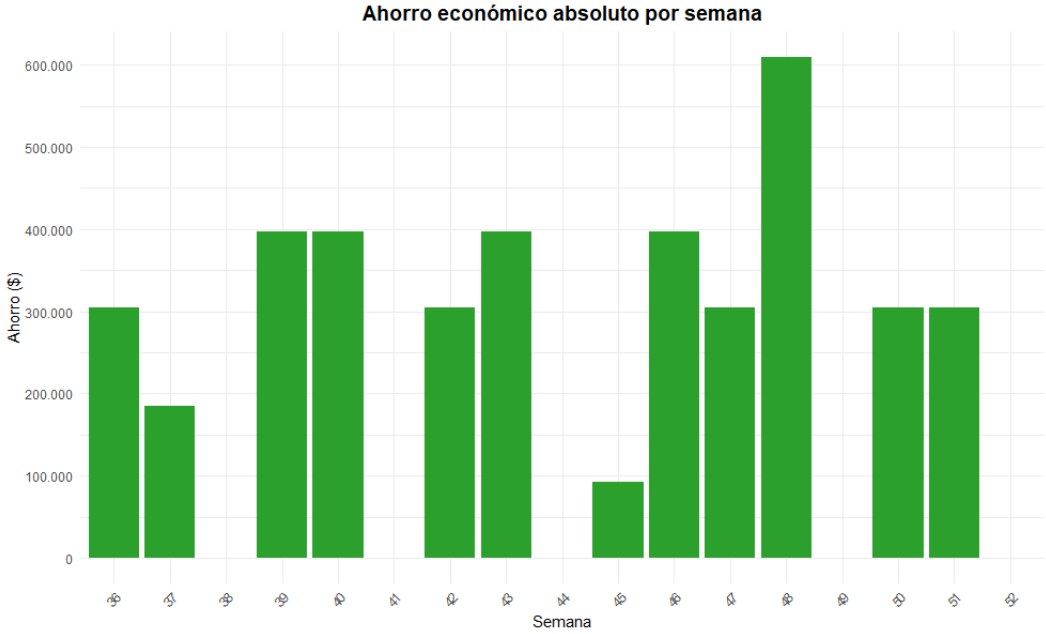

Figura 22: Ahorro en pesos por semana producto de la optimización

Estas cifras no sólo evidencian el éxito de la metodología propuesta, sino que confirman que el enfoque adoptado permite transformar un modelo de predicción de demanda en una herramienta concreta de ahorro y eficiencia operativa real.

En definitiva, este conjunto de resultados resume y valida todo el desarrollo anterior del trabajo: desde la correcta anticipación de la demanda, pasando por la construcción de una lógica de reasignación realista, hasta su aplicación concreta con beneficios tangibles en cortes semanales. La magnitud de los ahorros obtenidos por semana refuerza la idea de que, en contextos de alta presión logística y márgenes ajustados, la optimización basada en datos puede marcar una diferencia estructural en la rentabilidad de la operación.

6.3.4. Métricas acumuladas y promedios semanales

Luego de analizar el desempeño semana por semana, este apartado presenta una síntesis acumulada de los resultados obtenidos en las 17 semanas del set de *testing*. A partir de los valores totales de camiones utilizados y costos logísticos variables, se obtienen además las variaciones y los promedios semanales definidos en las métricas 5 a 8 del marco metodológico.

Indicador	Antes de optimizar	Después de optimizar	Reducción absoluta	Reducción porcentual
Camiones utilizados	152	134	18	11,84%
Costo variable total	$8.189.100	$4.186.080	$4.003.020	48,88%

Tabla 45: Camiones, costos variables y sus variaciones totales tras optimizar

La optimización permitió ahorrar más de 4 millones de pesos en costos logísticos variables a lo largo del período analizado, y reducir el uso de camiones en 18 unidades, sin afectar el volumen total de pallets entregado. De esta manera, en términos porcentuales, la reducción de camiones ($\overline{\Delta C^{\%}}$) alcanzó un 11,84%, y el ahorro respecto al costo variable original ($\overline{A^{\%}}$) fue de 48,88%.

Métrica	Valor
Reducción promedio de camiones por semana ($\overline{\Delta C}$)	1,06 camiones
Ahorro económico promedio por semana ($\overline{A}$)	$235.473

Tabla 46: Promedio de reducción de camiones y de ahorro económico tras optimizar

Estos resultados reafirman el impacto sistemático del modelo a lo largo del tiempo. No se trata de mejoras aisladas o excepcionales, sino de una consistencia operativa en la optimización, que se refleja tanto en el uso de recursos como en los costos asociados.

En conjunto con los resultados semanales, esta visión acumulada demuestra que la metodología propuesta logra un beneficio sostenido, con mejoras logísticas y económicas tangibles que justifican plenamente su aplicación y escalabilidad futura.

6.4. Conclusiones sobre los resultados de la optimización logística

A lo largo de este capítulo se abordó con profundidad el impacto que puede generar la combinación entre un modelo predictivo preciso y una estrategia de redistribución logística bien diseñada. La evidencia empírica obtenida en el período de testing permitió validar tanto la lógica del enfoque metodológico como los beneficios concretos de su implementación.

En primer lugar, el caso ilustrativo de la semana 36 permitió visualizar paso a paso el funcionamiento de la lógica de reasignación de pallets, destacando cómo, a partir de las predicciones del modelo, fue posible anticipar entregas y evitar el uso de un camión adicional, lo que implicó un ahorro logístico significativo. Si bien no se logró reducir un segundo camión, el resultado alcanzado permitió una regularización de cargas a lo largo de la semana, generando un uso más equilibrado de los recursos operativos.

En segundo lugar, se destacó el rol del modelo predictivo como una herramienta no solo técnica sino también comercial. El modelo permitió reducir la carga de negociación con los clientes, evitando ofertas irrelevantes y facilitando acuerdos comerciales sobre productos de alta probabilidad de aceptación. A través de los indicadores definidos, se demostró que, en promedio, cada semana se realizaron entre 5 y 6 ofrecimientos aceptados por los clientes, con un volumen medio de más de 4 pallets por combinación, lo que evidencia una alta precisión y utilidad práctica del modelo.

Finalmente, las métricas semanales y acumuladas permiten dimensionar el verdadero alcance de esta solución. En las 17 semanas evaluadas, se logró:

- → Mantener en todos los casos la integridad logística, es decir, la redistribución no generó sobreentregas ni modificó el total semanal de productos previstos.
- → Reducir el uso de 18 camiones sin alterar el volumen total de pallets entregado.
- → Ahorrar más de $4 millones en costos logísticos variables.

Estos resultados promedian un ahorro de $235.473 semanales y la reducción de 1,06 camiones por semana, métricas que reflejan una consistencia operativa, no un éxito circunstancial. La metodología demostró ser robusta, replicable y escalable, lo podría habilitar su uso futuro en otras zonas geográficas, otros grupos de clientes, o incluso con otros tipos de productos.

En resumen, el proceso de optimización logística planteado logró impactar en los principales indicadores operativos del negocio, ofreciendo una solución eficiente, sustentable y basada en datos reales. Esto refuerza la hipótesis central de este trabajo: que una buena predicción de la demanda, aplicada con criterio operativo, permite generar mejoras concretas en la planificación, la rentabilidad y la calidad del servicio.

7. Conclusión general

Esta tesis abordó de manera integral la problemática del pronóstico de demanda y la optimización logística en la distribución de productos lácteos. A partir del análisis detallado de datos históricos y el desarrollo de modelos predictivos avanzados, se propuso una solución basada en datos que conecta la anticipación de la demanda con decisiones logísticas concretas, permitiendo generar mejoras operativas tangibles en la planificación del reparto.

En primer lugar, se construyó un modelo de *machine learning* —específicamente *XGBoost*— capaz de predecir la demanda semanal por cliente y producto, utilizando variables tanto internas como externas. Se exploraron múltiples configuraciones y estrategias de optimización de hiperparámetros, y se validó el modelo sobre un conjunto de *test* con métricas robustas de desempeño. El modelo final alcanzó un error medio absoluto (MAE) de 20,66 unidades y un R^2 del 0,71, lo que indica una capacidad predictiva sólida aun en un contexto marcado por alta variabilidad y proporción de ceros.

En segundo lugar, se aplicó esta predicción como insumo para una simulación logística secuencial, orientada a redistribuir de forma anticipada los pallets semanales dentro de los días de operación, con el fin de maximizar la ocupación de camiones y reducir la cantidad total de envíos. Esta redistribución respetó en todo momento el volumen efectivamente demandado, sin alterar el *mix* de productos ni exceder la cantidad observada en la realidad. La metodología empleada logró reducir en promedio 1,06 camiones por semana y generar un ahorro económico semanal estimado de $235.473 en costos variables de transporte, sin afectar el nivel de servicio.

Los resultados obtenidos validan empíricamente la hipótesis central del trabajo: que una estrategia basada en datos, que combine predicción y optimización, puede mejorar significativamente la eficiencia operativa de la distribución, incluso en contextos de alta concentración comercial y variabilidad en la demanda.

Más allá de los logros alcanzados, el trabajo deja abiertos caminos de mejora y continuidad. En particular, se identifican las siguientes líneas de desarrollo futuro:

- → Modelos aún más robustos: Explorar algoritmos más complejos y configuraciones con mayor poder de generalización, por ejemplo, utilizando 10 *folds* en la validación cruzada para reducir el riesgo de sobreajuste.
- → Actualización continua del modelo: Incorporar semanalmente nuevas observaciones al entrenamiento del modelo, permitiendo una recalibración dinámica que se ajuste a cambios estructurales en los patrones de consumo.
- → Predicción semanal continua: Reemplazar la lógica actual de cortes semanales por calendario por una estructura de predicción continua, en la que se proyecte la demanda para los próximos 5 o 6 días hábiles a partir de cada día de operación. Esto permite construir un set de datos más amplio y granular, con ventanas móviles que se actualizan diariamente, pero manteniendo siempre la lógica de pronóstico por semana.
  Esta reformulación habilita un esquema más dinámico de planificación logística y comercial, y podría integrarse con un sistema de alertas operativas diarias para que preventistas y fleteros ajusten en tiempo real su oferta de productos según la proyección semanal vigente desde ese día.

Finalmente, se considera que el enfoque desarrollado en esta tesis no solo es aplicable al caso específico analizado, sino que podría ser replicado y adaptado a otros contextos logísticos similares dentro del sector de consumo masivo, especialmente en categorías de productos perecederos y/o de alta rotación. No obstante, es importante destacar que la dinámica del sector lácteo presenta particularidades específicas —como la caducidad acotada, los cortes semanales definidos por la frecuencia de reparto y la alta sensibilidad a la variabilidad de la demanda— que condicionan tanto la granularidad del modelo como su frecuencia de actualización. Por esta razón, cualquier intento de aplicar esta metodología en otros rubros debe partir de una evaluación cuidadosa del contexto operativo y comercial de cada industria, ajustando el diseño del modelo a las condiciones propias del sistema logístico en cuestión.

La combinación de analítica avanzada, conocimiento del negocio y soluciones pragmáticas habilita mejoras operativas con impacto directo en la rentabilidad, ofreciendo así una herramienta poderosa para enfrentar los desafíos crecientes de la distribución en mercados dinámicos.

Referencias

Advanced Logistics. (1 de February de 2025). *The Hidden Savings: How Data-Driven Logistics Optimization is Transforming Supply Chain Costs*. Obtenido de https://advancedlogistics.us/the-hidden-savings-how-data-driven-logistics-optimization-is-transforming-supply-chain-costs/

Ámbito Financiero. (s.f.). *Dólar blue histórico*. Obtenido de https://www.ambito.com/contenidos/dolar-informal-historico.html

Bergstra, J., & Bengio, Y. (2012). Random Search for Hyper-Parameter Optimization. *Journal of Machine Learning Research*, 284.

Bischl, B., Binder, M., Lang, M., Pielok, T., Richter, J., Coors, S., . . . Thomas, J. (2021). Hyperparameter Optimization: Foundations, Algorithms, Best Practices and Open Challenges.

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, (págs. 785-794).

Chongstitvatana, C., & Vithitsoontorn, P. (2022). Demand Forecasting in Production Planning for Dairy Products Using Machine Learning and Statistical Method. *International Electrical Engineering Congress*.

Deniz, N., & Ozceylan, E. (2023). A bibliometric and social network analysis of data-driven heuristic methods for logistics problems. *Journal of Industrial and Management Optimization*, 5671-5689.

Dineva, K., & Atanasova, T. (2023). Forecasting Weekly Cow Milk Production Using a Multivariate Time Series Approach. *23rd SGEM International Multidisciplinary Scientific GeoConference 2023.* Sofia: Bulgarian Academy of Sciences.

DNL. (s.f.). *Dirección Nacional de Lechería*. Obtenido de https://www.magyp.gob.ar/sitio/areas/ss_lecheria/

EmergentCold LatAm. (2025). Obtenido de Logística de productos lácteos: desafíos en el transporte y almacenamiento: https://emergentcoldlatam.com/logistica/logistica-de-productos-lacteos/

Goli, A., Khademi Zare, H., Tavakkoli-Moghaddam, R., & Sadeghieh, A. (2018). A comprehensive model of demand prediction based on hybrid artificial intelligence and metaheuristic algorithms: A case study in dairy industry. *Journal of Industrial and Systems Engineering*, 190-203.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction (Second Edition). New York: Springer.

Hyndman, R., & Athanasopoulos, G. (2020). *Forecasting: Principle and Practice*. Obtenido de https://otexts.com/fpp3/accuracy.html

INDEC. (s.f.). *Índice de precios al consumidor*. Obtenido de https://www.indec.gob.ar/indec/web/Nivel4-Tema-3-5-31

Jain, M. (15 de March de 2016). *Complete guide to parameter tuning in XGBoost (with codes in Python)*. Obtenido de Analytics Vidhya: https://www.analyticsvidhya.com/blog/2016/03/complete-guide-parameter-tuning-xgboost-with-codes-python/

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An Introduction to Statistical Learning: with Applications in R.* Springer.

Juan, A., Panadero, J., de la Torre, R., Reyes-Rubiano, L., Faulin, J., & Latorre, I. (2019). Simulation-based optimization in transportation and logistics: comparing sample average approximation with simheuristics. *2019 Winter Simulation Conference*, (págs. 1906-1917).

Kuhn, M., & Johnson, K. (2013). Applied predictive modeling. New York: Springer.

Meteostat. (s.f.). *Meteostat*. Obtenido de https://meteostat.net/es/place/ar/retiro?s=87585

Ministerio de Agricultura, Ganadería y Pesca. (2001). Obtenido de Análisis de la cadena de Productos Lácteos: https://alimentosargentinos.magyp.gob.ar/contenido/sectores/lacteos/productos/01_lacteos/Lacteos_02.htm

Nguyen, T. (2023). *Machine Learning Model for Forecasting Perishable Foods in Retail Business.* Tilburg: Tilburg University.

Observatorio de la Cadena Láctea Argentina (OCLA). (2019). *Estructura de la comercialización de productos lácteos*. Obtenido de https://www.ocla.org.ar/noticias/14887873-estructura-de-la-comercializacion-de-productos_lacteos

Probst, P., Boulesteix, A., & Bischl, B. (2018). Tunability: Importance of hyperparameters of machine learning algorithms. *Journal of Machine Learning Research*.