



Universidade de Vigo

Trabajo Fin de Máster

Optimización MILP en retailing

Sabela Fernández Villar

Máster en Técnicas Estadísticas

Curso 2017-2018

Propuesta de Trabajo Fin de Máster

Título en galego: Optimización MILP en retailing
Título en español: Optimización MILP en retailing
English title: MILP Optimization in retailing
Modalidad: Modalidad B
Autora: Sabela Fernández Villar, Universidade de Santiago de Compostela
Directora: María Luisa Carpenle Díaz, Universidade da Coruña
Tutor: Ángel Manuel González Rueda, INDRA SOFTWARE LABS
Breve resumen del trabajo: En este informe se profundiza en el modelo de optimización para la distribución del inventario, empleado por Zara en la actualidad. Dicho modelo se divide en dos partes para poder gestionar el tamaño de los datos, un primer modelo que maximiza los ingresos en cada país y un segundo modelo que minimiza los costes a nivel de tienda y referencia. Con el objetivo de reducir el tiempo computacional, se programan una serie de heurísticas para poder prescindir del primero de los modelos. Finalmente se comprueba si se obtienen resultados similares a los que proporciona el modelo actual.
Recomendaciones: El alumno debería haber cursado la asignatura de Programación Matemática y sentirse cómodo programando en R/AMPL y Java/C o algún otro lenguaje similar.

Doña María Luisa Carpenté Díaz, Profesora titular de la Universidade da Coruña, don Ángel Manuel González Rueda, Consultor data scientist senior de INDRA SOFTWARE LABS, informan que el Trabajo Fin de Máster titulado

Optimización MILP en retailing

fue realizado bajo su dirección por doña Sabela Fernández Villar para el Máster en Técnicas Estadísticas. Estimando que el trabajo está terminado, dan su conformidad para su presentación y defensa ante un tribunal.

En Santiago de Compostela, a 4 de septiembre de 2018.

La directora:

El tutor:

Doña María Luisa Carpenté Díaz

Don Ángel Manuel González Rueda

La autora:

Doña Sabela Fernández Villar

Agradecimientos

Expresar mi más sincero agradecimiento a mis tutores Luisa Carpenté y Ángel Manuel González, por resolver todas las dudas que me han ido surgiendo a lo largo de la elaboración del presente trabajo, y por sus correcciones y aportaciones realizadas, las cuales permitieron aumentar la calidad del trabajo.

Índice general

Resumen	XI
1. Introducción	1
2. Conceptos Previos	3
2.1. Trayectoria de la Compañía	3
2.2. Introducción a la Terminología Básica	3
2.2.1. Unidades en Almacén	4
2.2.2. Campañas de Rebajas y Temporadas	4
2.2.3. Equipo de Organización	5
2.2.4. Grupos y Clústers de Rebajas	5
2.3. Proceso de Distribución de Rebajas Heredado	6
2.3.1. Selección de la Referencia	6
2.3.2. Elección de la Tienda	7
2.3.3. Tamaño de Envío a Tienda	7
2.4. Proceso de Distribución Actual	7
2.4.1. Comercialización y Distribución de la Temporada Regular	7
2.4.2. Comercialización y Distribución de la Temporada de Rebajas	8
3. Modelo Inicial	11
3.1. Archivos en Python	11
3.2. Modelo I: Formulación de la Maximización de Ingresos	12
3.2.1. Índices y Conjuntos de Índices	12
3.2.2. Observación. Ilustración de los Grupos y Clúster de Precios	12
3.2.3. Parámetros	13
3.2.4. Variables de Decisión	15
3.2.5. Observación. Ilustración de las Variables x_{mgnk}^w e y_{mgnk}^w	15
3.2.6. Función Objetivo	16
3.2.7. Restricciones del Modelo I	17
3.2.8. Desarrollo del Modelo I	21
3.3. Observación. Diferencias código Python y AMPL	23
3.3.1. Variación en la Función Objetivo	23
3.3.2. Variación en las Restricciones	23
3.3.3. Ejemplo básico	23
3.4. Modelo II: Formulación para Minimizar Costes	27
3.4.1. Índices y Conjuntos de Índices	27
3.4.2. Parámetros	28
3.4.3. Variables de Decisión	30
3.4.4. Función Objetivo	30
3.4.5. Restricciones	31
3.4.6. Desarrollo del Modelo II	33
3.5. Observación. Diferencias código Python y AMPL	34
3.5.1. Ejemplo Básico	35
3.5.2. Observación. Problema variables no enteras	37

4. Puesta en Funcionamiento del Modelo	39
4.1. Generando los Datos	39
4.1.1. Generando Inventarios	40
4.1.2. Generando Demandas	41
4.1.3. Generando Precios	43
4.2. Pruebas Computacionales e Implementación Práctica	44
4.2.1. Salida del Modelo I	44
4.2.2. Salida del Modelo II	46
5. Profundizando en el Modelo	49
5.1. Archivos en Python	49
5.2. Heurística para el Cálculo de la Demanda	49
5.3. Primera Heurística para el Cálculo del Envío	52
5.4. Segunda Heurística para el Cálculo del Envío	55
5.5. Tiempo de Ejecución	58
5.6. Comparación de Resultados	59
5.7. Conclusiones	59
A. Modelo de Predicción de la Demanda	63
A.1. Construcción del conjunto de datos de demanda	63
A.2. Forma del Modelo de Predicción	65
A.3. Método Residual para la Estimación de Parámetros	66
A.4. Procedimiento para la estimación de Parámetros	66
B. Código en Python: Bmgn	69
C. Código en Python: Modelo I	71
D. Código en Python: Brj	75
E. Código en Python. Modelo II	79
F. Código en Python: Datos del Modelo	81
G. Código en Python: Heurística III	87
Bibliografía	89

Resumen

Resumen en español

En este informe se profundiza en el modelo de optimización para la distribución del inventario, empleado por Zara en la actualidad. Dicho modelo se divide en dos partes para poder gestionar el tamaño de los datos, un primer modelo que maximiza los ingresos en cada país y un segundo modelo que minimiza los costes a nivel de tienda y referencia. Con el objetivo de reducir el tiempo computacional, se programan una serie de heurísticas para poder prescindir del primero de los modelos. Finalmente se comprueba si se obtienen resultados similares a los que proporciona el modelo actual.

English abstract

This report examines in greater depth the optimization model for inventory distribution currently used by Zara. This model is divided into two parts in order to manage the size of the data, a first model that maximizes revenue in each country and a second model that minimizes costs at the store and reference level. In order to reduce the computational time, a series of heuristics are programmed in order to dispense with the first of the models. Finally, it is checked whether similar results are obtained to those provided by the current model.

Capítulo 1

Introducción

La integración de la planificación de inventario y las estrategias de ventas es una realidad en el sector minorista. Tanto, que se puede considerar una necesidad. En el mundo de la moda rápida, las estrategias de inventario son fundamentales para el éxito. El inventario es considerado como un artículo perecedero que conduce a períodos de venta cortos. Una práctica común de los minoristas es liquidar el inventario no vendido fijando precios de liquidación (Wang & Webster 2009). De 1925 a 1965 el promedio se mantuvo estable en el 6 %, pero a mediados de los años noventa había superado el 26 % (Mantrala & Rao 2001).

Este aumento en las unidades de liquidación se debe principalmente a la incertidumbre de la demanda, que se sigue de decisiones de descuento tomadas de forma dinámica. Mantrala y Rao (2001) afirman que en la década de 1990 se produjo una explosión de la variedad de productos por el lado de la oferta y que los gustos de los consumidores se diversificaron por el lado de la demanda, aumentando su volatilidad. En la actualidad, los minoristas emplean políticas de reducción del inventario de liquidación. Para ello se centran en maximizar los ingresos. Esto ha provocado un aumento de las estrategias dinámicas de fijación de precios en los últimos años para facilitar estas necesidades (Keskinocak & Elmaghraby 2003).

La integración de la comercialización y el control de inventarios fue propuesta por primera vez por Whitin en 1955 y desde entonces ha recibido una amplia atención en la literatura de comercialización, economía y gestión de inventarios (Smith & Achabal 1998). En 1991, por ejemplo, Eliasberg y Steinberg diseñaron un modelo integrado de decisión de producción de mercadotecnia conjunta. También se han analizado las estrategias de fijación de precios con respecto a las reducciones del precio de rebajas encontradas en (Gallego & van Ryzin 1994), así como (Feng & Gallego 1995), entre otros.

A continuación se presentará el caso de Zara, un minorista de moda rápida con sede en España. Este proyecto parte del artículo *Clearance pricing optimization at Zara* de Rodolfo Carboni Borrasé, para entender el contexto del problema. Se estudiará el modelo de optimización que emplea Zara en la actualidad, un problema de programación lineal entera mixta dividido en dos partes. Una vez terminada dicha familiarización y análisis, se trabajará en la mejora de algoritmos que actualmente se encuentran en funcionamiento, de forma que la calidad de la solución se mantenga y se reduzca el tiempo computacional. El caso de negocio de la empresa Zara es muy similar al de otros minoristas donde los ingresos necesitan ser maximizados. Sin embargo, su modelo de negocio y su cadena de suministro integrada verticalmente hacen que este caso sea único. La solución propuesta sugiere una metodología más sofisticada para coordinar la distribución de inventarios y las decisiones de descuento.

Este proyecto se divide en cinco capítulos. En el Capítulo 2, se presentan una serie de conceptos previos, necesarios para entender los capítulos posteriores. En el Capítulo 3, se describe con detalle el modelo de optimización empleado en la actualidad por Zara, el cual consta de dos partes. Se exponen una serie de ejemplos básicos con el fin de que la familiarización con el problema sea más rápida. En el capítulo 4, se generan datos aleatorios con los cuales se pondrá en funcionamiento el modelo de optimización que ha sido programado en lenguaje Python. Finalmente en el Capítulo 5, se programan algoritmos con el fin de mejorar el tiempo computacional, intentando que la calidad de la solución se mantenga. Todo esto irá acompañado de los códigos en lenguaje Python que han sido programados para recrear el modelo.

Capítulo 2

Conceptos Previos

2.1. Trayectoria de la Compañía

En 1975, Amancio Ortega Gaona abrió la primera tienda de Zara en La Coruña, España. Rápidamente ganó popularidad entre los clientes, ya que ofrecía ropa de diseño de alta calidad a precios razonables. En 1989, ya contaba con 82 tiendas en España y presencia internacional en Portugal, París y Nueva York. El grupo Inditex, con empresa matriz Zara, comenzó a cotizar en bolsa en 2001 y fue el tercer minorista de ropa más grande. Hoy, Inditex es el minorista de moda rápida más grande del mundo por ventas, superando a GAP Inc. en 2008. Inditex tiene ocho formatos comerciales, incluidos Zara, Pull & Bear, Massimo Dutti, Bershka, Stradivarius, Oysho, Zara Home y Uterqüe. En 2008, Inditex tenía 89.122 empleados en 72 países con 4.264 tiendas y más de 140 nacionalidades.

Zara es la cadena insignia de Inditex con unos ingresos de 6.800 millones de euros. Es conocida por sus innovadoras tendencias en moda y rápida respuesta a la demanda del mercado. La clave del éxito de Zara y su rápido crecimiento ha sido debido a su cadena de suministro virtualmente integrada que incluye abastecimiento textil, diseño, producción, distribución y venta al por menor. Los diseñadores de Zara buscan inspiración para nuevas tendencias en los diseñadores de moda de lujo y trabajan junto con los equipos de ventas regionales de moda para responder a las demandas de los clientes. La flexibilidad de la cadena de suministro, desde el aprovisionamiento hasta la distribución, permite que las nuevas tendencias de moda lleguen a las tiendas semanalmente. Por ejemplo, el equipo logístico es capaz de aceptar pedidos de una tienda y entregar la mercancía dentro de 24 horas para tiendas europeas y dentro de 48 horas para tiendas americanas o asiáticas.

En la última década, Zara ha crecido de 507 tiendas en 39 países a más de 1.300 tiendas en 73 países, triplicando sus ingresos. Para prosperar con la gran expansión, Zara amplió su red de distribuidores en todo el mundo a otros proveedores europeos y asiáticos. Con el rápido crecimiento, Zara se vio obligada a cambiar de escala su cadena de suministro y mejorar sus procesos actuales para mantener su ventaja competitiva.

Para facilitar las estrategias de crecimiento, en 2006, Zara inició una asociación con el programa Leaders for Global Operations (LGO) en el MIT para desarrollar e incorporar nuevos pronósticos, inventarios y precios y modelos de distribución a su cadena de suministro. Las tiendas Zara tienen tres departamentos para Hombres y Mujeres y Niños. *Zara Woman* representa el sesenta por ciento de los ingresos de Zara y, por lo tanto, ha sido seleccionada para recrear el modelo de optimización.

2.2. Introducción a la Terminología Básica

En esta sección se introduce la principal terminología empleada por la gente de negocio de Inditex, que será necesaria para poder entender con facilidad los modelos planteados.

2.2.1. Unidades en Almacén

El *Stock-keeping unit* o SKU, sin traducción literal al español, podría denominarse “código de artículo”, representa para una empresa la unidad mínima de un producto que puede ser vendida, comprada, o gestionada al inventario. Las unidades en almacén de Zara están determinadas por una codificación de **referencia**, **color** y **tamaño** que es visible en cualquiera de las etiquetas de precios de sus prendas.

La codificación de referencia está compuesta por un número de cuatro dígitos que identifica el modelo de la prenda y un número de tres dígitos que identifica su calidad, donde el modelo se relaciona con el patrón de diseño de la prenda y la calidad con el material textil del que está hecha. Del mismo modo, un número de tres dígitos determina el color. Un solo dígito indica el tamaño, pero en las etiquetas impresas se utiliza un símbolo más intuitivo para el cliente, como “M”, “L” o “XL”. Por ejemplo, el código “5927/024/716 S”, identifica de manera única a una prenda como modelo 5927 con calidad 024 (o equivalente referencia 5927/024) en el color 716 y tamaño pequeño. Además, para un seguimiento muy detallado del inventario y las ventas de Zara, las designaciones de país, tienda, campaña y temporada acompañan a las designaciones de referencia, tamaño y color.

2.2.2. Campañas de Rebajas y Temporadas

Como se ilustra en la Figura 2.1, Zara divide sus colecciones en dos campañas anuales: una **campaña de verano**, que se introduce gradualmente desde el inicio del año, y una **campaña de invierno**, que se introduce desde mediados de año.

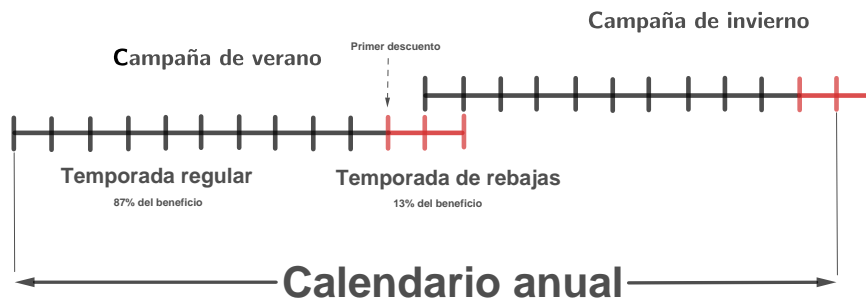


Figura 2.1: Campañas de venta de Zara.

Cada **temporada regular**, en negro en la Figura 2.1, tiene una duración de aproximadamente cinco meses, después de los cuales se vende todo el inventario restante de la temporada actual a un precio con descuento. Las **temporadas de rebajas**, en rojo en la Figura 2.1, se extienden aproximadamente durante dos meses y coexisten temporalmente con la introducción de la nueva temporada de ventas a medida que se realiza la transición de la campaña de verano a la de invierno, y viceversa. Durante la temporada de rebajas, los precios de las prendas se revisan semanalmente y se emiten descuentos de precios que aumentan gradualmente en respuesta a las ventas observadas y a los niveles de inventario restantes.

Como se ilustra en la Figura 2.1, una cantidad considerable de los ingresos de Zara se obtienen durante la temporada de rebajas. Sin embargo, al examinar la distribución de los ingresos, la importancia de las temporadas de rebajas se ve socavada por el hecho de que las prendas se venden a un precio reducido. Para visualizar mejor la importancia de la temporada de rebajas y su impacto en los beneficios netos de Zara, la Figura 2.2 ilustra las distribuciones de inventario vendido durante las temporadas de rebajas y regulares. De todo el inventario que está disponible en la campaña de verano o invierno, aproximadamente más del ochenta por ciento se vende durante las ventas de la temporada regular, mientras que el veinte

por ciento restante (más bajo que el promedio de la industria) se ofrece a un precio con descuento durante la temporada de rebajas. De este total, el noventa y cuatro por ciento del inventario se vende durante la temporada de rebajas, y a un precio medio de reducción del cincuenta por ciento, mientras que el seis por ciento restante se vende por un valor de rescate a otros distribuidores que operan fuera del mercado objetivo de Zara.

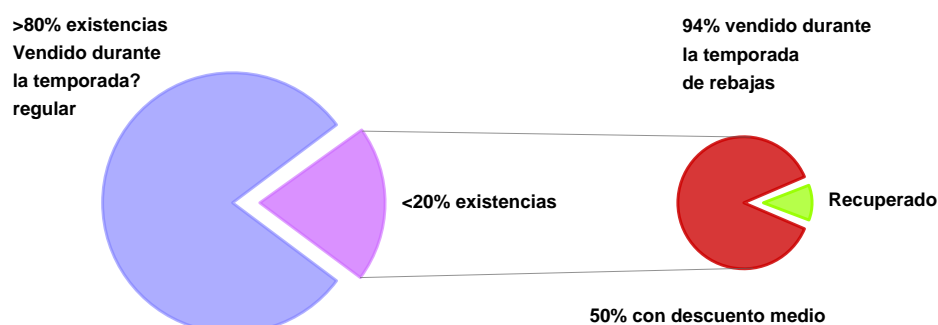


Figura 2.2: Distribución de las ventas del inventario.

2.2.3. Equipo de Organización

Hay tres grupos que colaboran para determinar la distribución del inventario durante la temporada de rebajas:

- **Equipo de Gestión del Rendimiento (EGR):** Estima las ventas esperadas de cada tienda tanto en ingresos como en ventas unitarias. Son responsables de mantener y actualizar el programa de envíos de la tienda cada semana.
- **Grupo de distribución:** Organiza el calendario de transporte a los centros de distribución (CD) para crear órdenes de trabajo. Debido a que el calendario especifica el número de unidades para cada tienda, corresponde al centro de distribución tomar y seleccionar las unidades de almacén para distribuir las SKUs a cada tienda.
- **Gerentes comerciales:** Son responsables de las ventas en tienda y expertos en el mercado de los países y regiones que gestionan. Están en contacto diario con los gerentes de tienda y están familiarizados con las necesidades de inventario de la tienda. Una vez que se libera el programa de envíos, trabajan con el EGR y el grupo de distribución para realizar cualquier ajuste en la programación. Los gerentes comerciales también pueden bloquear ciertas tendencias en una región determinada. Un ejemplo común es el bloqueo de la distribución de abrigos al Medio Oriente durante los meses de verano. Debido a las condiciones climáticas cálidas, los clientes son menos propensos a comprar abrigos a pesar de que tengan un precio rebajado.

2.2.4. Grupos y Clústers de Rebajas

Durante la temporada de rebajas, todas las referencias se agrupan en uno de los veintidós **grupos de rebajas**, que se definen de acuerdo a los tipos generales de prendas y las propiedades de los productos, por ejemplo, chaquetas básicas, vestidos TRF ¹, camisetas y polos, etc. Además, las referencias dentro de un grupo de rebajas se agrupan en **clúster** o **conglomerados de precios** según sus precios en la temporada regular. Por ejemplo, el clúster de precios 59,90 euros dentro del grupo de rebajas de faldas

¹TRF significa Trafaluc, que es uno de los cuatro departamentos principales de la colección de ropa de mujer de Zara. La sección de TRF está dirigida a clientes más jóvenes que las colecciones WOMAN o STUDIO, que son más clásicas.

de TRF incluye todas las referencias dentro de este grupo cuyo precio fue de 59,90 euros durante la temporada regular.

Las ventas e inventarios de la temporada de rebajas se controlan diariamente, y las nuevas decisiones de fijación de precios se toman aproximadamente cada semana. De esta manera, la temporada de rebajas se divide en varios períodos de rebajas, donde el final de un período y el comienzo del siguiente se determinan cada vez que se toman nuevas decisiones de fijación de precios. Los períodos de rebajas pueden o no corresponderse a una semana natural y, por lo tanto, se definen sólo en función de los momentos en que se toman las decisiones de fijación de precios.

2.3. Proceso de Distribución de Rebajas Heredado

El método de predicción utilizado anteriormente a la creación de los modelos de optimización se basa en datos de campañas y temporadas similares. Por lo tanto, al estimar las ventas para la temporada de rebajas del invierno actual se utilizan los datos de ventas de rebajas del invierno del año anterior. La predicción se calcula teniendo en cuenta las ventas del año anterior y la tendencia actual de ventas para esa misma campaña y temporada. El EGR gestiona un calendario de los envíos, que se manda al centro de distribución para decidir qué mercancías recibirá cada tienda. El centro de distribución utiliza un proceso en tres etapas para determinar los transportes. Estos pasos son: **seleccionar una referencia**, **elegir la tienda** a la que enviar la referencia y calcular el **tamaño del envío**.

2.3.1. Selección de la Referencia

El primer paso es seleccionar una referencia para distribuir. El sistema de recolección de Zara se basa en introducir una referencia y asignar la cantidad a enviar a todas las tiendas. Para elegir la referencia r , el equipo de distribución considera que tienen prioridad las mercancías con el mayor **índice de éxito**, la menor **relación existencias-ventas** y el mayor número de **unidades en el inventario**.

■ Índice de éxito:

Cada referencia tiene un índice de éxito IE_r , que es la proporción de ventas totales de una referencia ante la oportunidad total de ventas para la temporada actual. Esto es una estimación del éxito de ventas durante la vida útil del artículo. El rango del índice de éxito es de 0 a 1 y se utiliza como porcentaje de las existencias vendidas en una tienda determinada. Un índice de éxito de 0.8 significa que se ha vendido el 80 por ciento del inventario total disponible para la tienda.

$$IE_r = \frac{Ventas_r}{Inventario_r + Ventas_r} \quad (2.1)$$

La Ecuación (2.1) describe esta relación, en la que:

- $Inventario_r$: son las existencias actuales disponibles en una tienda determinada.
- $Ventas_r$: es el total de ventas de esa referencia.

■ Relación existencias-ventas:

También se considera que tienen prioridad las referencias con una baja relación entre existencias disponibles y ventas. Dicha relación, o rotación Rot_r , es un índice directo de las existencias que deben estar disponibles en cada tienda en un momento dado. Divide las existencias al final de un período entre las ventas durante ese período. La rotación es el número de días que el inventario actual permanecerá disponible en una tienda si la tasa de ventas permanece igual durante ese período.

Esto difiere del índice de éxito, ya que sólo se captura en un período determinado y no en toda la temporada. En Zara, la relación existencias-ventas se estima diaria o semanalmente. La Ecuación (2.2) es el cálculo de la rotación para un período dado.

$$Rot_r = \frac{Inventario_r}{Ventas_r} \quad (2.2)$$

Teniendo en cuenta el índice de éxito IE_r , y la relación entre existencias-ventas Rot_r , el CD está finalmente eligiendo enviar los artículos que están vendiendo muy bien primero. Cuanto antes estén estos artículos en la tienda, más probable será que se vendan durante la primera semana de rebajas y no tendrán que ser descontados muchas veces.

- **Unidades en inventario:**

Por último, las referencias con más inventario disponible tienen prioridad de envío. Los artículos con más existencias disponibles son los que no tuvieron éxito durante la temporada regular. Si hay un exceso de cantidad de dichos artículos en los centros de distribución, es probable que se utilicen para completar pedidos de tienda debido al volumen de la mercancía.

2.3.2. Elección de la Tienda

Para elegir qué tiendas deben recibir un envío de una referencia elegida, el equipo del CD considera primero aquellas tiendas que tienen las ventas unitarias totales más altas de la referencia y las ventas más altas de esa referencia la semana anterior. Las tiendas que han sido bloqueadas por el gerente comercial tendrán una alerta en el sistema y no permitirán al usuario asignar una lista de recolección a esa tienda. Un método final para elegir una tienda se basa en el conocimiento del CD sobre ese mercado. Por ejemplo, si la referencia es un par de pantalones cortos, y se sabe que históricamente Japón vende muchos pantalones cortos, las tiendas japonesas tendrán prioridad de estos envíos con respecto a otro país.

2.3.3. Tamaño de Envío a Tienda

El tamaño del envío a cada tienda se basa en el nivel de inventario actual de esa tienda y sus ventas la semana anterior. Convencionalmente los envíos de las tiendas son más grandes para la temporada de rebajas que para la temporada regular, ya que la mercancía debe reponer existencias para toda la temporada de rebajas y no sólo para una semana de ventas. Por ejemplo, para un producto del cual hay menos de mil artículos disponibles, el CD sólo lo enviará a las tiendas con las ventas históricas más altas.

2.4. Proceso de Distribución Actual

En la actualidad, a raíz de la colaboración con los profesores Gallien y Caro, Zara ha desarrollado modelos de optimización con la finalidad de maximizar los ingresos y minimizar los costes.

2.4.1. Comercialización y Distribución de la Temporada Regular

Las decisiones sobre comercialización son tomadas por los directores de ventas. La mercancía está centralizada en Zara desde la distribución de inventarios hasta el almacenamiento en tiendas. Durante la temporada regular, la filosofía para la distribución del inventario de Zara utiliza un pronóstico para predecir las ventas de la semana posterior basándose en ventas históricas. Se utiliza un modelo de previsión de la demanda y optimización creado por Gallien y Caro (2007) para maximizar los ingresos para una referencia, y además determina cómo asignar el inventario a cada tienda. El sistema informático de

Zara también permite a los gerentes de tienda realizar pedidos de mercancías adicionales a través de un asistente digital personal (ADP).

2.4.2. Comercialización y Distribución de la Temporada de Rebajas

Zara colaboró con los profesores Gallien y Caro (2007) para ayudarles una vez más en un modelo de optimización, pero esta vez para la temporada de rebajas. El enfoque adoptado por los profesores fue desarrollar un proceso en dos etapas.

Fijación de Precios

La primera etapa consistía en establecer una política de fijación de precios para garantizar que los artículos durante la temporada de rebajas se tasaran de manera que se maximizaran los ingresos. En 2007, LGO Rodolfo Carboni en conjunto con Gallien y Caro trabajaron en un modelo de optimización que facilitaría las decisiones de precios (Carboni 2008).

Distribución del Inventario

La segunda etapa se basaba en asegurar que las tiendas recibieran la mercancía adecuada para la temporada de rebajas. La combinación de tener tiendas con la mercancía adecuada y a un precio justo en el mercado facilitaría la meta de Zara de maximizar los ingresos de rebajas. Esta etapa será el eje central de este proyecto, en la cual nos centraremos en capítulos posteriores.

La solución empleada por Zara para el proceso de la distribución de la mercancía en la temporada de rebajas se divide en dos partes: utilizar un pronóstico de la demanda para las ventas de final de temporada y de la temporada de rebajas, y emplear un modelo de optimización que coordina las decisiones de precios con la comercialización de la tienda durante la temporada de rebajas, como se puede ver en la Figura 2.3. En el Capítulo 3 se revisan los dos modelos de optimización actuales, basados en la programación lineal, cuyo objetivo es determinar las reposiciones al final de la temporada, así como la distribución del inventario de rebajas.

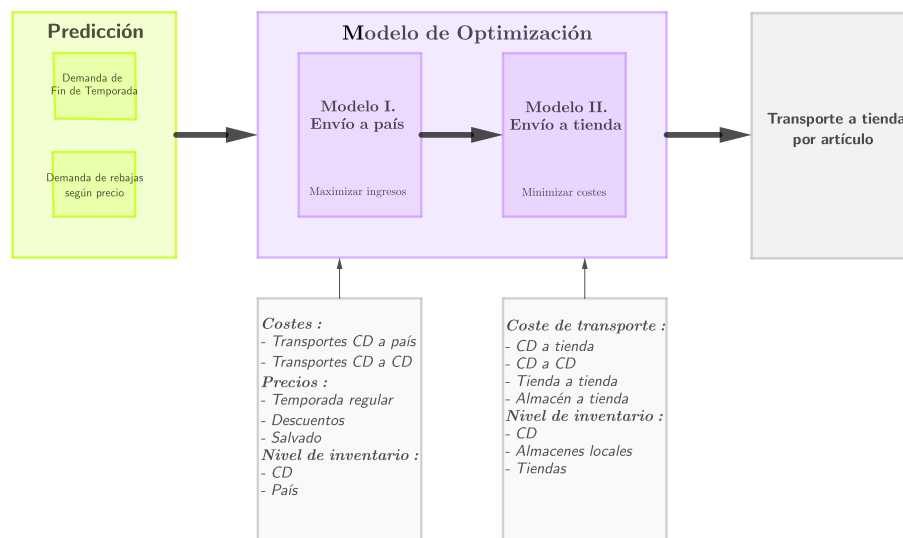


Figura 2.3: Esquema de los modelos.

Como se refleja en la Figura 2.3, el **Modelo de optimización I** utiliza las salidas obtenidas a partir de un modelo de predicción de la demanda (puede verse en el Apéndice A), desarrollado por Caro y Gallien (2007), para decidir cuánta mercancía se enviaría a cada uno de los 73 países, con el fin de maximizar los ingresos globales. Se tienen en cuenta los gastos de transporte desde los centros de distribución centrales a cada país, así como el coste de transbordo entre cada centro de distribución.

El **Modelo de optimización II** utiliza la salida que proporciona el primer modelo para desagregar los envíos de país a nivel de tienda y artículo, minimizando al mismo tiempo los costes. En este modelo, también se consideran los gastos de transporte a las tiendas desde los centros de distribución, almacenes de país y otras tiendas. Las ventajas del modelo son que utiliza un pronóstico de la demanda, considera precios y costes de descuento y determina los envíos a una tienda a nivel de referencia.

Capítulo 3

Modelo Inicial

El modelo de optimización se divide en dos modelos para poder gestionar el tamaño de los datos. El objetivo del primer modelo es maximizar las ganancias globales para cada grupo y clúster de precios determinando la asignación del inventario a cada país. Utiliza la predicción de la demanda en cada uno de los distintos puntos de precio desde el precio de temporada regular hasta los distintos descuentos de precio durante la temporada de rebajas. El resultado de este modelo será el número de unidades que se enviarán a un país en un determinado grupo y clúster de precios.

En el segundo modelo se asigna cada grupo y clúster, según un nivel de referencia, a cada tienda minimizando los costes de envío. Con este modelo, los usuarios pueden determinar qué referencia enviar a cada tienda desde los centros de distribución y almacenes locales, así como qué referencias transferir de una tienda a otra. Estos dos modelos están diseñados para funcionar unas semanas antes de que empiece la temporada de rebajas, para asignar el número total de unidades por referencia que enviar a cada tienda para las ventas de fin de temporada y temporada de rebajas. Los dos modelos fueron formulados por el profesor Felipe Caro (2009).

Para diferenciar entre las variables y los parámetros del modelo, las primeras se escribirán en negrita. Además, para distinguir entre la temporada regular y la temporada de rebajas, las variables y parámetros con tilde harán referencia a la temporada regular. Por ejemplo, \tilde{D}_{mgn} , y D_{mgn} , hacen referencia a la previsión de ventas para un país determinado m , grupo g y clúster n durante la temporada regular y la de rebajas respectivamente.

3.1. Archivos en Python

Los modelos de optimización fueron formulados usando un modelo de programación matemática lineal entera mixta codificado en el lenguaje de programación Python y resuelto usando el solver de programación lineal Gurobi. El código de cada modelo de optimización está contenido en un fichero `.py`.

- `ModeloI.py` y `ModeloII.py`: Estos archivos contienen el modelo de optimización cuya formulación incluye las variables, parámetros, función objetivo y restricciones.
- `Modelo_datos.py`: El propósito de este módulo es generar los parámetros de entrada en los dos modelos. Los datos de entrada incluyen las estimaciones de demanda, la posición de inventario, precios y constantes del modelo.
- `Bmgn.py` y `Brj.py`: El propósito de estos dos módulos es calcular, con la información del módulo `Modelo_datos.py`, los parámetros de entrada relativos a los requisitos máximos y mínimos de envío, B_{mgn} , B_{rj} y L_j , que se describen en secciones posteriores de este capítulo.

3.2. Modelo I: Formulación de la Maximización de Ingresos

El objetivo de este modelo de optimización es maximizar las ganancias globales al tomar decisiones de distribución de inventarios basadas en un sistema de precios dinámico. El modelo tiene en cuenta cada grupo de productos y cada clúster de precios para determinar la asignación del inventario a cada país.

Las estimaciones de inventario y demanda se toman del sistema POS¹ en cada período con el fin de aumentar la exactitud de los datos. Para las reducciones de precio en temporada de rebajas se supone que las decisiones de determinación de precios son permanentes, por lo que no pueden aumentar a lo largo del tiempo. Se introducirán primero las notaciones generales del modelo, seguidas de la definición de los parámetros.

3.2.1. Índices y Conjuntos de Índices

- $m \in \mathcal{M}$: Cada país dentro de la red de distribución. El índice m también se utilizará para denotar almacenes locales para aquellos países en los que existen. En 2015, fueron abiertas 77 nuevas tiendas de Zara, hasta llegar a las 2162 en 88 países.
- $a \in \mathcal{A}$: Principales centros de distribución ubicados en España. Por tanto, $a(m)$ indica el centro de distribución que abastece al país m y $a(j)$ el centro de distribución que abastece a la tienda j . $\mathcal{M}(a) \subseteq \mathcal{M}$ y $\mathcal{T}(a) \subseteq \mathcal{J}$ denotan el conjunto de países y tiendas respectivamente, abastecidas por los centros de distribución. La empresa Zara dispone de tres centros de distribución, ubicados en España, cuyos códigos son “10001”, “10018” y “10026”, que serán empleados más adelante, una vez formulado el modelo de optimización.
- $g \in \mathcal{G}$: Grupos de ventas de rebajas que representan cada línea de productos y están numerados del 1 al 21.
- $n \in \mathcal{N}(g)$: Clúster de precios en un grupo determinado g . Cada grupo g tiene agrupaciones de precios distintas que representan varias referencias r .
- $k \in \mathcal{K} = \{k : k \geq 1\}$: Los precios de rebajas están indexados por k . Donde $\mathcal{P}(m)$ denota el conjunto de precios de rebajas que aumenta por convenio, es decir, $p_{mg1} \leq p_{mg2} \leq \dots \leq p_{mgk}$, donde p_{mg1} , es el precio de la liquidación final en el país m .
- $w \in \mathcal{W} = \{w : 0 \leq w < W_r\}$: Períodos comprendidos en la temporada de rebajas (de *week* en inglés), donde $w = 0$ representa el primer período de la temporada de rebajas y $w = W_r$ el último período de la temporada.

3.2.2. Observación. Ilustración de los Grupos y Clúster de Precios

Se presenta un ejemplo ilustrativo para entender los conceptos de **grupo de rebajas** y de **clúster de precios**. Como ya se ha explicado en la Subsección 2.2.4, los grupos de rebajas se definen de acuerdo a los tipos generales de prendas y las propiedades de los productos. Además, las referencias dentro de un grupo de rebajas se agrupan en clúster o conglomerados de precios según sus precios en la temporada regular.

Se accede a la página web de Zara <https://www.zara.com> y se seleccionan cuatro artículos pertenecientes al grupo de ventas de **vestidos de mujer**, los que se muestran en la Figura 3.1. Dentro del grupo de ventas de vestidos de mujer, los vestidos 1 y 2 se pueden clasificar en el **clúster de precios de 29.95 euros** y los vestidos 3 y 4 en el **clúster de precios de 39.95 euros**.

¹El sistema POS, del inglés *Point of Sale*, o Punto de Venta en español, es un conjunto de herramientas de hardware y software, que le permiten a los negocios, facturar sus ventas, facilitando llevar el control de su flujo de caja, inventarios, compras, etc. Además de mejorar la atención al cliente en el punto de venta.



Figura 3.1: Vestidos de Zara.

3.2.3. Parámetros

Cada país tiene distintos grupos de precios basados en su moneda y en la demanda del mercado local. Por ejemplo, un artículo en el centro de distribución puede clasificarse en el clúster de 9,99 euros. Ese mismo artículo en un país diferente puede tener un precio diferente para la moneda local del país. Para asignar el inventario se estandarizaron los clúster en las categorías de precios de los centros de distribución situados en España. Por lo tanto, los clúster se agruparon de la misma forma que en los centros de distribución y los precios vienen dados en euros.

- \tilde{P}_{mgn} : Precio de temporada regular en euros, para un clúster n en un país m y grupo g .
- $psaldero_g$: Precio de liquidación para cada grupo g al final de la temporada de rebajas.

Al asignar el inventario de los centros de distribución, el modelo incluye mercancías para la temporada actual en tránsito o pedidos al proveedor. Es importante contar este inventario ya que sólo se venderá durante la temporada actual y se distribuirá para el período actual, así como para la temporada de rebajas. Cualquier cambio en el cumplimiento del proveedor se actualiza semanalmente y se refleja en el recuento de inventario. Por lo tanto, el inventario disponible en el período actual se indica como sigue:

- \tilde{I}_{agn} : Inventario disponible en unidades, al principio del período actual en la distribución, en el CD a y en el grupo g y clúster n . El inventario también incluye la mercancía entrante para la temporada regular actual.
- \tilde{I}_{mgn} : Inventario disponible en unidades, al principio del período actual en el país m , grupo g y clúster n .

La demanda para cada país, grupo y clúster, según el precio de rebajas, se obtiene a partir del modelo de predicción de la demanda (para más detalles en el modelo de predicción y la obtención de dicho parámetro véase el Apéndice A).

- $D_{mgnk}^0 := \sum_{r \in \mathcal{N}(g)} D_{mrk}^0$: Demanda estimada en unidades para el primer período de la temporada de rebajas para el clúster n , en el país m y grupo g , a un precios de liquidación p_{mgk} .
- $\tilde{D}_{mgn} := \sum_{r \in \mathcal{N}(g)} D_{mr}$: Ventas estimadas en unidades para el final de la temporada regular para el clúster n , en el país m y grupo g .

Para garantizar que el inventario se distribuya entre varios países se establece una restricción máxima de transporte.

- B_{mgn} : Expedición máxima para el clúster n , en el país m y grupo g . Hay cuatro pasos para calcular este parámetro, que se especifican en el Algoritmo 1. El código en Python del Algoritmo 1, que se emplea para el cálculo del parámetro B_{mgn} se puede ver con detalle en el Apéndice B.

Para calcular el índice de éxito del país, primero se deben estimar los índices de éxito de todas las aperturas, cierres o ampliaciones de tiendas nuevas. Cada año hay nuevas aperturas de tiendas y para contabilizar las ventas esperadas de estas tiendas, debe asignarse a la nueva tienda un nivel de ventas de una tienda equivalente que se haya abierto durante al menos un año.

Algoritmo 1 Cálculo de B_{mgn}

Entrada: $I_{mgn}, I_{agn}, Ventas_{mgn}$

- 1: **for** $(m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N}$ **do**
- 2: Calcular $IE_{mgn} = \frac{Ventas_{mgn}}{I_{mgn} + Ventas_{mgn}}$
- 3: **end for**

- 4: **for** $(m, g, n) \in \mathcal{M}_{gn} \times \mathcal{G} \times \mathcal{N}$ **do**
- 5: Calcular la media (μ_{mgn}) y desviación típica (σ_{mgn}) de IE_{mgn}
- 6: **if** $IE_{mgn} < \mu_{mgn} - 1,5\sigma_{mgn}$ **then**
- 7: $B_{mgn} = 0$ y $\mathcal{M}_{gn} = \mathcal{M}_{gn} \setminus \{m\}$
- 8: **end if**
- 9: **end for**

- 10: **for** $(m, g, n) \in \mathcal{M}_{gn} \times \mathcal{G} \times \mathcal{N}$ **do**
- 11: Calcular $B_{mgn} = \sum_{a \in A} \tilde{I}_{agn} \cdot v_{mgn}$, donde $v_{mgn} = \frac{Ventas_{mgn}}{\sum_{m' \in \mathcal{M}_{gn}} Ventas_{m'gn}}$
 En donde v_{mgn} mide el porcentaje de ventas de cada tienda para un clúster de precios determinado en su respectivo país.

- 12: Se define X_{mgn} el envío mínimo a nivel de país, grupo y clúster.
- 13: **if** $B_{mgn} < X_{mgn}$ **then**
- 14: $B_{mgn} = 0$ y $\mathcal{M}_{gn} = \mathcal{M}_{gn} \setminus \{m\}$
- 15: Recalcular B_{mgn}
- 16: **end if**
- 17: **end for**

Los siguientes parámetros o constantes se pueden ajustar en base a la política y los estándares del negocio:

- C_E : Porcentaje del precio del artículo que representa los gastos de envío por unidad de los centros de distribución al país m .
- C_A : Porcentaje del precio del artículo que representa los gastos de transporte por unidad de un centro de distribución a otro.
- D : Número de días en un período de rebajas.
- W_r : Número de períodos en la temporada de rebajas.
- S_A : Porcentaje de existencias en los centros de distribución que se puede salvar.
- S_T : Porcentaje del total de existencias mundiales que pueden ser recuperadas.
- Q_m : Cantidad mínima de transporte en unidades por país m .
- $desc_min$: Descuento mínimo para la primera semana de venta. Esto obliga al modelo a hacer un descuento mínimo por el que el precio para cada clúster debe reducirse con respecto a la temporada regular.

- **tiempo**: Factor tiempo que disminuye la cantidad de ventas de liquidación del país cada semana. Un factor de tiempo de 0.80 representa que la cantidad máxima de ventas del periodo de rebajas actual puede ser el 80 % de las ventas estimadas de la semana anterior.

3.2.4. Variables de Decisión

- q_{mgn} : Flujo de inventario en unidades desde los centros de distribución al país m para el grupo g y el clúster n .
- I_{mgn}^w : Inventario disponible en unidades para el período w durante la temporada de rebajas en el país m , del clúster n en el grupo g .
- \tilde{d}_{mgn} : Ventas previstas en unidades durante la temporada regular en el país m , del clúster n en el grupo g .
- d_{mgnk}^w : Ventas previstas en unidades para el período w durante la temporada de rebajas en el país m para el clúster n en el grupo g a precio de liquidación k .
- $q_{aa'gn}$: Flujo de inventario en unidades desde el centro de distribución a hasta a' para el clúster n en el grupo g .
- $x_{mgnk}^w \in \{0, 1\}$: Variable de decisión binaria que asume el valor 1 para indicar si en el país m a un clúster n , en el grupo g se le asigna un precio de liquidación inferior o igual a p_{mgk} durante el período de rebajas w . En otras palabras, si durante el período de rebajas w' , el precio del clúster n' en el grupo g' y país m' es $p_{mg'k'}$, entonces todas las variables $x_{m'g'n'k'}^{w'}$ de manera que $k \geq k'$ se establecerán en 1.
- $y_{mgnk}^w \in \{0, 1\}$: $y_{mgn1}^w = x_{mgn1}^w$; $y_{mgnk}^w = x_{mgnk}^w - x_{mgnk-1}^w \forall k > 1$: Variable de decisión binaria definida en función de la variable x_{mgnk}^w previamente definida, que asume el valor de 1, si en el país m , el precio p_{mgk} se asigna al clúster n , en el grupo g durante el período de rebajas w .

Como se discutirá en breve, la implementación de las variables binarias x_{mgnk}^w e y_{mgnk}^w permite establecer limitaciones de modelo que responden a las necesidades particulares del negocio de Zara durante la temporada de rebajas.

- $Lsaldero_{mg}$: Inventario total que se recuperará en el país m del grupo g .

3.2.5. Observación. Ilustración de las Variables x_{mgnk}^w e y_{mgnk}^w

Para entender de forma clara la construcción de la variable x_{mgnk}^w se presentan dos ejemplos, cuyo esquema de precios en la temporada de rebajas se detalla en la Figura 3.2 y la Figura 3.3.

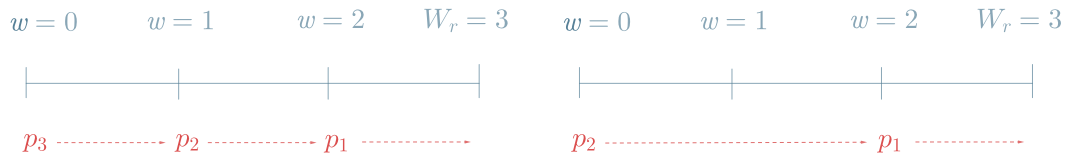


Figura 3.2: Precios rebajas primer caso.

Figura 3.3: Precios rebajas segundo caso.

Si se consideran fijos los índices de país, grupo y clúster, la variable resultante depende solamente de dos índices x_k^w . De este modo, construyendo una matriz con los términos de esta variable, se obtiene una forma sencilla e intuitiva de representarla. Dicha matriz dispone de tantas filas como precios y columnas

como etapas haya en la temporada de rebajas (habrá tantas matrices como combinaciones entre países y clústers). Para los dos esquemas de rebajas vistos, se construye su respectiva matriz con elementos x_k^w .

$$\begin{pmatrix} x_1^0 & x_1^1 & x_1^2 \\ x_2^0 & x_2^1 & x_2^2 \\ x_3^0 & x_3^1 & x_3^2 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

Figura 3.4: Matriz de x_k^w primer caso.

$$\begin{pmatrix} x_1^0 & x_1^1 & x_1^2 \\ x_2^0 & x_2^1 & x_2^2 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

Figura 3.5: Variable x_k^w segundo caso.

Del mismo modo que se hizo para la variable x_{mgnk}^w , una vez fijados los índices relativos al país, grupo y clúster, se construyen las matrices de variables con elementos y_k^w para los dos casos citados anteriormente.

$$\begin{pmatrix} y_1^0 & y_1^1 & y_1^2 \\ y_2^0 & y_2^1 & y_2^2 \\ y_3^0 & y_3^1 & y_3^2 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

Figura 3.6: Matriz de y_k^w primer caso.

$$\begin{pmatrix} y_1^0 & y_1^1 & y_1^2 \\ y_2^0 & y_2^1 & y_2^2 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}$$

Figura 3.7: Matriz de y_k^w segundo caso.

3.2.6. Función Objetivo

La función objetivo del Modelo I consiste en maximizar las ganancias, maximizando la suma de las ventas totales y restando los costes asociados. Se van a considerar tres elementos que pueden generar beneficios y dos elementos de costes de transporte. La función objetivo del Modelo I vendrá dada por:

$$\max \text{Ventas_FdT} + \text{Ventas_R} + \text{Ventas_S} - \text{Costes_CE} - \text{Costes_CDE} \quad (3.1)$$

En donde los elementos de la Ecuación (3.1) se definen en las siguientes líneas:

- **Ventas_FdT**: Representa los ingresos totales generados por las ventas de fin de temporada o de temporada regular.

$$\text{Ventas_FdT} = \sum_{\substack{(m,g) \in \mathcal{M} \times \mathcal{G} \\ n \in \mathcal{N}(g)}} \tilde{P}_{mgn} \times \tilde{d}_{mgn} \quad (3.2)$$

- **Ventas_R**: Esta variable representa los ingresos generados durante la temporada de rebajas.

$$\text{Ventas_R} = \sum_{\substack{(m,g) \in \mathcal{M} \times \mathcal{G} \\ n \in \mathcal{N}(g) \\ 0 \leq w \leq W_r, k \geq 1}} p_{mgk} \times d_{mgnk}^w \quad (3.3)$$

- **Ventas_S**: Representa los ingresos generados por la venta del exceso de inventario en cada país y centros de distribución al precio de liquidación.

$$Ventas_S = \sum_{g \in \mathcal{G}} \left(\sum_{m \in \mathcal{M}} Lsaldero_{mg} + \sum_{\substack{a \in \mathcal{A} \\ n \in \mathcal{N}(g)}} \tilde{I}_{agn} - \sum_{\substack{m \in \mathcal{M} \\ n \in \mathcal{N}(g)}} q_{mgn} \right) \times psaldero_g \quad (3.4)$$

- **Costes_CE**: Esta variable es el coste de envío total desde el centro de distribución a al país m .

$$Costes_CE = C_E \cdot \sum_{\substack{m \in \mathcal{M} \\ g \in \mathcal{G} \\ n \in \mathcal{N}(g)}} (\tilde{P}_{mgn} \times q_{mgn}) \quad (3.5)$$

- **Costes_CDE**: Representa el coste de envío total desde el centro de distribución a al centro de distribución a' .

$$Costes_CDE = C_A \cdot \sum_{\substack{a \neq a' \in \mathcal{A} \\ g \in \mathcal{G} \\ n \in \mathcal{N}(g)}} (\tilde{P}_{m(a)gn} \times q_{aa'gn}) \quad (3.6)$$

3.2.7. Restricciones del Modelo I

Las restricciones siguientes son restricciones de movimiento de inventario para los centros de distribución y los países:

- **Restricción de balance de inventario desde los CD:**

La primera restricción garantiza que los transportes efectuados desde los centros de distribución no exceden el inventario disponible. El inventario disponible en un centro de distribución, que viene dado en la primera parte de la inecuación, se compone de las unidades de inventario en el período actual más las unidades de inventario que llegan desde otros centros de distribución. Esta cantidad debe ser siempre mayor que las unidades que salen de ese centro de distribución a todos los países que abastece y a otros centros de distribución.

$$\tilde{I}_{agn} + \sum_{a' \in \mathcal{A} \setminus \{a\}} q_{a'agn} \geq \sum_{m \in \mathcal{M}(a)} q_{mgn} + \sum_{a' \in \mathcal{A} \setminus \{a\}} q_{aa'gn} \quad \forall (a, g, n) \in \mathcal{A} \times \mathcal{G} \times \mathcal{N} \quad (3.7)$$

- **Restricción de balance de inventario en el país:**

En la restricción (3.8) se asegura un equilibrio del flujo de inventario para cada país. El inventario disponible en el primer período de la temporada de rebajas en un país para un clúster determinado, se compone de las unidades de inventario disponibles en el período actual y de las unidades de inventario que llegan desde el centro de distribución menos las ventas previstas durante la temporada regular.

$$I_{mgn}^0 = \tilde{I}_{mgn} + q_{mgn} - \tilde{d}_{mgn} \quad \forall (m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \quad (3.8)$$

- **Restricción del flujo de inventario en rebajas:**

La tercera restricción modela los niveles iniciales de inventario de cada período de rebajas descontando las ventas del período anterior.

$$I_{mgn}^{w+1} = I_{mgn}^w - \sum_{k \geq 1} d_{mgnk}^w \quad \forall (m, g, n, w) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \times \mathcal{W} \quad (3.9)$$

- **Restricción de las ventas esperadas en la temporada regular:**

La estimación de ventas esperadas en la temporada regular por el Modelo de Optimización I está limitada por las ventas estimadas por el Modelo de Predicción.

$$\tilde{d}_{mgn} \leq \tilde{D}_{mgn} \quad \forall (m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \quad (3.10)$$

- **Restricción de las ventas esperadas en la temporada de rebajas**

De igual forma que se modela en la restricción (3.10), la estimación de ventas esperadas en la temporada de rebajas por el Modelo de Optimización I, no debe superar las ventas estimadas por el Modelo de Predicción en el primer período de rebajas, que se multiplican por un factor de tiempo, cuyo fin es modelar la disminución de la cantidad de ventas en rebajas para cada período. El objetivo de la variable y_{mgnk}^w es convertir a cero la demanda de un período en el cual un precio de rebajas no estará en vigencia.

$$d_{mgnk}^w \leq D_{mgnk}^0 \cdot y_{mgnk}^w \cdot tiempo^w \quad \forall (m, g, n, k, w) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \times \mathcal{K} \times \mathcal{W} \quad (3.11)$$

- **Restricción de envío máximo:**

El flujo de envío no debe superar la expedición máxima a un país. La restricción máxima de envío se utiliza para asegurar que las mercancías se asignen a varios países. Este cálculo sirve como un factor esencial para impedir que los países reciban un envío para un determinado grupo o clúster.

$$q_{mgn} \leq B_{mgn} \quad \forall (m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \quad (3.12)$$

- **Restricción de envío mínimo:**

El flujo de inventario debe ser mayor que la cantidad mínima de transporte a un país. Cualquier artículo en los centros de distribución que tenga un nivel de *stock* inferior a este envío mínimo, no se enviaría a los países y, en última instancia, se vendería a precio de liquidación. El requisito mínimo de envío está determinado por la política del centro de distribución.

$$\sum_{g \in \mathcal{G}, n \in \mathcal{N}(g)} q_{mgn} \geq Q_m \quad \forall m \in \mathcal{M} \quad (3.13)$$

■ **Restricción de ingresos de liquidación:**

La restricción (3.14) indica que los ingresos generados por la venta del exceso de inventario, no deben superar las ganancias que se obtienen si se vende un porcentaje fijado del total de las existencias en inventario y en tienda en la temporada actual.

$$\sum_{\substack{a \in \mathcal{A} \\ g \in \mathcal{G} \\ n \in \mathcal{N}(g)}} \tilde{I}_{agn} - \sum_{\substack{m \in \mathcal{M} \\ g \in \mathcal{G} \\ n \in \mathcal{N}(g)}} q_{mgn} + \sum_{\substack{m \in \mathcal{M} \\ g \in \mathcal{G} \\ n \in \mathcal{N}(g)}} I_{mgn}^{W_r} \leq \left(\sum_{\substack{a \in \mathcal{A} \\ g \in \mathcal{G} \\ n \in \mathcal{N}(g)}} \tilde{I}_{agn} + \sum_{\substack{m \in \mathcal{M} \\ g \in \mathcal{G} \\ n \in \mathcal{N}(g)}} \tilde{I}_{mgn} \right) \cdot S_T \quad (3.14)$$

■ **Restricción de liquidación de países:**

El inventario total que se recuperará para cada país está restringido por la cantidad de inventario que se deja sin vender al final de la temporada de rebajas, como se indica en la restricción (3.15).

$$L_{saldero}_{mg} \leq \sum_{n \in \mathcal{N}(g)} I_{mgn}^{W_r} \quad \forall (m, g) \in \mathcal{M} \times \mathcal{G} \quad (3.15)$$

■ **Restricción de asignación de precios en los clústers:**

La restricción (3.16) asegura que el orden inicial de cada clúster de precios se mantenga durante todo el período de rebajas. Por ejemplo, si el precio de una referencia r fue superior durante la temporada regular que el de una referencia r' , nunca será inferior al de la referencia r' durante la temporada de rebajas.

$$x_{mgnk}^w \leq x_{mgn+1k}^w \quad \forall (m, g, n, k, w) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \times \mathcal{K} \times \mathcal{W} \quad (3.16)$$

■ **Restricción de precios:**

Con el fin de seguir la política de fijación de precios de liquidación de Zara, la restricción (3.17) garantiza que el precio de un artículo nunca podrá aumentar a medida que avance el período de rebajas. Esto significa que el precio de liquidación para el período $w + 1$ sólo puede ser igual o inferior al precio de liquidación en el período w .

$$x_{mgnk}^w \leq x_{mgnk}^{w+1} \quad \forall (m, g, n, k, w) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \times \mathcal{K} \times \mathcal{W} \quad (3.17)$$

■ **Restricción de descuentos de grupos:**

La restricción (3.18) impone que el precio que estará en vigencia en la primera etapa del período de rebajas será el mayor precio que verifica la condición del descuento mínimo.

$$x_{mgnk}^0 = 1 \quad \forall (m, g, n, \max(k)) \ni p_{mgk} < \tilde{P}_{mgn} \cdot (1 - \text{desc_min}) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \times \mathcal{K} \quad (3.18)$$

■ **Restricción de menor precio:**

La restricción (3.19) asigna a cero a la variable x_{mgnk} asociada a un precio menor que el que está vigente en un período de rebajas determinado.

$$x_{mgnk-1}^w \leq x_{mgnk}^w \quad \forall (m, g, n, k, w) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \times \mathcal{K} \times \mathcal{W} \quad (3.19)$$

■ **Restricción de descuentos de la primera semana:**

Esta restricción impone que solo estará en vigencia un único precio para cada período de rebajas. Junto a la restricción (3.18) garantizan que se descontarán todos los clústers durante la primera semana de la temporada de rebajas.

$$\sum_{k \in \mathcal{K}} y_{mgnk}^0 = 1 \quad \forall (m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \quad (3.20)$$

■ **Restricción estructural:**

En la restricción (3.21) se define la variable binaria y_{mgnk}^w . Como ya se ha visto en la Sección 4.1.3, asume el valor de 1, si el precio p_{mgk} se asigna al clúster n , en el grupo g , en el país m durante el período de rebajas w .

$$y_{mgnk}^w = x_{mgnk}^w - x_{mgnk-1}^w \quad \forall (m, g, n, k, w) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \times \mathcal{K} \times \mathcal{W} \quad (3.21)$$

■ **Dominio de las variables:**

En la restricción (3.22) se especifica que las variables y_{mgnk}^w y x_{mgnk}^w son binarias.

$$y_{mgnk}^w, x_{mgnk}^w \in \{0, 1\} \quad \forall (m, g, n, k, w) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \times \mathcal{K} \times \mathcal{W} \quad (3.22)$$

La Ecuación (3.23) impone que las ventas esperadas y niveles de existencias tengan valores positivos.

$$d_{mgnk}^w, I_{mgnk}^w \geq 0 \quad \forall (m, g, n, k, w) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \times \mathcal{K} \times \mathcal{W} \quad (3.23)$$

La Ecuación (3.24) impone que todos los envíos y niveles de demanda tengan valores positivos.

$$q_{aa'gn}, q_{mgn}, \tilde{d}_{mgn} \geq 0 \quad \forall (a, a', m, g, n) \in \mathcal{A} \times \mathcal{A} \times \mathcal{M} \times \mathcal{G} \times \mathcal{N}, \quad a \neq a' \quad (3.24)$$

A modo de resumen se tabulan los valores de los parámetros y las variables del Modelo I en la Tabla 3.1 y Tabla 3.2 respectivamente.

Parámetro	Descripción
\tilde{P}_{mgn}	Precio de temporada regular para un clúster en un país y grupo.
$psaldero_g$	Precio de rebajas para cada grupo al final de la temporada de rebajas.
\tilde{I}_{agn}	Inventario disponible al principio del período actual en un CD, grupo y clúster.
\tilde{I}_{mgn}	Inventario disponible al principio del período actual en un país, grupo y clúster.
D_{mgnk}^w	Demanda estimada para un período en rebajas en un país, grupo y clúster para un precio.
\tilde{D}_{mgn}	Ventas estimadas al final de la temporada regular en un país, grupo y clúster.
B_{mgn}	Expedición máxima de un grupo y clúster para un país.
C_E	Porcentaje del precio de un artículo que hace referencia a los gastos de envío por unidad de los CD a un país.
C_A	Porcentaje del precio de un artículo que hace referencia a los gastos de transporte por unidad de un CD a otro.
D	Número de días en un período de rebajas.
W_r	Número de períodos en la temporada de rebajas.
S_A	Porcentaje de existencias en los CD que se puede salvar.
S_T	Porcentaje del total de existencias mundiales que pueden ser recuperadas.
Q_m	Cantidad mínima de transporte en unidades por país.
$desc_min$	Descuento mínimo para la primera semana de venta.
$tiempo$	Factor tiempo que disminuye la cantidad de ventas de liquidación del país cada semana.

Tabla 3.1: Parámetros del modelo I.

Variable	Descripción
q_{mgn}	Flujo de inventario desde los CD a un país, grupo y clúster.
I_{mgn}^w	Inventario disponible para un período en rebajas en un país grupo y clúster.
\tilde{d}_{mgn}	Ventas previstas en la temporada regular para un país, grupo y clúster.
d_{mgnk}^w	Ventas previstas para un período de rebajas en un país, grupo y clúster para un precio.
$q_{aa'gn}$	Flujo de inventario desde un CD a otro para un grupo y clúster.
x_{mgnk}^w	Variable binaria que indica si en un país, a un clúster se le asigna un precio menor o igual que p_{mgk} en un período de rebajas.
y_{mgnk}^w	Variable binaria que indica si el precio p_{mgk} se asigna a un país y clúster en un período de rebajas.
$Lsaldero_{mg}$	Inventario total que se recuperará en un país y grupo.

Tabla 3.2: Variables del modelo I.

3.2.8. Desarrollo del Modelo I

En esta subsección se presenta una versión conjunta del primer modelo, visto con detalle en la Sección 3.2, con el fin de tener una visión más general de este. Puede verse el código en Python del Modelo I en el Apéndice C.

$$\max \quad \mathbf{Ventas_FdT} + \mathbf{Ventas_R} + \mathbf{Ventas_S} \\ - \mathbf{Costes_CE} - \mathbf{Costes_CDE}$$

con:

$$\mathbf{Ventas_FdT} = \sum_{\substack{(m,g) \in \mathcal{M} \times \mathcal{G} \\ n \in \mathcal{N}(g)}} \tilde{P}_{mgn} \times \tilde{d}_{mgn}$$

$$\mathbf{Ventas_R} = \sum_{\substack{(m,g) \in \mathcal{M} \times \mathcal{G} \\ n \in \mathcal{N}(g) \\ 0 \leq w \leq W_r, k \geq 1}} p_{mgk} \times d_{mgnk}^w$$

$$\begin{aligned}
Ventas_S &= \sum_{g \in \mathcal{G}} \left(\sum_{m \in \mathcal{M}} Lsaldero_{mg} + \sum_{\substack{a \in \mathcal{A} \\ n \in \mathcal{N}(g)}} \tilde{I}_{agn} \right. \\
&\quad \left. - \sum_{\substack{m \in \mathcal{M} \\ n \in \mathcal{N}(g)}} \mathbf{q}_{mgn} \right) \times psaldero_g \\
Costes_CE &= C_E \cdot \sum_{\substack{m \in \mathcal{M} \\ g \in \mathcal{G} \\ n \in \mathcal{N}(g)}} (\tilde{P}_{mgn} \times \mathbf{q}_{mgn}) \\
Costes_CDE &= C_A \cdot \sum_{\substack{a \neq a' \in \mathcal{A} \\ g \in \mathcal{G} \\ n \in \mathcal{N}(g)}} (\tilde{P}_{m(a)gn} \times \mathbf{q}_{aa'gn})
\end{aligned}$$

s. a.

$$\begin{aligned}
\tilde{I}_{agn} + \sum_{a' \in \mathcal{A} \setminus \{a\}} \mathbf{q}_{a'agn} &\geq & \forall (a, g, n) \in \mathcal{A} \times \mathcal{G} \times \mathcal{N} \\
&\geq \sum_{m \in \mathcal{M}(a)} \mathbf{q}_{mgn} + \sum_{a' \in \mathcal{A} \setminus \{a\}} \mathbf{q}_{aa'gn} \\
\mathbf{I}_{mgn}^0 &= \tilde{I}_{mgn} + \mathbf{q}_{mgn} - \tilde{\mathbf{d}}_{mgn} & \forall (m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \\
\mathbf{I}_{mgn}^{w+1} &= \mathbf{I}_{mgn}^w - \sum_{k \geq 1} \mathbf{d}_{mgnk}^w & \forall (m, g, n, w) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \times \mathcal{W} \\
\tilde{\mathbf{d}}_{mgn} &\leq \tilde{\mathbf{D}}_{mgn} & \forall (m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \\
\mathbf{d}_{mgnk}^w &\leq \mathbf{D}_{mgnk}^0 \cdot \mathbf{y}_{mgnk}^w \cdot tiempo^w & \forall (m, g, n, k, w) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \times \mathcal{K} \times \mathcal{W} \\
\mathbf{q}_{mgn} &\leq B_{mgn} & \forall (m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \\
\sum_{g \in \mathcal{G}, n \in \mathcal{N}(g)} \mathbf{q}_{mgn} &\geq Q_m & \forall m \in \mathcal{M} \\
\sum_{\substack{a \in \mathcal{A} \\ g \in \mathcal{G} \\ n \in \mathcal{N}(g)}} \tilde{I}_{agn} - \sum_{\substack{m \in \mathcal{M} \\ g \in \mathcal{G} \\ n \in \mathcal{N}(g)}} \mathbf{q}_{mgn} + \sum_{\substack{m \in \mathcal{M} \\ g \in \mathcal{G} \\ n \in \mathcal{N}(g)}} \mathbf{I}_{mgn}^{W_r} &\leq \\
&\leq \left(\sum_{\substack{a \in \mathcal{A} \\ g \in \mathcal{G} \\ n \in \mathcal{N}(g)}} \tilde{I}_{agn} + \sum_{\substack{m \in \mathcal{M} \\ g \in \mathcal{G} \\ n \in \mathcal{N}(g)}} \tilde{I}_{mgn} \right) \cdot S_T \\
Lsaldero_{mg} &\leq \sum_{n \in \mathcal{N}(g)} \mathbf{I}_{mgn}^{W_r} & \forall (m, g) \in \mathcal{M} \times \mathcal{G} \\
\mathbf{x}_{mgnk}^w &\leq \mathbf{x}_{mgn+1k}^w & \forall (m, g, n, k, w) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \times \mathcal{K} \times \mathcal{W} \\
\mathbf{x}_{mgnk}^w &\leq \mathbf{x}_{mgnk}^{w+1} & \forall (m, g, n, k, w) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \times \mathcal{K} \times \mathcal{W} \\
\mathbf{x}_{mgnk}^0 &= 1 & \forall (m, g, n, \max(k)) \ni p_{mgk} < \tilde{P}_{mgn} \cdot (1 - desc_min)) \\
&& \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \times \mathcal{K} \\
\mathbf{x}_{mgnk-1}^w &\leq \mathbf{x}_{mgnk}^w & \forall (m, g, n, k, w) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \times \mathcal{K} \times \mathcal{W} \\
\sum_{k \in \mathcal{K}} \mathbf{y}_{mgnk}^0 &= 1 & \forall (m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \\
\mathbf{y}_{mgnk}^w &= \mathbf{x}_{mgnk}^w - \mathbf{x}_{mgnk-1}^w & \forall (m, g, n, k, w) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \times \mathcal{K} \times \mathcal{W} \\
\mathbf{y}_{mgnk}^w, \mathbf{x}_{mgnk}^w &\in \{0, 1\} & \forall (m, g, n, k, w) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \times \mathcal{K} \times \mathcal{W} \\
\mathbf{d}_{mgnk}^w, \mathbf{I}_{mgnk}^w &\geq 0 & \forall (m, g, n, k, w) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \times \mathcal{K} \times \mathcal{W} \\
\mathbf{q}_{aa'gn}, \mathbf{q}_{mgn}, \tilde{\mathbf{d}}_{mgn} &\geq 0 & \forall (a, a', m, g, n) \in \mathcal{A} \times \mathcal{A} \times \mathcal{M} \times \mathcal{G} \times \mathcal{N}, \quad a \neq a'
\end{aligned}$$

3.3. Observación. Diferencias código Python y AMPL

El código en lenguaje AMPL disponible en la tesis *Clearance pricing optimization at Zara* presenta ciertas diferencias con el modelo descrito en la Subsección 3.2.8.

3.3.1. Variación en la Función Objetivo

La función objetivo varía debido a que la variable **Ventas_R** dada en la Ecuación (3.25), se modifica ligeramente, multiplicando su totalidad por un factor de tiempo que disminuye las ventas conforme avanza el período de rebajas. De esta forma se obtiene la nueva variable:

$$\mathbf{Ventas_R} = \sum_{\substack{(m,g) \in \mathcal{M} \times \mathcal{G} \\ n \in \mathcal{N}(g) \\ 0 \leq w \leq W_r, k \geq 1}} p_{mgk} \times \mathbf{d}_{mgnk}^w \times \text{tiempo}^w \quad (3.25)$$

3.3.2. Variación en las Restricciones

La restricción (3.12), que hace referencia al envío máximo, no se incluye en el código en AMPL. En lugar de ella se incluyen las dos restricciones que se definen en las siguientes líneas.

- **Restricción de distribución del inventario:**

Se incluye una variante de la restricción (3.12), en la que el lado derecho de la desigualdad se multiplica por el inventario total en los CD para el clúster n . Esta nueva restricción obliga a que el inventario se distribuya proporcionalmente a cada país basándose en el índice de éxito.

$$\mathbf{q}_{mgn} \leq \sum_{a \in \mathcal{A}} \tilde{I}_{agn} \cdot B_{mgn} \quad \forall (m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \quad (3.26)$$

- **Restricción de exceso máximo de inventario salvado:**

Otra restricción que se considera es la imposición de una cantidad máxima de exceso de inventario que se puede salvar en un centro de distribución.

$$\sum_{\substack{a \in \mathcal{A} \\ g \in \mathcal{G} \\ n \in \mathcal{N}(g)}} \tilde{I}_{agn} - \sum_{\substack{m \in \mathcal{M}(a) \\ g \in \mathcal{G} \\ n \in \mathcal{N}(g)}} \mathbf{q}_{mgn} \leq \sum_{\substack{a \in \mathcal{A} \\ g \in \mathcal{G} \\ n \in \mathcal{N}(g)}} \tilde{I}_{agn} \cdot S_A \quad (3.27)$$

3.3.3. Ejemplo básico

En esta subsección se formula un ejemplo básico, con datos de entrada no reales, para ayudar a entender el funcionamiento del modelo y además comprobar que la salida que nos proporciona es lógica.

La empresa Zara dispone de dos centros de distribución, abasteciendo cada uno de ellos a un país. Con el único fin de simplificar el modelo y facilitar su comprensión se trabaja con un solo grupo y clúster. Los conjuntos e índices del Modelo I se reducen a los siguientes:

- $\mathcal{M} = \{m_1, m_2\}$
- $\mathcal{A} = \{a_1, a_2\}$
- $\mathcal{K} = \{1, 2\}$
- $\mathcal{W} = \{0, 1\}$

Se dispone de 6 unidades de cierto artículo en el inventario del primer centro de distribución y de 2 unidades en el primer país. En el segundo país así como en el segundo centro de distribución no se dispone de ninguna unidad de este artículo. Solamente se puede realizar una venta en cada etapa, esto es, la demanda en la temporada regular y cada una de las etapas de rebajas estará acotada por 1 unidad. El primer paso es calcular la expedición máxima para cada país, B_m . Para ello seguimos los pasos que se han detallado en la Subsección 3.3. Debido a que las Ventas en un país son similares a la demanda, para el cálculo de B_m se sustituye el parámetro $Ventas_m$ por la demanda en cada país D_m .

Algoritmo 2 : Cálculo de B_m

Entrada: I_m, I_a, D_m

1: **for** $m \in \mathcal{M}$ **do**

2: $IE_{m_1} = \frac{1}{3}$ e $IE_{m_2} = 1$.

3: **end for**

4: **for** $m \in \mathcal{M}$ **do**

5: $\mu_m = \frac{2}{3}$ y $\sigma_m = \frac{1}{3}$

6: $IE_m > \mu_m - 1,5\sigma_m \forall m \in \mathcal{M}$

7: $B_{m_1} = 3$ uds y $B_{m_2} = 3$ uds.

8: Se define $X_{m_1} = 1$ y $X_{m_2} = 1$

9: $B_m > X_m \forall m \in \mathcal{M}$

10: **end for**

La situación que se describe se recoge en la Figura 3.8. Además detalla el precio para la temporada regular y los posibles precios para la temporada de rebajas.

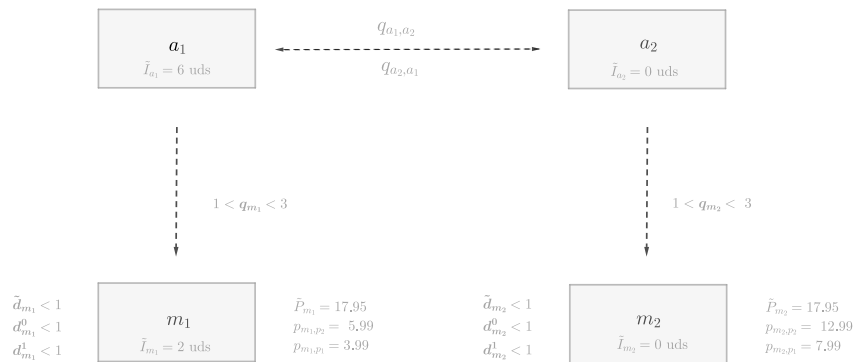


Figura 3.8: Esquema del ejemplo.

Los parámetros de entrada se resumen en la Tabla 3.3 y la Tabla 3.4, y las variables de salida que nos proporciona el modelo una vez ejecutado se detallan en la Tabla 3.5 y la Tabla 3.6.

Parámetro	Valor
\tilde{P}_{m_1}	17.95 euros
\tilde{P}_{m_2}	17.95 euros
\tilde{I}_{m_1}	2 uds
\tilde{I}_{m_2}	0 uds
\tilde{D}_{m_1}	1 ud
\tilde{D}_{m_2}	1 ud
\tilde{B}_{m_1}	3 uds
\tilde{B}_{m_2}	3 uds
\tilde{I}_{a_1}	6 uds
\tilde{I}_{a_2}	0 uds
$D_{m_1 1}^0$	1 ud
$D_{m_1 2}^0$	1 ud
$D_{m_2 1}^0$	1 ud
$D_{m_2 2}^0$	1 ud
$p_{m_1 1}$	3.99 euros
$p_{m_1 2}$	5.99 euros
$p_{m_2 1}$	7.99 euros
$p_{m_2 2}$	12.99 euros

Tabla 3.3: Parámetros modelo I.

Constante	Valor
C_E	0.02
C_A	0.01
D	7 días
W_r	2
S_A	1
S_T	1
Q	1
$desc_min$	0.1
$tiempo$	1
$psaldero$	1 euro

Tabla 3.4: Constantes modelo I.

Variable	Valor
q_{m_1}	1 uds
q_{m_2}	3 uds
$I_{m_1}^0$	2 uds
$I_{m_2}^0$	2 uds
$I_{m_1}^1$	1 uds
$I_{m_2}^1$	1 uds
$I_{m_1}^2$	0 uds
$I_{m_2}^2$	0 uds
\tilde{d}_{m_1}	1 uds
\tilde{d}_{m_2}	1 uds
$d_{m_1 1}^0$	0 uds
$d_{m_2 1}^0$	0 uds
$d_{m_1 2}^0$	1 uds
$d_{m_2 2}^0$	1 uds
$d_{m_1 1}^1$	0 uds
$d_{m_2 1}^1$	0 uds
$d_{m_1 2}^1$	1 uds
$d_{m_2 2}^1$	1 uds
$q_{a_1 a_2}$	3 uds
$q_{a_2 a_1}$	0 uds
$Lsaldero_{m_1}$	0 uds
$Lsaldero_{m_2}$	0 uds

Tabla 3.5: Variables continuas modelo I.

Variable	Valor
$x_{m_1 1}^0$	0
$x_{m_1 1}^1$	0
$x_{m_1 2}^1$	1
$x_{m_2 1}^0$	0
$x_{m_2 2}^0$	1
$x_{m_2 1}^1$	0
$x_{m_2 2}^1$	1
$y_{m_1 1}^0$	0
$y_{m_1 2}^0$	1
$y_{m_1 1}^1$	0
$y_{m_1 2}^1$	1
$y_{m_2 1}^0$	0
$y_{m_2 2}^0$	1
$y_{m_2 1}^1$	0
$y_{m_2 2}^1$	1

Tabla 3.6: Variables binarias modelo I.

Sin necesidad de ejecutar el modelo ya se intuye que la distribución de inventario que maximiza los ingresos consiste en trasladar tres unidades del centro de distribución a_1 hasta a_2 . De esta forma se en-

viaría una de las tres unidades restantes en el primer centro de distribución (una vez realizado el traslado al segundo centro de distribución) al país m_1 y tres unidades del centro de distribución a_2 al país m_2 . Las 2 unidades restantes no se moverían del centro de distribución a_1 . Debido a que las existencias del artículo en cuestión son más que las que se pueden vender, el precio del artículo en cada país no influye en cuanto a las unidades que se envían a cada uno de los países.

Las variables binarias del modelo nos proporcionan el esquema general de precios de rebajas que maximiza los ingresos. Al igual que se hizo en la Sección 3.2.5 se representan las matrices con elementos las variables x_{mk}^w e y_{mk}^w , que nos dará una visión más intuitiva del esquema de precios. Debido a que solamente se trabaja con un clúster, el número de matrices de elementos x_{mk}^w e y_{mk}^w será de una para cada país.

$$\begin{pmatrix} x_{m_1 1}^0 & x_{m_1 1}^1 \\ x_{m_1 2}^0 & x_{m_1 2}^1 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix}$$

Figura 3.9: Matriz de x_{mk}^w en el país m_1 .

$$\begin{pmatrix} x_{m_2 1}^0 & x_{m_2 1}^1 \\ x_{m_2 2}^0 & x_{m_2 2}^1 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix}$$

Figura 3.10: Matriz de x_{mk}^w en el país m_2 .

$$\begin{pmatrix} y_{m_1 1}^0 & y_{m_1 1}^1 \\ y_{m_1 2}^0 & y_{m_1 2}^1 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix}$$

Figura 3.11: Matriz de y_{mk}^w en el país m_1 .

$$\begin{pmatrix} y_{m_2 1}^0 & y_{m_2 1}^1 \\ y_{m_2 2}^0 & y_{m_2 2}^1 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix}$$

Figura 3.12: Matriz de y_{mk}^w en el país m_2 .

Como ya se intuía, la variable y_{mk}^w nos muestra que lo más beneficioso para maximizar los ingresos, en ambos países, es fijar el único grupo de precios a p_2 en la primera etapa de rebajas y mantenerlo toda la temporada de rebajas, no siendo necesario descontarlo a p_1 . En la Figura 3.13 y la Figura 3.14 se presenta un esquema con los envíos y los precios vigentes en cada etapa de rebajas para cada país de la red.

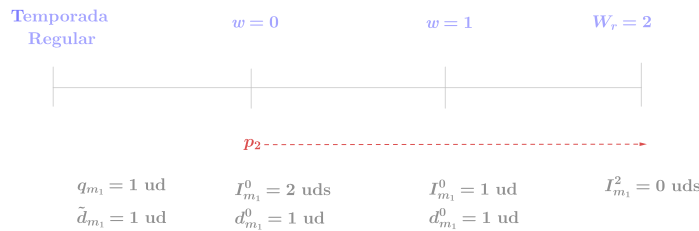
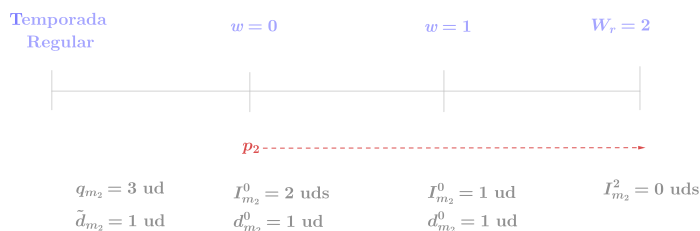


Figura 3.13: Esquema del país m_1 .

Figura 3.14: Esquema del país m_2 .

En último lugar, se desagregan los términos de la función objetivo, es decir, se obtiene el valor de cada uno de los tres tipos de ventas (regular, rebajas y liquidación), así como los dos tipos de costes de envío (de centro de distribución a país o entre centros de distribución), vistos en la Ecuación (3.1). Estos valores se recogen en la Tabla 3.7. Finalmente se obtiene un valor para la función objetivo de 73.8 euros de beneficio total.

Variable	Valor
<i>Ventas_FdT</i>	35.9 euros
<i>Ventas_R</i>	37.96 euros
<i>Ventas_S</i>	2 euros
<i>Costes_CE</i>	1.436 euros
<i>Costes_CDE</i>	0.5385 euros

Tabla 3.7: Parámetros del Modelo I.

3.4. Modelo II: Formulación para Minimizar Costes

El propósito del Modelo II es distribuir la mercancía a cada tienda minimizando los costes de envío. Con el resultado del Modelo I, que dicta la cantidad total a enviar a cada país para un nivel de grupo y clúster de precios, este modelo desagrega los transportes, a nivel de artículo-tienda.

En este modelo, los índices (g, n) , referentes al grupo de rebajas y clúster de precios serán fijos. Además, el modelo de optimización se centra en un grupo y clúster a la vez debido al exceso de datos, ya que el número de referencias en un grupo y clúster puede variar de 1 a 100 y se distribuyen en más de 1200 tiendas en 73 países.

3.4.1. Índices y Conjuntos de Índices

A continuación se introduce notación adicional para el Modelo II, además de la ya empleada en el primer modelo.

- $r \in \mathcal{R}(n)$: Referencias en un clúster de precios determinado n . Una referencia se clasifica por su precio original de temporada regular y se agrupa en clúster de precios n .
- $j \in \mathcal{J}$: Tiendas dentro de la red de distribución. Donde $m(j)$ denota el país de la tienda j y $\mathcal{T}(m) \subseteq \mathcal{J}$ denota el conjunto de tiendas del país m . A 11 de agosto de 2017, Zara tenía 2232 tiendas en el mundo, distribuidas en 88 mercados en los cinco continentes. Por número de tiendas, el primer país es España, donde se encuentran 449 tiendas.

- $l \in \mathcal{L}$: Plataformas logísticas ² para cada tienda. Las tiendas son capaces de enviar y recibir *merchandising* en liquidación de tiendas dentro de su país o en una región del país. Por lo tanto, $l(j)$ denota la plataforma logística para una tienda determinada y $\mathcal{T}(l) \subseteq \mathcal{J}$ denota el conjunto de tiendas asociadas a la plataforma l . En la Figura se pueden ver las plataformas logísticas de las que dispone Inditex, ubicadas todas ellas en España.



Figura 3.15: Plataformas logísticas de Inditex.

3.4.2. Parámetros

Los siguientes parámetros de inventario inicial son similares a los del Modelo I, sin embargo, estos se encuentran a un nivel referencia y no de grupo y clúster.

- \tilde{I}_{ra} : Inventario en unidades al principio del período actual para la referencia r en el centro de distribución a .
- \tilde{I}_{rj} : Inventario en unidades al principio del período actual para la referencia r en la tienda j .
- \tilde{I}_{rm} : Inventario en unidades al principio del período actual para la referencia r en el país m .

El primer modelo de optimización proporciona una salida del envío total a un país por grupo y clúster, en la variable de decisión q_{mgn} . En este modelo es importante determinar primero cómo distribuir esta cantidad a cada tienda. El inventario total en el país más el transporte entrante del centro de distribución se asigna a cada tienda en función del porcentaje de ventas de cada tienda en su respectivo país, que viene denotado por v_{jgn} . El inventario actual de la tienda también se tiene en cuenta como se muestra en la Ecuación (3.28).

- λ_j : Cantidad total en unidades a transportar a cada tienda j .

$$\lambda_j = (\tilde{I}_{m(j)gn} + q_{m(j)gn}) \cdot v_{jgn} - \sum_{r \in \mathcal{R}} \tilde{I}_{rj} \quad \text{donde, } v_{jgn} = \frac{Ventas_{jgn}}{\sum_{j' \in \mathcal{T}(m(j))} Ventas_{j'gn}} \quad (3.28)$$

Similar al cálculo del envío máximo B_{mgn} , en el Modelo I, hay una variable similar para el envío máximo por tienda y referencia. Este parámetro garantiza que las referencias estén bien asignadas a cada tienda y permite bloquear ciertas referencias a nivel de tienda.

- b_j : Cantidad máxima en unidades a transportar a cada tienda j .

²Una plataforma logística es una zona especializada que cuenta con la infraestructura y los servicios necesarios para facilitar las actividades relativas al transporte, empaque y distribución, para tránsito nacional y/o internacional de mercancías donde distintos agentes coordinan sus acciones en beneficio de la competitividad de los productos que hacen uso de la infraestructura.

- B_{rj} : Cantidad máxima de transporte en unidades para la tienda j de la referencia r . Hay seis pasos en el cálculo de este parámetro como se explica a continuación. El código en Python del Algoritmo 3 se encuentra disponible en el Apéndice D.

Algoritmo 3 : Cálculo de B_{rj}

Entrada: $q_{mgn}, \tilde{I}_{ra}, Ventas_{rj}, I_{rj}$

```

1: for  $(r, j) \in \mathcal{R} \times \mathcal{J}$  do
2:   Se define  $\mathcal{J}_r$  el conjunto de todas las tiendas para la referencia  $r$ .
3:   if  $q_{mgn} = 0$  para algún  $m \in \mathcal{M}$  then
4:      $B_{rj} = 0$  para todo  $j \in \mathcal{T}(m)$  y  $\mathcal{J}_r = \mathcal{J}_r - \{j\}$ .
5:   end if
6: end for

```

```

7: for  $(r, j) \in \mathcal{R} \times \mathcal{J}_r$  do
8:   Calcular  $IE_{rj}, \mu_{rj}$  y  $\sigma_{rj}$ 
9: end for
10: for  $(r, j) \in \mathcal{R} \times \mathcal{J}_r$  do
11:   if  $IE_{rj} < \mu_{rj} - 1,5\sigma_{rj}$  then
12:      $B_{rj} = 0$  y  $\mathcal{J}_r = \mathcal{J}_r - \{j\}$ 
13:   end if
14: end for

```

```

15: for  $r \in \mathcal{R}$  do
16:   Sea  $H = \frac{\sum_{a \in \mathcal{A}} \tilde{I}_{ra}}{X_r}$  el número máximo de tiendas que pueden recibir una referencia  $r$ .
   Sea  $|\mathcal{J}_r|$  el número de tiendas del conjunto  $\mathcal{J}_r$ .
   Sea  $X_r$  la cantidad mínima de envío que puede tener una referencia específica.
17:   if  $H < |\mathcal{J}_r|$  then
18:     Se retiran las  $|\mathcal{J}_r| - H$  tiendas de  $\mathcal{J}_r$  que tengan el índice de éxito más bajo  $IE_{rj}$  y se establece
      $B_{rj} = 0$ .
19:   end if
20: end for

```

```

21: for  $(r, j) \in \mathcal{R} \times \mathcal{J}_r$  do
22:   Sea  $[\lambda_j]^+ = \max \{\lambda_j, 0\} \forall j \in \mathcal{J}_r$ .
23:   Calcular  $B_{rj} = \frac{[\lambda_j]^+}{\sum_{j' \in \mathcal{J}_r} [\lambda_{j'}]^+} \cdot \sum_{a \in \mathcal{A}} \tilde{I}_{ra}$ .
24: end for

```

```

25: for  $(r, j) \in \mathcal{R} \times \mathcal{J}_r$  do
26:   if  $B_{rj} < X_r$  then
27:      $B_{rj} = 0$  y  $\mathcal{J}_r = \mathcal{J}_r - \{j\}$ 
28:   end if
29: end for

```

```

30: for  $(r, j) \in \mathcal{R} \times \mathcal{J}_r$  do
31:   Recalcular  $B_{rj}$ .
32:   Para evitar problemas de infactibilidad, se verifica que se cumpla:

```

$$\sum_{r \in \mathcal{R}(n)} B_{rj} \geq q_{m(j)gn} \cdot v_{jgn} \quad \forall j \in \mathcal{J}_r \quad (3.29)$$

```

33: end for

```

Los siguientes parámetros son constantes que se pueden ajustar en base a la política y los estándares del negocio. En este modelo son los costes asociados al envío y manipulación.

- C_T : Porcentaje del precio del artículo que representa el coste del transporte entre las plataformas logísticas y tiendas.

- C_L : Porcentaje del precio del artículo que representa el coste de transporte entre plataformas logísticas y plataformas logísticas.
- C_{AT} : Porcentaje del precio del artículo que representa el coste de transporte desde el almacén de un país a la tienda.
- R_S : Relación de holgura para un envío máximo por tienda o referencia.

3.4.3. Variables de Decisión

Todas las variables de decisión en este modelo son cantidades de mercancía a transportar de un punto a otro. A continuación se muestran los envíos del centro de distribución y los almacenes de país.

- q_{raj} : Unidades enviadas de la referencia r desde el centro de distribución a a la tienda j .
- q_{rmj} : Unidades enviadas para la referencia r desde el almacén del país m a la tienda j .
- $q_{raa'}$: Transbordo realizado en unidades para la referencia r entre el centro de distribución a y a' .

Durante la temporada de rebajas, las tiendas pueden recibir mercancías de otras tiendas dentro de su país. La mayoría de los países no tienen un almacén nacional que pueda facilitar dichas transferencias entre tiendas, por lo que hay plataformas logísticas en cada país donde se mandan los envíos y se envían posteriormente a otro sitio. Cada tienda tiene una sola plataforma logística y los envíos se pueden transferir entre una sola plataforma logística y otra. Las siguientes variables de decisión facilitan el almacenamiento utilizando estas plataformas logísticas.

- q_{rlj} : Unidades enviadas de la referencia r desde la plataforma logística l hasta la tienda j .
- q_{rjl} : Unidades enviadas de la referencia r desde la tienda j a la plataforma logística l .
- $q_{rl'l'}$: Transbordo realizado en unidades de la referencia r entre las plataformas logísticas l y l' .

3.4.4. Función Objetivo

La función objetivo consiste en minimizar los gastos de transporte al distribuir mercancías a cada tienda. Existen cinco elementos básicos en este modelo, cada uno de los cuales representa costes asociados con el movimiento de mercancías entre centros de distribución, tiendas y plataformas logísticas. Todos los costes se basan en costes unitarios por referencia. De esta forma, la función objetivo del Modelo II viene dada por:

$$\min \text{CDaCD} + \text{Tienda.PL} + \text{PLaPL} + \text{CD.Tienda} + \text{Almacen.Tienda} \quad (3.30)$$

En donde los elementos de la Ecuación (3.30) se definen del siguiente modo:

- CDaCD : Coste de envío de una referencia entre un centro de distribución y otro. Este coste incluye los costes de gestión tales como el reemplazamiento de etiquetas de precios en situaciones en que la mercancía es enviada a un país con un precio diferente. Es el mismo coste C_A que en el del Modelo I.

$$\text{CDaCD} = C_A \cdot \sum_{\substack{r \in \mathcal{R} \\ a, a' \in \mathcal{A}}} q_{raa'} \quad (3.31)$$

- **Tienda_PL**: Coste de envío para distribuir mercancías de una tienda a otra a través de una plataforma logística. El coste de envío incluye los costes de manipulación de materiales para preparar el envío en una tienda y recibir el envío en otra tienda.

$$\mathbf{Tienda_PL} = C_T \cdot \sum_{\substack{r \in \mathcal{R} \\ j \in \mathcal{J}}} \mathbf{q}_{rl(j)j} + \mathbf{q}_{rjl(j)} \quad (3.32)$$

- **PLaPL**: Coste de envío de un artículo desde una plataforma logística a otra. Un ejemplo de ello es que una tienda en Nueva York tiene un exceso de mercancía que se puede vender en una tienda en Miami que enviará los artículos a su plataforma logística noreste. Esa plataforma entonces enviaría el material a la plataforma logística del sur y permitiría que la mercancía se distribuyera a las tiendas que se encuentran en la región sur de los Estados Unidos.

$$\mathbf{PLaPL} = C_L \cdot \sum_{\substack{r \in \mathcal{R} \\ l, l' \in \mathcal{L}}} \mathbf{q}_{rl l'} \quad (3.33)$$

- **CD_Tienda**: Coste de envío desde los centros de distribución a las tiendas, siendo el coste C_E del Modelo I similar. El coste de envío incluye los costes de transporte y manipulación de materiales.

$$\mathbf{CD_Tienda} = C_E \cdot \sum_{\substack{r \in \mathcal{R} \\ j \in \mathcal{J}}} \mathbf{q}_{ra(j)j} \quad (3.34)$$

- **Almacen_Tienda**: Representa el coste de envío de mercancías desde el almacén del país local a una tienda. Hay sólo unos pocos países que tienen un gran número de tiendas que necesitan un almacén para soportar las ventas.

$$\mathbf{Almacen_Tienda} = C_{AT} \cdot \sum_{\substack{r \in \mathcal{R} \\ j \in \mathcal{J}}} \mathbf{q}_{rm(j)j} \quad (3.35)$$

3.4.5. Restricciones

En el Modelo II hay cinco restricciones principales que se basan principalmente en el flujo de las existencias y las restricciones en los envíos.

- **Restricción de balance de inventario desde los CD:**

La restricción (3.36) garantiza que los transportes desde los centros de distribución no excedan el inventario disponible. Es decir, la mercancía que sale de un centro de distribución, bien a una tienda o bien a otro centro de distribución, nunca debe superar las unidades que hay en dicho centro más las que llegan desde otros centros.

$$\tilde{I}_{ra} + \sum_{a' \in \mathcal{A} \setminus \{a\}} \mathbf{q}_{ra'a} \geq \sum_{j \in \mathcal{T}(a)} \mathbf{q}_{raj} + \sum_{a' \in \mathcal{A} \setminus \{a\}} \mathbf{q}_{raa'} \quad \forall (r, a) \in \mathcal{R}(n) \times \mathcal{A} \quad (3.36)$$

- **Restricción de flujo de inventario desde las plataformas logísticas:**

La restricción (3.37) garantiza que las unidades que salen de una plataforma logística, bien a otras plataformas o a las tiendas, deben ser las mismas que las que entran en dicha plataforma. De esto se deriva que toda la mercancía fluye a través de una plataforma logística y no se almacena en ella.

$$\sum_{l' \in \mathcal{L}} \mathbf{q}_{rl'l} + \sum_{j \in \mathcal{T}(l)} \mathbf{q}_{rjl} = \sum_{l' \in \mathcal{L}} \mathbf{q}_{rl'l'} + \sum_{j \in \mathcal{T}(l)} \mathbf{q}_{rlj} \quad \forall (r, l) \in \mathcal{R}(n) \times \mathcal{L} \quad (3.37)$$

- **Restricción de balance de inventario en almacén:**

La Restricción (3.38) es una ecuación de balance de inventario que asegura que los envíos desde los almacenes de los países no excedan el inventario disponible.

$$\tilde{I}_{rm} \geq \sum_{j \in \mathcal{T}(m)} \mathbf{q}_{rmj} \quad \forall (r, m) \in \mathcal{R}(n) \times \mathcal{M} \quad (3.38)$$

- **Restricción de balance de inventario en tienda:**

La restricción (3.39) garantiza que el inventario disponible en una tienda más las unidades de inventario que entran sea, como mínimo, la cantidad total de transporte asignada a esa tienda, que se compone de las unidades que envía a la plataforma logística y la cantidad de unidades a transportar a cada una de las otras tiendas.

$$\sum_{r \in \mathcal{R}(n)} I_{rj} + \mathbf{q}_{ra(j)j} + \mathbf{q}_{rm(j)j} + \mathbf{q}_{rl(j)j} \geq \lambda_j + \sum_{r \in \mathcal{R}(n)} \mathbf{q}_{rjl(j)} \quad \forall j \in \mathcal{J} \quad (3.39)$$

- **Restricción de cantidad máxima de envío:**

La restricción (3.40) garantiza que los envíos de una referencia se distribuyan a varias tiendas según el requisito máximo de envío.

$$\mathbf{q}_{ra(j)j} + \mathbf{q}_{rm(j)j} + \mathbf{q}_{rl(j)j} \leq B_{rj} \quad \forall (r, j) \in \mathcal{R}(n) \times \mathcal{J} \quad (3.40)$$

- **Restricción de no negatividad:**

La restricción (3.41) requiere que todos los envíos sean positivos.

$$\mathbf{q}_{raa'}, \mathbf{q}_{raj}, \mathbf{q}_{rmj}, \mathbf{q}_{rlj}, \mathbf{q}_{rjl}, \mathbf{q}_{rl'l} \geq 0 \quad \forall (r, j, a, l, l') \in \mathcal{R}(n) \times \mathcal{J} \times \mathcal{A} \times \mathcal{L} \quad (3.41)$$

Finalmente, a modo de resumen, se tabulan los valores de los parámetros y variables del Modelo II en la Tabla 3.8 y Tabla 3.9 respectivamente.

Parámetro	Descripción
\tilde{I}_{ra}	Inventario al principio del período actual para una referencia en un CD.
\tilde{I}_{rj}	Inventario al principio del período actual para una referencia en una tienda.
\tilde{I}_{rm}	Inventario al principio del período actual para una referencia en un país.
λ_j	Cantidad total de un clúster en un grupo g a transportar a cada tienda.
b_j	Cantidad máxima en unidades a transportar a cada tienda j .
B_{rj}	Cantidad máxima de transporte para una referencia a una tienda.
C_T	Porcentaje del precio del artículo que representa el coste de transporte entre las PL y las tiendas.
C_L	Porcentaje del precio del artículo que representa el coste de transporte entre dos PL.
C_{AT}	Porcentaje del precio del artículo que representa el coste de transporte entre el almacén de un país y una tienda.
R_S	Relación de holgura para un envío máximo por tienda o referencia.

Tabla 3.8: Parámetros del modelo II.

Variable	Descripción
q_{raj}	Unidades enviadas de una referencia desde un CD a una tienda.
q_{rmj}	Unidades enviadas de una referencia desde el almacén de un país a una tienda.
$q_{raa'}$	Transbordo realizado para una referencia entre dos CD.
q_{rlj}	Unidades enviadas de una referencia desde una PL a una tienda.
q_{rjl}	Unidades enviadas de una referencia desde una tienda a una PL.
$q_{rl'}$	Unidades enviadas de una referencia desde una PL a otra.

Tabla 3.9: Variables del modelo II.

3.4.6. Desarrollo del Modelo II

Como ya se ha realizado en la Subsección 3.2.8 con el Modelo I, se presenta una versión completa del Modelo II, con el fin de tener una visión más general de este. El código en Python del Modelo II se encuentra disponible en el Apéndice E.

$$\min \quad CDaCD + Tienda_PL + PLaPL \\ + CD_Tienda + Almacen_Tienda$$

en donde:

$$CDaCD = c_A \cdot \sum_{\substack{r \in \mathcal{R} \\ a, a' \in \mathcal{A}}} q_{raa'}$$

$$Tienda_PL = c_T \cdot \sum_{\substack{r \in \mathcal{R} \\ j \in \mathcal{J}}} q_{rl(j)j} + q_{rjl(j)}$$

$$PLaPL = c_L \cdot \sum_{\substack{r \in \mathcal{R} \\ l, l' \in \mathcal{L}}} q_{rl'}$$

$$CD_Tienda = c_E \cdot \sum_{\substack{r \in \mathcal{R} \\ j \in \mathcal{J}}} q_{ra(j)j}$$

$$Almacen_Tienda = c_{AT} \cdot \sum_{\substack{r \in \mathcal{R} \\ j \in \mathcal{J}}} q_{rm(j)j}$$

s. a.

$$\begin{aligned}
\tilde{I}_{ra} + \sum_{a' \in A} \mathbf{q}_{ra'a} &\geq \sum_{j \in \mathcal{T}(a)} \mathbf{q}_{raj} + \sum_{a' \in A} \mathbf{q}_{raa'} && \forall (r, a) \in \mathcal{R}(n) \times \mathcal{A} \\
\sum_{l' \in \mathcal{L}} \mathbf{q}_{rl'l} + \sum_{j \in \mathcal{T}(l)} \mathbf{q}_{rjl} &= \sum_{l' \in \mathcal{L}} \mathbf{q}_{rl'l} + \sum_{j \in \mathcal{T}(l)} \mathbf{q}_{rlj} && \forall (r, l) \in \mathcal{R}(n) \times \mathcal{L} \\
\tilde{I}_{rm} &\geq \sum_{j \in \mathcal{T}(m)} \mathbf{q}_{rmj} && \forall (r, m) \in \mathcal{R}(n) \times \mathcal{M} \\
\sum_{r \in \mathcal{R}(n)} I_{rj} + \mathbf{q}_{ra(j)j} + \mathbf{q}_{rm(j)j} + \mathbf{q}_{rl(j)j} &\geq \lambda_j + \sum_{r \in \mathcal{R}(n)} \mathbf{q}_{rjl(j)} && \forall j \in \mathcal{J} \\
\mathbf{q}_{ra(j)j} + \mathbf{q}_{rm(j)j} + \mathbf{q}_{rl(j)j} &\leq B_{rj} && \forall (r, j) \in \mathcal{R}(n) \times \mathcal{J} \\
\mathbf{q}_{raa'}, \mathbf{q}_{raj}, \mathbf{q}_{rmj}, \mathbf{q}_{rlj}, \mathbf{q}_{rjl}, \mathbf{q}_{rl'l} &\geq 0 && \forall (r, j, a, l, l') \in \mathcal{R}(n) \times \mathcal{J} \times \mathcal{A} \times \mathcal{L} \times \mathcal{L}
\end{aligned}$$

3.5. Observación. Diferencias código Python y AMPL

Del mismo modo que se ha hecho en la Sección 3.3, en este apartado se comentan las diferencias del modelo descrito con el código en AMPL. Las restricciones (3.37), (3.39) y (3.40) se sustituyen por las siguientes:

- **Restricción de flujo de entrada a cada tienda:**

La restricción (3.42) asegura que el flujo de entrada a cada tienda es menor que la cantidad total de transporte asignada a esa tienda.

$$\sum_{r \in \mathcal{R}} \mathbf{q}_{rm(j)j} + \mathbf{q}_{ra(j)j} \geq \max\{\lambda_j, 0\} \quad \forall j \in \mathcal{J} \quad (3.42)$$

- **Restricción de requisito de envío al almacén:**

Esta restricción obliga a que el inventario restante en un centro de distribución después de enviar unidades a las tiendas y a otros centros de distribución y recibir inventario de otros centros tiene que ser menor que un porcentaje del inventario que había inicialmente en el centro de distribución.

$$\sum_{r \in \mathcal{R}} (\tilde{I}_{ra} - \sum_{j \in \mathcal{J}} \mathbf{q}_{ra(j)j} + \sum_{a' \in \mathcal{A} \setminus \{a\}} \mathbf{q}_{ra'a} - \sum_{a' \in \mathcal{A} \setminus \{a'\}} \mathbf{q}_{raa'}) \leq 0,2 \cdot \sum_{r \in \mathcal{R}} \tilde{I}_{ra} \quad \forall a \in \mathcal{A} \quad (3.43)$$

- **Restricción de ratio máximo por tienda:**

Se impone un envío máximo a cada tienda, que viene dado por la cantidad máxima de unidades a transportar a cada tienda más un porcentaje del inventario total en el CD.

$$\mathbf{q}_{rm(j)j} + \sum_{a \in \mathcal{A}} \mathbf{q}_{ra(j)j} \leq b_j + \sum_{a \in \mathcal{A}} \tilde{I}_{ra} \cdot R_S \quad \forall (r, j) \in \mathcal{R} \times \mathcal{J} \quad (3.44)$$

3.5.1. Ejemplo Básico

Continuando con el ejemplo presentado en la Subsección 3.3.3, la empresa Zara dispone de dos tiendas en cada uno de los países. Las tiendas ubicadas en el primer país estarán asociadas a una misma plataforma logística y las ubicadas en el segundo a otra, como se muestra en la Figura 3.16. Para simplificar el modelo se trabaja con una referencia en el único grupo y clúster. Los demás conjuntos e índices se reducen a los siguientes:

$$\blacksquare \mathcal{J} = \{j_1, j_2, j_3, j_4\}$$

$$\blacksquare \mathcal{L} = \{l_1, l_2\}$$

El primer paso es calcular el número de unidades a enviar a cada tienda, λ_j , cantidad que estará acotada por la variable q_m multiplicada por el peso de la tienda en cada país, tal y como se indica en la Ecuación (3.28). Finalmente se obtienen valores de $\lambda_{j_1} = \lambda_{j_2} = \lambda_{j_3} = \lambda_{j_4} = 1,5$.

El siguiente paso es calcular el número máximo de unidades que se pueden enviar a cada una de las tiendas, B_j . Para ello seguimos los pasos que se han detallado en el Algoritmo 3, descrito en la Subsección 3.4.2. Como ya se hizo en la Subsección 3.3.3, se sustituye el parámetro de $Ventas_j$ por la demanda D_j .

Algoritmo 4 : Cálculo de $B_{r,j}$

Entrada: $q_m, \tilde{I}_a, D_j, I_j$

```

1: for  $j \in \mathcal{J}$  do
2:   Se define  $\mathcal{J}$  el conjunto de todas las tiendas para la única referencia.
3:    $q_m > 0 \forall m \in \mathcal{M}$ 
4: end for

```

```

5: for  $j \in \mathcal{J}$  do
6:    $IE_{j_1} = IE_{j_2} = IE_{j_3} = IE_{j_4} = 1, \mu_j = 1$  y  $\sigma_j = 0$ 
7: end for

```

```

8: for  $j \in \mathcal{J}$  do
9:    $IE_j > \mu_j - 1,5\sigma_j$ 
10: end for

```

```

11:  $H = 6$  uds,  $|\mathcal{J}| = 4$ 
12:  $H > |\mathcal{J}|$ 

```

```

13: for  $j \in \mathcal{J}$  do
14:    $B_{j_1} = B_{j_2} = B_{j_3} = B_{j_4} = 1,5$  uds
15: end for

```

```

16: for  $j \in \mathcal{J}$  do
17:   Se define  $X = 1$ . Por tanto  $B_j > X \forall j \in \mathcal{J}$ 
18: end for

```

```

19: for  $j \in \mathcal{J}$  do
20:   Se verifica que se cumple la Ecuación (3.29).
21: end for

```

Estos datos configuran los parámetros de entrada del Modelo II. La situación que se describe se recoge en la Figura 3.16.

Variable	Valor
qa_1j_1	1 uds
qa_1j_2	0 uds
qa_2j_3	1.5 uds
qa_2j_4	1.5 uds
qm_1j_1	0.5 uds
qm_1j_2	1.5 uds
qm_2j_3	0 uds
qm_2j_4	0 uds
qa_1a_2	3 uds
qa_2a_1	0 uds
ql_1j_1	0 uds
ql_1j_2	0 uds
ql_2j_3	0 uds
ql_2j_4	0 uds
qj_1l_1	0 uds
qj_2l_1	0 uds
qj_3l_2	0 uds
ql_4l_2	0 uds
ql_1j_1	0 uds
ql_1j_2	0 uds
ql_1l_2	0 uds
ql_2l_1	0 uds

Tabla 3.12: Variables continuas Modelo II.

En último lugar, se desagregan los términos de la función objetivo, es decir, se obtiene el valor de cada uno de los cinco tipos de costes (entre centros de distribución, de centro de distribución a tienda, de tienda a plataforma, entre plataformas y de almacén a tienda), tal y como aparece en la Ecuación (3.30). Estos valores se recogen en la Tabla 3.13. Siguiendo este esquema el coste de transporte es de 0.13 euros.

Variable	Valor
$CDaCD$	0.03 euros
$Tienda_PL$	0 euros
$PLaPL$	0 euros
CD_Tienda	0.08 euros
$Almacen_Tienda$	0.02 euros

Tabla 3.13: Parámetros del Modelo I.

3.5.2. Observación. Problema variables no enteras

Debido a que se trabaja con envíos en unidades, la obtención de variables no enteras es un problema a tener en cuenta. Para solucionar esto se deciden truncar los valores de las variables, ya que al trabajar con cantidades masivas de datos esto no supondrá un cambio muy grande en el valor de la función objetivo. Si se aplica este procedimiento se obtienen los valores de los parámetros y variables truncados, reflejados en la Tabla 3.14 y la Tabla 3.15.

Parámetro	Valor
\tilde{I}_{a_1}	6 uds
\tilde{I}_{a_2}	0 uds
\tilde{I}_{j_1}	0 uds
\tilde{I}_{j_2}	0 uds
\tilde{I}_{j_3}	0 uds
\tilde{I}_{j_4}	0 uds
\tilde{I}_{m_1}	2 uds
\tilde{I}_{m_2}	0 uds
λ_{j_1}	1 ud
λ_{j_2}	1 ud
λ_{j_3}	1 ud
λ_{j_4}	1 ud
B_{j_1}	1 ud
B_{j_2}	1 ud
B_{j_3}	1 ud
B_{j_4}	1 ud

Tabla 3.14: Parámetros truncados.

Variable	Valor
$q_{a_1 j_1}$	1 ud
$q_{a_1 j_2}$	0 uds
$q_{a_2 j_3}$	1 ud
$q_{a_2 j_4}$	1 ud
$q_{m_1 j_1}$	0 uds
$q_{m_1 j_2}$	1 ud
$q_{m_2 j_3}$	0 uds
$q_{m_2 j_4}$	0 uds
$q_{a_1 a_2}$	3 uds
$q_{a_2 a_1}$	0 uds
$q_{l_1 j_1}$	0 uds
$q_{l_1 j_2}$	0 uds
$q_{l_2 j_3}$	0 uds
$q_{l_2 j_4}$	0 uds
$q_{j_1 l_1}$	0 uds
$q_{j_2 l_1}$	0 uds
$q_{j_3 l_2}$	0 uds
$q_{l_4 l_2}$	0 uds
$q_{l_1 j_1}$	0 uds
$q_{l_1 j_2}$	0 uds
$q_{l_1 l_2}$	0 uds
$q_{l_2 l_1}$	0 uds

Tabla 3.15: Variables truncadas.

Capítulo 4

Puesta en Funcionamiento del Modelo

Hasta ahora se ha enunciado y explicado detenidamente el procedimiento que emplea Zara en la actualidad para determinar la distribución del inventario. Llegados a este punto se ejecutan el Modelo I y el Modelo II con la finalidad de obtener las salidas de los dos modelos y gráficos.

Mencionar que para este proyecto no ha sido posible el convenio acordado con la empresa Indra. Por tanto los datos relativos a la empresa Zara han tenido que ser generados de manera aleatoria. Por ello se dedicará la Sección 4.1 de este capítulo a explicar brevemente como ha sido el proceso de generación de datos.

4.1. Generando los Datos

En esta sección se explica brevemente el proceso de simulación de los datos. Recordar que debido a la poca información disponible se hace imposible que este procedimiento sea muy minucioso. Se han creado una serie de algoritmos con los que se pretende generar los datos de forma lógica. En las siguientes líneas se explican los pasos que se han seguido, así como las distribuciones que se han empleado para generar cada uno de los parámetros.

El módulo en el que se programan los algoritmos que generan los parámetros de entrada en los dos modelos es `Modelo_datos.py`. Puede verse con detalle en el Apéndice E.

Para ver de forma clara los principales conjuntos e índices de cada modelo, estos se recogen en la Tabla 5.2 y la Tabla 4.2. Para el Modelo II el número de referencias es a nivel de grupo y clúster.

Conjunto	Valor
\mathcal{M}	73 países
\mathcal{A}	3 centros de distribución
\mathcal{G}	10 grupos
$\mathcal{N}(g)$	50 clústers
\mathcal{K}	3 posibles precios
\mathcal{W}	4 períodos

Tabla 4.1: Parámetros del Modelo I.

Constante	Valor
$\mathcal{R}(n)$	100 referencias
\mathcal{J}	1200 tiendas
\mathcal{L}	3 plataformas logísticas

Tabla 4.2: Parámetros del Modelo II.

4.1.1. Generando Inventarios

En primer lugar se generan los inventarios de cada referencia, para cada centro de distribución, país y tienda. Para ello se sigue una distribución exponencial truncada, para que los valores que se obtienen no tengan mucha dispersion. Una vez obtenido el nivel medio de inventario, se varía en torno a una distribución normal para cada una de las referencias en un mismo grupo y clúster. En el caso de que se tengan valores negativos, el algoritmo lo suple cogiendo valores absolutos.

Ejemplo:

Por ejemplo, para el primer clúster, en el caso de la variable \tilde{I}_{rm} , la distribución $Exp(300)$ definida en el intervalo $(100,600)$, proporciona el nivel de inventario medio para todas las referencias de dicho clúster, en un cierto país. Este inventario, para cada referencia, se varía siguiendo una distribución normal de media ese valor. En la Figura 4.1a se representa un diagrama de barras con los valores medios de inventario para los 9 primeros países y el intervalo en el que varían las unidades de cada referencia. En la Figura 4.1b se grafica, para el primer país, un histograma de las unidades disponibles para cada una de las referencias, que como se observa sigue una distribución similar a la gaussiana.

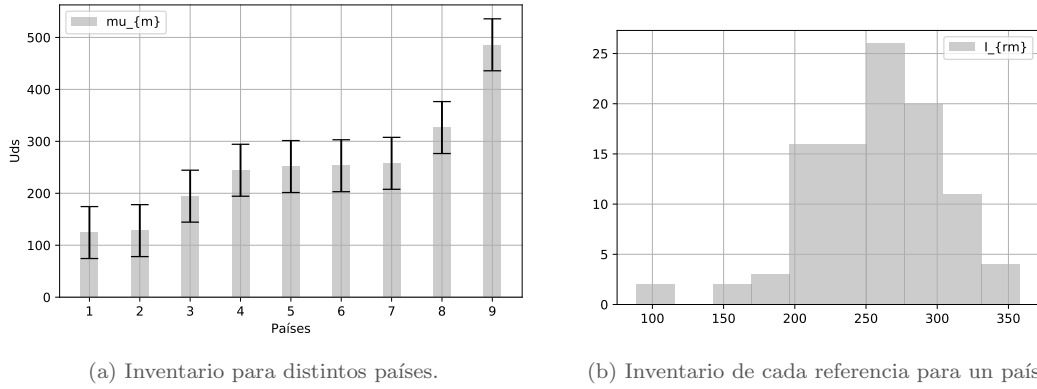


Figura 4.1: Construcción del nivel de inventario.

Los algoritmos que se han empleado para generar los inventarios $\tilde{I}_{r(gn)a}$, $\tilde{I}_{r(gn)m}$ e $\tilde{I}_{r(gn)j}$ son similares. Cada uno de ellos se ejecuta una vez para cada grupo y clúster.

Algoritmo 5 Generar \tilde{I}_{ra}

- 1: **for** $a \in \mathcal{A}$ **do**
 - 2: Nivel medio de inventario en el CD:
 $\mu_a \sim Exp(5 \cdot 10^8)$ en el intervalo $(10^8, 10^{10})$.
 - 3: **for** $r \in \mathcal{R}$ **do**
 - 4: $\tilde{I}_{r(gn)a} \sim N(\mu_a, 1000)$
 - 5: **if** $\tilde{I}_{r(gn)a} < 0$ **then**
 - 6: $\tilde{I}_{r(gn)a} = |\tilde{I}_{r(gn)a}|$
 - 7: **end if**
 - 8: **end for**
 - 9: **end for**
-

Algoritmo 6 Generar \tilde{I}_{rm}

```

1: for  $m \in \mathcal{M}$  do
2:   Nivel medio de inventario en el país:
    $\mu_m \sim \text{Exp}(300)$  en el intervalo (100,600).
3:   for  $r \in \mathcal{R}$  do
4:      $\tilde{I}_{r(gn)m} \sim N(\mu_m, 50)$ 
5:     if  $\tilde{I}_{r(gn)m} < 0$  then
6:        $\tilde{I}_{r(gn)m} = |\tilde{I}_{r(gn)m}|$ 
7:     end if
8:   end for
9: end for

```

Algoritmo 7 Generar \tilde{I}_{rj}

```

1: for  $j \in \mathcal{J}$  do
2:   Nivel medio de inventario en la tienda:
    $\mu_j \sim \text{Exp}(100)$  en el intervalo (0,200).
3:   for  $r \in \mathcal{R}$  do
4:      $\tilde{I}_{r(gn)j} \sim N(\mu_j, 50)$ 
5:     if  $\tilde{I}_{r(gn)j} < 0$  then
6:        $\tilde{I}_{r(gn)j} = |\tilde{I}_{r(gn)j}|$ 
7:     end if
8:   end for
9: end for

```

Una vez generados valores a nivel de referencia es fácil obtener los inventarios a un nivel de clúster, simplemente sumando los valores de las referencias en cada uno de ellos. Por tanto, para el Modelo I los inventarios a nivel de grupo y clúster se crean según las ecuaciones 4.1 y 4.2.

$$\tilde{I}_{agn} = \sum_{r \in \mathcal{R}(n)} \tilde{I}_{r(gn)a} \quad \forall (a, g, n) \in \mathcal{A} \times \mathcal{G} \times \mathcal{N} \quad (4.1)$$

$$\tilde{I}_{mgn} = \sum_{r \in \mathcal{R}(n)} \tilde{I}_{r(gn)m} \quad \forall (m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \quad (4.2)$$

En la Figura 4.2a se comparan las unidades en inventario de tres clústers distintos para los tres centros de distribución. En la Figura 4.2b se realiza esta misma comparación para los tres primeros países.

4.1.2. Generando Demandas

Para generar las demandas se ha decidido utilizar una distribución similar a la empleada para simular los inventarios. Para ello, se varían las unidades de inventario siguiendo una distribución normal. En primer lugar se han generado las demandas para cada referencia en cada tienda, como se indica en la Ecuación (4.3).

$$\tilde{D}_{r(gn)j} \sim |N(\tilde{I}_{r(gn)j}/4, \tilde{I}_{r(gn)j}/8)| \quad \forall (r, g, n, j) \in \mathcal{R} \times \mathcal{G} \times \mathcal{N} \times \mathcal{J} \quad (4.3)$$

Las demandas \tilde{D}_{jgn} y \tilde{D}_{mgn} se obtienen fácilmente sumando la demanda de cada una de las tiendas.

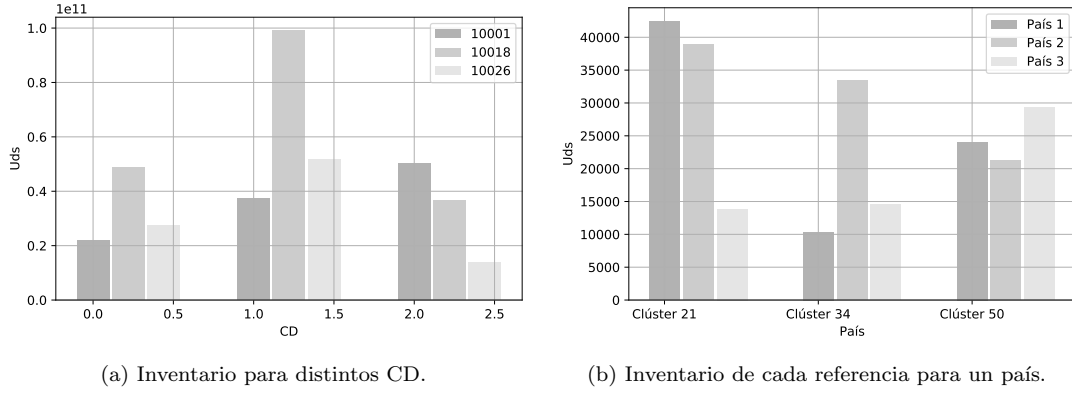


Figura 4.2: Construcción del nivel de inventario.

$$\tilde{D}_{jgn} = \sum_{r \in \mathcal{R}(n)} \tilde{D}_{r(gn)j} \quad \forall (j, g, n) \in \mathcal{J} \times \mathcal{G} \times \mathcal{N} \quad (4.4)$$

$$\tilde{D}_{mgn} = \sum_{j \in \mathcal{J}(m)} \tilde{D}_{r(gn)j} \quad \forall (m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \quad (4.5)$$

Para generar la demanda en el primer período de rebajas D_{mgnk}^0 , con la finalidad de tener en cuenta que la demanda aumenta en la temporada de rebajas, se siguen los pasos que se especifican en el Algoritmo 8.

Algoritmo 8 Generar D_{mgnk}^0

- 1: **for** $(m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N}$ **do**
 - 2: Generar un vector aleatorio $D_{mgn}^0 \sim N(2 \cdot \tilde{D}_{mgn}, \tilde{D}_{mgn}/10)$ de dimensión $|\mathcal{K}|$.
 - 3: Ordenar los elementos de D_{mgn}^0 en sentido creciente.
 - 4: **for** $k \in \mathcal{K}$ **do**
 - 5: Asignar a la menor demanda el mayor precio y así sucesivamente.
 - 6: **end for**
 - 7: **end for**
-

En la Figura 4.3 se comparan las unidades de demanda en la temporada regular y los distintos períodos de rebajas, para tres países y clústers distintos. Se puede observar que se ha considerado que la demanda sea mayor en la temporada de rebajas y sea decreciente según se avanza en el período de rebajas.

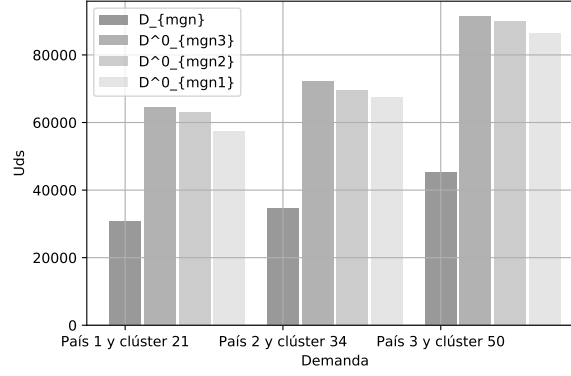


Figura 4.3: Construcción del nivel de demanda.

4.1.3. Generando Precios

Por último, se generan los precios para los dos modelos. Se genera el conjunto de precios para la temporada regular, \tilde{P}_{mgn} , siguiendo una distribución normal, en donde la notación N^+ hace referencia a la distribución gaussiana truncada inferiormente en 0.

$$\tilde{P}_{mgn} \sim N^+(30, 20) \quad \forall (m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \quad (4.6)$$

Para generar los posibles precios en rebajas se han creado algoritmos que tengan en cuenta la jerarquía de precios. Es decir, los precios de temporada regular son mayores que los precios en rebajas para un clúster o grupo determinado. Además los precios en rebajas se ordenan en sentido decreciente.

Algoritmo 9 Generar posibles precios

- 1: **for** $(m, g) \in \mathcal{M} \times \mathcal{G}$ **do**
 - 2: Calcular $pmin_{mg} = \min\{\tilde{P}_{mgn}, n \in \mathcal{N}(g)\}$
 - 3: Generar vector $p_{mg} \sim N(25, 10)$ truncado superiormente en $pmin_{mg}$, de dimensión $|\mathcal{K}|$.
 - 4: Ordenar los elementos de p_{mg} en sentido decreciente.
 - 5: Asignar en este orden a los precios $p_{mg1}, p_{mg2}, \dots, p_{mgk}$.
 - 6: **end for**
-

En último lugar se calcula el parámetro $psaldero_g$ de forma similar que se ha hecho para los precios en rebajas.

Algoritmo 10 Generar $psaldero_g$

- 1: **for** $g \in \mathcal{G}$ **do**
 - 2: Calcular $pmin_g = \min\{p_{mg}, m \in \mathcal{M}\}$
 - 3: Generar $psaldero_g \sim N(5, 3)$ truncado superiormente en $pmin_g$.
 - 4: **end for**
-

En la Figura 4.4a se representan los distintos precios de temporada regular y períodos de rebajas para tres países y tres clústers distintos. En la Figura 4.4b se representa un histograma para ver el rango de los precios en temporada regular, P_{mgn} , para el primer país.

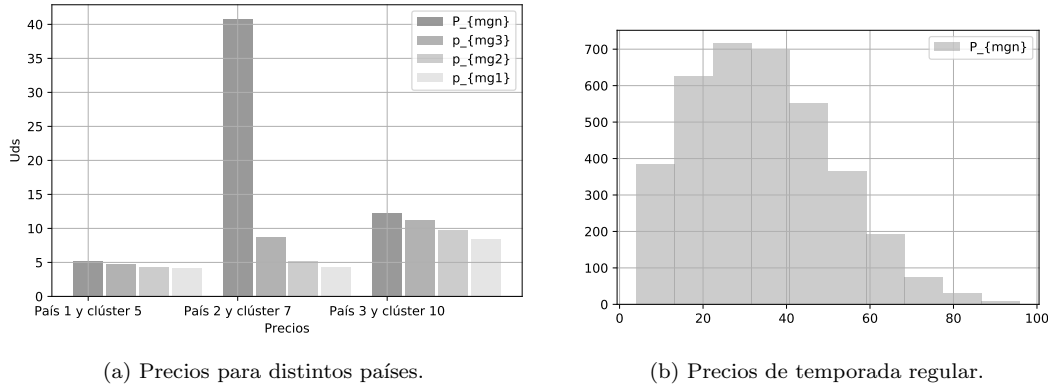


Figura 4.4: Precios.

4.2. Pruebas Computacionales e Implementación Práctica

En esta sección se implementan los dos modelos de optimización y se discuten los principales resultados obtenidos. Los módulos de los que se parte son `ModeloI.py` y `ModeloII.py` que fueron presentados en la Sección 3.2 y Sección 3.4.

4.2.1. Salida del Modelo I

El Modelo I se gestiona a nivel de grupo y clúster para todos los países de la red y su función es maximizar los ingresos globales. El resultado del modelo proporciona una estimación de ventas y envíos a nivel de país, de CD a CD y de CD a país. En este primer modelo, la variable de decisión más importante es q_{mgn} , ya que determina el total de envíos de un CD a un país y es el principal dato de entrada para el Modelo II.

En la Tabla 4.3 se recoge el flujo de inventario para el clúster 3, en el grupo 1. En este ejemplo, el modelo nos devuelve para cada país, su posición de inventario desde la fecha actual (Inv_Reg) hasta el final de la temporada de rebajas (Inv_Final). $Ventas_Reg$ y $Ventas_Reb$ son las ventas previstas para la temporada regular y la temporada de rebajas respectivamente. Inv_Reb es el inventario estimado al inicio de la temporada de rebajas.

Para el país 1, el modelo sugiere enviar 224025 unidades (q_{mgn}) para cubrir todas las ventas sin dejar un exceso de inventario en el país. Se puede observar que el inventario para la primera etapa de rebajas es igual a la diferencia entre el inventario en temporada regular y las ventas en temporada regular, más las unidades enviadas desde el centro de distribución, como se refleja en la Ecuación (4.7).

$$Inv_Reb = q_{mgn} + Inv_Reg - Ventas_Reg \quad (4.7)$$

En el caso del país 3, dado que la demanda en temporada regular es mayor que el inventario en el país, las unidades que quedan sin vender se venderían una vez que se recibe el envío desde el centro de distribución. Las ganancias totales, que vienen dadas por la función objetivo, en este caso tienen un valor de $1,7605 \cdot 10^{13}$.

País	Grupo	Clúster	Inv_Reg	$Ventas_Reg$	q_{mgn}	Inv_Reb	$Ventas_Reb$	Inv_Final
1	1	3	36559	28704	224025	231880	231880	0
2	1	3	53927	28721	0	25205	25205	0
3	1	3	21777	21777	341931	322893	322892	0
4	1	3	10974	10974	370730	339337	339337	0

Tabla 4.3: Transportes por país detallados del Modelo I.

Se representan los valores de q_{mgn} para el clúster 3, en el grupo 1, en los 9 primeros países. En la Figura 4.5b se representan los valores de expedición máxima B_{mgn} . Se observa que el transporte es prácticamente proporcional a la expedición máxima en cada país. Sin embargo, el cambio en la escala es considerable, esto seguramente sea debido a la forma en la que se generan los datos.

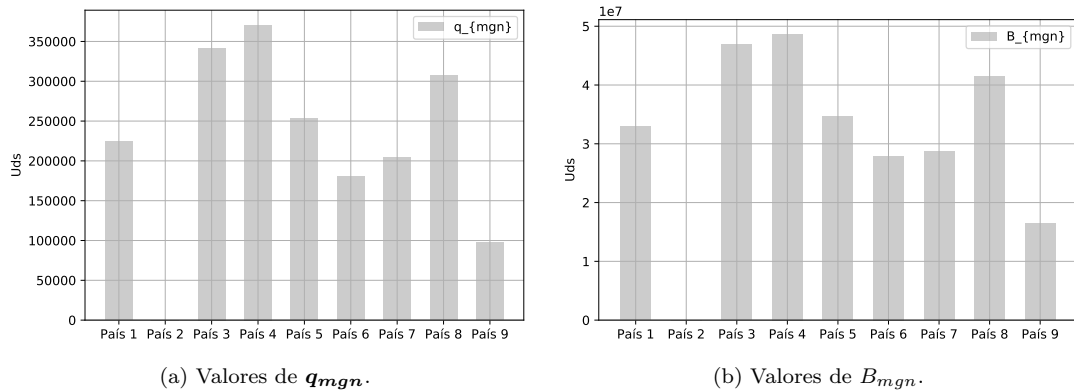


Figura 4.5: Envío de CD a país.

Para ver como evoluciona el nivel de demanda a lo largo del tiempo, se representa para el clúster 3 en el grupo 1, para los tres primeros países, el nivel de demanda en temporada regular (\tilde{d}_{mgn}) y en las distintas etapas de la temporada de rebajas ($d_{mgnk}^0, d_{mgnk}^1, d_{mgnk}^2$ y d_{mgnk}^3). Se observa en la Figura 4.6, que la demanda en temporada regular es menor que la demanda en rebajas en los tres países. Además, el modelo de optimización devuelve el mismo valor para la demanda en los tres períodos de rebajas para el país 1 y 3 (ya que se le ha asignado valor 1 al parámetro de *tiempo*). En el país 2, la demanda a partir del segundo período es cero.

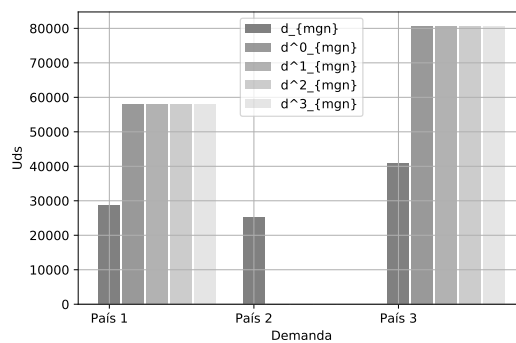


Figura 4.6: Evolución de la demanda para los tres primeros países.

De la misma forma, para ver la evolución temporal del inventario, se representan en la Figura 4.7 los valores del nivel de inventario en la temporada regular (\tilde{I}_{mgn}) y al inicio de cada período de rebajas ($I_{mgn}^0, I_{mgn}^1, I_{mgn}^2, I_{mgn}^3$ e I_{mgn}^4) para los tres primeros países y el clúster 3, en el grupo 1. El nivel de inventario es menor en la temporada regular y crece una vez que se envía mercancía desde los centros de distribución. Se observa que en el país 2, el inventario se agota en el segundo período de rebajas, como ya se podía intuir a partir de la Figura 4.6. En el resto de países el inventario se agota en el último período de rebajas, como ya se mostraba en la Tabla 4.3.

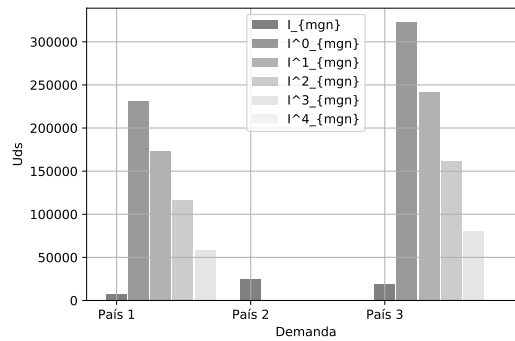


Figura 4.7: Evolución del inventario para los tres primeros países.

4.2.2. Salida del Modelo II

El Modelo II utiliza la recomendación de envío para cada país del Modelo I y determina el tamaño del envío por tienda y artículo. La función objetivo pretende minimizar los gastos de transporte entre tiendas, almacenes y centros de distribución.

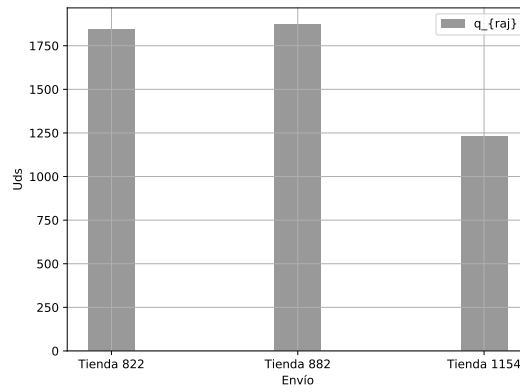
En la Tabla 4.4 se recoge el flujo de inventario de distintas tiendas para distintas referencias¹, en el clúster 3 y grupo 1. En este ejemplo, el modelo nos devuelve para cada tienda, el número de unidades de la referencia en inventario en la fecha actual (*Stock*), los transportes desde el centro de distribución a la tienda y del almacén del país a tienda. Para la tienda 1, el inventario de la referencia que se especifica es suficiente y no es necesario el traslado de unidades a esta tienda. En el caso de la tienda 627, se realiza un traslado desde el almacén del país. Para la tienda 1105 el traslado se realiza directamente desde el centro de distribución a la tienda. Los costes totales, que vienen dados por la función objetivo, en este caso tienen un valor de 18739 euros.

País	Tienda	Grupo	Clúster	Artículo	Stock	País-Tienda	DC-Tienda
1	1	1	3	1	134	0	0
39	627	1	3	99	52	0	9
68	1105	1	3	57	50	8	0

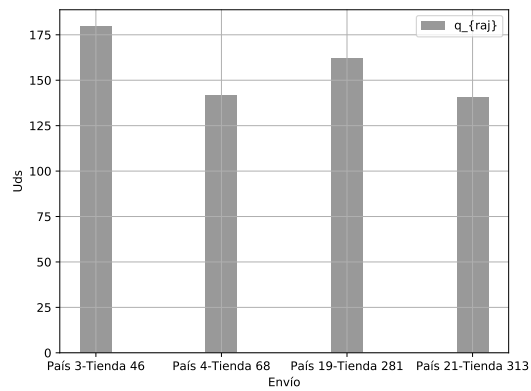
Tabla 4.4: Transportes por tienda detallados del Modelo II.

En la Figura 4.8 se representan valores de la variable que dicta los envíos desde los centros de distribución a las tiendas, q_{raj} , para la primera referencia desde el centro de distribución 10026.

¹La notación para las referencias que se emplea en este capítulo no es la explicada en la Sección 2.2.1, sin embargo al no disponer de información se deciden numerar las referencias con números desde 1 hasta 100.

Figura 4.8: Valores de q_{raj} .

Para visualizar los envíos desde el almacén del país a las tiendas, en la Figura 4.9 se representan valores de la variable q_{rmj} para la referencia 34. Se observa que estos valores son menores que los envíos desde el centro de distribución al país, q_{raj} .

Figura 4.9: Valores de q_{rmj} .

Capítulo 5

Profundizando en el Modelo

Como ya se ha visto en capítulos anteriores, el primer modelo sugiere un esquema general de los precios de rebajas, y acota, en cierto modo, los artículos de cada referencia que serán enviados a cada país. Por tanto, el primer modelo proporciona simplemente una aproximación de los traslados definitivos, que vendrán dados por el Modelo II. El objetivo de este capítulo será describir una serie de heurísticas con el propósito de prescindir del primer modelo o simplificarlo, con el único fin de reducir el tiempo computacional. Posteriormente se valorará si dichas modificaciones en el modelo son razonables para este problema.

En este capítulo se describen una serie de heurísticas, con las que se pretende omitir el Modelo I. En primer lugar, se describe una primera heurística que dicta los precios en cada período de rebajas así como las demandas en cada uno de ellos. A continuación, se presentan un par de heurísticas, que emplean como variable de entrada la demanda calculada con la primera heurística, y devuelven las unidades enviadas desde los centros de distribución a los países, q_{mgn} .

5.1. Archivos en Python

Las heurísticas que se describen a lo largo de este capítulo se encuentran en los siguientes ficheros `.py`.

- `HeuristicaI.py`: En este archivo se programa la primera heurística, que devuelve valores aproximados de la demanda en temporada regular \tilde{d}_{mgn} y en rebajas d_{mgnk}^w .
- `HeuristicaII.py`: Este archivo contiene la segunda heurística, que calcula los envíos a los países desde los centros de distribución, q_{mgn} , con datos de entrada dados por el módulo `HeuristicaI.py`.
- `HeuristicaIII.py`: La tercera heurística calcula la variable q_{mgn} , con datos de entrada dados por el módulo `HeuristicaI.py`.

5.2. Heurística para el Cálculo de la Demanda

La primera modificación probada ha sido una heurística que devuelve la demanda en la temporada regular y en cada una de las etapas de rebajas, así como los precios que están en vigencia en cada una de ellas. Dicha heurística calcula la variable de demanda en la temporada regular como la demanda obtenida con el modelo de predicción, que serán datos de entrada para las dos heurísticas que calculan el parámetro q_{mgn} , que se describirán en secciones posteriores.

$$\tilde{d}_{mgn} \approx \tilde{D}_{mgn} \quad \forall (m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \quad (5.1)$$

En la temporada de rebajas, la heurística considera que el precio que está en vigencia en cada etapa siempre es menor que el de la etapa anterior. Es decir, el precio para cada clúster n , en el grupo g , se descuenta en cada una de las etapas de rebajas. Para que esto ocurra las variables \mathbf{y}_{mgnk}^w que cumplen la condición $k = |\mathcal{K}| - w$ se asignan a 1. En el caso de que haya más etapas que precios, a las restantes etapas se les asigna el menor precio, es decir, \mathbf{y}_{mgn1}^w se asigna a 1.

De esta forma, la demanda en los distintos períodos de rebajas, \tilde{d}_{mgnk}^w , viene dada por la demanda para el primer período, calculada en el modelo de predicción, multiplicada por $\mathbf{y}_{mgnk}^w \cdot tiempo^w$.

$$\tilde{d}_{mgnk}^w \approx D_{mgnk}^0 \cdot \mathbf{y}_{mgnk}^w \cdot tiempo^w \quad \forall (w, m, g, n, k) \in \mathcal{W} \times \mathcal{M} \times \mathcal{G} \times \mathcal{N} \times \mathcal{K} \quad (5.2)$$

Se describe el pseudocódigo en el Algoritmo 11.

Algoritmo 11 Heurística para el cálculo de la demanda

Entrada: \tilde{D}_{mgn} , D_{mgnk}^0 , *tiempo*

- 1: **for** $w \in \{0, 1, \dots, \min\{W_r, |\mathcal{K}|\}\}$ **do**
- 2: **for** $k \in \mathcal{K}$ **do**
- 3: **if** $k = |\mathcal{K}| - w$ **then**
- 4: $\mathbf{y}_{mgnk}^{h_1, w} = 1$
- 5: **else**
- 6: $\mathbf{y}_{mgnk}^{h_1, w} = 0$
- 7: **end if**
- 8: **end for**
- 9: **end for**
- 10: **if** $W_r > |\mathcal{K}|$ **then**
- 11: **for** $w \in \{\min\{W_r, |\mathcal{K}|\} + 1, \dots, \max\{W_r, |\mathcal{K}|\}\}$ **do**
- 12: **for** $k \in \mathcal{K}$ **do**
- 13: **if** $k = 1$ **then**
- 14: $\mathbf{y}_{mgn1}^{h_1, w} = 1$
- 15: **else**
- 16: $\mathbf{y}_{mgnk}^{h_1, w} = 0$
- 17: **end if**
- 18: **end for**
- 19: **end for**
- 20: **end if**
- 21: $\tilde{d}_{mgn}^{h_1} = \tilde{D}_{mgn}$
- 22: $\tilde{d}_{mgnk}^{h_1, w} = D_{mgnk}^0 \cdot \mathbf{y}_{mgnk}^{h_1, w} \cdot tiempo^w$

En las siguientes líneas se presenta el código en Python del Algoritmo 11. Este código está disponible en el módulo `HeuristicaI.py`.

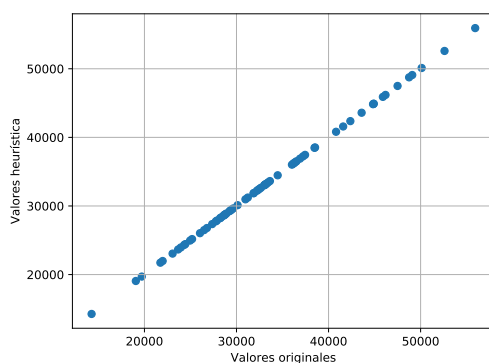
Código en Python:

```

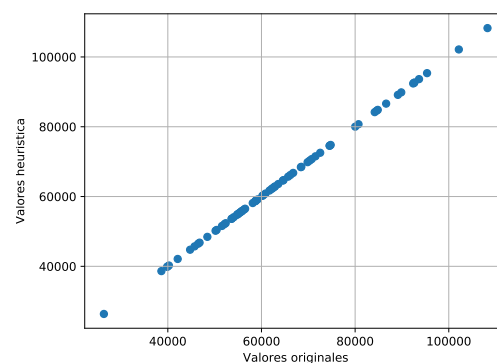
1 from sim_datos import Wr, PAIS, GRUPO, CLUSTER, setCLUSTER, setPRECIOS, Dmgn, D_mgn
2 import itertools
3
4 ## Variable y:
5 y = {}
6 for m in PAIS:
7     for g in GRUPO:
8         for n in CLUSTER[g]:
9             for w in range(min(Wr, len(setPRECIOS))):
10                 for k in setPRECIOS:
11                     if k == len(setPRECIOS) - w:
12                         y[w, m, g, n, k] = 1
13                     else:
14                         y[w, m, g, n, k] = 0
15             if Wr > len(setPRECIOS):
16                 for w in range(len(setPRECIOS), Wr):
17                     for k in setPRECIOS:
18                         if k == 1:
19                             y[w, m, g, n, k] = 1
20                         else:
21                             y[w, m, g, n, k] = 0
22
23 ## Demanda en temporada regular:
24 dmgn = Dmgn
25
26 ## Demanda en temporada de rebajas:
27 d_mgn = {}
28 for m in PAIS:
29     for g in GRUPO:
30         for n in CLUSTER[g]:
31             for k in setPRECIOS:
32                 for w in range(Wr):
33                     d_mgn[w, m, g, n, k] = D_mgn[m, g, n, k] * y[w, m, g, n, k] * tiempo^w

```

Para los 9 primeros países se representa un ajuste lineal de las demandas obtenidas por el Modelo I frente a las obtenidas por la heurística. En el caso de la temporada de rebajas se representa la demanda en el primer período, con precio p_3 . Se observa que en el caso de la temporada regular, las demandas son coincidentes, esto se debe a que la demanda que se obtiene por el Modelo I es igual a la calculada por el modelo de predicción, \hat{D}_{mgn} , y por tanto a la obtenida con el Algoritmo 11. En el caso de la temporada de rebajas el ajuste también es exacto, por el mismo motivo.



(a) Demandas temporada regular.



(b) Demandas en rebajas.

Figura 5.1: Demanda obtenida por el Modelo I y el Algoritmo 11.

5.3. Primera Heurística para el Cálculo del Envío

Obtenidos valores para la demanda en el Algoritmo 11, se ha probado el funcionamiento de una heurística sencilla que calcule una variable q_{mgn} aproximada a la obtenida por el Modelo I. Para la creación de este algoritmo se parte de la restricción de balance de inventario del Modelo I, dada en la Ecuación (3.8).

$$I_{mgn}^0 = \tilde{I}_{mgn} + q_{mgn} - \tilde{d}_{mgn} \quad \forall (m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \quad (5.3)$$

Pasando el valor que se quiere calcular al lado izquierdo de la igualdad queda:

$$q_{mgn} = I_{mgn}^0 - \tilde{I}_{mgn} + \tilde{d}_{mgn} \quad \forall (m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \quad (5.4)$$

Las variables I_{mgn}^0 e \tilde{d}_{mgn} que son calculadas por el Modelo I en este caso son desconocidas. Para la construcción del algoritmo se van a intentar aproximar por valores cercanos a ellas. Para el caso de la demanda \tilde{d}_{mgn} , se emplea la variable calculada en el Algoritmo 11, $\tilde{d}_{mgn}^{h_1}$. Suponiendo que se vende la mayor parte de las unidades disponibles en inventario al final de la temporada de rebajas, la variable I_{mgn}^0 se puede aproximar por la demanda en toda la temporada de rebajas, que ha sido calculada en el Algoritmo 11. Lo descrito aparece reflejado en las ecuaciones (5.5) y (5.6).

$$\tilde{d}_{mgn} \approx \tilde{d}_{mgn}^{h_1} \quad \forall (m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \quad (5.5)$$

$$I_{mgn}^0 \approx \sum_{w=0}^{W_r-1} \sum_{k \in \mathcal{K}} d_{mgnk}^{h_1, w} \quad \forall (m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \quad (5.6)$$

De esta forma se tiene:

$$q_{mgn} \approx \sum_{w=0}^{W_r-1} \sum_{k \in \mathcal{K}} d_{mgnk}^{h_1, w} - \tilde{I}_{mgn} + \tilde{d}_{mgn}^{h_1} \quad \forall (m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \quad (5.7)$$

Para que el algoritmo sea más exacto, se importa el parámetro de expedición máxima B_{mgn} . En el caso de que el valor de B_{mgn} sea nulo, se igualará a cero el valor de q_{mgn} para dicho país y clúster. Además, por razones de factibilidad se obliga a que q_{mgn} sea menor que B_{mgn} . En este algoritmo, las variables calculadas en el Algoritmo 11, $d_{mgnk}^{h_1, w}$ y $\tilde{d}_{mgn}^{h_1}$, se consideran parámetros de entrada, por ello no se denotarán en negrita. Se puede ver el pseudocódigo en el Algoritmo 12.

Algoritmo 12 Primera heurística para el cálculo del envío

Entrada: $B_{mgn}, \tilde{d}_{mgn}^{h1}, d_{mgnk}^{h1,w}$

- 1: **for** $(m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N}$ **do**
- 2: **if** $B_{mgn} = 0$ **then**
- 3: $q_{mgn}^{h2} = 0$
- 4: **else**
- 5: $q_{mgn}^{h2} = \sum_{w=0}^{Wr-1} \sum_{k \in \mathcal{K}} d_{mgnk}^{h1,w} - \tilde{I}_{mgn} + \tilde{d}_{mgn}^{h1}$
- 6: **end if**
- 7: **if** $q_{mgn}^{h2} \geq B_{mgn}$ **then**
- 8: $q_{mgn}^{h2} = B_{mgn}$
- 9: **end if**
- 10: **end for**

En las siguientes líneas se presenta el código en Python del Algoritmo 12. Este código está disponible en el módulo `HeuristicaII.py`.

Código en Python:

```

1 from Modelo_datos import PAIS, GRUPO, CLUSTER, setPRECIOS, Imgn, Wr
2 from Bmgn import Bmgn
3 from HeuristicaI import dmgn, d_mgn
4
5 qmgn = {}
6 for m in PAIS:
7     for g in GRUPO:
8         for n in CLUSTER[g]:
9             if Bmgn[m, g, n] == 0:
10                qmgn[m, g, n] = 0
11            else:
12                qmgn[m, g, n] = sum( sum( d_mgn[w, m, g, n, k] for k in setPRECIOS) for w in
13                                   list(range(0, Wr))) + dmgn[m, g, n] - Imgn[m, g, n]
14            if qmgn[m, g, n] >= Bmgn[m, g, n]:
15                qmgn[m, g, n] = Bmgn[m, g, n]

```

Se representa para los nueve primeros países, en el clúster 3, la nueva variable q_{mgn}^{h2} , frente a la variable que se obtiene por el Modelo I, q_{mgn} . Se observa que la diferencia entre los dos valores es mínima.

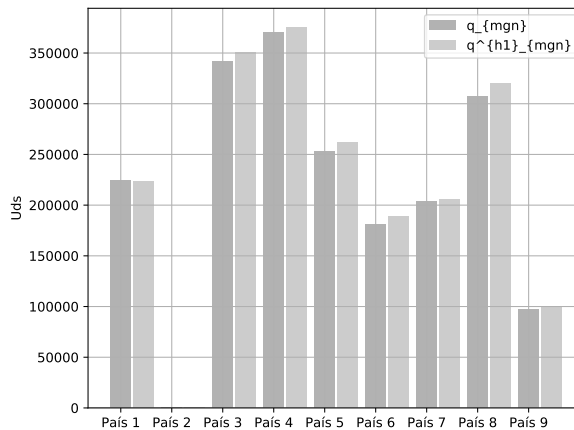


Figura 5.2: Valores de q_{mgn} frente a q_{mgn}^{h2} .

Para asegurarse si el nuevo proceso se alinea con la decisión del modelo actual, ambos pronósticos se trazan en un ajuste bivariable, para el clúster 3. Como se puede ver en la Figura 5.3, la solución dada por el Algoritmo 12, con las demandas calculadas en el Algoritmo 11, se ajusta bastante bien al proceso actual.

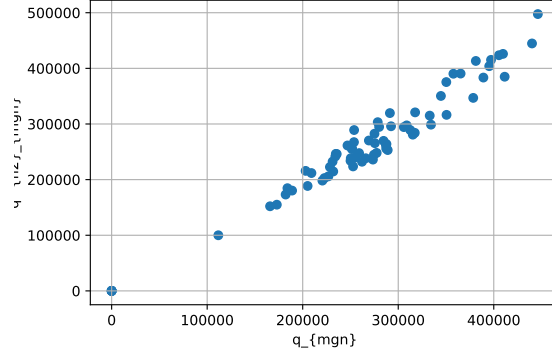


Figura 5.3: Ajuste bivariable de q_{mgn} frente a q_{mgn}^{h2} .

Otra forma de evaluar la eficacia de los algoritmos 11 y 12 es calcular las ganancias derivadas de las ventas de inventario, sin restar los costes, para el procedimiento original y empleando las heurísticas. Las ganancias totales se obtendrían sumando los ingresos que se tienen de las ventas de inventario, dados en la función objetivo del Modelo I y restando los gastos, es decir, el valor de la función objetivo del Modelo II. En la Ecuación (5.8) se representan los beneficios obtenidos de las ventas, sin restar costes.

$$Ventas_FdT + Ventas_R + Ventas_S \quad (5.8)$$

siendo:

$$\begin{aligned} Ventas_FdT &= \sum_{\substack{(m,g) \in \mathcal{M} \times \mathcal{G} \\ n \in \mathcal{N}(g)}} \tilde{P}_{mgn} \times \tilde{d}_{mgn} \\ Ventas_R &= \sum_{\substack{(m,g) \in \mathcal{M} \times \mathcal{G} \\ n \in \mathcal{N}(g) \\ 0 \leq w \leq W_r, k \geq 1}} p_{mgk} \times d_{mgnk}^w \\ Ventas_S &= \sum_{g \in \mathcal{G}} \left(\sum_{m \in \mathcal{M}} Lsaldero_{mg} + \sum_{\substack{a \in \mathcal{A} \\ n \in \mathcal{N}(g)}} \tilde{I}_{agn} - \sum_{\substack{m \in \mathcal{M} \\ n \in \mathcal{N}(g)}} q_{mgn} \right) \times psaldero_g \end{aligned}$$

En la Tabla se desagregan los términos de la Ecuación 5.8 el procedimiento original y empleando los algoritmos 11 y 12.

Las ganancias que se obtienen por el modelo original son de $1.76038 \cdot 10^{13}$ euros, mientras que empleando las dos heurísticas son de $1.76037 \cdot 10^{13}$ euros. Es decir, habría una diferencia de 10^8 euros. Por tanto, el importe total dado por los dos métodos varía en un porcentaje menor que el 0.00057%. Por lo que, los algoritmos 11 y 12, parecen ser adecuados para comparar en secciones posteriores los costes de envío, dados en la función objetivo del Modelo II, con los obtenidos por el procedimiento actual.

	Proceso original	Heurística
<i>Ventas_FdT</i>	$4.06224 \cdot 10^9$	$4.06224 \cdot 10^9$
<i>Ventas_R</i>	$1.02932 \cdot 10^{10}$	$1.02932 \cdot 10^{10}$
<i>Ventas_S</i>	$1.75894 \cdot 10^{13}$	$1.75893 \cdot 10^{13}$

Tabla 5.1: Parámetros del Modelo I.

5.4. Segunda Heurística para el Cálculo del Envío

En esta sección se describe una nueva heurística, que utiliza como variables de entrada, las demandas obtenidas en el Algoritmo 11, y devuelve como variable de salida el envío desde los centros de distribución a los países, q_{mgn} .

Se define la variable Dem_fal_{mgn} , como el número de unidades faltantes en el inventario de un país para completar la demanda existente en la temporada regular. En el caso de que esta demanda fuese mayor que B_{mgn} se elegiría el menor valor de los dos, para ello se crea la variable Dem_max_{mgn} . Mencionar que la esta heurística emplea como parámetro de entrada, la demanda en temporada regular calculada con el Algoritmo 11, $\tilde{d}_{mgn}^{h_1}$.

$$Dem_fal_{mgn} = \text{máx}\{\tilde{d}_{mgn}^{h_1} - \tilde{I}_{mgn}, 0\} \quad \forall (m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \quad (5.9)$$

$$Dem_max_{mgn} = \text{mín}\{Dem_fal_{mgn}, B_{mgn}\} \quad \forall (m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \quad (5.10)$$

La base de esta heurística es maximizar las ventas. Por tanto, se recorre $\tilde{P}_{mgn} \cdot \tilde{d}_{mgn}^{h_1}$ en sentido decreciente, para completar antes las demandas de los artículos que proporcionan más ganancias. En el caso de que el centro de distribución a asociado al país disponga de las unidades suficientes para abastecer la demanda del país, se envían las unidades requeridas y se restan del inventario en el centro de distribución. Si el centro de distribución a asociado al país no dispone de las unidades suficientes para abastecer a dicho país, se busca otro centro de distribución a' que pueda enviar las unidades restantes para completar la demanda. Para ello se define el conjunto de los centros de distribución que pueden abastecer al centro de distribución a .

$$\mathcal{A}_{a'a} = \{a' \in \mathcal{A} : \exists q_{aa'}\} \quad (5.11)$$

Se recorren los centros del conjunto $\mathcal{A}_{a'a}$ de forma que $I_{a'gn}$ se ordene en sentido decreciente. Es decir, se recorren primero los centros que dispongan de más inventario para el clúster n en el grupo g . De esta forma se va completando la demanda máxima faltante, $Dem_max_fal_{mgn}$ que viene dada en la Ecuación (5.12).

$$Dem_max_fal_{mgn} = Dem_max_{mgn} - \tilde{I}_{a(m)gn} \quad \forall (m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \quad (5.12)$$

Para que el algoritmo sea más exacto, antes del comienzo de la temporada de rebajas, se actualiza el nivel de inventario en el país (aunq el inventario real sería con el envío desde el centro de distribución).

$$\tilde{I}_{mgn} = \text{máx}\{\tilde{I}_{mgn} - \tilde{d}_{mgn}^{h_1}, 0\} \quad \forall (m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \quad (5.13)$$

Para el cálculo del envío en la temporada de rebajas, el procedimiento es similar. Para cada período de la temporada de rebajas se define la variable $\mathbf{Dem_fal_r_mgn}$, como el número de unidades faltantes en el inventario de un país para completar la demanda existente en cada periodo. En el caso de que esta demanda fuese mayor que B_{mgn} se elegiría el menor valor de los dos, para ello se crea la variable $\mathbf{Dem_max_r_mgn}$, como ya se ha hecho para la temporada regular. El Algoritmo 13 emplea como dato de entrada la demanda para cada período de la temporada de rebajas calculada con el Algoritmo 11, \tilde{d}_{mgnk}^{w,h_1} .

$$\mathbf{Dem_fal_r_mgn} = \max\left\{\sum_{k \in \mathcal{K}} \tilde{d}_{mgnk}^{h_1,w} - \tilde{I}_{mgn}, 0\right\} \quad \forall (m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \quad (5.14)$$

$$\mathbf{Dem_max_r_mgn} = \min\{\mathbf{Dem_fal_r_mgn}, B_{mgn}\} \quad \forall (m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N} \quad (5.15)$$

Se recorre $p_{mgk} \cdot \tilde{d}_{mgnk}^{h_1,w}$ en sentido decreciente, para completar antes las demandas de los artículos que proporcionan más ganancias. A continuación se aplica el mismo procedimiento que en la temporada regular para actualizar el valor de los envíos q_{mgn} . El pseudocódigo de la heurística descrita se detalla en el Algoritmo 13 (el código en Python está disponible en el Apéndice G).

Algorithm 13 Segunda heurística para el cálculo del envío

Entrada: $\tilde{P}_{mgn}, \tilde{I}_{mgn}, \tilde{I}_{agn}, B_{mgn}, \tilde{d}_{mgn}^{h_1}, \tilde{d}_{mgnk}^{h_1,w}$

- 1: **for** $(m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N}(g)$ **do**
- 2: $\mathbf{Dem_fal_mgn} = \max\{\tilde{d}_{mgn}^{h_1} - \tilde{I}_{mgn}, 0\}$
- 3: $\mathbf{Dem_max_mgn} = \min\{\mathbf{Dem_fal_mgn}, B_{mgn}\}$
- 4: **end for**
- 5: **for** (m, g, n) recorriendo $\tilde{P}_{mgn} \cdot \tilde{d}_{mgn}^{h_1}$ en sentido decreciente **do**
- 6: **if** $\tilde{I}_{a(m)gn} \geq \mathbf{Dem_max_mgn}$ **then**
- 7: $q_{mgn}^{h_3} = \mathbf{Dem_max_mgn}$
- 8: $\tilde{I}_{a(m)gn} = \tilde{I}_{a(m)gn} - \mathbf{Dem_max_mgn}$
- 9: **else**
- 10: $\mathbf{Dem_max_fal_mgn} = \mathbf{Dem_max_mgn} - \tilde{I}_{a(m)gn}$
- 11: $q_{mgn}^{h_3} = \tilde{I}_{a(m)gn}$
- 12: $\tilde{I}_{a(m)gn} = 0$
- 13: $\mathcal{A}_{a'a} = \{a' \in \mathcal{A} : \exists q_{aa'}\}$
- 14: $ord(\mathcal{A}_{a'a})$: $\mathcal{A}_{a'a}$ ordenado según $\tilde{I}_{a'gn}$
- 15: **for** $a' \in ord(\mathcal{A}_{a'a})$ **do**
- 16: $d = \min\{\tilde{I}_{a'gn}, \mathbf{Dem_max_fal_mgn}\}$
- 17: $q_{mgn}^{h_3} = q_{mgn}^{h_3} + d$
- 18: $\tilde{I}_{a'gn} = \tilde{I}_{a'gn} - d$
- 19: $q_{a'agn} = q_{a'agn} + d$
- 20: **if** $q_{mgn}^{h_3} = \mathbf{Dem_max_mgn}$ **then**
- 21: **break**
- 22: **end if**
- 23: **end for**
- 24: **end if**
- 25: **end for**
- 26: **for** $(m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N}(g)$ **do**
- 27: $\tilde{I}_{mgn} = \max\{\tilde{I}_{mgn} - \tilde{d}_{mgn}^{h_1}, 0\}$
- 28: **end for**
- 29: **for** $w \in \{0, \dots, W_r\}$ **do**
- 30: **for** $(m, g, n) \in \mathcal{M} \times \mathcal{G} \times \mathcal{N}(g)$ **do**

```

31:    $Dem\_fal\_r_{mgn} = \max\{\sum_{k \in \mathcal{K}} d_{mgnk}^{h_1,w} - \tilde{I}_{mgn}, 0\}$ 
32:    $Dem\_max\_r_{mgn} = \min\{Dem\_fal\_r_{mgn}, B_{mgn}\}$ 
33: end for
34: for  $(m, g, n)$  recorriendo  $\tilde{p}_{mgk} \cdot d_{mgnk}^{h_1,w}$  en sentido decreciente do
35:   if  $\tilde{I}_{a(m)gn} \geq Dem\_max\_r_{mgn}$  then
36:      $q_{mgn}^{h_3} = q_{mgn}^{h_3} + Dem\_max\_r_{mgn}$ 
37:      $\tilde{I}_{a(m)gn} = \tilde{I}_{a(m)gn} - Dem\_max\_r_{mgn}$ 
38:   else
39:      $Dem\_max\_fal\_r_{mgn} = Dem\_max\_r_{mgn} - \tilde{I}_{a(m)gn}$ 
40:      $q_{mgn}^{h_3} = q_{mgn}^{h_3} + \tilde{I}_{a(m)gn}$ 
41:      $\tilde{I}_{a(m)gn} = 0$ 
42:     for  $a' \in ord(\mathcal{A}_{a'})$  do
43:        $d = \min\{\tilde{I}_{a'gn}, Dem\_max\_fal\_r_{mgn}\}$ 
44:        $q_{mgn}^{h_3} = q_{mgn}^{h_3} + d$ 
45:        $\tilde{I}_{a'gn} = \tilde{I}_{a'gn} - d$ 
46:        $q_{a'agn} = q_{a'agn} + d$ 
47:       if  $q_{mgn}^{h_3} = Dem\_max\_r_{mgn}$  then
48:         break
49:       end if
50:     end for
51:   end if
52: end for
53:  $\tilde{I}_{mgn} = \max\{\tilde{I}_{mgn} - \sum_{k \in \mathcal{K}} d_{mgnk}^{h_1,w}, 0\}$ 
54: end for

```

En la Figura 5.4 se puede observar como para los nueve primeros países, en el clúster 3, la diferencia entre la nueva variable $q_{mgn}^{h_3}$, obtenida por el Algoritmo 13, y la variable que se obtiene por el Modelo I, q_{mgn} , no es muy grande, excepto para los países 2 y 9. Se observa que para el País 2, el valor de $q_{mgn}^{h_3}$ que se obtiene es cero, frente a las 250 000 unidades que sugiere enviar el Modelo I. Esto quizás sea debido a que no se dispone de inventario en los centros de distribución para realizar el envío a dicho país.

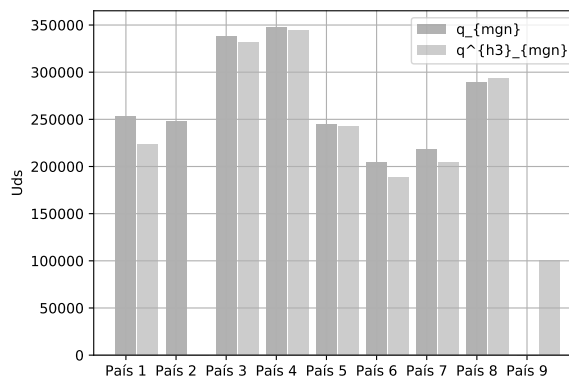


Figura 5.4: Valores de q_{mgn} frente a $q_{mgn}^{h_3}$.

Para asegurarse si el nuevo proceso se alinea con la decisión del modelo actual, ambos pronósticos se trazan en un ajuste bivariable, para el clúster 3. Como se puede comprobar en la Figura 5.5, la variable $q_{mgn}^{h_3}$ obtenida por el Algoritmo 13, con las demandas calculadas por el Algoritmo 11, se ajusta bastante bien al proceso actual, excepto en ciertos puntos, como ya se podía visualizar en la Figura 5.4.

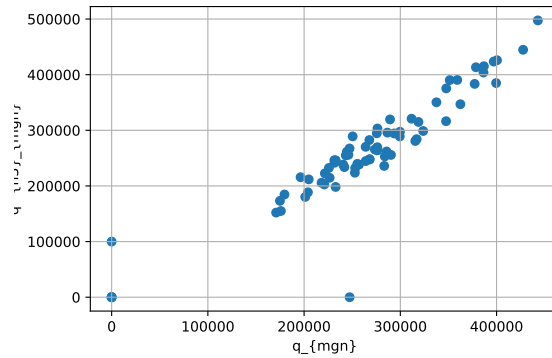


Figura 5.5: Ajuste bivariable de q_{mgn} frente a $q_{mgn}^{h_3}$.

En la Tabla se desagregan los términos de la Ecuación 5.8 el procedimiento original y empleando los algoritmos 11 y 13.

	Proceso original	Heurística
<i>Ventas_FdT</i>	$4.06224 \cdot 10^9$	$4.06224 \cdot 10^9$
<i>Ventas_R</i>	$1.02932 \cdot 10^{10}$	$1.02932 \cdot 10^{10}$
<i>Ventas_S</i>	$1.75894 \cdot 10^{13}$	$1.75893 \cdot 10^{13}$

Tabla 5.2: Parámetros del Modelo I.

Se calculan las ganancias, descritas en la Ecuación (5.8), empleando los algoritmos 11 y 13, obteniendo un valor de $1.76037 \cdot 10^{13}$ euros. Mientras que por el procedimiento original se obtenían ganancias de $1.76038 \cdot 10^{13}$ euros. La diferencia es de 10^8 euros, igual a la obtenida por el primer procedimiento. Los algoritmos 11 y 13, también parecen ser adecuados para comparar en secciones posteriores los costes de envío, dados en la función objetivo del Modelo II, con los obtenidos por el procedimiento actual.

5.5. Tiempo de Ejecución

En esta sección se comparan los tiempos de ejecución del Modelo I con las tres heurísticas que se han descrito en este capítulo. Para ello se importa el módulo `time` de Python. En la Tabla 5.3 aparecen los tiempos de ejecución de cada uno de los algoritmos.

Algoritmo	Tiempo
Modelo I	55.06 seg
Algoritmo 11	0.07 seg
Algoritmo 12	0.03 seg
Algoritmo 13	0.18 seg

Tabla 5.3: Tiempos de ejecución.

El tiempo de ejecución del procedimiento propuesto en la Sección 5.2, que es la suma de los tiempos de los algoritmos 11 y 12, es de 0.1 seg. Mientras que el tiempo de ejecución del procedimiento descrito en la Sección 5.3, que es la suma de los tiempos de los algoritmos 11 y 13, es de 0.25 seg. Se ha reducido el tiempo de ejecución considerablemente, respecto al procedimiento actual, por tanto, los dos procesos proporcionan tiempos muy buenos. Solo queda probar que el valor de la función objetivo del Modelo II, con los datos de entrada calculados por las heurísticas sea similar a la obtenida con el modelo actual.

5.6. Comparación de Resultados

Por último, en la Tabla 5.4, se comparan los valores obtenidos de la función objetivo del Modelo II, si se considera como dato de entrada el parámetro q_{mgn} original, el parámetro obtenido por el Algoritmo 12, $q_{mgn}^{h_2}$, y el obtenido por el Algoritmo 13, $q_{mgn}^{h_3}$.

En la Figura 5.6 se compara para los cinco primeros países, el valor que se obtiene para la función objetivo del Modelo II, empleando los tres procedimientos. Se observa que los costes de envío que se obtienen si se emplean los algoritmos 11 y 12 son mayores que los costes del proceso original. Lo contrario que ocurre para el caso de los algoritmos 11 y 13, que se obtienen valores menores de la función objetivo. Esto debe ser debido a que, a pesar de que aproxima bastante bien, el algoritmo no ajusta bien algunos valores de los envíos, y por ello se obtienen costes de envío inferiores a los del Modelo I. Además, no tendría sentido que el empleo de una heurística reduciese de tal forma los costes, frente a los obtenidos a partir de un modelo de optimización, que es un procedimiento más complejo.

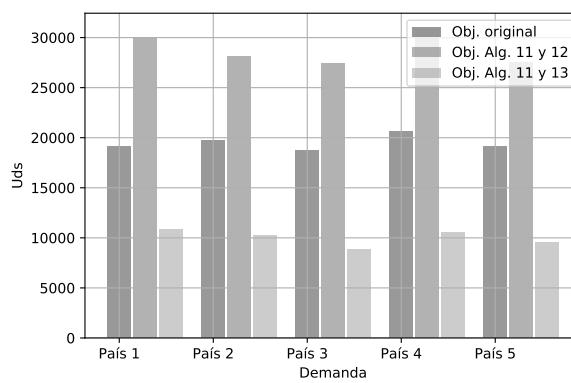


Figura 5.6: Función objetivo para los cinco primeros países.

Para corroborar lo visto en la Figura 5.6, en la Tabla 5.5 se recoge la media y desviación típica de la función objetivo del Modelo II que proporcionan los tres procedimientos vistos. Se observa que la media varía en un porcentaje en torno al 60% y un 81% del valor obtenido por el modelo actual, para el primer y segundo proceso respectivamente.

5.7. Conclusiones

El estudio del caso de la empresa Zara en este proyecto ha consistido en el intento de mejorar el tiempo computacional del modelo empleado por Zara en la actualidad, teniendo en cuenta las reducciones de precios. Para ello se han probado dos modificaciones, la primera de ellas ha sido la sustitución del Modelo I por dos heurísticas que proporcionan buenas aproximaciones para los envíos desde los centros de distribución a los países y para las demandas. Además reducen considerablemente el tiempo computacional. Sin embargo, el valor de la función objetivo del Modelo II, introduciendo como parámetros de entrada los obtenidos con la heurística, no es cercano al que se obtiene con el proceso actual.

Una segunda modificación que se ha probado, ha consistido en el empleo de la primera heurística para la obtención de las demandas y así considerarlas como parámetros de entrada de una nueva heurística que devuelva el valor de los envíos. Al igual que ocurría con el primer procedimiento descrito, se obtienen buenas aproximaciones para los envíos desde los centros de distribución a los países y se reduce de forma considerable el tiempo computacional. Sin embargo, el valor de la función objetivo del Modelo II se aleja considerablemente del obtenido con el modelo original.

Clúster	Valor original	Alg. 11 y 12	Alg. 11 y 13
1	19109.47450	30041.34726	10863.34790
2	19733.38903	28086.21267	10213.75140
3	18739.95173	27451.02799	8866.43438
4	20579.64195	30887.67252	10500.11980
5	19137.31426	27493.61872	9544.83409
6	14310.75241	26828.75240	7900.70110
7	14494.44497	27596.93983	9534.44519
8	15591.89666	29658.69024	10801.41679
9	15697.85979	27077.03846	8256.44082
10	15853.83069	26576.67840	9552.56404
11	18812.48669	29280.15103	11440.18994
12	18228.72903	29005.64345	10560.35530
13	19047.31591	27791.45757	9089.46468
14	17380.81884	26160.36631	8602.58427
15	18618.78385	27234.55575	9695.23923
16	16580.59794	24654.49984	7050.21443
17	17724.84572	27704.57537	9423.65676
18	19375.60749	28083.70514	9219.99014
19	19886.53855	30218.95635	10802.88433
20	17393.93810	27290.52337	8380.05093
21	17673.47052	29766.73159	10398.05835
22	14406.00572	23739.30889	6185.17592
23	18298.07974	29091.56497	9460.52556
24	16058.31690	25684.76106	7149.47276
25	17901.68701	26779.51899	9190.79740
26	17769.37528	28358.23057	9184.30583
27	18648.98400	28630.52832	8210.19632
28	19893.19606	29784.24365	11551.80049
29	18003.78560	28445.59508	9364.86645
30	19599.02864	26057.48887	7916.75504
31	14530.84499	24650.32188	7475.48781
32	14408.49956	26814.85886	8186.05303
33	14847.75661	25082.32727	7503.03813
34	17477.75863	30053.91645	11676.85807
35	18483.26139	30958.26041	11333.02187
36	14142.11974	25821.69440	8215.72712
37	16604.59663	26615.22238	9317.06001
38	17270.03377	26754.21512	8687.04927
39	17111.08760	27221.60857	9359.72210
40	18691.30047	26298.43904	8447.35629
41	13663.90471	24894.80654	7112.61357
42	16215.01873	29028.74048	11097.23588
43	18746.69152	28855.61650	9556.36666
44	19635.74783	30356.71047	11512.32484
45	16328.36179	27269.39020	7788.80352
46	14582.78102	26058.98637	8120.56814
47	15956.81107	25778.24050	8163.81200
48	17197.84261	29359.06216	10399.19710
49	16012.88620	30434.02269	11803.47790
50	14557.04501	27370.63004	8584.66043

Tabla 5.4: Valores de la función objetivo del Modelo II.

Momentos	Obj. Original	Obj. Alg. 11 y 12	Obj. Alg. 11 y 13
Media	17220	27702	9265
Desv. Típica	1856	1768	1377

Tabla 5.5: Momentos del error en la función objetivo del Modelo II.

Apéndice A

Modelo de Predicción de la Demanda

A.1. Construcción del conjunto de datos de demanda

Se definen los siguientes índices y conjuntos, que serán imprescindibles para enunciar el modelo más adelante:

- $m \in \mathcal{M}$: Cada país dentro de la red de distribución.
- $g \in \mathcal{G}$: Cada grupo de ventas de rebajas.
- $r \in \mathcal{R}$: Cada artículo dentro de un determinado grupo de ventas g .
- $j \in \mathcal{J}$: Cada tienda dentro de un determinado país m .
- $s \in \mathcal{S}(r)$: Combinaciones de tamaño-color disponibles para cada artículo $r \in \mathcal{R}$. Un SKU (artículo completamente especificado) corresponde entonces a un par $(r, s) \in \mathcal{R} \times \mathcal{S}(r)$, que para simplificar se escribirá rs .
- $w \in \mathcal{W}$: Período de rebajas que suele ser de una semana (en inglés *week*). Por convención, se escribe $w = 0$ y $w = w_a$ para denotar el primer período y el período actual de ventas de liquidación respectivamente, y se escribe $w < 0$ para representar las semanas durante la Temporada Regular de ventas, es decir, antes de las ventas de rebajas.
- $d \in \mathcal{D}(w)$: Día en el período w . Además, $w(d)$ denotará la semana en la que cae el día d .
- $t \in \mathcal{T}$: Denota un día de la semana. Siendo $T(d) \in \mathcal{T} := \{Lun, \dots, Dom\}$ el día de la semana para d .
- $k \in \mathcal{K} = \{k : k \geq 1\}$: Los precios de rebajas están indexados por k . Donde $\mathcal{P}(m)$ denota el conjunto de precios de rebajas que aumenta por convenio, es decir, $p_{mg1} \leq p_{mg2} \leq \dots \leq p_{mgk}$, donde p_{g1} , es el precio de la liquidación final del grupo g en el país m .
- $Ventas_{mr}^w := \sum_{s \in \mathcal{S}(r), j \in \mathcal{J}, d \in \mathcal{D}(w)} Ventas_{rsj}^d$ son las ventas del artículo r , en el período w , en el país m agregadas según tallas, colores y tiendas. Donde $Ventas_{rsj}^d$ son las ventas para un SKU rs en la tienda j , en el día d .
- $I_{mr}^w := \sum_{s \in \mathcal{S}(r), j \in \mathcal{J}} I_{rsj}^w$: Posición de inventario del artículo r disponible en todo el país m al principio del período w , donde I_{rsj}^d es la posición de inventario del SKU rs en la tienda j al principio del día d .

El cálculo exacto de la tasa de demanda λ_{mr}^w presenta ciertas dificultades, ya que puede diferir de las ventas debido a la censura de la demanda, causada por el agotamiento de existencias, y los efectos estacionales que afectan a la demanda, pero que no están relacionados con los cambios de precios. Por lo

tanto, el primer paso es desestacionalizar los datos, utilizando factores de estacionalidad diarios definidos como:

$$\delta_d := \frac{\sum_{r \in \mathcal{R}} \text{Ventas}_{mr}^{w(d)}}{\overline{\text{Ventas}}} \cdot \gamma \frac{T_{t(d)}}{\sum_{t \in \mathcal{T}} T_{t(d)}}, \quad \forall d \in \mathcal{D} \quad (\text{A.1})$$

La Ecuación (A.1) define los factores de desestacionalización δ_d . En donde:

- $\overline{\text{Ventas}}$: Promedio de las ventas en el país dado.
- $T_t := \frac{\sum_{d \in \mathcal{D}, s \in \mathcal{S}(r), j \in \mathcal{J}} 1_{\{t(d)=t\}} \text{Ventas}_{rsj}^d}{\sum_{d \in \mathcal{D}} 1_{\{t(d)=t\}}}$: Promedio de ventas para un día t .
- 1_E : Función indicadora asociada al evento E .

En otras palabras, el factor δ_d captura las variaciones de ventas diarias esperadas para Zara a nivel de país ¹. Aunque estos componentes deben actualizarse cada temporada, se ha observado que se mantienen bastante constantes a lo largo de los años (ver Carboni 2009), lo que demuestra la robustez de la fórmula y valida su uso.

La corrección de la censura de la demanda es más complicada, incluso para un solo SKU (ver Besbes y Muharremoglu 2010). Además, debido a la política de inventario de Zara, un artículo r suele dejar de mostrarse cuando faltan algunos de sus tamaños o colores clave, incluso si se dispone de existencias para otros tamaños/colores, lo que agrava el efecto de censura. Para capturar esto, se recurre de nuevo a Caro y Gallien (2010) y se define el indicador $1_{DNE_{rsj}^d}$ que es igual a uno si el SKU rs no se muestra en la tienda j el día d . Aquí DNE_{rsj}^d significa Días de No Exposición, y formalmente el evento se define como:

$$DNE_{rsj}^d := \{\{I_{rsj}^d = 0\} \text{ o } \{\min_{\bar{s} \in \mathcal{S}^+(r)} I_{r\bar{s}j}^d = 0 \text{ y } \max_{\bar{s} \in \mathcal{S}(r)} \text{Ventas}_{r\bar{s}j}^d = 0\}\} \quad (\text{A.2})$$

donde $\mathcal{S}^+(r)$ dispone de los tamaños/colores clave para el artículo r . Es decir, la Ecuación (A.2) indica que no hay existencias en inventario para el SKU rs , o bien, que no hay existencias en los tamaños/colores clave del mismo. Por último, se estima la tasa de demanda para el artículo r , en el país m y período w como:

$$\lambda_{mr}^w = \frac{\text{Ventas}_{mr}^w}{\text{Tiempo}_{mr}^w}, \quad (\text{A.3})$$

siendo:

$$\text{Tiempo}_{mr}^w := \sum_{s \in \mathcal{S}(r), j \in \mathcal{J}} \alpha_{rsj} \cdot \sum_{d \in \mathcal{D}(w)} \delta_d (1 - 1_{DNE_{rsj}^d}) \quad (\text{A.4})$$

En la Ecuación (A.4) se tiene:

- $\alpha_{rsj} := \sum_{w < 0} \text{Ventas}_{rsj}^w / \sum_{w < 0} \text{Ventas}_r^w$: Parámetros de desagregación, que indican el peso relativo de las ventas de la tienda j para los SKU rs .
- $\text{Ventas}_{rsj}^w := \sum_{d \in \mathcal{D}(w)} \text{Ventas}_{rsj}^d$: Ventas de un SKU rs en la tienda j en el período w .

Nótese que la Ecuación (A.4) es un promedio ponderado de los “días en exhibición” desestacionalizados en el período w . Por lo tanto, la relación en la Ecuación (A.3) corresponde a una tasa de demanda normalizada. Nótese también que Zara reasigna con frecuencia el inventario antes de las ventas de liquidación. Cuando eso pasa, se usa:

$$\tilde{\alpha}_{rsj} = \frac{\sum_{w < 0, j \in \mathcal{J}} \text{Ventas}_{rsj}^w}{\sum_{w < 0} \text{Ventas}_r^w} \frac{\sum_{w < 0, r \in \mathcal{R}, s \in \mathcal{S}(r)} \text{Ventas}_{rsj}^w}{\sum_{w < 0, r \in \mathcal{R}} \text{Ventas}_r^w}, \quad (\text{A.5})$$

para desagregar la tasa de demanda a nivel de SKU y tienda. En otro caso, se usaría $\tilde{\alpha}_{rsj} = \alpha_{rsj}$.

¹Hay dos tipos de estacionalidad inherentes a las tendencias de ventas de Zara: la estacionalidad intra-semanal e inter-semanal. La estacionalidad intra-semanal se refiere a la estacionalidad observada de un día de la semana a otro, por ejemplo, las ventas un viernes tienden a ser mayores que las ventas un lunes. La estacionalidad inter-semanal mide los diferentes patrones de ventas de una semana de la temporada regular a la otra, y da cuenta de las tendencias de ventas regulares dentro del año y otros eventos especiales como los días festivos nacionales, que pueden afectar a las ventas observadas.

A.2. Forma del Modelo de Predicción

El principal inconveniente a la hora de predecir la demanda en periodos futuros λ_{mr}^w es que solo se disponen de datos de temporada regular para ajustar los parámetros del Modelo de Predicción, ya que son los únicos datos disponibles en el momento en que se ejecuta el modelo antes del inicio del período de rebajas. Para superar esto, es necesario un procedimiento de estimación en dos etapas, que se describirá brevemente. El proceso recién descrito resultó en la selección de la siguiente forma funcional:

$$\begin{aligned}\lambda_{mr}^w &= F(C_r, E_{mr}^w, \lambda_{mr}^w, I_{mr}^w, p_{mr}^w) \\ &= \exp(\beta_{0mr} + \beta_1 \ln(C_r) + \beta_2 E_{mr}^w + \beta_3 \ln(\lambda_{mr}^w) + \beta_4 \ln(\min(1, \frac{I_{mr}^w}{f \cdot C_r \cdot S_{mr}})) + \beta_5 \ln(\frac{p_{mr}^w}{p_{mr}^T}))\end{aligned}\quad (\text{A.6})$$

donde la variable dependiente es la tasa de demanda del artículo r , en el periodo w y país m . Los regresores son:

- **Cantidad de compra** (C_r): Tamaño de la compra realizada para un artículo r (en unidades). Se selecciona esta variable por su correlación con el componente “moda” de un artículo. Normalmente, los artículos de bajo contenido de moda (también conocidos como “básicos”) se compran en grandes cantidades, mientras que los artículos más de moda se compran deliberadamente en pequeñas cantidades, y debido a que la compra de cada artículo cubre toda la red de tiendas de Zara, esta variable es la misma en todos los países. Existe una correlación positiva entre esta variable y el nivel de demanda observado.
- **Edad de un artículo** (E_{mr}^w): Número de días desde que el artículo fue introducido en las tiendas. La selección de esta variable es intuitiva porque las ventas típicamente alcanzan su punto máximo poco después de la introducción de un producto y luego disminuyen gradualmente a medida que pasan las semanas. Esta variable puede ser dependiente del país.
- **Demanda del período anterior** (λ_{mr}^w): La tasa de demanda se correlaciona automáticamente y no se esperan cambios bruscos en el nivel de demanda de una referencia de una semana a otra, por lo que se incluye como una variable en el modelo con un desfase de un período, es decir, el pronóstico para el nivel de demanda de la presente semana es una función del nivel de demanda medido en la semana anterior.
- **Efecto de surtido roto** ($\min\{1, \frac{I_{mr}^w}{f \cdot C_r \cdot S_{mr}}\}$): En la venta al por menor se ha documentado bien que la tasa de demanda de un artículo suele disminuir cuando el inventario desciende por debajo de un cierto nivel. Este hecho se conoce como el efecto del surtido roto y es especialmente prevalente en las prendas de vestir ya que, cuando el inventario es bajo, los artículos restantes son generalmente aquellos que son menos atractivos para los clientes (ver Smith y Achabal 1998 y Sección A-3 en Fisher y Raman 2010). Para incorporar esto en nuestro modelo, se define un umbral obtenido del producto de:
 - C_r : el número de colores en los que se ofrece una referencia.
 - S_{mr} : el número de tallas en las que se ofrece.
 - f : un escalar que se ajusta individualmente para cada país y grupo de rebajas.

De esta manera, los niveles de Posición de Inventario, en cuanto a colores y tallas para una referencia dada, que están por debajo de este umbral son capturados por esta variable. Aunque este parámetro se puede definir para un almacén individual, se calibra para todo el país con el fin de mantener la regresión agregada a ese nivel.

- **Descuento de precio** ($\frac{p_{mr}^w}{p_{mr}^T}$): El precio es obviamente un motor de ventas clave en el período de liquidación. La selección de esta variable específica refleja, sin embargo, que los clientes son más sensibles a la reducción de precio que a la reducción de precio absoluta. De hecho, una práctica habitual en Zara y otros minoristas del sector de la confección es hacer publicidad de rebajas específicas (expresadas en porcentajes negativos) utilizando carteles colocados en diversas zonas

de la tienda. Además, Zara muestra deliberadamente el precio actual p_{mr}^w junto con el precio de temporada regular p_{mr}^T en la etiqueta del artículo, de modo que los clientes saben inmediatamente lo grande que es el descuento. El regresor seleccionado captura estas características como la relación entre estos dos precios.

A.3. Método Residual para la Estimación de Parámetros

Los parámetros $\beta_{0mr}, \beta_1, \dots, \beta_5$, en la Ecuación (A.6) son los coeficientes de la regresión. En particular, β_5 representa la elasticidad del precio, que en este modelo se considera constante e idéntica para todos los artículos del grupo de productos dado para cada país. Para linealizar el modelo de regresión, se toman logaritmos en la Ecuación (A.6). Nótese que el término de error en el modelo linealizado se convierte en un factor de error multiplicativo en el modelo original, esta transformación aparentemente inocua tiene consecuencias importantes cuando se realiza el pronóstico, como se verá más adelante.

Una observación importante es que al aplicar la ecuación (A.6) para predecir la demanda durante las ventas de rebajas, las frecuencias deseables en las que deben actualizarse los coeficientes de la regresión varían según los diferentes regresores. Específicamente, mientras que $\beta_{0mr}, \beta_1, \beta_2, \beta_3$ pueden ser estimados una vez usando datos de la temporada regular, es deseable actualizar la estimación de β_4 y β_5 con más frecuencia, en parte porque inicialmente hay muy pocos datos disponibles para la mayoría de los artículos de la temporada actual. Esto motiva un procedimiento de estimación en dos etapas: en la Etapa 1 los coeficientes para algunos regresores se estiman con datos estacionales regulares, mientras que en la Etapa 2 se estiman los coeficientes de las otras variables explicativas, tales como la elasticidad de precio, y se actualizan periódicamente a medida que se dispone de una base de datos de ventas de liquidación.

Más específicamente, en la Etapa 1 se usan datos de la temporada de ventas regulares ($w < 0$) y se ejecuta la regresión:

$$\ln(\lambda_{mr}^w) = \beta_{0mr} + \beta_1 \ln(C_r) + \beta_2 E_{mr}^w + \beta_3 \ln(\lambda_{mr}^w) + \beta_4 \ln(\min\{1, \frac{I_{mr}^w}{f \cdot C_r \cdot S_{mr}}\}) + u_{mr}^w, \quad \forall r \in R, m \in \mathcal{M}, w < 0 \quad (\text{A.7})$$

con el término de error u_{mr}^w , de donde se obtiene el conjunto de parámetros $\tilde{\beta}_{0mr}, \tilde{\beta}_1, \tilde{\beta}_2, \tilde{\beta}_3$. En la Etapa 2, se utilizan datos de ventas de liquidación ($w \geq 0$) para calcular los residuos:

$$\phi_r^w = \ln(\lambda_{mr}^w) - \tilde{\beta}_{0mr} - \tilde{\beta}_1 \ln(C_r) - \tilde{\beta}_2 E_{mr}^w - \tilde{\beta}_3 \ln(\lambda_{mr}^w), \quad \forall r \in R, m \in \mathcal{M}, w \geq 0 \quad (\text{A.8})$$

Finalmente se realiza una regresión de los residuos con respecto al efecto de surtido roto y las reducciones de precio:

$$\phi_{mr}^w = \beta_4 \ln(\min\{1, \frac{I_{mr}^w}{f \cdot C_r \cdot S_{mr}}\}) + \beta_5^w \ln(\frac{p_{mr}^w}{p_{mr}^T}) + \epsilon_{mr}^w, \quad \forall r \in R, m \in \mathcal{M}, w \geq 0 \quad (\text{A.9})$$

donde ϵ_{mr}^w el término de error de la regresión. Esto nos proporciona los parámetros estimados $\hat{\beta}_4$ y $\hat{\beta}_5$. Al igual que en Smith et al (1994), este procedimiento de dos etapas se basa en el supuesto pragmático de que los efectos en toda la estación, es decir, los estimados en la primera etapa, son más estables que los parámetros clave que se actualizan en la segunda etapa. Por lo tanto, en la Ecuación (A.8) se eliminan los efectos de los regresores no actualizados de la Etapa 1 para obtener los residuos que contienen sólo los efectos de los regresores actualizados. Se incluye β_4^t en la regresión de la Etapa 1 para mejorar el ajuste, pero no se tiene en cuenta en la Ecuación (A.8) ya que el efecto de surtido roto es particularmente relevante durante las ventas de liquidación y, por lo tanto, el parámetro debe actualizarse.

A.4. Procedimiento para la estimación de Parámetros

En primer lugar, se ejecuta el proceso agrupando los datos de temporadas anteriores, lo que genera el conjunto de parámetros estimados $\{\tilde{\beta}_4^{P,w}, \tilde{\beta}_5^{P,w}\}_{w \geq 0}$, donde se utiliza el superíndice P para indicar que proviene de datos de temporadas pasadas. Por supuesto, a partir de la Etapa 1 también se obtienen el

conjunto de parámetros $\tilde{\beta}_{0mr}^P, \tilde{\beta}_1^P, \tilde{\beta}_2^P, \tilde{\beta}_3^P$, pero estos sólo juegan un papel en el cálculo de los residuos.

El procedimiento de dos etapas se ejecuta de nuevo utilizando los datos de la temporada actual hasta el período más reciente disponible ($w = w_a - 1$). A partir de la Etapa 1 se obtienen el conjunto de parámetros estimados $\tilde{\beta}_{0r}^A, \tilde{\beta}_1^A, \tilde{\beta}_2^A, \tilde{\beta}_3^A$ y a partir de la Etapa 2, $\{\tilde{\beta}_4^{A,w}, \tilde{\beta}_5^{A,w}\}_{0 \leq w < w_a}$, donde se utiliza el superíndice A para denotar que proviene de datos de la temporada actual. Finalmente, con el fin de incorporar este conocimiento en las previsiones que se efectúen para períodos posteriores, se utiliza un método de ponderación o suavizado exponencial, en el que los parámetros $\hat{\beta}_4^w$ y $\hat{\beta}_5^w$, utilizados para la predicción para el período actual, se basan en la suma ponderada del parámetro estimado para el período anterior con datos históricos, y el parámetro medido para el período anterior con datos de la temporada actual:

$$\hat{\beta}_4^w = \omega \bar{\beta}_4^{P,w-1} + (1 - \omega) \hat{\beta}_4^{A,w-1} \quad (\text{A.10})$$

$$\hat{\beta}_5^w = \omega \bar{\beta}_5^{P,w-1} + (1 - \omega) \hat{\beta}_5^{A,w-1} \quad (\text{A.11})$$

El parámetro de ponderación ω es un número entre cero y uno. El valor de este parámetro se establece para cada país y grupo de rebajas basándose en datos históricos, de manera que se minimiza la diferencia cuadrada entre los parámetros predichos y los reales medidos. Este valor suele situarse en torno a 0.8, lo que significa que la mayor parte del peso se coloca en el valor del parámetro medido durante el período de rebajas anterior.

Con los parámetros $\bar{\beta}_4^w$ y $\bar{\beta}_5^w$ obtenidos de las ecuaciones (A.10) y (A.11), se puede obtener la predicción de la tasa de demanda para el período actual ($w = w_a$) de acuerdo a la fórmula ²:

$$\tilde{\lambda}_{mr}^w \approx \exp(\tilde{\beta}_{0mr}^A + \tilde{\beta}_1^A \ln(C_r) + \tilde{\beta}_2^A E_{mr}^w + \tilde{\beta}_3^A \ln(\lambda_{mr}^w) + \hat{\beta}_4^w \ln(\min(1, \frac{I_{mr}^w}{f \cdot C_r \cdot S_{mr}}))) + \hat{\beta}_5^w \ln(\frac{p_{mr}^w}{p_{mr}^T})) \quad (\text{A.12})$$

Una vez que se ha ajustado un modelo para cada grupo de rebajas y cada país, se pueden realizar previsiones para cada referencia introduciendo las variables para cada una de ellas. Sin embargo, para hacer las predicciones en períodos de rebajas futuros, el precio actual al que se venderá la referencia durante ese período es desconocido. Por esta razón, se debe hacer una predicción del nivel de demanda λ_{mrk} para cada referencia r y posible precio k en la lista de precios de rebajas.

Sin embargo, una vez que se hacen las predicciones del nivel de demanda, es necesario convertirlas en unidades reales de ventas. Para poder convertir las predicciones del nivel de demanda en cifras de ventas, se necesita: determinar la duración del período de rebajas para el cual se realizará la previsión, y considerar el inventario existente para cada referencia y su distribución en color, tamaño y tienda. Para ello se siguen los siguientes pasos:

- (1) **Eliminación de estacionalidad intra-semanal:** Se multiplica la tasa de demanda λ_{mrk} por los factores de estacionalidad diarios δ_d , calculados en la Ecuación (A.1). En la práctica, sin embargo, debido a que las listas de precios se generan y envían semanalmente a las tiendas, esta suma se suele simplificar y se sustituye por una constante de siete días.
- (2) **Desagregación de la tasa de demanda nivel de SKU y de tienda:** Este producto se multiplica por los parámetros de desagregación de la demanda α_{rsj} o $\tilde{\alpha}_{rsj}$, calculados en la Sección 3.1.
- (3) **Conversión en ventas unitarias:** Finalmente, se utiliza la distribución Gamma para convertir este cálculo en ventas unitarias, que denotamos por D_{mrk}^w , y se define como la demanda estimada en unidades para una referencia r , en un período w , en el país m y a precio de liquidación k ³. Esta estimación es la entrada para el modelo de optimización, que se desarrollará en el Capítulo 4.

²La ecuación (A.12) ignora el hecho de que la expectativa del término error de regresión (ϵ_{mr}^w) suele ser mayor que uno cuando la transformación logarítmica es inversa. Por lo tanto, se suele multiplicar la estimación dada en (A.12) por un factor de corrección. Generalmente se utiliza el factor de medición H^w introducido por Duan (1983).

³Para referirse a las ventas unitarias al final de la temporada durante la temporada regular ($w < 0$), se emplea la notación D_{mr} , ya que no depende del precio p_{mgk} .

Apéndice B

Código en Python: Bmgn

```
1 import numpy as np
2 import scipy.stats
3 from Modelo_datos import GRUPO, CLUSTER, setCLUSTER, PAIS, DATOS_CD, Dmgn, Imgn, Iagn
4 np.random.seed(3)
5
6 PAIS_gn = {}
7 for g in GRUPO:
8     for n in CLUSTER[g]:
9         PAIS_gn[g,n] = list(range(1,74))
10
11 ## PASO 1: Cálculo de IE_mgn:
12 IE_mgn = {}
13 for m in PAIS:
14     for g in GRUPO:
15         for n in CLUSTER[g]:
16             IE_mgn[m,g,n] = Dmgn[m,g,n]/(Imgn[m,g,n] + Dmgn[m,g,n])
17
18 ## PASO 2: Eliminar países con IE bajo:
19 media = np.mean(list(IE_mgn.values()))
20 sd = np.std(list(IE_mgn.values()))
21 Bmgn = {}
22 for m in PAIS:
23     for g in GRUPO:
24         for n in CLUSTER[g]:
25             if IE_mgn[m,g,n] < (media - 1.5*sd):
26                 Bmgn[m,g,n] = 0
27                 PAIS_gn[g,n].remove(m)
28
29 ## PASO 3: Construcción de parámetros vmgn y Bmgn:
30 vmgn = {}
31 for g in GRUPO:
32     for n in CLUSTER[g]:
33         for m in PAIS_gn[g,n]:
34             vmgn[m,g,n] = Dmgn[m,g,n]/sum(Dmgn.values())
35             Bmgn[m,g,n] = sum(Iagn[a,g,n] for a in DATOS_CD)*vmgn[m,g,n]
36
37 ## PASO 4: Recalcular Bmgn:
38 mu = 10
39 sigma = 3
40 N = len(PAIS)*len(setCLUSTER)
41 lower = 0
42 upper = 20000
43
44 Xmgn = {}
45 for g in GRUPO:
46     for n in CLUSTER[g]:
47         for m in PAIS_gn[g,n]:
48             Xmgn[m,g,n] = scipy.stats.truncnorm.rvs((lower-mu)/sigma,
49                                                     (upper-mu)/sigma, loc=mu, scale=sigma)
50
51 for g in GRUPO:
52     for n in CLUSTER[g]:
53         for m in PAIS_gn[g,n]:
```

```
53         if Bmgn[m, g, n] < Xmgn[m, g, n]:
54             Bmgn[m, g, n] = 0
55             PAIS_gn[g, n].remove(m)
vmgn = {}
57 for g in GRUPO:
58     for n in CLUSTER[g]:
59         for m in PAIS_gn[g, n]:
60             vmgn[m, g, n] = Dmgn[m, g, n]/sum(Dmgn.values())
61             Bmgn[m, g, n] = sum(Iagn[a, g, n] for a in DATOS_CD)*vmgn[m, g, n]
```

Apéndice C

Código en Python: Modelo I

Una vez enunciado, se presenta el código en Python del Modelo I, disponible en el módulo `ModeloI.py`. Se ha resuelto empleando el solver lineal Gurobi.

```
1 from gurobipy import *
2 import sys
3 from matplotlib import pyplot as plt
4 Bmgn = eval(open("Bmgn.txt").read())
5
6 def solve(GRUPO, CLUSTER, setCLUSTER, PAIS, PAIS_CD, DATOS_CD, setPRECIOS, PRECIOS, Pmgn,
7 psaldero, Iagn, Imgn, Drj, Djgn, Dmgn, D_mgn, CE, CA, D, Wr, SA, ST, Qm, desc_min,
8 tiempo, holgura, n_precios):
9
10     # Modelo:
11     modeloI = Model("ModeloI")
12
13     # -----
14     # VARIABLES DE DECISIÓN:
15
16     # Flujo de inventario desde CD al país m, cluster n y grupo g:
17     q_mgn = modeloI.addVars(PAIS, [(g, CLUSTER[g][i]) for g in GRUPO for i in range(0, len(
18 CLUSTER[g]))], lb=0, name="q_mgn")
19
20     # Inventario en cada país al principio de la temporada de rebajas:
21     L_mgn = modeloI.addVars(list(range(0, Wr+1)), PAIS, [(g, CLUSTER[g][i]) for g in GRUPO
22 for i in range(0, len(CLUSTER[g]))], lb=0, name="L_mgn")
23
24     # Ventas esperadas para la temporada regular:
25     d_mgn = modeloI.addVars(PAIS, [(g, CLUSTER[g][i]) for g in GRUPO for i in range(0, len(
26 CLUSTER[g]))], lb=0, name="d_mgn")
27
28     # Ventas esperadas durante las ventas de rebajas en el país m, clúster n y grupo g:
29     d_l_mgn = modeloI.addVars(list(range(0, Wr)), PAIS, [(g, CLUSTER[g][i]) for g in GRUPO for
30 i in range(0, len(CLUSTER[g]))], setPRECIOS, lb=0, name="d_l_mgn")
31
32     # Flujo desde un CD a otro CD para el clúster n en el grupo g:
33     q_agm = modeloI.addVars(DATOS_CD, DATOS_CD, [(g, CLUSTER[g][i]) for g in GRUPO for i in
34 range(0, len(CLUSTER[g]))], lb=0, name="q_agm")
35
36     # Seleccionar precios mayores o iguales que el precio de decisión:
37     x = modeloI.addVars(list(range(0, Wr)), PAIS, [(g, CLUSTER[g][i]) for g in GRUPO for i
38 in range(0, len(CLUSTER[g]))], setPRECIOS, lb=0, vtype=GRB.BINARY, name="x")
39
40     # Decidir precio para la venta de rebajas:
41     y = modeloI.addVars(list(range(0, Wr)), PAIS, [(g, CLUSTER[g][i]) for g in GRUPO for i
42 in range(0, len(CLUSTER[g]))], setPRECIOS, lb=0, vtype=GRB.BINARY, name="y")
43
44     # Inventario del país que puede ser recuperado:
45     L_saldero = modeloI.addVars(PAIS, GRUPO, lb=0, name="L_saldero")
46
47     modeloI.update()
```

```

40 # VARIABLES FUNCIÓN OBJETIVO:
41 # -----
42 # Ganancias totales de la temporada regular y rebajas:
VENTAS_FdT = sum(sum(sum(Pmgn[m,g,CLUSTER[g][i]]*dmgn[m,g,CLUSTER[g][i]] for m in
PAIS) for i in range(0,len(CLUSTER[g]))) for g in GRUPO)
44
45 # Beneficios de rebajas totales:
46 VENTAS_R = sum( sum( sum( (sum( sum( d_mgn[w,m,g,CLUSTER[g][i],k] for w in list(range
(0, Wr))) for i in range(0,len(CLUSTER[g]))) *PRECIOS[m,g,k] for g in GRUPO) for m in
PAIS) for k in setPRECIOS)
48
49 # Beneficios totales del inventario recuperado:
VENTAS_S = sum(( sum(L_saldero[m,g] for m in PAIS) + sum( sum( Iagn[cd,g,CLUSTER[g][
i]] for cd in DATOS_CD) for i in range(0,len(CLUSTER[g]))) - sum( sum( q_mgn [m,g,
CLUSTER[g][i]] for m in PAIS) for i in range(0,len(CLUSTER[g]))) *psaldero[g] for g
in GRUPO)
50
51 # Costes totales de envío de los países:
52 COSTES_CE = CE* ( sum( sum( sum( Pmgn[m,g,CLUSTER[g][i]]*q_mgn[m,g,CLUSTER[g][i]] for
m in PAIS) for i in range(0,len(CLUSTER[g]))) for g in GRUPO)
54
55 # Costes totales de envío desde los CD:
COSTES_CDE = CA* ( sum( sum( sum( sum( sum( Pmgn[PAIS_CD[cd][j],g,CLUSTER[g][i]]*
q_agn[cd,cd2,g,CLUSTER[g][i]] for cd2 in (DATOS_CD[:DATOS_CD.index(cd)] + DATOS_CD[
DATOS_CD.index(cd)+1:]) for j in range(0,len(PAIS_CD[cd]))) for cd in DATOS_CD) for
i in range(0,len(CLUSTER[g]))) for g in GRUPO)
56
57 # FUNCIÓN OBJETIVO:
58 # -----
59 modeloI.setObjective(VENTAS_FdT+ VENTAS_R + VENTAS_S - COSTES_CE - COSTES_CDE,
60 GRB.MAXIMIZE)
62
63 ## RESTRICCIONES:
64 ## -----
65 # Restricción de balance de inventario desde los CD:
modeloI.addConstrs((Iagn[cd,g,CLUSTER[g][i]] + sum(q_agn[cd2,cd,g,CLUSTER[g][i]] for
cd2 in (DATOS_CD[:DATOS_CD.index(cd)] + DATOS_CD[DATOS_CD.index(cd)+1:])) >= sum(
q_mgn[PAIS_CD[cd][j],g,CLUSTER[g][i]] for j in range(0,len(PAIS_CD[cd]))) + sum(q_agn[
cd,cd2,g,CLUSTER[g][i]] for cd2 in (DATOS_CD[:DATOS_CD.index(cd)] + DATOS_CD[
DATOS_CD.index(cd)+1:])) for cd in DATOS_CD for g in GRUPO for i in range(0,len(
CLUSTER[g])), "envios_cd")
66
67 # Restricción de balance de inventario en el país:
68 modeloI.addConstrs((L_mgn[0,m,g,CLUSTER[g][i]] == Imgn[m,g,CLUSTER[g][i]] + q_mgn[m,g
,CLUSTER[g][i]] - dmgn[m,g,CLUSTER[g][i]] for m in PAIS for g in GRUPO for i in range
(0,len(CLUSTER[g])), "balance_inv")
70
71 # Restricción del flujo de inventario en rebajas:
modeloI.addConstrs((L_mgn[w+1,m,g,CLUSTER[g][i]] == L_mgn[w,m,g,CLUSTER[g][i]] - sum(
d_mgn[w,m,g,CLUSTER[g][i],k] for k in setPRECIOS) for w in list(range(0, Wr)) for m
in PAIS for g in GRUPO for i in range(0,len(CLUSTER[g])), "inv_estimado")
72
73 # Restricción de las ventas esperadas en la temporada regular:
74 modeloI.addConstrs((dmgn[m,g,CLUSTER[g][i]] <= Dmgn[m,g,CLUSTER[g][i]]
for m in PAIS for g in GRUPO for i in range(0,len(CLUSTER[g])),
75 "ventas_temp_reg")
76
77 # Restricción de las ventas esperadas en la temporada de rebajas:
78 modeloI.addConstrs((d_mgn[w,m,g,CLUSTER[g][i],k] <= D_mgn[m,g,CLUSTER[g][i],k]*y[w,m,
g,CLUSTER[g][i],k]*tiempo[m]**(w-1) for w in list(range(0, Wr)) for m in PAIS for g
in GRUPO for i in range(0,len(CLUSTER[g])) for k in setPRECIOS), "precio_demanda")
80
81 # Restricción de envío máximo:
82 modeloI.addConstrs((q_mgn[m,g,CLUSTER[g][i]] <= Bmgn[m,g,CLUSTER[g][i]] for m in
PAIS for g in GRUPO for i in range(0,len(CLUSTER[g])), "ventas_reb_esp")
84
85 for b in Bmgn.values():
if b < 0:
86 sys.exit("El parámetro Bmgn tiene que ser positivo")

```

```

88 # Restricción de envío mínimo:
    modeloI.addConstrs((sum(sum(q_mgn[m,g,CLUSTER[g][i]] for i in range(0,len(CLUSTER[
90 g]))) for g in GRUPO) >= Qm for m in PAIS), "envio_min")

92 # Restricción de ingresos de liquidación:
    modeloI.addConstrs((sum(sum(sum(L_mgn[Wr,m,g,CLUSTER[g][i]] for m in PAIS) for i
    in range(0,len(CLUSTER[g]))) for g in GRUPO) + sum(sum(sum(Iagn[cd,g,CLUSTER[g][i
    ]] for cd in DATOS_CD) for i in range(0,len(CLUSTER[g]))) for g in GRUPO) - sum(sum(
    sum(q_mgn[m,g,CLUSTER[g][i]] for m in PAIS) for i in range(0,len(CLUSTER[g]))) for
    g in GRUPO) <= (sum(sum(sum(Iagn[cd,g,CLUSTER[g][i]] for cd in DATOS_CD) for i
    in range(0,len(CLUSTER[g]))) for g in GRUPO) + sum(sum(sum(Img[n][m,g,CLUSTER[g][i
    ]] for m in PAIS) for i in range(0,len(CLUSTER[g]))) for g in GRUPO))* ST for m in
    PAIS for g in GRUPO), "exceso_inv_total")

94 # Restricción de liquidación de países::
    modeloI.addConstrs((L_saldero[m,g] <= sum(L_mgn[Wr,m,g,CLUSTER[g][i]] for i in range
    (0,len(CLUSTER[g]))) for m in PAIS for g in GRUPO), "inv_restante")

96 # Restricción de asignación de precios en los clústers:
98 modeloI.addConstrs((x[w,m,g,CLUSTER[g][i],k] <= 1 for w in list(range(0, Wr)) for m
    in PAIS for g in GRUPO for i in range(0,len(CLUSTER[g])) for k in setPRECIOS if
    CLUSTER[g][i]==list(CLUSTER.values())[i-1][i-1]), "clusters_ordenados")

100 modeloI.addConstrs((x[w,m,g,CLUSTER[g][i],k] <= x[w,m,g,CLUSTER[g][i+1],k] for w in
    list(range(0, Wr)) for m in PAIS for g in GRUPO for i in range(0,len(CLUSTER[g])-1)
    for k in setPRECIOS if CLUSTER[g][i]!=list(CLUSTER.values())[i-1][i-1]), "
    clusters_ordenados")

102 # Restricción de precios:
    modeloI.addConstrs((x[w,m,g,CLUSTER[g][i],k] <= x[w+1,m,g,CLUSTER[g][i],k] for w in
    list(range(0, Wr-1)) for m in PAIS for g in GRUPO for i in range(0,len(CLUSTER[g]))
    for k in setPRECIOS), "precios_decrecientes")

104 # Restricción de descuentos de grupos:
106 precios_k={}
    for m in PAIS:
108         for g in GRUPO:
            precios_k[m,g] = float(PRECIOS[m,g,n_precios])
110
    for m,g,i in zip(PAIS,GRUPO, range(0,len(CLUSTER[g]))):
112         if precios_k[m,g] < Pmgn[m,g,CLUSTER[g][i]]*(1-desc_min):
            con_name='{}{}{}{}{}{}{}{}{}{}'.format(' ',m,' ',g, ' ',CLUSTER[g][i], ' ')
114             modeloI.addConstr(x[0,m,g,CLUSTER[g][i],n_precios] == 1,
                "descuento_inicial" + con_name)
116
118 # Restricción estructural:
    modeloI.addConstrs((x[w,m,g,CLUSTER[g][i],setPRECIOS[setPRECIOS.index(k)-1]] <= x[w,m,
    g,CLUSTER[g][i],k] for w in list(range(0, Wr)) for m in PAIS for g in GRUPO for i in
    range(0,len(CLUSTER[g])) for k in setPRECIOS[1:]), "precio_k_o_mayor")

120 # Restricción de descuentos de la primera semana:
    modeloI.addConstrs((sum(y[0,m,g,CLUSTER[g][i],k] for k in setPRECIOS) == 1 for m in
    PAIS for g in GRUPO for i in range(0,len(CLUSTER[g]))), "elegir")
122
124 # Definición de y:
    modeloI.addConstrs((y[w,m,g,CLUSTER[g][i],k] == x[w,m,g,CLUSTER[g][i],k] - x[w,m,g,
    CLUSTER[g][i],setPRECIOS[setPRECIOS.index(k)-1]] for w in list(range(1, Wr)) for m in
    PAIS for g in GRUPO for i in range(0,len(CLUSTER[g])) for k in setPRECIOS[1:]), "y")
126
    modeloI.addConstrs((y[w,m,g,CLUSTER[g][i],k] == x[w,m,g,CLUSTER[g][i],k] for w in
    list(range(0, Wr)) for m in PAIS for g in GRUPO for i in range(0,len(CLUSTER[g])) for
    k in setPRECIOS[1:]), "y")
128
130 # Optimización:
    modeloI.optimize()
132
    if modeloI.status == GRB.Status.OPTIMAL:
        print(modeloI.getVars())
        print('Optimal objective: %g' % modeloI.objVal)

```


Apéndice D

Código en Python: Brj

```
import numpy as np
2 import sys
import itertools
4 from matplotlib import pyplot as plt
np.random.seed(3)
6 from Modelo_datos import TIENDA, GRUPO, CLUSTER, setCLUSTER, PAIS, PAIS_T, T_PAIS, REF,
    DATOS_CD, Dmgn, D_mgn, Wr, Irj, Imgn, Ira, Drj, Djgn
from Bmgn import Bmgn
8
## Importar del Modelo I:
10 q_mgn = eval(open("q_mgn.txt").read())
qmgn = {}
12 i = 0
while True:
14     try:
        m = (list(itertools.chain.from_iterable(itertools.repeat(x, len(setCLUSTER)) for
        x in PAIS))))[i]
16         g = (list(itertools.chain.from_iterable(itertools.repeat(x, 5) for x in GRUPO))*
        len(PAIS))[i]
        n = (setCLUSTER * len(PAIS))[i]
18         qmgn[m,g,n] = q_mgn[i]
        i = i + 1
20     except IndexError:
        break
22
## Cantidad mínima a transportar a cada tienda L_j:
24 ## -----
26 vjgn = {}
for j in TIENDA:
28     for g in GRUPO:
        for n in CLUSTER[g]:
30         vjgn[j,g,n] = Djgn[j,g,n]/ sum(Djgn[j,g,n] for j in T_PAIS[PAIS_T[j]])
32 Lj = {}
for g in GRUPO:
34     for n in CLUSTER[g]:
        for j in TIENDA:
36         Lj[j,g,n] = (Imgn[PAIS_T[j],g,n] + qmgn[PAIS_T[j],g,n])*vjgn[j,g,n]
            - sum(Irj[r,j,g,n] for r in REF)
38
## Cantidad máxima a transportar a cada tienda L_j:
40 ## -----
42 Jr = {}
Brj = {}
44 IE_rj = {}
media = {}
46 sd = {}
Xr = {}
48 H = {}
suma_r = {}
```

```

50 suma = {}
52 for g in GRUPO:
53     for n in CLUSTER[g]:
54
55         ## PASO 1: Fijar Brj a cero si qmgn es cero:
56         for r in REF:
57             Jr[r,g,n] = list(range(1,1201))
58
59         for m in PAIS:
60             if qmgn[m,g,n] == 0:
61                 for r in REF:
62                     for j in T_PAIS[m]:
63                         Brj[r,j,g,n] = 0
64                         Jr[r,g,n].remove(j)
65
66         ## PASO 2: Retirar tiendas IErj bajo:
67         for r in REF:
68             for j in Jr[r,g,n]:
69                 IE_rj[r,j,g,n] = Drj[r,j,g,n]/(I_rj[r,j,g,n] + Drj[r,j,g,n])
70
71         media[g,n] = np.mean(list(IE_rj.values()))
72         sd[g,n] = np.std(list(IE_rj.values()))
73
74         for r in REF:
75             for j in Jr[r,g,n]:
76                 if IE_rj[r,j,g,n] < (media[g,n] - 1.5*sd[g,n]):
77                     Brj[r,j,g,n] = 0
78                     Jr[r,g,n].remove(j)
79
80         ## PASO 3: Requisitos mínimos de envío:
81         for r in REF:
82             Xr[r,g,n] = 1
83             H[r,g,n] = int(sum(Ira[r,a,g,n] for a in DATOS_CD)/Xr[r,g,n])
84             if H[r,g,n] < len(Jr[r,g,n]):
85                 for i in range(0, len(Jr[r,g,n])-H[r,g,n]-1):
86                     Brj[r,sorted(Jr[r,g,n],g,n)[i]] = 0
87                     Jr[r,g,n].remove(sorted(Jr[r,g,n])[i])
88
89         ## PASO 4: Cálculo de Brj
90         for r in REF:
91             suma_r[r,g,n] = sum(abs(Lj[j,g,n]) for j in Jr[r,g,n])
92
93         for r in REF:
94             for j in Jr[r,g,n]:
95                 Brj[r,j,g,n] = abs(Lj[j,g,n])/suma_r[r,g,n]*sum(Ira[r,a,g,n]
96                     for a in DATOS_CD)
97
98         ## PASO 5: Asegurar que se cumpla requisito mínimo envío:
99         for r in REF:
100             for j in Jr[r,g,n]:
101                 if Brj[r,j,g,n] < Xr[r,g,n]:
102                     Brj[r,j,g,n] = 0
103                     Jr[r,g,n].remove(j)
104
105         ## PASO 6:
106         ## Recálculo de Brj:
107         for r in REF:
108             suma_r[r,g,n] = sum(abs(Lj[j,g,n]) for j in Jr[r,g,n])
109
110         for r in REF:
111             for j in Jr[r,g,n]:
112                 Brj[r,j,g,n] = abs(Lj[j,g,n])/suma_r[r,g,n]*sum(Ira[r,a,g,n]
113                     for a in DATOS_CD)
114
115         ## Comprobar que se verifica la factibilidad:
116         for j in Jr[r,g,n]:
117             suma[j,g,n] = sum(Brj[r,j,g,n] for r in REF)
118
119         for j in Jr[r,g,n]:
120             if suma[j,g,n] < qmgn[PAIS_T[j],g,n]*vjgn[j,g,n]:

```



```
sys.exit("No se verifica la factibilidad de Brj")
```


Apéndice E

Código en Python. Modelo II

Una vez enunciado el problema, se presenta el código en Python del Modelo II, disponible en el módulo `ModeloII.py`. Se ha resuelto empleando el solver lineal Gurobi.

```
1 from gurobipy import *
2 from Brj import Brj, Lj
4 def solve (PAIS, REF, TIENDA, DATOS_CD, PL, T_PL, PL_T, T_PAIS, PAIS_T, CD_T, T_CD, I_rm, I_rj, I_ra,
5           CA, CT, CL, CE, CAT, RS, obj, g, n):
6
7     # Modelo:
8     modeloII = Model("ModeloII")
9
10    #
11    #-----
12
13    # Flujo de inventario desde los CD a las tiendas:
14    q_raj = modeloII.addVars(REF, [(CD_T[j], j) for j in TIENDA], lb=0, name="q_raj")
15
16    # Flujo de inventario desde el almacén local a la tienda:
17    q_rmj = modeloII.addVars(REF, [(PAIS_T[j], j) for j in TIENDA], lb=0, name="q_rmj")
18
19    # Flujo de inventario desde la plataforma logística a la tienda:
20    q_rl j = modeloII.addVars(REF, [(PL_T[j], j) for j in TIENDA], lb=0, name="q_rl j")
21
22    # Flujo de invenatrio desde la tienda a la plataforma logística:
23    q_rj l = modeloII.addVars(REF, [(j, PL_T[j]) for j in TIENDA], lb=0, name="q_rj l")
24
25    # Flujo desde un CD a CD para el clúster n en el GRUPO g:
26    q_ raa = modeloII.addVars(REF, DATOS_CD, DATOS_CD, lb=0, name="q_ raa")
27
28    # Flujo de una PL a otra PL:
29    q_ rll = modeloII.addVars(REF, PL, PL, lb=0, name="q_ rll")
30
31    modeloII.update()
32
33    #
34    #-----
35
36    # Costes totales de transporte de CD:
37    CDaCD = CA * (sum( sum( sum( q_ raa [r, cd, cd2] for r in REF) for cd2 in (DATOS_CD[:
38    DATOS_CD.index(cd)]+DATOS_CD[DATOS_CD.index(cd)+1:])) for cd in DATOS_CD))
39
40    # Costes totales transporte Tienda a PL:
41    Tienda_PL = CT * (sum( sum( q_ rl j [r, PL_T[j], j] + q_ rj l [r, j, PL_T[j]] for r in REF) for
42    j in TIENDA))
43
44    # Costes totales de transporte de PL a PL:
45    PLaPL = CL * (sum( sum( sum( q_ rll [r, l, l2] for r in REF) for l2 in (PL[:PL.index(1)]+
46    PL[PL.index(1)+1:])) for l in PL))
47
48    # Total DC transfer costs var
```

```

46 CD_Tienda = CE * (sum( sum( q_raj[r,CD-T[j],j] for r in REF) for j in TIENDA))
48 # Total DC transfer costs var
Almacen_Tienda = CAT * ( sum( sum( q_rmj[r, PAIS-T[j], j] for r in REF) for j in
TIENDA))
50
52 # FUNCIÓN OBJETIVO:
53 #-----
54 modeloII.setObjective(CDaCD + Tienda_PL + PLaPL + CD_Tienda + Almacen_Tienda , GRB.
MINIMIZE)
56
58 ### RESTRICCIONES:
59 #-----
60
61 # Restricción de balance de inventario desde los CD:
62 modeloII.addConstrs(( I_ra[r,cd] + sum( q_raa[r,cd2,cd] for cd2 in DATOS.CD) >= sum(
sum(q_raj [r,cd,j] for j in [(T_CD[cd][i])]) for i in range(0,len(T_CD[cd]))) + sum(
q_raa [r,cd,cd2] for cd2 in DATOS.CD) for cd in DATOS.CD for r in REF), "
cd_transportes")
64
65 # Restricción de flujo de inventario desde las plataformas logísticas:
66 modeloII.addConstrs(( sum( q_rl1[r,l2,l] for l2 in PL) + sum(sum(q_rjl [r,j,l] for
j in [(T_PL[l][i])]) for i in range(0,len(T_PL[l]))) == sum(q_rl1 [r,l,l2] for l2 in
PL) + sum(sum(q_rlj [r,l,j] for j in [(T_PL[l][i])]) for i in range(0,len(T_PL[l])))
for r in REF for l in PL), "flujo_inv")
68
69 # Restricción de balance de inventario en almacén:
70 modeloII.addConstrs(( sum(sum(q_rmj [r,m,j] for j in [(T_PAIS[m][i])]) for i in
range(0,len(T_PAIS[m]))) <= I_rm[r,m] for r in REF for m in PAIS), "balance_inv")
72
73 # Restricción de balance de inventario en tienda:
74 modeloII.addConstrs(( sum( I_rj[r,j] + q_raj[r,CD-T[j],j] + q_rmj[r,PAIS-T[j],j]
+ q_rlj [r,PL-T[j],j] for r in REF) >= Lj[j] + sum( q_rjl [r,j,PL-T[j]] for r in REF
) for j in TIENDA), "flujo_inv_tienda")
76
77 # Restricción de cantidad máxima de envío:
78 modeloII.addConstrs(( q_raj[r,CD-T[j],j] + q_rmj[r,PAIS-T[j],j] + q_rlj [r,PL-T[j],j
] <= B_rj[r,j] for r in REF for j in TIENDA), "flujo_inv_tienda")
79
80 # Optimización:
81 modeloII.optimize()
82 print(modeloII.getVars())
83 if modeloII.status == GRB.Status.OPTIMAL:
84     print('Optimal objective: %g' % modeloII.objVal)
85     obj[n] = modeloII.objVal

```

Apéndice F

Código en Python: Datos del Modelo

Se presenta el código en lenguaje Python disponible en el módulo `Modelo_datos.py`, en el que se simulan los conjuntos y datos para los dos modelos de optimización.

```
1 import numpy as np
2 import itertools
3 import sys
4 from scipy.stats import truncexpon, truncnorm
5 import matplotlib.pyplot as plt
6 np.random.seed(3)
7
8 ##   CONJUNTOS MODELO I:
9 ##-----
10
11 ## Creación del conjunto GRUPO:
12 GRUPO = list(range(1,11))
13
14 ## Creación del conjunto setCLUSTER:
15 setCLUSTER = list(range(1,51))
16
17 ## Creación del conjunto CLUSTER:
18 CLUSTER = {}
19 for g in GRUPO:
20     CLUSTER[g] = list(range(1 + 5*(g-1), 6 + 5*(g-1)))
21
22 ## Creación del conjunto PAIS:
23 PAIS = list(range(1,74))
24
25 ## Creación del conjunto DATOS_CD:
26 DATOS_CD = [10001,10018,10026]
27
28 ##### Se sortean los países que abastece cada CD:
29 pais_cd = np.random.multinomial(len(PAIS), np.ones(len(DATOS_CD))/len(DATOS_CD),
30                                 size=1)[0]
31 cd = []
32 for i in range(0, len(pais_cd)):
33     cd = cd + pais_cd[i]*[DATOS_CD[i]]
34
35 ## Creación del conjunto PAIS_CD:
36 CD_PAIS = {}
37 i = 0
38 while True:
39     try:
40         m = PAIS[i]
41         CD_PAIS[m] = cd[i]
42         i = i + 1
43     except IndexError:
44         break
45
46 ## Creación del conjunto CD_PAIS:
47 PAIS_CD = {10001: list(range(1, pais_cd[0]+1)), 10018: list(range(pais_cd[0]+1,
48                                 pais_cd[0]+pais_cd[1]+1)), 10026: list(range(pais_cd[0] + pais_cd[1] + 1,
```

```

        len(PAIS) + 1))}
50
51 ## Creación del conjunto setPRECIOS:
52 setPRECIOS = list(range(1,4))
53
54 ## CONJUNTOS MODELO II:
55 ## -----
56
57 ## Creación del conjunto TIENDA:
58 TIENDA = list(range(1,1201))
59
60 ## Creación del conjunto PL:
61 PL = list(range(1,4))
62
63 ## Creación del conjunto REF:
64 REF = list(range(1,101))
65
66 ## Se sorteian las tiendas en cada país:
67 tienda_pais = np.random.multinomial(len(TIENDA), np.ones(len(PAIS))/len(PAIS), size=1)[0]
68 pais = []
69 for i in range(0,len(tienda_pais)):
70     pais = pais + tienda_pais[i]*[PAIS[i]]
71
72 ## Creación del conjunto PAIS_T:
73 PAIS_T = {}
74 i = 0
75 while True:
76     try:
77         j = TIENDA[i]
78         PAIS_T[j] = pais[i]
79         i = i + 1
80     except IndexError:
81         break
82
83 ## Creación del conjunto T_PAIS:
84 T_PAIS = {}
85 for k, v in PAIS_T.items():
86     T_PAIS[v] = T_PAIS.get(v, [])
87     T_PAIS[v].append(k)
88
89 ## Se sorteian las tiendas a las que abastece cada PL:
90 tienda_pl = np.random.multinomial(len(TIENDA), np.ones(len(PL))/len(PL))[0]
91 pl = []
92 for i in range(0,len(tienda_pl)):
93     pl = pl + list(itertools.chain.from_iterable(itertools.repeat(x, tienda_pl[i])
94         for x in PL))
95
96 ## Creación del conjunto PL_T:
97 PL_T = {}
98 i = 0
99 while True:
100     try:
101         j = TIENDA[i]
102         PL_T[j] = pl[i]
103         i = i + 1
104     except IndexError:
105         break
106
107 ## Creación del conjunto T_PL:
108 T_PL = {}
109 for k, v in PL_T.items():
110     T_PL[v] = T_PL.get(v, [])
111     T_PL[v].append(k)
112
113 ## Se sorteian las tiendas a las que abastece cada CD:
114 tienda_cd = np.random.multinomial(len(TIENDA), np.ones(len(DATOS_CD))/len(DATOS_CD))[0]
115 cd = []
116 for i in range(0,len(tienda_cd)):
117     cd = cd + list(itertools.chain.from_iterable(itertools.repeat(x, tienda_cd[i])
118         for x in DATOS_CD))

```

```

120 ## Creación del conjunto CD_T:
121 CD_T = {}
122 i = 0
123 while True:
124     try:
125         j = TIENDA[i]
126         CD_T[j] = cd[i]
127         i = i + 1
128     except IndexError:
129         break
130
131 ## Creación del conjunto T_CD:
132 T_CD = {}
133 for k, v in CD_T.items():
134     T_CD[v] = T_CD.get(v, [])
135     T_CD[v].append(k)
136
137 ##
138 ##-----
139
140 # Inventario en el cd al principio del período actual
141 Ia = {}
142 Ira = {}
143 for g in GRUPO:
144     for n in CLUSTER[g]:
145
146         for a in DATOS_CD:
147             lower, upper, scale = 100000000, 1000000000, 500000000
148             Ia[a,g,n] = truncexpon.rvs(b=(upper-lower)/scale, loc=lower, scale=scale)
149
150         for r in REF:
151             for a in DATOS_CD:
152                 Ira[r,a,g,n] = abs(np.random.normal(loc=Ia[a,g,n], scale=1000, size=None)
153 )
154
155 # Inventario en el almacén del país al principio del período actual
156 Im = {}
157 Irm = {}
158 for g in GRUPO:
159     for n in CLUSTER[g]:
160         for m in PAIS:
161             lower, upper, scale = 100, 600, 300
162             Im[m,g,n] = truncexpon.rvs(b=(upper-lower)/scale, loc=lower, scale=scale)
163         for r in REF:
164             for m in PAIS:
165                 Irm[r,m,g,n] = abs(np.random.normal(loc=Im[m,g,n], scale=50))
166
167 # Inventario en tienda al principio del período actual
168 Ij = {}
169 Irj = {}
170 for g in GRUPO:
171     for n in CLUSTER[g]:
172         for j in TIENDA:
173             lower, upper, scale = 0, 200, 100
174             Ij[j,g,n] = truncexpon.rvs(b=(upper-lower)/scale, loc=lower, scale=scale)
175         for r in REF:
176             for j in TIENDA:
177                 Irj[r,j,g,n] = abs(np.random.normal(loc=Ij[j,g,n], scale=50))
178
179 Iagn = {}
180 for a in DATOS_CD:
181     for g in GRUPO:
182         for n in CLUSTER[g]:
183             Iagn[a,g,n] = sum(Ira[r,a,g,n] for r in REF)
184
185 Iagn = {}
186 for m in PAIS:
187     for g in GRUPO:
188         for n in CLUSTER[g]:
189             Imgn[m,g,n] = sum(Irm[r,m,g,n] for r in REF)

```

```

190  ##      SIMULACIÓN DE PRECIOS:
191  ##      -----
192
193  # Precios temporada regular:
194  mu = 30
195  sigma = 20
196  lower = 4
197  upper = 100
198  Pmgn = {}
199  for m in PAIS:
200      for g in GRUPO:
201          for n in CLUSTER[g]:
202              Pmgn[m, g, n] = truncnorm.rvs((lower[1]-mu[1])/sigma[1],
203              (upper[1]-mu[1])/sigma[1], loc=mu[1],
204              scale=sigma[1])
205
206  # Precios en rebajas:
207  precio_min={}
208  for m in PAIS:
209      for g in GRUPO:
210          precio_min[m, g] = min(Pmgn[m, g, n] for n in CLUSTER[g])
211
212  mu = 25
213  sigma = 10
214  N = len(setPRECIOS)
215  lower = 4
216  upper = precio_min
217  PRECIOS = {}
218  for m in PAIS:
219      for g in GRUPO:
220          PRECIOS[m, g] = sorted(list(truncnorm.rvs((lower-mu)/sigma, (upper[m, g]-mu)/
221          sigma,
222          loc=mu, scale=sigma, size=N)))
223  for m in PAIS:
224      for g in GRUPO:
225          for k in setPRECIOS:
226              PRECIOS[m, g, k] = PRECIOS[m, g][k-1]
227
228  # Psaldero:
229  mu = 5
230  sigma = 3
231  lower = 0.5
232  precio_min={}
233  psaldero = {}
234  for g in GRUPO:
235      precio_min[g] = min(PRECIOS[m, g, k] for m in PAIS for k in setPRECIOS)
236      psaldero[g] = truncnorm.rvs((lower-mu)/sigma, (precio_min[g]-mu)/sigma, loc=mu,
237      scale=sigma)
238
239  ##      SIMULACIÓN DE DEMANDAS:
240  ##      -----
241
242  Drj = {}
243  for r in REF:
244      for j in TIENDA:
245          for g in GRUPO:
246              for n in CLUSTER[g]:
247                  Drj[r, j, g, n] = abs(np.random.normal(loc=int(Irj[r, j, g, n]/4),
248                  scale=int(Irj[r, j, g, n]/8)))
249
250  Djgn = {}
251  for j in TIENDA:
252      for g in GRUPO:
253          for n in CLUSTER[g]:
254              Djgn[j, g, n] = sum(Drj[r, j, g, n] for r in REF)
255
256  Dmgn = {}
257  for m in PAIS:

```



```

258     for g in GRUPO:
260         for n in CLUSTER[g]:
                Dmgn[m,g,n] = sum(Djgn[j,g,n] for j in T_PAIS[m])

262 D = {}
D_mgn = {}

264 for m in PAIS:
266     for g in GRUPO:
268         for n in CLUSTER[g]:
                D[m,g,n] = sorted(np.random.normal(loc=Dmgn[m,g,n]*2, scale=Dmgn[m,g,n]/10,
                size=len(setPRECIOS)))
270         for k in setPRECIOS:
                D_mgn[m,g,n,k] = D[m,g,n][-k]

272
274 ## -----
                Constantes del modelo I:
                ## -----

276 CE=0.02
CA=0.01
278 SA=1
ST=1
280 Qm=1
desc_min=0.1
282 Wr=4
holgura=0.9999
284 tiempo={}
for m in PAIS:
286     tiempo[m]=1
D=7
288 n_precios=3

290 ## -----
                RESTRICCIONES PARÁMETROS MODELO I:
                ## -----
292 # Opción GRUPO WOMAN:

294 for p in Pmgn.values():
296     if p < 0:
                sys.exit("El vector de parámetros Pmgn debe ser positivo")

298 for d in D_mgn.values():
300     if d < 0:
                sys.exit("El vector de parámetros D_mgn debe ser positivo")

302 if CE < 0:
304     sys.exit("El parámetro CE tiene que ser positivo")

306 if CA < 0:
308     sys.exit("El parámetro CA tiene que ser positivo")

310 if D < 0:
312     sys.exit("El parámetro D tiene que ser positivo")

314 if SA > 1 or SA < 0:
316     sys.exit("El parámetro SA debe estar comprendido entre 0 y 1")

318 if ST > 1 or ST < 0:
320     sys.exit("El parámetro ST debe estar comprendido entre 0 y 1")

322 if Qm < 0:
324     sys.exit("El parámetro Qm tiene que ser positivo")

326 if desc_min > 1 or desc_min < 0:
328     sys.exit("El parámetro de descuento mínimo tiene que ser mayor o igual que 1")

330 for t in tiempo.values():
332     if t > 1 or t < 0:

```

```

328     sys.exit("El vector de parámetros tiempo tiene que ser mayor o igual que 1")
330 if holgura > 1 or holgura < 0:
331     sys.exit("El parámetro holgura tiene que ser mayor o igual que 1")
332
333 ##      Constantes del modelo II:
334 ##-----
335
336 CA = 0.01
337 CT = 0.01
338 CL = 0.01
339 CE = 0.02
340 CAT = 0.01
341 RS = 1
342
343 ##      RESTRICCIONES PARÁMETROS MODELO II:
344 ##-----
345
346 # Opción GRUPO WOMAN:
347
348 if CA < 0:
349     sys.exit("El parámetro CA tiene que ser positivo")
350
351 if CE < 0:
352     sys.exit("El parámetro CE tiene que ser positivo")
353
354 if CT < 0:
355     sys.exit("El parámetro CT tiene que ser positivo")
356
357 if CL < 0:
358     sys.exit("El parámetro CL tiene que ser positivo")
359
360 if CAT < 0:
361     sys.exit("El parámetro CAT tiene que ser positivo")
362
363 if RS < 0:
364     sys.exit("El parámetro RS tiene que ser positivo")
365
366
367 #      IMPORTAR MODELO I:
368 #-----
369 import ModeloI
370 ModeloI.solve(GRUPO, CLUSTER, setCLUSTER, PAIS, PAIS_CD, DATOS_CD, setPRECIOS, PRECIOS,
371              Pmgn, psaldero, Iagn, Imgn, Drj, Djgn, Dmgn, D_mgn, CE, CA, D, Wr, SA, ST, Qm,
372              desc_min, tiempo, holgura, n_precios)
373
374 #      IMPORTAR MODELO II:
375 #-----
376 obj = {}
377 for g in GRUPO:
378     for n in CLUSTER[g]:
379         I_rm = {}
380         for r in REF:
381             for m in PAIS:
382                 I_rm[r,m] = Irm[r,m,g,n]
383         I_ra = {}
384         for r in REF:
385             for a in DATOS_CD:
386                 I_ra[r,a] = Ira[r,a,g,n]
387         I_rj = {}
388         for r in REF:
389             for j in TIENDA:
390                 I_rj[r,j] = Irj[r,j,g,n]
391
392 import ModeloII
393 ModeloII.solve(PAIS, REF, TIENDA, DATOS_CD, PL, T_PL, PL_T, T_PAIS, PAIS_T, CD_T,
394               T_CD, I_rm, I_rj, I_ra, CA, CT, CL, CE, CA, RS, obj, g, n)

```

Apéndice G

Código en Python: Heurística III

Se presenta el código en lenguaje Python disponible en el módulo `HeuristicaIII.py`.

```
1 from sim_datos import PAIS,GRUPO,CLUSTER,CD_PAIS,DATOS_CD,PRECIOS,setPRECIOS,Pmgn,Iagn,
   Imgn,Wr
2 from Bmgn import Bmgn
3 from HeuristicaI import dmgn_h,d_mgn_h
4 from collections import OrderedDict
5
6 ganancias = {}
7
8 Dem_fal = {}
9 for m in PAIS:
10     for g in GRUPO:
11         for n in CLUSTER[g]:
12             ganancias[m,g,n] = Pmgn[m,g,n]*dmgn_h[m,g,n]
13             Dem_fal[m,g,n] = max(dmgn_h[m,g,n]-Imgn[m,g,n],0)
14
15 ganancias_ord = OrderedDict(sorted(ganancias.items(), key=lambda x: x[1], reverse=True))
16
17 Dem_max = {}
18 for m in PAIS:
19     for g in GRUPO:
20         for n in CLUSTER[g]:
21             Dem_max[m,g,n] = min(Dem_fal[m,g,n],Bmgn[m,g,n])
22
23 qagn = {}
24 Dem_max_fal = {}
25
26 qmgn = {}
27
28 for (m,g,n) in ganancias_ord:
29     if Iagn[CD_PAIS[m],g,n] >= Dem_max[m,g,n]:
30         qmgn[m,g,n] = Dem_max[m,g,n]
31         Iagn[CD_PAIS[m],g,n] = Iagn[CD_PAIS[m],g,n] - Dem_max[m,g,n]
32     else:
33         Dem_max_fal[m,g,n] = Dem_max[m,g,n] - Iagn[CD_PAIS[m],g,n]
34         qmgn[m,g,n] = Iagn[CD_PAIS[m],g,n]
35         Iagn[CD_PAIS[m],g,n] = 0
36     A = DATOS_CD[0:DATOS_CD.index(CD_PAIS[m])] + DATOS_CD[DATOS_CD.index(CD_PAIS[m]),
37         len(DATOS_CD)-1]
38
39     for a2 in A:
40         Iagn_ord = OrderedDict(sorted(Iagn[a2,g,n].items(), key=lambda x: x[1],
41             reverse=True))
42     for a2 in Iagn_ord:
43         d = min(Iagn[a2,g,n],Dem_max_fal[m,g,n])
44         qmgn[m,g,n] = qmgn[m,g,n] + d
45         Iagn[a2,g,n] = Iagn[a2,g,n] - d
46         qagn[a2,CD_PAIS[m],g,n] = qagn[a2,CD_PAIS[m],g,n] + d
47         if qmgn[m,g,n] == Dem_max[m,g,n]:
48             exit
```

```

50 for m in PAIS:
52     for g in GRUPO:
54         for n in CLUSTER[g]:
55             Imgn[m,g,n] = max(Imgn[m,g,n] - dmgn_h[m,g,n], 0)
56
57 for w in list(range(0,Wr)):
58
59     ganancias_r = {}
60     Dem_fal_r = {}
61
62     for m in PAIS:
63         for g in GRUPO:
64             for n in CLUSTER[g]:
65                 for k in setPRECIOS:
66                     if d_mgn_h[w,m,g,n,k] != 0:
67                         ganancias_r[m,g,n] = PRECIOS[m,g,k]*d_mgn_h[w,m,g,n,k]
68
69     for m in PAIS:
70         for g in GRUPO:
71             for n in CLUSTER[g]:
72                 Dem_fal_r[m,g,n] = max(sum(d_mgn_h[w,m,g,n,k] for k in setPRECIOS)-Imgn[m,g,n],0)
73
74     ganancias_r_ord = OrderedDict(sorted(ganancias_r.items(), key=lambda x: x[1], reverse=True))
75
76     Dem_max_r = {}
77     for m in PAIS:
78         for g in GRUPO:
79             for n in CLUSTER[g]:
80                 Dem_max_r[m,g,n] = min(Dem_fal_r[m,g,n], Bmgn[m,g,n])
81
82     Dem_max_fal_r = {}
83
84     for (m,g,n) in ganancias_r_ord:
85         if Iagn[CD_PAIS[m],g,n] >= Dem_max_r[m,g,n]:
86             qmgn[m,g,n] = qmgn[m,g,n] + Dem_max_r[m,g,n]
87             Iagn[CD_PAIS[m],g,n] = Iagn[CD_PAIS[m],g,n] - Dem_max_r[m,g,n]
88         else:
89             Dem_max_fal_r[m,g,n] = Dem_max_r[m,g,n] - Iagn[CD_PAIS[m],g,n]
90             qmgn[m,g,n] = qmgn[m,g,n] + Iagn[CD_PAIS[m],g,n]
91             Iagn[CD_PAIS[m],g,n] = 0
92             A = DATOS_CD[0:DATOS_CD.index(CD_PAIS[m])] + DATOS_CD[DATOS_CD.index(CD_PAIS[m]),len(DATOS_CD)-1]
93
94             for a2 in A:
95                 Iagn_ord = OrderedDict(sorted(Iagn[a2,g,n].items(), key=lambda x: x[1], reverse=True))
96
97                 for a2 in Iagn_ord:
98                     d = min(Iagn[a2,g,n], Dem_max_fal_r[m,g,n])
99                     qmgn[m,g,n] = qmgn[m,g,n] + d
100                     Iagn[a2,g,n] = Iagn[a2,g,n] - d
101                     qagn[a2,CD_PAIS[m],g,n] = qagn[a2,CD_PAIS[m],g,n] + d
102                     if qmgn[m,g,n] == Dem_max_r[m,g,n]:
103                         exit
104
105     for m in PAIS:
106         for g in GRUPO:
107             for n in CLUSTER[g]:
108                 Imgn[m,g,n] = max(Imgn[m,g,n] - sum(d_mgn_h[w,m,g,n,k] for k in setPRECIOS), 0)

```

Bibliografía

- [1] Besbes, O., Muharremoglu, A. (2010) On the Price of Demand Censoring in the Newsvendor Problem. Working Paper. Columbia University, NY.
- [2] Carboni, R. (2008, junio) Clearance Pricing Optimization at Zara.
- [3] Carboni, R. (2009) Clearance Pricing Optimization at Zara. Tesis, MIT.
- [4] Caro, F. (2009, noviembre 10) In preparation for: Coordination of Inventory Distribution and Price Markdowns for Clearance Sales. *Coordination of Inventory Distribution and Price Markdowns for Clearance Sales*.
- [5] Caro, F. & Gallien, J. (2007, diciembre 10) In preparation for: Clearance Pricing for a Fast-Fashion Retail Network. *Clearance Pricing Optimization at Zara*.
- [6] Caro, F. & Gallien, J. (2011, junio 1) Clearance Pricing Optimization for a Fast-Fashion Retailer.
- [7] Duan, N. (1983) A Nonparametric Retransformation Method. *Journal of the American Statistical Association* 78(383): 605-610.
- [8] Feng, Y., & Gallego, G. (1995) Optimal Starting Times for End-of-Season Sales and Optimal Stopping Times for Promotional Fares. *Management Science*, 41(8), 1371-1391.
- [9] Gallego, G., & van Ryzin, G. (1994). Optimal Dynamic Pricing of Inventories with Stochastic Demand over Finite Horizons. *Management Science*, 40 (8), 999-1020.
- [10] Keskinocak, P., & Elmaghraby, W. (2003). Dynamic Pricing in the Presence of Inventory Considerations: Research Overview, Current. *Management Science*, 49 (10), 1287-1309.
- [11] Manning, W., Mullahy, J. (2001) Estimating log models: to transform or not to transform. *Journal of Health Economics* 20: 461-494.
- [12] Mantrala, M. K., & Rao, S. (2001). A Decision-Support System that Helps Retailers Decide Order Quantities and Markdowns for Fashion Goods. *Interfaces*, 31 (3), S146-S165.
- [13] Smith, S. A., & Achabal, D. D. (1998). Clearance Pricing and Inventory Policies for Retail Chains. *Management Science*, 44 (3), 285-300.
- [14] Smith S., McIntyre, S., Achabal, D. (1994) A Two-Stage Sales Forecasting Procedure Using Discounted Least Squares. *Journal of Marketing Research* 31(1): 44-56.
- [15] Parra, O. (2004, Junio) Coordination of Inventory Distribution & Price Markdowns for Clearance Sales at Zara.
- [16] Wang, C. X., & Webster, S. (2009). Markdown money contracts for perishable goods with clearance pricing. *European Journal of Operational Research* , 196 (3), 1113-1122.
- [17] <https://charhadas.com/articulo/5226455-trafaluc-lo-mejor-de-zara-para-las-teens>. Accedido 1 de mayo de 2018.
- [18] <https://prezi.com/uwk5jklq3imm/plataformas-logisticas-y-operadores-logisticos>. Accedido 1 de mayo de 2018.

- [19] <https://vendiendo.co/blogs/sistema-de-pos-punto-venta>. Accedido 2 de abril de 2018.
- [20] <https://Wikipedia.es>. Accedido 15 de agosto de 2018.
- [21] <https://Zara.com>. Accedido 15 de agosto de 2018.