



Universidad Internacional de La Rioja  
Facultad de Economía y Empresa

Máster Universitario en Inteligencia de Negocio

**Aplicación de Business Intelligence en el  
canal de distribución de una empresa  
tabacalera**

Trabajo fin de estudio presentado por:	Agustín Daniel Contreras Blanco Marc Garcia Torregrosa
Tipo de trabajo:	Proyecto de Inteligencia de Negocio
Modalidad:	Grupal
Director/a:	Serhiy Lyalkov Lyalkova
Fecha:	16 de julio de 2025

## RESUMEN

Este Trabajo de Fin de Máster tiene como objetivo el diseño y desarrollo de un sistema de Business Intelligence (BI) aplicado al canal de distribución de la empresa tabacalera Altadis Imperial Brands, que ha facilitado los datos necesarios para la realización del proyecto.

Mediante herramientas como RStudio para el análisis estadístico y Power BI para la visualización de datos, se llevó a cabo un proceso de análisis, tratamiento y modelado de los datos operativos proporcionados por la empresa.

A lo largo del presente proyecto se emplearon modelos descriptivos, predictivos y de segmentación con el fin de transformar grandes volúmenes de datos en información relevante para la compañía. Esto ha permitido la identificación de patrones de consumo y entrega, segmentar los distintos puntos de venta en función de su rendimiento y detectar características geográficas clave para Imperial Brands.

Las conclusiones del estudio refuerzan la utilidad de los sistemas BI como herramienta clave en la mejora de la eficiencia operativa y toma de decisiones. Se recomienda utilizar las entregas como referencia para planificar la reposición de productos, segmentar puntos de venta para la personalización de estrategias comerciales, y revisar la calidad de los datos operativos ya que se detectaron inconsistencias en algunos periodos. Asimismo, también se propone adaptar el portafolio de productos ofrecidos a los patrones de consumo identificados.

En conjunto, el trabajo realizado demuestra cómo la implementación efectiva de BI no solo proporciona una ventaja competitiva, sino que también favorece la alineación de la estrategia comercial con el comportamiento real del mercado.

**Palabras clave:** Altadis Imperial Brands, retail tabacalero, Business Intelligence, RStudio, PowerBI

## ABSTRACT

This Master's Thesis aims to design and develop a Business Intelligence (BI) system applied to the distribution channel of the tobacco company Altadis Imperial Brands, which provided the necessary data for the execution of the project.

Using tools such as RStudio for statistical analysis and Power BI for data visualization, a comprehensive process was carried out involving the analysis, processing, and modelling of the operational data provided by the company.

Throughout the project, descriptive, predictive, and segmentation models were employed to transform large volumes of data into relevant information for the company. This allowed for the identification of consumption and delivery patterns, the segmentation of retail outlets based on their performance, and the detection of geographic characteristics relevant to Imperial Brands' strategy.

The conclusions of the study reinforce the usefulness of BI systems as key tools for improving operational efficiency and decision-making. It is recommended to use delivery data as a reference for planning product replenishment, to segment points of sale in order to personalize commercial strategies, and to review the quality of operational data due to inconsistencies identified in certain periods. Additionally, adapting the product portfolio to the consumption patterns identified is also proposed.

Overall, this project demonstrates how an effective BI implementation not only provides a competitive advantage but also helps align commercial strategy with the actual behaviour of the target market.

**Keywords:** Altadis Imperial Brands, tobacco retail, Business Intelligence, RStudio, Power BI

## ORGANIZACIÓN DEL TRABAJO EN GRUPO

El trabajo en grupo se ha estructurado de la siguiente manera:

<b>Epígrafe</b>	<b>Alumno responsable</b>
Introducción	Agustín Daniel Contreras Blanco
Marco teórico	Trabajo colaborativo
Objetivos del TFM	Trabajo colaborativo
Tratamiento de datos	Trabajo colaborativo
Análisis exploratorio de datos	Trabajo colaborativo
Modelado de datos en Power BI	Marc Garcia Torregrosa
Modelos predictivos con RStudio	Agustín Daniel Contreras Blanco
Modelos de segmentación con Rstudio	Agustín Daniel Contreras Blanco
Conclusiones	Marc Garcia Torregrosa
Recomendaciones	Marc Garcia Torregrosa

## ÍNDICE

RESUMEN.....	2
ABSTRACT .....	3
ORGANIZACIÓN DEL TRABAJO EN GRUPO .....	4
ÍNDICE .....	5
TABLA DE ILUSTRACIONES.....	8
1. INTRODUCCIÓN .....	12
2. MARCO TEÓRICO .....	13
2.1. EL SECTOR DEL TABACO EN EL RETAIL.....	13
2.2. HERRAMIENTAS DE ANÁLISIS EN BUSINESS Intelligence.....	14
3. OBJETIVOS DEL TFM .....	15
3.1. OBJETIVO GENERAL .....	15
3.2. OBJETIVOS ESPECÍFICOS .....	15
3.2.1. Análisis y calidad de los datos con RStudio.....	15
3.2.2. Creación de dashboards de control con Power BI .....	15
3.2.3. Desarrollo de modelos analíticos y predictivos con RStudio .....	15
3.2.4. Generación de insights estratégicos .....	16
4. TRATAMIENTO DE DATOS.....	17
4.1. EXCEL.....	17
4.2. RSTUDIO .....	18
4.3. DICCIONARIO DE VARIABLES .....	19
5. ANÁLISIS EXPLORATORIO DE DATOS .....	22
5.1. ANÁLISIS DESCRIPTIVO.....	22
5.1.1. Tabla Affiliated_Outlets .....	22

5.1.2. Tabla DeliveryDay.....	23
5.1.3. Tabla SalesDay.....	24
5.1.4. Tabla OoSDay .....	24
5.1.5. Tabla RouteDay .....	25
5.1.6. Tabla Product .....	25
5.2. ANÁLISIS DE PREPROCESAMIENTO ÁGIL: GRÁFICOS Y MATRIZ DE CORRELACIÓN .....	26
5.2.1. Gráficos de caja: Ventas, entregas y total de ventas por tienda .....	26
5.2.2. Histograma de unidades vendidas y entregadas .....	28
5.2.3. Ventas y entregas semanales.....	30
5.2.4. Tiempo medio entre entregas por tienda.....	32
5.2.5. Top 10 productos con más roturas .....	33
5.2.6. Top 10 tiendas con más roturas.....	34
5.2.7. Matriz de correlación .....	35
6. MODELADO DE DATOS EN POWER BI .....	38
6.1. TRATAMIENTO Y LIMPIEZA DE LOS DATOS.....	38
6.2. VISUALIZACIÓN DE LOS DATOS.....	40
6.2.1. Dashboard 1: Análisis de ventas .....	41
6.2.2. Dashboard 2: Análisis de Productos.....	42
6.3. RESULTADOS OBTENIDOS EN POWER BI .....	43
6.3.1. Ventas por provincias.....	43
6.3.2. Ventas por zona.....	43
6.3.3. Evolución de las ventas .....	44
6.3.4. Ventas por tipo de producto .....	44
6.3.5. Peso de tipo de producto por provincias .....	45

6.3.6. Peso de tipo de producto por zona .....	45
6.3.7. Evolución de las ventas por tipo de producto .....	46
7. MODELOS PREDICTIVOS CON RSTUDIO .....	47
7.1. MODELO PREDICTIVO DE VENTAS .....	47
7.1.1. Predicción de la serie posterior a mayo de 2015 .....	50
7.2. MODELO PREDICTIVO DE ENTREGAS.....	51
8. MODELOS DE SEGMENTACIÓN CON RSTUDIO.....	56
8.1. SEGMENTACIÓN CON K-MEANS SIN NORMALIZAR.....	56
8.2. SEGMENTACIÓN CON K-MEANS CON DATOS NORMALIZADOS.....	62
9. CONCLUSIONES.....	65
10. RECOMENDACIONES .....	68
11. REFERENCIAS .....	70

## TABLA DE ILUSTRACIONES

Ilustración 1: Duplicado en la tabla de Producto .....	17
Ilustración 2: Corrección del duplicado .....	17
Ilustración 3: Fórmula para transformar en formato Fecha .....	17
Ilustración 4: Formato Fecha transformado.....	18
Ilustración 5: Error en el formato fecha en R .....	18
Ilustración 6: Código para transformar las variables Fecha en R .....	18
Ilustración 7: Variable Fecha corregida en R .....	19
Ilustración 8: Librerías R .....	22
Ilustración 9: Código copia de las tablas originales.....	22
Ilustración 10: Análisis descriptivo Affiliatet_Outlets .....	23
Ilustración 11: Análisis descriptivo DeliveryDay.....	23
Ilustración 12: Análisis descriptivo SalesDay .....	24
Ilustración 13: Análisis descriptivo OoSDay .....	24
Ilustración 14: Análisis descriptivo RouteDay .....	25
Ilustración 15: Análisis descriptivo Product .....	25
Ilustración 16: Código para eliminar los valores negativos.....	26
Ilustración 17: Código gráfico de Caja de Ventas .....	26
Ilustración 18: Gráfico de Caja de Ventas .....	27
Ilustración 19: Código gráfico de Caja de Entregas .....	27
Ilustración 20: Gráfico de Caja de Entregas.....	27
Ilustración 21: Código gráfico de Caja de Ventas por Tiendas .....	28
Ilustración 22: Gráfico de Caja de Ventas por Tiendas.....	28



Ilustración 23: Código gráfico Histograma de Ventas .....	29
Ilustración 24: Gráfico Histograma de Ventas.....	29
Ilustración 25: Código gráfico Histograma de Entregas .....	29
Ilustración 26: Gráfico Histograma de Entregas.....	30
Ilustración 27: Código gráfico Evolución de Ventas en el tiempo .....	30
Ilustración 28: Gráfico Evolución de Ventas en el tiempo .....	31
Ilustración 29: Código gráfico Evolución de Entregas en el tiempo .....	32
Ilustración 30: Gráfico Evolución de Entregas en el tiempo .....	32
Ilustración 31: Código tiempo medio entre entregas por tienda.....	33
Ilustración 32: Tiempo medio entre entregas por tienda .....	33
Ilustración 33: Código gráfico top 10 Productos con más Roturas .....	33
Ilustración 34: Gráfico top 10 Productos con más Roturas.....	34
Ilustración 35: Código gráfico top 10 Tiendas con más Roturas .....	35
Ilustración 36: Gráfico top 10 Productos con más Roturas.....	35
Ilustración 37: Código Matriz de Correlación .....	36
Ilustración 38: Matriz de Correlación .....	37
Ilustración 39: Modelo de conexiones de las Tablas en PowerBI .....	38
Ilustración 40: Transformación del tipo de variable de Número a Texto en PowerBI.....	39
Ilustración 41: Creación de la nueva columna Provincia.....	39
Ilustración 42: Creación de la nueva columna especificando España como país .....	40
Ilustración 43: Dashboard 1.....	41
Ilustración 44: Dashboard 2.....	42
Ilustración 45: Gráficos de Ventas por Provincias.....	43

Ilustración 46: Gráfico Ventas por Zona .....	44
Ilustración 47: Gráfico de Evolución de las Ventas .....	44
Ilustración 48: Gráfico Ventas por Tipo de Producto .....	45
Ilustración 49: Gráfico de Tipo de Producto por Provincias.....	45
Ilustración 50: Gráfico Peso de Tipo de Producto por Zona.....	46
Ilustración 51: Gráfico Evolución de las Ventas por Tipo de Productos.....	46
Ilustración 52: Código de ventas semanales .....	47
Ilustración 53: Gráfico de ventas semanales.....	48
Ilustración 54: Predicción de ventas semanales .....	48
Ilustración 55: Resultados de la predicción de ventas .....	49
Ilustración 56: Validación del modelo de ventas .....	50
Ilustración 57: Métricas del error .....	50
Ilustración 58: Código y resultados de la predicción de la serie incompleta .....	51
Ilustración 59: Gráfico de predicción de la serie incompleta.....	51
Ilustración 60: Código de entregas semanales.....	51
Ilustración 61: Código de la predicción de entregas .....	52
Ilustración 62: Resultados de la predicción de entregas.....	53
Ilustración 63: Validación del modelo de entregas .....	54
Ilustración 64: Métricas del error .....	54
Ilustración 65: Código de la predicción final .....	55
Ilustración 66: Resultados de la predicción final.....	55
Ilustración 67: Creación del dataset a segmentar.....	56
Ilustración 68: Código del modelo K-Means sin normalizar.....	57

Ilustración 69: Gráfico del método del codo .....	57
Ilustración 70: Resultados del modelo K-Means sin normalizar .....	58
Ilustración 71: Coeficiente de Silouette del modelo sin normalizar .....	61
Ilustración 72: Segmentación por ventas y roturas de stock .....	62
Ilustración 73: Código del modelo K-Means normalizado .....	63
Ilustración 74: Coeficiente de Silouette del modelo normalizado .....	63
Ilustración 75: Resultados del modelo K-Means sin normalizar .....	64

## 1. INTRODUCCIÓN

El sector tabacalero es una industria altamente regulada, aunque también competitiva, donde la eficiencia en la gestión de la cadena de suministro y la capacidad para poder predecir la demanda suelen ser factores clave para el éxito de las compañías. En este contexto, herramientas como el Business Intelligence (BI) se han convertido en un pilar fundamental en lo relativo a la transformación de datos operativos en conocimiento estratégico sobre el estado de las compañías, permitiendo así a las mismas la optimización de sus procesos internos y la mejora en la toma de decisiones.

Este Trabajo Fin de Máster (TFM) se enmarca en este ámbito, con el objetivo de desarrollar un sistema de BI para Altadis Imperial Brands, una de las principales empresas del sector tabacalero, utilizando las tecnologías de visualización Power BI y de análisis RStudio.

El proyecto, se centra en el análisis de datos operativos facilitados por el grupo Imperial Brands, abarcando desde la limpieza y exploración de datos inicial hasta la implementación de modelos predictivos y de segmentación.

Teniendo eso en cuenta, la metodología empleada combina técnicas de análisis descriptivo además de visualización y modelado de datos, con el fin de extraer insights que puedan contribuir a mejorar la estrategia comercial y la gestión del canal de distribución con los establecimientos asociados.

Entre los desafíos abordados destacan la presencia de valores atípicos, la corrección del formato de los datos y la integración de distintas fuentes de información, aspectos clave a la hora de garantizar la consistencia y calidad de los resultados obtenidos.

La relevancia de este proyecto radica en su enfoque práctico y alineado con las necesidades reales de una empresa líder en un sector del retail con particularidades muy concretas como es el del tabaco. Los resultados obtenidos no solo reafirman la utilidad de las herramientas de BI en la industria, sino que también ofrecen nueva información sobre los procesos internos de Altadis y proponen recomendaciones concretas para la mejora de estos.

## 2. MARCO TEÓRICO

### 2.1. EL SECTOR DEL TABACO EN EL RETAIL

El mercado del tabaco es una industria altamente regulada con un gran impacto económico a nivel mundial. Estudios recientes destacan la importancia de la distribución en el sector retail, dado que los productos de tabaco dependen de canales de venta específicos, principalmente estancos y puntos de venta con recargo.

Dada la complejidad de la industria del retail y su constante transformación impulsada por la implementación de procesos cada vez más eficientes (PricewaterhouseCoopers, 2015), se requieren soluciones cada vez más sofisticadas como las que ofrece el Business Intelligence (BI). Según Solano (2018), el BI agrupa prácticas, capacidades y metodologías que facilitan a las empresas la toma de decisiones óptimas.

La competencia en la industria tabacalera está determinada por diversos factores como los costes de producción, la fijación de precios, el control del stock y la adaptabilidad a las tendencias de consumo, especialmente con la introducción de productos de nueva generación, como los dispositivos de vapeo. A todo esto, se le ha de sumar el efecto del grado de exposición y el tipo de planteamiento por parte de cada una de las principales empresas respecto a los distintos mercados donde operan (Uzcátegui-Sánchez y Camino-Mogro, 2017).

En este contexto, el análisis de datos desempeña un papel crucial al permitir a las empresas tabacaleras mejorar la gestión de su cadena de suministro, prever la demanda y ajustar sus estrategias comerciales según los hábitos de compra de los consumidores. La incorporación de datos geográficos y segmentación avanzada han demostrado ser clave para mejorar la eficiencia operativa y la personalización de la oferta (Imperial Brands, 2024).

## 2.2. HERRAMIENTAS DE ANÁLISIS EN BUSINESS INTELLIGENCE

El uso de herramientas de Business Intelligence (BI) es esencial para transformar los datos en información estratégica. Power BI se ha consolidado como una de las plataformas líderes en la visualización y análisis de datos en tiempo real, permitiendo a las empresas supervisar indicadores clave de rendimiento (KPIs) y tomar decisiones basadas en evidencia (Pérez, Ortega y Bastidas, 2023). A través de dashboards interactivos, Power BI facilita la exploración de datos sobre ventas, inventarios y distribución, optimizando la eficiencia de las operaciones comerciales (Microsoft, 2023).

Por otro lado, R es un lenguaje de programación ampliamente utilizado, en el ámbito académico y en la ciencia de datos por su analítica avanzada. Debido a sus capacidades estadísticas y de machine learning, R es ideal para la implementación de modelos predictivos como ARIMA (para la previsión de ventas o demanda) o K-Means (para la segmentación de clientes) además de otros tipos de modelo (Hyndman y Athanasopoulos, 2021). Estos modelos permiten identificar patrones en los datos, facilitando a las empresas la comprensión de su situación actual y adaptarse mejor a posibles cambios que pudiesen darse en el mercado. Además, facilitan el diseño de estrategias más precisas y personalizadas.

### 3. OBJETIVOS DEL TFM

#### 3.1. OBJETIVO GENERAL

Desarrollar un sistema de Business Intelligence para Imperial Brands con el fin de transformar datos operativos en conocimiento estratégico, optimizando la gestión del canal de distribución y la toma de decisiones mediante análisis avanzados, visualizaciones efectivas y modelos predictivos.

#### 3.2. OBJETIVOS ESPECÍFICOS

##### 3.2.1. Análisis y calidad de los datos con RStudio

Se realizará un análisis descriptivo de los datos mediante el uso de RStudio con el objetivo de evaluar su consistencia y calidad para el posterior tratamiento y análisis. Este análisis previo permitirá detectar posibles outliers, valores nulos y otros posibles descuadres que pudiese haber en los distintos conjuntos de datos. Una vez realizado este paso, los datos podrán enriquecerse con fuentes externas, como información geográfica.

##### 3.2.2. Creación de dashboards de control con Power BI

Se desarrollarán dashboards interactivos que permitan a la empresa supervisar su desempeño de manera eficiente. A través de estos paneles, se implementarán KPIs y métricas clave para evaluar principalmente la evolución de las ventas en distintas zonas geográficas o según el tipo de producto. Además, se diseñarán visualizaciones interactivas que faciliten el análisis del comportamiento de los clientes y la identificación de oportunidades de mejora, contribuyendo a una toma de decisiones más informada y estratégica.

##### 3.2.3. Desarrollo de modelos analíticos y predictivos con RStudio

Se implementará el modelo analítico ARIMA para las predicciones de ventas y entregas del grupo Imperial Brands mediante el uso de RStudio. Para segmentar a los clientes y enfocar estrategias más precisas en grupos específicos, se aplicará el modelo de clustering K-Means. Este enfoque permitirá identificar patrones de comportamiento y agrupar clientes con

características similares, optimizando la personalización de las estrategias comerciales y mejorando la eficiencia en la toma de decisiones.

#### 3.2.4. Generación de insights estratégicos

Por último, a partir de la información generada a lo largo del proyecto, se recopilarán y analizarán los insights estratégicos obtenidos mediante las herramientas de análisis. El objetivo es proporcionar a la empresa información valiosa que facilite la toma de decisiones y la definición de estrategias efectivas. De esta manera, el sistema de BI propuesto contribuirá a la optimización del canal de distribución y a la mejora en la toma de decisiones estratégicas dentro de Imperial Brands.



## 4. TRATAMIENTO DE DATOS

### 4.1. EXCEL

En la tabla Product, concretamente la variable Product\_Code, *Natu122* aparece dos veces. Sin embargo, no se trata de un duplicado exacto, ya que su formato es diferente en cada caso. Por ello, decidimos cambiar el nombre de uno de ellos para diferenciarlos según su formato, ya que no correspondían al mismo producto.

46	Inte755	114	ASL
47	Inte943	283	ASL
48	Inte947	388	ASL
49	Natu079	85	ASL
50	Natu122	283	ATA
51	Natu122	283	ASL
52	Natu408	85	ASL
53	Natu461	85	ASL
54	Natu508	213	ATA

**Ilustración 1: Duplicado en la tabla de Producto**

Se le ha asignado 1221 al del Formato ATA y 1222 al del formato ASL para distinguirlos.

nte755	114	ASL
nte943	283	ASL
nte947	388	ASL
Natu079	85	ASL
Natu1221	283	ATA
Natu1222	283	ASL
Natu408	85	ASL
Natu461	85	ASL
Natu508	213	ATA

**Ilustración 2: Corrección del duplicado**

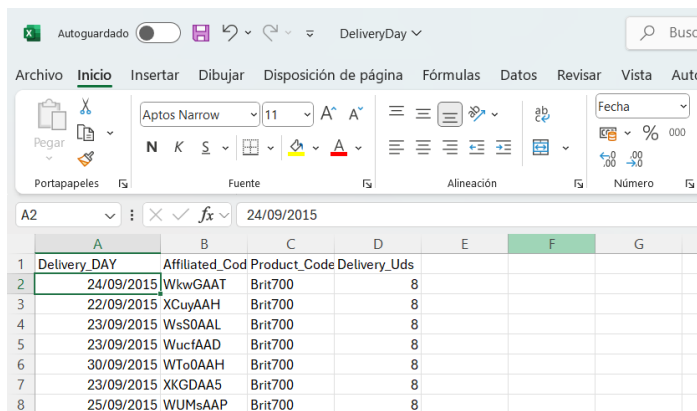
Por tanto, el modelo y la estructura de las tablas quedaría de la siguiente manera:

Se observó que en las tablas DeliveryDay, OoSDay, RouteDay y SalesDay, la variable que debía seguir un formato Fecha no estaba correctamente, así que se aplicó la siguiente fórmula dentro del propio archivo Excel:

	A	B	C	D	E	F	G	H	I
1	Delivery_DAY	Affiliated_Cod	Product_Code	Delivery_Uds					
2		20150924	WkwGAAT	Brit700	8	24/09/2015			
3		20150922	XCuyAAH	Brit700	8	22/09/2015			
4		20150923	WsS0AAL	Brit700	8	23/09/2015			
5		20150923	WuctAAD	Brit700	8	23/09/2015			
6		20150930	WTo0AAH	Brit700	8	30/09/2015			
7		20150923	XKGSAA5	Brit700	8	23/09/2015			

**Ilustración 3: Fórmula para transformar en formato Fecha**

De este modo, ya tendremos el delivery\_DAY en formato fecha. Se hizo también de igual manera con el resto de las tablas.



	A	B	C	D	E	F	G
	Delivery_DAY	Affiliated_Cod	Product_Code	Delivery_Uds			
2	24/09/2015	WkwGAAT	Brit700	8			
3	22/09/2015	XCuyAAH	Brit700	8			
4	23/09/2015	WsSOAAL	Brit700	8			
5	23/09/2015	WucfAAD	Brit700	8			
6	30/09/2015	WT00AAH	Brit700	8			
7	23/09/2015	XKGDAAS	Brit700	8			
8	25/09/2015	WUMsAAP	Brit700	8			

Ilustración 4: Formato Fecha transformado

## 4.2. RSTUDIO

Lo primero que nos dimos cuenta, era que el tipo de variable que detectaba el programa en las fechas era "POSIXct".

```
> str(DeliveryDay)
tibble [615,141 × 4] (S3: tbl_df/tbl/data.frame)
 $ Delivery_DAY      : POSIXct[1:615141], format: "2015-09-24" "2015-09-22" ...
 $ Affiliated_Code   : chr [1:615141] "WkwGAAT" "XCuyAAH" "WsSOAAL" "WucfAAD" ...
 $ Product_Code      : chr [1:615141] "Brit700" "Brit700" "Brit700" "Brit700" ...
 $ Delivery_Uds      : num [1:615141] 8 8 8 8 8 8 8 8 8 ...
```

Ilustración 5: Error en el formato fecha en R

Por ese motivo, se transformaron aquellas tablas que tenían la variable fecha con el siguiente comando.

```
reparto$Delivery_DAY <-as.Date(reparto$Delivery_DAY)
oos$OoS_DAY <-as.Date(oos$OoS_DAY)
ruta$Route_DAY<-as.Date(ruta$Route_DAY)
ventas$Sales_DAY<-as.Date(ventas$Sales_DAY)
```

Ilustración 6: Código para transformar las variables Fecha en R

De este modo, ya se disponía de las variables en el formato adecuado.

```
> str(reparto)
tibble [615,141 × 4] (S3: tbl_df/tbl/data.frame)
 $ Delivery_DAY   : Date[1:615141], format: "2015-09-24" "2015-09-22" ...
 $ Affiliated_Code: chr [1:615141] "wkwGAAT" "XCuyAAH" "WSS0AAL" "WucfAAD" ...
 $ Product_Code   : chr [1:615141] "Brit700" "Brit700" "Brit700" "Brit700" ...
 $ Delivery_Uds    : num [1:615141] 8 8 8 8 8 8 8 8 8 ...
```

Ilustración 7: Variable Fecha corregida en R

### 4.3. DICCIONARIO DE VARIABLES

En este apartado se realiza un diccionario de variables de las tablas que Imperial Brands ha proporcionado:

- **Tabla Affiliated\_Outlets:** En esta tabla se proporciona información sobre los establecimientos afiliados (clientes de Imperial Brands).

VARIABLE	TIPO	REFERENCIA
Affiliated_Code	chr	Código del establecimiento afiliado.
Affiliated_NAME	chr	Nombre del establecimiento afiliado.
POSTALCODE	int	Código postal del establecimiento afiliado.
Engage	int	Nivel de compromiso del establecimiento afiliado.
Management_Cluster	int	Segmento operativo al que pertenece el establecimiento afiliado.
Location	chr	Tipo de localización del establecimiento afiliado.
Tam_m2	chr	Tamaño del establecimiento afiliado (en m <sup>2</sup> ).

- **Tabla Product:** Se incluyen los detalles de los productos comercializados con los clientes.

VARIABLE	TIPO	REFERENCIA
Product_Code	chr	Código del producto comercializado.
SIZE	num	Volumen del producto en unidades de consumo.
Format	chr	Código de formato del producto.

- **Tabla DeliveryDay:** Se corresponde a datos de entregas realizadas o reposición de mercancía, representando las compras realizadas por los estancos.

VARIABLE	TIPO	REFERENCIA
Delivery_DAY	Date	Fecha de entrega de mercancía.
Affiliated_Code	chr	Código del establecimiento afiliado.
Product_Code	chr	Código del producto entregado.
Delivery_Uds	num	Número de unidades entregadas.

- **Tabla OosDay:** Registro diario de incidencias de out of stock (roturas de stock) en los establecimientos.

VARIABLE	TIPO	REFERENCIA
OoS_DAY	Date	Día de rotura de stock.
Affiliated_Code	chr	Código del establecimiento.
Product_Code	chr	Código del producto.

- **Tabla RouteDay:** Información sobre las rutas de entregas realizadas.

VARIABLE	TIPO	REFERENCIA
Route_DAY	Date	Fecha de realización de la ruta.
Affiliated_Code	chr	Código del establecimiento.

- **Tabla SalesDay:** Datos sobre las ventas diarias registradas en cada establecimiento.

VARIABLE	TIPO	REFERENCIA
Sales_DAY	Date	Fecha de las ventas del establecimiento.
Affiliated_Code	chr	Código del establecimiento.
Product_Code	chr	Código del producto vendido.
Sales_Uds	num	Número de unidades vendidas.

## 5. ANÁLISIS EXPLORATORIO DE DATOS

Se cargan las librerías que se van a necesitar a lo largo del trabajo.

```
library(readxl)
library(readr)
library(dplyr)
library(ggplot2)
library(tidyr)
library(lubridate)
library(corrplot)
```

Ilustración 8: Librerías R

Se realiza una copia de todas las tablas para evitar la sobreescritura en las tablas originales.

```
tiendas <- Affiliated_Outlets
reparto <- DeliveryDay
oos <- OoSDay
ruta <- RouteDay
ventas <- SalesDay
producto <- Product
```

Ilustración 9: Código copia de las tablas originales

### 5.1. ANÁLISIS DESCRIPTIVO

En este apartado se emplearán dos funciones básicas para el análisis exploratorio de los datos: `str()` y `summary()`. La función `str()` permite identificar la estructura interna de los objetos en R, especificando el tipo de dato de cada variable. Por ejemplo, las variables con el tipo `Date` corresponden a fechas, `chr` indica cadenas de texto (caracteres), y `num` se refiere a variables numéricas. Por otro lado, la función `summary()` proporciona un resumen estadístico básico que facilita la comprensión general del comportamiento de cada variable.

#### 5.1.1. Tabla `Affiliated_Outlets`

En la tabla `Affiliated_Outlets`, las variables `Affiliated_Code`, `Affiliated_NAME`, `Location` y `Tam_m2` se presentan como cadenas de caracteres, sin valores resumidos estadísticamente. La variable `POSTALCODE`, de tipo numérico entero, muestra un rango que va desde 1.001 hasta 50.800. En cuanto a `Engage`, sus valores oscilan entre 1 y 3, con una media de 2,06 y una mediana igual a 2. Por último, `Management_Cluster`, también de tipo entero, abarca valores entre 0 y 4, con una media de 1,87 y una mediana de 2.

```
> str(tiendas)
tibble [3,583 × 7] (S3: tbl_df/tbl/data.frame)
 $ Affiliated_Code   : chr [1:3583] "WjpeAAD" "Wjs7AAD" "WjvHAAT" "WjwRAAT" ...
 $ Affiliated_NAME   : chr [1:3583] "JULIO-704" "CAMI-741" "PORTUARIOS-047" "PONIENTE-682" ...
 $ POSTALCODE        : num [1:3583] 18613 3801 33212 8940 3815 ...
 $ Engage            : num [1:3583] 2 2 1 2 1 2 1 2 2 2 ...
 $ Management_Cluster : num [1:3583] 4 0 0 1 2 4 2 4 1 0 ...
 $ Location          : chr [1:3583] "VACATIONAL" "ANY" "ANY" "ESCAPE" ...
 $ Tam_m2            : chr [1:3583] ">20m2" ">20m2" "5-10m2" "<2m2" ...

> summary(tiendas)
  Affiliated_Code   Affiliated_NAME   POSTALCODE      Engage      Management_Cluster
Length:3583        Length:3583        Min.   : 1001    Min.   :1.00    Min.   :0.000
Class :character    Class :character    1st Qu.:11510    1st Qu.:2.00    1st Qu.:0.000
Mode  :character    Mode  :character    Median :27001    Median :2.00    Median :2.000
                                Mean  :25129    Mean  :2.06    Mean  :1.874
                                3rd Qu.:36874    3rd Qu.:2.00    3rd Qu.:4.000
                                Max.   :50800    Max.   :3.00    Max.   :4.000

  Location          Tam_m2
Length:3583        Length:3583
Class :character    Class :character
Mode  :character    Mode  :character
```

Ilustración 10: Análisis descriptivo Affiliatet\_Outlets

### 5.1.2. Tabla DeliveryDay

En la tabla DeliveryDay, la variable Delivery\_DAY, de tipo fecha, abarca un período comprendido entre el 9 de marzo y el 16 de octubre de 2015, con una mediana situada el 25 de junio del mismo año. Las variables Affiliated\_Code y Product\_Code son de tipo carácter. En cuanto a Delivery\_Uds, variable numérica que recoge la cantidad entregada, sus valores se sitúan entre -345 y 794 unidades, con una media de 8,55 y una mediana de 8. Es destacable la presencia de valores negativos en esta variable, lo cual podría estar relacionado con devoluciones o correcciones de inventario.

```
> str(reparto)
tibble [615,141 × 4] (S3: tbl_df/tbl/data.frame)
 $ Delivery_DAY      : Date[1:615141], format: "2015-09-24" "2015-09-22" "2015-09-23" ...
 $ Affiliated_Code   : chr [1:615141] "WkwGAAT" "XCuyAAH" "WsSOAAL" "WucfAAD" ...
 $ Product_Code      : chr [1:615141] "Brit700" "Brit700" "Brit700" "Brit700" ...
 $ Delivery_Uds      : num [1:615141] 8 8 8 8 8 8 8 8 8 8 ...

> summary(reparto)
  Delivery_DAY      Affiliated_Code   Product_Code      Delivery_Uds
Min.   :2015-03-09    Length:615141    Length:615141    Min.   : -345.000
1st Qu.:2015-04-29    Class :character    Class :character    1st Qu.:   4.000
Median :2015-06-25    Mode  :character    Mode  :character    Median :   8.000
Mean   :2015-06-24                                Mean  :   8.553
3rd Qu.:2015-08-20                                3rd Qu.:   8.000
Max.   :2015-10-16                                Max.   : 794.000
```

Ilustración 11: Análisis descriptivo DeliveryDay

### 5.1.3. Tabla SalesDay

En la tabla SalesDay, la variable Sales\_DAY, de tipo fecha, comprende un intervalo que va del 9 de marzo al 6 de septiembre de 2015. Las variables Affiliated\_Code y Product\_Code son de tipo carácter. Por su parte, Sales\_Uds, que recoge la cantidad de unidades vendidas, presenta valores entre -15 y 110, con una mediana de 1 unidad y una media de 2,05. Al igual que en la tabla DeliveryDay, se observan valores negativos, lo que podría estar asociado a anulaciones o ajustes en los registros de ventas.

```
> str(ventas)
tibble [1,048,575 × 4] (S3: tbl_df/tbl/data.frame)
 $ Sales_DAY      : Date[1:1048575], format: "2015-08-26" "2015-08-26" "2015-08-28" ...
 $ Affiliated_Code: chr [1:1048575] "WjxnAAD" "WjkSAAT" "WU41AAH" "XCvxAAH" ...
 $ Product_Code   : chr [1:1048575] "Brit627" "Brit627" "Brit700" "Brit700" ...
 $ Sales_Uds      : num [1:1048575] 1 1 7 5 1 1 3 1 1 1 ...

> summary(ventas)
  Sales_DAY      Affiliated_Code  Product_Code      Sales_Uds
Min.   :2015-03-09  Length:1048575  Length:1048575  Min.   :-15.000
1st Qu.:2015-03-28  Class :character  Class :character 1st Qu.:  1.000
Median :2015-04-18  Mode  :character  Mode  :character Median :  1.000
Mean   :2015-04-26                                     Mean  :  2.047
3rd Qu.:2015-05-08                                     3rd Qu.:  3.000
Max.   :2015-09-06                                     Max.   :110.000
```

Ilustración 12: Análisis descriptivo SalesDay

### 5.1.4. Tabla OoSDay

En la tabla OosDay, la variable OoS\_DAY, correspondiente a fechas de incidencias de rotura de stock, abarca desde el 9 de marzo hasta el 4 de octubre de 2015. La mediana se sitúa en el 13 de junio y la media en el 16 de junio del mismo año. Las variables Affiliated\_Code y Product\_Code son de tipo carácter.

```
> str(oos)
tibble [262,278 × 3] (S3: tbl_df/tbl/data.frame)
 $ OoS_DAY        : Date[1:262278], format: "2015-03-26" "2015-03-26" "2015-03-26" ...
 $ Affiliated_Code: chr [1:262278] "WUZ9AAP" "Wl2rAAD" "WqjSAAT" "WjqcAAD" ...
 $ Product_Code   : chr [1:262278] "Natu079" "Dome363" "Natu079" "Dome363" ...

> summary(oos)
  OoS_DAY      Affiliated_Code  Product_Code
Min.   :2015-03-09  Length:262278  Length:262278
1st Qu.:2015-04-15  Class :character  Class :character
Median :2015-06-13  Mode  :character  Mode  :character
Mean   :2015-06-16                                     Mean  :
3rd Qu.:2015-08-18                                     3rd Qu.:
Max.   :2015-10-04                                     Max.   :
```

Ilustración 13: Análisis descriptivo OoSDay



### 5.1.5. Tabla RouteDay

En la tabla RouteDay, la variable Route\_DAY, de tipo fecha, cubre un período que va del 9 de marzo al 11 de diciembre de 2015. La variable Affiliated\_Code es de tipo carácter.

```
> str(ruta)
tibble [94,858 × 2] (S3: tbl_df/tbl/data.frame)
 $ Route_DAY      : Date[1:94858], format: "2015-03-09" "2015-03-09" "2015-03-09" ...
 $ Affiliated_Code: chr [1:94858] "WjeGAAT" "WjeJAAT" "WjemAAD" "WjeuAAD" ...
> summary(ruta)
  Route_DAY      Affiliated_Code
Min.   :2015-03-09   Length:94858
1st Qu.:2015-05-19   Class :character
Median :2015-07-27   Mode  :character
Mean   :2015-07-25
3rd Qu.:2015-10-01
Max.   :2015-12-11
```

Ilustración 14: Análisis descriptivo RouteDay

### 5.1.6. Tabla Product

En la tabla Product, las variables Product\_Code y Format son de tipo carácter, mientras que SIZE es una variable numérica cuyos valores oscilan entre 29 y 1.415 unidades de consumo. La mediana se sitúa en 142, y la media en 214, lo que indica cierta dispersión en los volúmenes de los productos.

```
> str(producto)
tibble [60 × 3] (S3: tbl_df/tbl/data.frame)
 $ Product_Code: chr [1:60] "Brit090" "Brit555" "Brit627" "Brit700" ...
 $ SIZE       : num [1:60] 142 142 85 142 71 29 255 317 199 190 ...
 $ Format      : chr [1:60] "ASL" "ASL" "ASL" "ASL" ...
> summary(producto)
  Product_Code      SIZE      Format
Length:60         Min.   : 29   Length:60
Class :character   1st Qu.: 85   Class :character
Mode  :character   Median : 142  Mode  :character
                        Mean  : 214
                        3rd Qu.: 227
                        Max.   :1415
```

Ilustración 15: Análisis descriptivo Product

Como se ha observado, en las tablas ventas y reparto contenían algunos valores negativos. En el caso de las ventas, debidos a descuadres por pérdidas, hurtos, etc. Por otro lado, en las entregas debido a correcciones. Por ese motivo, se ha optado por sustituir esos datos negativos filtrándolos por los valores positivos.

Además, en ninguna de las tablas que la empresa nos ha facilitado se han encontrado valores nulos, lo que facilita la limpieza de los datos.

```
#Eliminamos los valores negativos  
ventas <- ventas %>% filter(Sales_Uds >= 0)  
reparto <- reparto %>% filter(Delivery_Uds >= 0)
```

Ilustración 16: Código para eliminar los valores negativos

## 5.2. ANÁLISIS DE PREPROCESAMIENTO ÁGIL: GRÁFICOS Y MATRIZ DE CORRELACIÓN

A continuación, con los datos existentes, se llevó a cabo la creación de gráficos con el objetivo de poder extraer las primeras conclusiones sobre ellos mediante un preprocesamiento ágil:

### 5.2.1. Gráficos de caja: Ventas, entregas y total de ventas por tienda

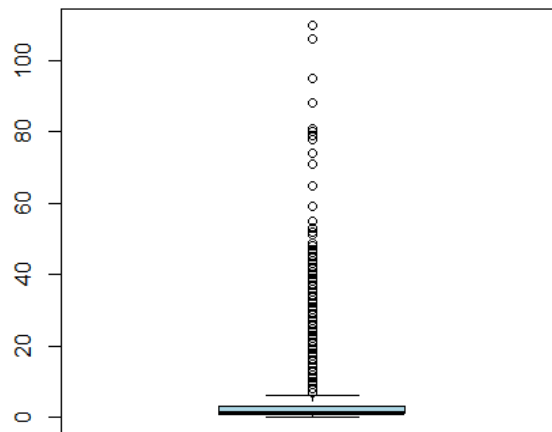
En primer lugar, se realizó un diagrama de caja de las unidades vendidas por producto y día para poder conocer su distribución. Las primeras conclusiones que se pueden extraer al respecto son que los datos presentan una elevada dispersión, con una cantidad de outliers significativa.

Sin embargo, gran parte de la muestra se encuentra concentrada en valores bastante cercanos y próximos a 0, donde se sitúa la mediana. Esto indicaría que la mayoría de los establecimientos venden relativamente pocas unidades de cada producto diariamente pero sí que hay algunos en concreto que suelen concentrar una mayor presencia de outliers, lo que podría hacer referencia a tiendas con mayor número de clientes o productos con una mayor demanda.

```
# Outliers en ventas  
boxplot(ventas$Sales_Uds, main = "Boxplot de Ventas - Detección de Outliers", col = "lightblue")
```

Ilustración 17: Código gráfico de Caja de Ventas

**Boxplot de Ventas - Detección de Outliers**



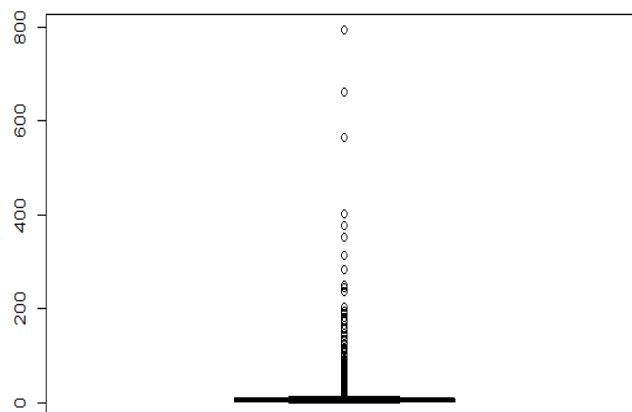
**Ilustración 18: Gráfico de Caja de Ventas**

En el caso de los datos de reposición de las mercancías por día, establecimiento y tipo de producto se encuentra un fenómeno similar al caso anteriormente mencionado, con una gran dispersión en los datos, pero debido al mismo fenómeno de variabilidad en la demanda de los distintos productos, número de clientes del establecimiento y necesidades de reposición concretas de cada día.

```
# Outliers en entregas  
boxplot(reparto$Delivery_Uds, main = "Boxplot de Entregas - Detección de Outliers", col = "lightgreen")
```

**Ilustración 19: Código gráfico de Caja de Entregas**

**Boxplot de Entregas - Detección de Outliers**



**Ilustración 20: Gráfico de Caja de Entregas**

Para visualizar los datos de manera clara, se realizó otro diagrama de caja del sumatorio de unidades vendidas por tienda. En este contexto, se observan unos datos mucho más concentrados y que aportan más información. En ellos, podemos observar un rango intercuartílico relativamente simétrico, con una gran concentración de valores entre las 100 y 900 unidades vendidas por establecimiento aproximadamente y la mediana situándose en torno a 400. Además de este fenómeno, se observa la presencia de valores atípicos, pero en mucha menor medida que en los gráficos anteriormente analizados.

```
# Boxplot de ventas por tienda
ventas %>%
  group_by(Affiliated_Code) %>%
  summarise(total_ventas = sum(Sales_Uds)) %>%
  ggplot(aes(x = "", y = total_ventas)) +
  geom_boxplot(fill = "lightblue") +
  theme_minimal() +
  labs(title = "Boxplot del Total de Ventas por Tienda", y = "Total Ventas")
```

Ilustración 21: Código gráfico de Caja de Ventas por Tiendas

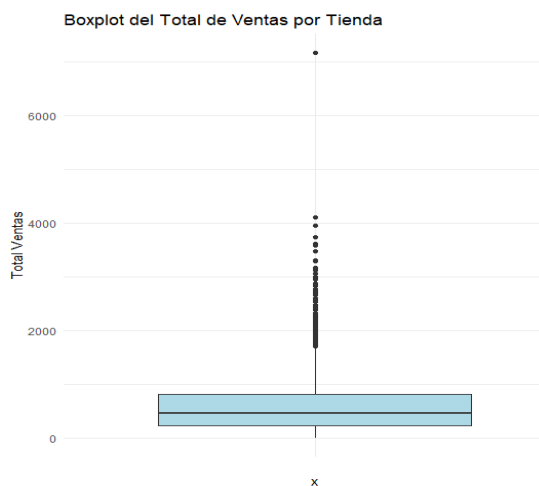


Ilustración 22: Gráfico de Caja de Ventas por Tiendas

### 5.2.2. Histograma de unidades vendidas y entregadas

Si observamos las distribuciones de las unidades vendidas y entregadas, comprobamos lo ya visto anteriormente, pero con mayor claridad.

En el caso de las unidades vendidas encontramos la mayor concentración muy cerca de 0 y van disminuyendo a medida que se acercan a 15, por lo que rara vez los clientes compran más de 15 unidades de un mismo producto en un día concreto.

```
# Histograma de ventas
ggplot(ventas, aes(x = Sales_Uds)) +
  geom_histogram(bins = 50, fill = "skyblue", color = "black") +
  theme_minimal() +
  labs(title = "Histograma de Unidades Vendidas", x = "Unidades Vendidas", y = "Frecuencia")
```

Ilustración 23: Código gráfico Histograma de Ventas

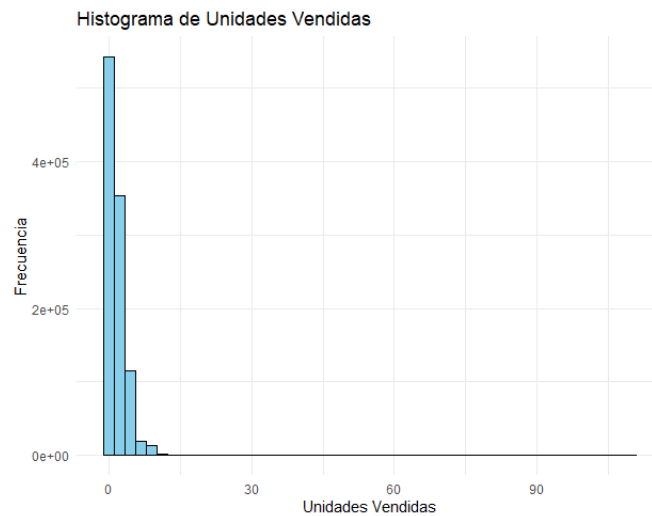


Ilustración 24: Gráfico Histograma de Ventas

En el caso de las unidades entregadas observamos exactamente el mismo patrón, pero con unidades de mayor magnitud, disminuyendo cada vez más medida que se acercan a 100 unidades entregadas.

```
# Histograma de entregas
ggplot(reparto, aes(x = Delivery_Uds)) +
  geom_histogram(bins = 50, fill = "lightgreen", color = "black") +
  theme_minimal() +
  labs(title = "Histograma de Unidades Entregadas", x = "Unidades Entregadas", y = "Frecuencia")
```

Ilustración 25: Código gráfico Histograma de Entregas

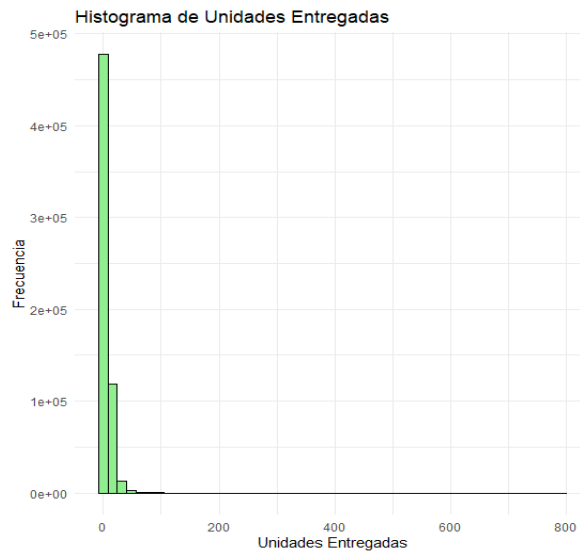


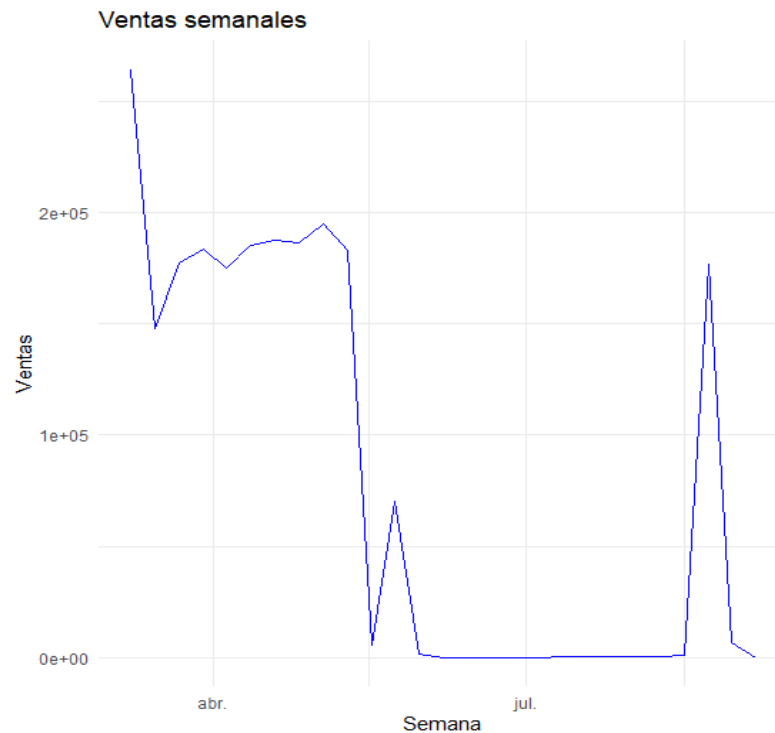
Ilustración 26: Gráfico Histograma de Entregas

### 5.2.3. Ventas y entregas semanales

Para representar la evolución en las ventas semanales, se ha creado un gráfico que representa la suma de unidades vendidas en el total de los establecimientos. En él se observa un patrón de ventas bastante irregular a lo largo del tiempo, con un mayor volumen en las primeras semanas del año para un posterior gran descenso durante los meses de verano y posterior ascenso de nuevo al final de ese periodo.

```
# Evolución de ventas en el tiempo
ventas %>%
  mutate(Semana = floor_date(Sales_DAY, "week")) %>%
  group_by(Semana) %>%
  summarise(Ventas_Semana = sum(Sales_Uds)) %>%
  ggplot(aes(x = Semana, y = Ventas_Semana)) +
  geom_line(color = "blue") +
  theme_minimal() +
  labs(title = "Ventas semanales", x = "Semana", y = "Ventas")
```

Ilustración 27: Código gráfico Evolución de Ventas en el tiempo



**Ilustración 28: Gráfico Evolución de Ventas en el tiempo**

En el caso de la evolución de entregas de productos a los distintos establecimientos observamos un patrón distinto. Y es que, el número de entregas semanales oscila entre las 12.000 y las 20.000 unidades durante todos los meses. En general, se encuentra un patrón semanal ciertamente irregular, pero sin embargo se sigue pudiendo apreciar un menor volumen de entregas durante los meses de verano.

En este caso, también se puede apreciar un gran descenso repentino aproximadamente a principios de abril. Si consultamos el calendario del año 2015, que es de cuando son los datos, se observa que en ese año el periodo de Semana Santa transcurrió durante la semana del 29 de marzo al 5 de abril, lo que explicaría ese descenso repentino en los repartos y posterior recuperación a la siguiente semana.

```
# Evolución de entregas en el tiempo
reparto %>%
  mutate(Semana = floor_date(Delivery_DAY, "week")) %>%
  group_by(Semana) %>%
  summarise(Entregas_Semana = sum(Delivery_Uds)) %>%
  ggplot(aes(x = Semana, y = Entregas_Semana)) +
  geom_line(color = "green") +
  theme_minimal() +
  labs(title = "Entregas semanales", x = "Semana", y = "Entregas")
```

Ilustración 29: Código gráfico Evolución de Entregas en el tiempo

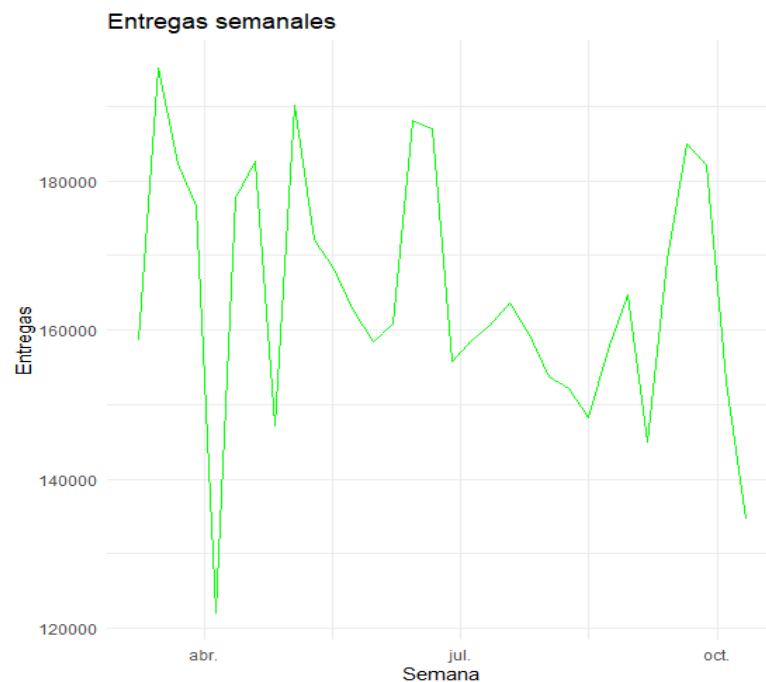


Ilustración 30: Gráfico Evolución de Entregas en el tiempo

#### 5.2.4. Tiempo medio entre entregas por tienda

Los resultados representan el número medio de días entre entregas por establecimiento. Sus valores oscilan entre 0 y 62 días, con una media de 2,64 y una mediana de 1,42. El primer cuartil se sitúa en 0,91 días y el tercero en 2,59, lo que indica que la mayoría de los establecimientos reciben entregas con una frecuencia relativamente alta (cada dos días o menos). Se detectan también 4 valores ausentes (NA's), que podrían corresponder a casos con una única entrega registrada o datos incompletos.



```
# Tiempo medio entre entregas por tienda
frecuencia_entregas <- reparto %>%
  arrange(Affiliated_Code, Delivery_DAY) %>%
  group_by(Affiliated_Code) %>%
  mutate(dias_entre = as.numeric(Delivery_DAY - lag(Delivery_DAY))) %>%
  summarise(media_dias_entre_entregas = mean(dias_entre, na.rm = TRUE))

summary(frecuencia_entregas$media_dias_entre_entregas)
```

Ilustración 31: Código tiempo medio entre entregas por tienda

```
> summary(frecuencia_entregas$media_dias_entre_entregas)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
 0.0000  0.9093  1.4183  2.6357  2.5854 62.0000     4
```

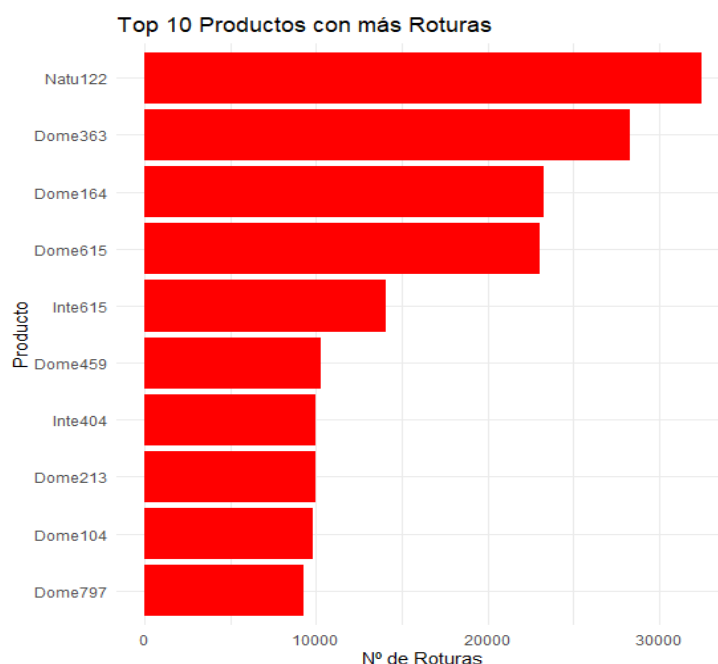
Ilustración 32: Tiempo medio entre entregas por tienda

#### 5.2.5. Top 10 productos con más roturas

En el caso del análisis de los productos que más roturas sufren hemos encontrado que hay 4 que han sufrido por encima de 20.000 roturas, bastante por encima del resto. Estos productos son Natu122, con más de 30.000; Dome363, con alrededor de 28.000 y finalmente Dome163 y Dome615 con alrededor de 23.000 roturas cada uno.

```
# Roturas por producto
oos %>%
  group_by(Product_Code) %>%
  summarise(Roturas = n()) %>%
  arrange(desc(Roturas)) %>%
  top_n(10, Roturas) %>%
  ggplot(aes(x = reorder(Product_Code, Roturas), y = Roturas)) +
  geom_bar(stat = "identity", fill = "red") +
  coord_flip() +
  theme_minimal() +
  labs(title = "Top 10 Productos con más Roturas", x = "Producto", y = "Nº de Roturas")
```

Ilustración 33: Código gráfico top 10 Productos con más Roturas



**Ilustración 34: Gráfico top 10 Productos con más Roturas**

#### 5.2.6. Top 10 tiendas con más roturas

En el análisis de las tiendas que han experimentado un mayor número de roturas de stock, se observa que el establecimiento XCWMAA5 destaca claramente con cerca de 750 incidencias, situándose como el que ha reportado más roturas durante el periodo analizado. Le siguen WtdoAAD y WuaxAAD, ambas con cifras cercanas a las 650 roturas, también significativamente superiores al resto. A partir del cuarto puesto, con WTIWAAX alrededor de 550 incidencias, la tendencia comienza a descender de forma progresiva. Las demás tiendas del top 10 presentan valores más homogéneos, con un volumen de roturas en torno a las 450 unidades, lo que sugiere que el impacto de las roturas está más concentrado en unas pocas localizaciones concretas.

```
# Roturas por tienda
oos %>%
  group_by(Affiliated_Code) %>%
  summarise(Roturas = n()) %>%
  arrange(desc(Roturas)) %>%
  top_n(10, Roturas) %>%
  ggplot(aes(x = reorder(Affiliated_Code, Roturas), y = Roturas)) +
  geom_bar(stat = "identity", fill = "orange") +
  coord_flip() +
  theme_minimal() +
  labs(title = "Top 10 Tiendas con más Roturas", x = "Tienda", y = "Nº de Roturas")
```

Ilustración 35: Código gráfico top 10 Tiendas con más Roturas

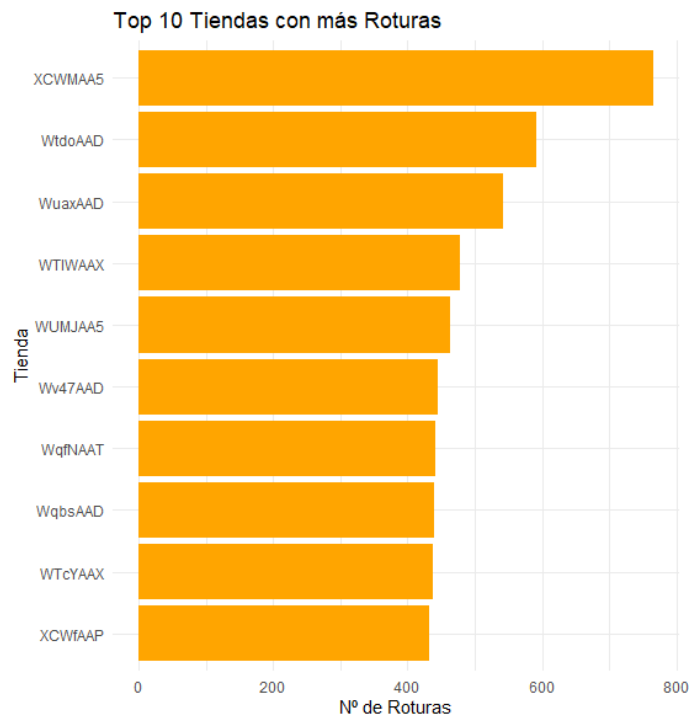


Ilustración 36: Gráfico top 10 Productos con más Roturas

### 5.2.7. Matriz de correlación

La matriz usa un esquema de color azul oscuro para las correlaciones altas y colores más claros para las más bajas.

Sales\_Total y Delivery\_Total: presentan una correlación muy alta (0.98), lo cual indica que el volumen de ventas está estrechamente relacionado con el volumen de entregas. Esta relación es esperable, ya que mayores entregas suelen corresponder a mayores ventas.

Sales\_Total y Engage: tienen una correlación moderada (0.55). Esto sugiere que los establecimientos con mayor nivel de compromiso tienden a registrar mayores ventas, aunque no de forma determinante.

Delivery\_Total y Engage: muestran una correlación muy similar (0.56), reforzando la idea anterior, pero en relación con el volumen de entregas.

Management\_Cluster muestra bajas correlaciones ( $\leq 0.12$ ) con todas las demás variables. Esto indica que el segmento operativo al que pertenece un establecimiento no está directamente relacionado con sus ventas, entregas o nivel de compromiso. Su impacto, si lo hay, podría ser más estructural o categórico que cuantitativo.

```
# MATRIZ DE CORRELACIÓN
# Cruzamos datos de ventas, reparto y tiendas
ventas_reparto <- ventas %>%
  group_by(Affiliated_Code) %>%
  summarise(Sales_Total = sum(Sales_Uds))

reparto_total <- reparto %>%
  group_by(Affiliated_Code) %>%
  summarise(Delivery_Total = sum(Delivery_Uds))

correlacion_data <- ventas_reparto %>%
  left_join(reparto_total, by = "Affiliated_Code") %>%
  left_join(tiendas %>% select(Affiliated_Code, Engage, Management_Cluster), by = "Affiliated_Code") %>%
  drop_na()

# Matriz de correlación
cor_matrix <- cor(correlacion_data %>% select(-Affiliated_Code))

# Graficar la matriz
corrplot(cor_matrix, method = "color", addCoef.col = "black", number.cex = 0.7, tl.cex = 0.8)
```

Ilustración 37: Código Matriz de Correlación

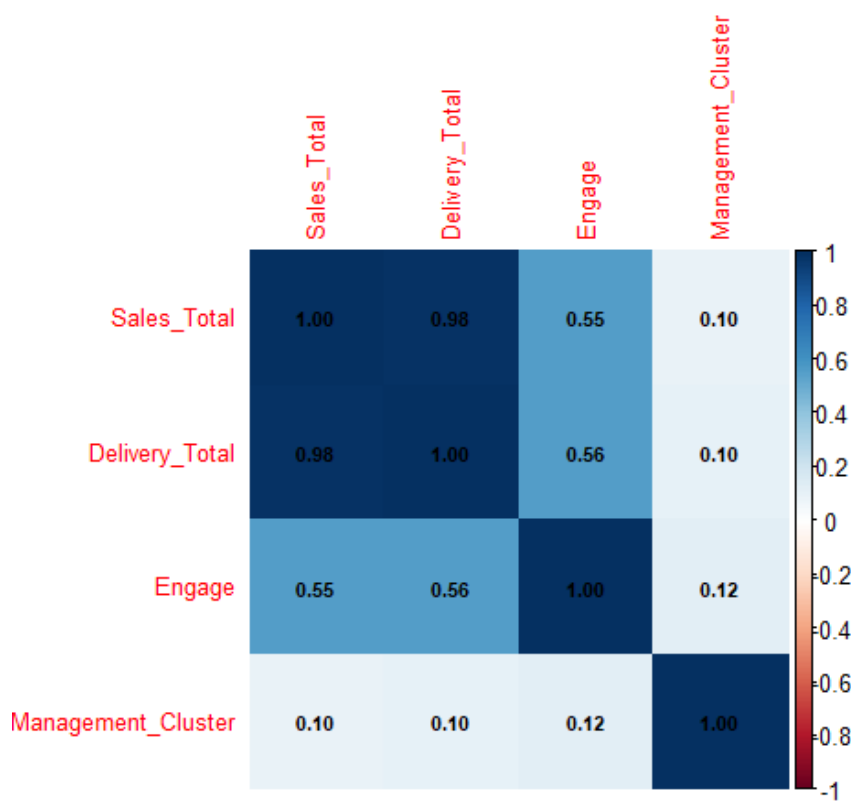
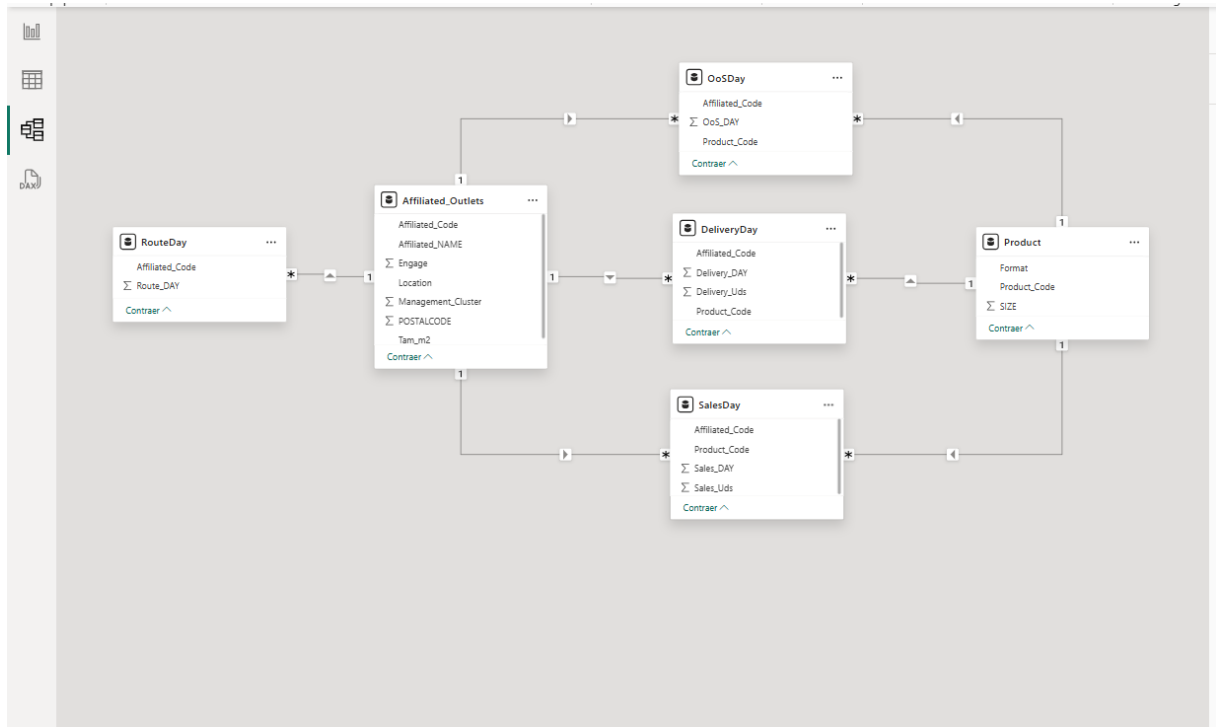


Ilustración 38: Matriz de Correlación

## 6. MODELADO DE DATOS EN POWER BI

### 6.1. TRATAMIENTO Y LIMPIEZA DE LOS DATOS

Cargamos todas las tablas a PowerBI y administramos las relaciones.



**Ilustración 39: Modelo de conexiones de las Tablas en PowerBI**

Ahora, se va a transformar todas las variables que han adquirido formato Número de manera predeterminada, a formato Texto en caso de que sean variables categóricas.

Nombre: POSTALCODE  
Tipo de datos: Texto  
Formato: Texto  
Resumen: No resumir  
Categoría de datos: Sin clasificar

Ordenar por columna: Ordenar  
Grupos de datos: Grupos  
Administrar relaciones: Relaciones  
Nueva columna: Cálculos

Tabla: Affiliated\_Outlets (3583 filas) Columna: POSTALCODE (2360 valores distintos)

Affiliated_Code	Affiliated_NAME	POSTALCODE	Engage	Management_Cluster	Location	Tam_m2
W0GAAAL	CONSTITUCION-341	41804	2	4	VILLAGE	5-10m2
WjuuAAD	SANTA-140	32300	2	4	VILLAGE	5-10m2
WizhAAD	JOSE-721	40005	2	4	VILLAGE	5-10m2
WkbmAAD	PABLO-770	41805	2	4	VILLAGE	5-10m2
W3QAAT	COMARQUES-437	43740	2	4	VILLAGE	5-10m2
W0IAAT	EXTREMADURA-666	41970	2	4	VILLAGE	5-10m2
WqZvAAL	NAVARRA-455	26141	2	4	VILLAGE	5-10m2
WQFAA1	HERMANOS-696	28970	2	4	VILLAGE	5-10m2
WskwAAD	VECINAL-508	7013	2	4	VILLAGE	5-10m2
W7IAAD	BERNARDO-788	36670	2	4	VILLAGE	5-10m2
WqHQAAT	DOCTORS-407	25220	2	4	VILLAGE	5-10m2
WtpBAAX	ALFEREZ-281	21890	2	4	VILLAGE	5-10m2
WtnKAAT	RAMON-575	46740	2	4	VILLAGE	5-10m2
W4IAAD	ESPAÑA-132	28990	2	4	VILLAGE	5-10m2
WqH6AAL	GRAN-530	08600	2	4	VILLAGE	5-10m2
WtdNAAX	NOVA-104	17185	2	4	VILLAGE	5-10m2
WTawAAH	BERNINCHES-223	19180	2	4	VILLAGE	5-10m2
Wtc7AAD	LUIS-311	02660	2	4	VILLAGE	5-10m2
WjpsAAD	FEDERICO-690	14720	2	4	VILLAGE	5-10m2
WIPCAA1	SALVADOR-750	07500	2	4	VILLAGE	5-10m2
WtnhAAH	ANDALUCIA-225	11560	2	4	VILLAGE	5-10m2
WkuIAAT	RAMON-702	03400	2	4	VILLAGE	5-10m2
WIZVAAT	JERONIMO-650	30800	2	4	VILLAGE	5-10m2
W3MAAT	GABRIEL-374	46780	2	4	VILLAGE	5-10m2
WslJAAL	MAJOR-486	08394	2	4	VILLAGE	5-10m2
WkuZAAL	TAIALA-611	17007	2	4	VILLAGE	5-10m2
WTeGAAX	ALAMEDA-021	21003	2	4	VILLAGE	5-10m2

Ilustración 40: Transformación del tipo de variable de Número a Texto en PowerBI

A continuación, para terminar con el tratamiento de los datos, a partir de la variable POSTALCODE que se encuentra en la Tabla Affiliated\_Outlets, se va a generar una nueva columna que nos indicará la provincia.

Tabla: Affiliated\_Outlets (3583 filas) Columna: POSTALCODE (2360 valores distintos)

Affiliated_Code	Affiliated_NAME	POSTALCODE	Engage	Management_Cluster	Location	Tam_m2	Ubicacion_Provincias	Provincia	Ubicacion_CP
W0GAAAL	CONSTITUCION-341	41804	2	4	VILLAGE	5-10m2	España, Sevilla	Sevilla	España, 41804
WjuuAAD	SANTA-140	32300	2	4	VILLAGE	5-10m2	España, Ourense	Ourense	España, 32300
WizhAAD	JOSE-721	40005	2	4	VILLAGE	5-10m2	España, Segovia	Segovia	España, 40005
WkbmAAD	PABLO-770	41805	2	4	VILLAGE	5-10m2	España, Sevilla	Sevilla	España, 41805
W3QAAT	COMARQUES-437	43740	2	4	VILLAGE	5-10m2	España, Tarragona	Tarragona	España, 43740
W0IAAT	EXTREMADURA-666	41970	2	4	VILLAGE	5-10m2	España, Sevilla	Sevilla	España, 41970
WqZvAAL	NAVARRA-455	26141	2	4	VILLAGE	5-10m2	España, La Rioja	La Rioja	España, 26141
WQFAA1	HERMANOS-696	28970	2	4	VILLAGE	5-10m2	España, Madrid	Madrid	España, 28970
WskwAAD	VECINAL-508	07013	2	4	VILLAGE	5-10m2	España, Baleares	Baleares	España, 07013
W7IAAD	BERNARDO-788	36670	2	4	VILLAGE	5-10m2	España, Pontevedra	Pontevedra	España, 36670
WqHQAAT	DOCTORS-407	25220	2	4	VILLAGE	5-10m2	España, Lleida	Lleida	España, 25220
WtpBAAX	ALFEREZ-281	21890	2	4	VILLAGE	5-10m2	España, Huelva	Huelva	España, 21890
WtnKAAT	RAMON-575	46740	2	4	VILLAGE	5-10m2	España, Valencia	Valencia	España, 46740
W4IAAD	ESPAÑA-132	28990	2	4	VILLAGE	5-10m2	España, Madrid	Madrid	España, 28990
WqH6AAL	GRAN-530	08600	2	4	VILLAGE	5-10m2	España, Barcelona	Barcelona	España, 08600
WtdNAAX	NOVA-104	17185	2	4	VILLAGE	5-10m2	España, Girona	Girona	España, 17185
WTawAAH	BERNINCHES-223	19180	2	4	VILLAGE	5-10m2	España, Guadalajara	Guadalajara	España, 19180
Wtc7AAD	LUIS-311	02660	2	4	VILLAGE	5-10m2	España, Albacete	Albacete	España, 02660
WjpsAAD	FEDERICO-690	14720	2	4	VILLAGE	5-10m2	España, Córdoba	Córdoba	España, 14720
WIPCAA1	SALVADOR-750	07500	2	4	VILLAGE	5-10m2	España, Baleares	Baleares	España, 07500
WtnhAAH	ANDALUCIA-225	11560	2	4	VILLAGE	5-10m2	España, Cádiz	Cádiz	España, 11560
WkuIAAT	RAMON-702	03400	2	4	VILLAGE	5-10m2	España, Alicante	Alicante	España, 03400
WIZVAAT	JERONIMO-650	30800	2	4	VILLAGE	5-10m2	España, Murcia	Murcia	España, 30800
W3MAAT	GABRIEL-374	46780	2	4	VILLAGE	5-10m2	España, Valencia	Valencia	España, 46780
WslJAAL	MAJOR-486	08394	2	4	VILLAGE	5-10m2	España, Barcelona	Barcelona	España, 08394
WkuZAAL	TAIALA-611	17007	2	4	VILLAGE	5-10m2	España, Girona	Girona	España, 17007
WTeGAAX	ALAMEDA-021	21003	2	4	VILLAGE	5-10m2	España, Huelva	Huelva	España, 21003

Ilustración 41: Creación de la nueva columna Provincia

Para que PowerBI entienda que el nombre de las provincias hace referencia a provincias de España, hay que especificarlo. Por ese motivo, se crea una nueva columna llamada Ubicación\_Provincia.

1 Ubicación_Provincias = "España, "&Affiliated_Outlets_Original[Provincia]									
Affiliated_Code	Affiliated_NAME	POSTALCODE	Engage	Management_Cluster	Location	Tam_m2	Ubicación_Provincias	Provincia	Ubicación_CP
W0pGAAAL	CONSTITUCION-341	41804	2	4	VILLAGE	5-10m2	España, Sevilla	Sevilla	España, 41804
W0uAAAD	SANTA-140	32300	2	4	VILLAGE	5-10m2	España, Ourense	Ourense	España, 32300
W02hAAD	JOSE-721	40005	2	4	VILLAGE	5-10m2	España, Segovia	Segovia	España, 40005
W0bmAAD	PABLO-770	41805	2	4	VILLAGE	5-10m2	España, Sevilla	Sevilla	España, 41805
W03QAAT	COMARQUES-437	43740	2	4	VILLAGE	5-10m2	España, Tarragona	Tarragona	España, 43740
W00IAAT	EXTREMADURA-666	41970	2	4	VILLAGE	5-10m2	España, Sevilla	Sevilla	España, 41970
W0ZvAAL	NAVARRA-455	26141	2	4	VILLAGE	5-10m2	España, La Rioja	La Rioja	España, 26141
W0QFAA1	HERMANOS-696	28970	2	4	VILLAGE	5-10m2	España, Madrid	Madrid	España, 28970
W0kwAAD	VEGICAL-508	07013	2	4	VILLAGE	5-10m2	España, Baleares	Baleares	España, 07013
W07IAAD	BERNARDO-788	36670	2	4	VILLAGE	5-10m2	España, Pontevedra	Pontevedra	España, 36670
W0hQAAT	DOCTORS-407	25220	2	4	VILLAGE	5-10m2	España, Lleida	Lleida	España, 25220
W0gBAAX	ALFEREZ-281	21890	2	4	VILLAGE	5-10m2	España, Huelva	Huelva	España, 21890
W0mKAAT	RAMON-575	46740	2	4	VILLAGE	5-10m2	España, Valencia	Valencia	España, 46740
W04IAAD	ESPAÑA-132	28990	2	4	VILLAGE	5-10m2	España, Madrid	Madrid	España, 28990
W06GAAL	GRAN-530	08600	2	4	VILLAGE	5-10m2	España, Barcelona	Barcelona	España, 08600
W0dNAAX	NOVA-104	17185	2	4	VILLAGE	5-10m2	España, Girona	Girona	España, 17185
W0awAAH	BERNINCHES-223	19180	2	4	VILLAGE	5-10m2	España, Guadalajara	Guadalajara	España, 19180
W0c7AAD	LUIS-311	02660	2	4	VILLAGE	5-10m2	España, Albacete	Albacete	España, 02660
W0psAAD	FEDERICO-690	14720	2	4	VILLAGE	5-10m2	España, Córdoba	Córdoba	España, 14720
W0PCAAT	SALVADOR-750	07500	2	4	VILLAGE	5-10m2	España, Baleares	Baleares	España, 07500
W0TnhAAH	ANDALUCIA-225	11560	2	4	VILLAGE	5-10m2	España, Cádiz	Cádiz	España, 11560
W0uIAAT	RAMON-702	03400	2	4	VILLAGE	5-10m2	España, Alicante	Alicante	España, 03400
W0ZVAAT	JERONIMO-650	30800	2	4	VILLAGE	5-10m2	España, Murcia	Murcia	España, 30800
W03MAAT	GABRIEL-374	46780	2	4	VILLAGE	5-10m2	España, Valencia	Valencia	España, 46780
W0uJAAL	MAJOR-486	08394	2	4	VILLAGE	5-10m2	España, Barcelona	Barcelona	España, 08394
W0LZAAL	TAIALA-611	17007	2	4	VILLAGE	5-10m2	España, Girona	Girona	España, 17007
W0eGAAX	ALAMEDA-021	21003	2	4	VILLAGE	5-10m2	España, Huelva	Huelva	España, 21003

Tabla: Affiliated Outlets Original (3583 filas) Columnas: Ubicación Provincias (48 valores distintos)

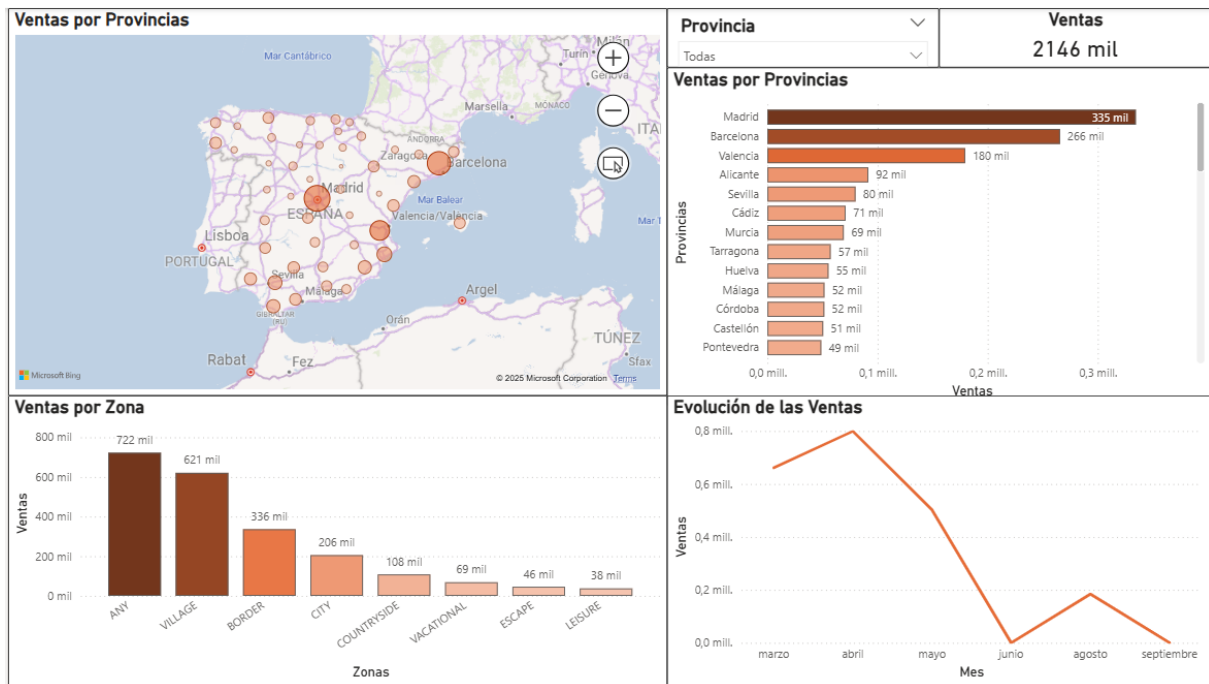
Ilustración 42: Creación de la nueva columna especificando España como país

## 6.2. VISUALIZACIÓN DE LOS DATOS

Una vez limpiados y tratado los datos, se han realizado dos páginas de dashboard para poder representar los datos. El primer dashboard, muestra un análisis de las ventas por provincias. El segundo, muestra un análisis de ventas en función del tipo de producto.



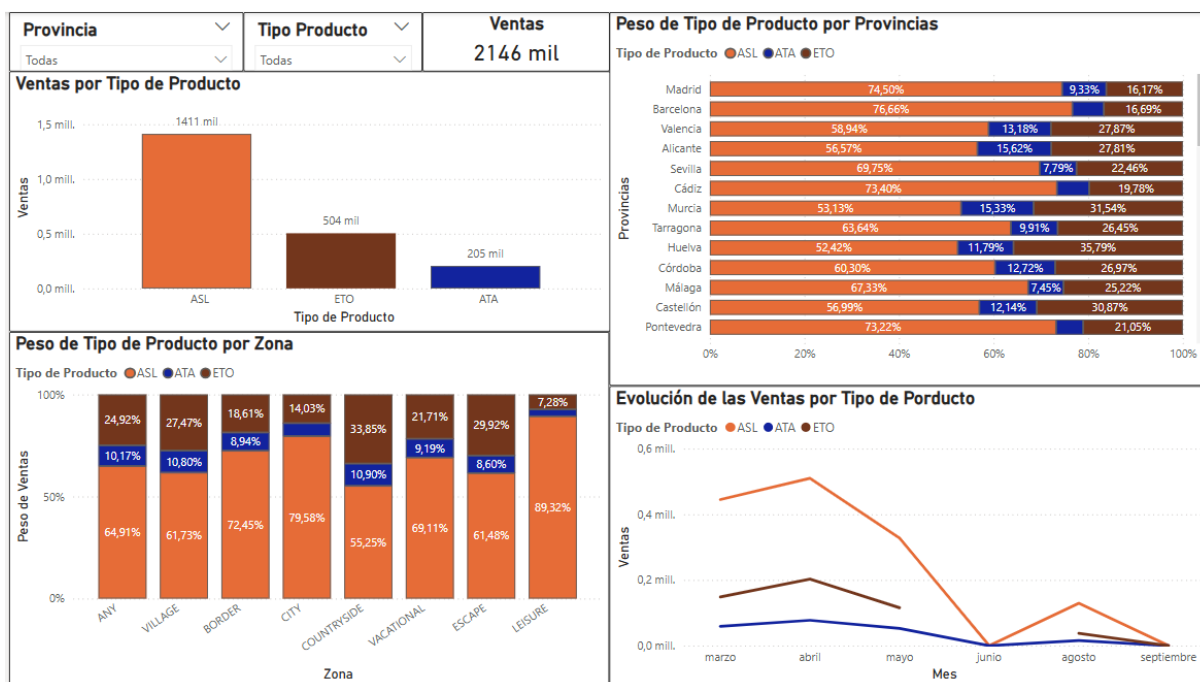
### 6.2.1. Dashboard 1: Análisis de ventas



**Ilustración 43: Dashboard 1**

Esta página muestra cuatro gráficos, una tarjeta (KPI) y una tarjeta de segmentación de los datos. El primer gráfico (arriba izquierda) muestra un mapa de burbujas. Cada burbuja es diferente en función de la dimensión del tamaño de ventas que se han producido en las distintas provincias de España. Para poder ver de una manera más gráfica el volumen de ventas por provincia, arriba derecha se encuentra el gráfico de barras apiladas que muestra el volumen de ventas por provincias. Abajo izquierda se encuentra un gráfico de columnas apiladas. Este gráfico muestra la cantidad de ventas por el tipo de zona. Abajo derecha, el gráfico de líneas muestra la evolución de las ventas a lo largo del tiempo. La tarjeta muestra la suma de ventas totales. Por último, la tarjeta de segmentación de los datos permite seleccionar únicamente las provincias en función a los intereses del usuario.

## 6.2.2. Dashboard 2: Análisis de Productos



**Ilustración 44: Dashboard 2**

Esta página muestra cuatro gráficos, una tarjeta (KPI) y dos tarjetas de segmentación de los datos. El gráfico de columnas apiladas (arriba izquierda), muestra las ventas en función del tipo de producto. Arriba derecha con el gráfico de barras 100% apiladas se puede observar el peso que tiene cada tipo de producto en función de la provincia. El gráfico de columnas 100% apiladas (abajo izquierda), muestra el peso de cada tipo de producto en función del tipo de zona. Abajo derecha, el gráfico de líneas muestra la evolución de ventas a lo largo del tiempo de cada tipo de producto. La tarjeta muestra la suma de ventas totales. Por último, las tarjetas de segmentación de los datos nos permiten seleccionar únicamente las provincias en función de nuestros intereses y el tipo de producto.

### 6.3. RESULTADOS OBTENIDOS EN POWER BI

Gracias a la representación gráfica de los datos, es posible extraer información relevante que puede aportar valor en futuras tomas de decisiones. No obstante, es importante señalar que los datos fueron proporcionados sin un diccionario que describa el significado de cada variable. Por ello, las interpretaciones realizadas se basan en estimaciones razonables sobre el posible contenido y propósito de dichas variables.

#### 6.3.1. Ventas por provincias

En el primer gráfico, podemos observar que las provincias que más ventas tienen son Madrid y Barcelona por mucha diferencia. Seguidas por las provincias de la Comunidad Valenciana como Valencia y Alicante. Por otro lado, las provincias de Zamora, Teruel y Soria, su volumen de ventas es muy reducido.

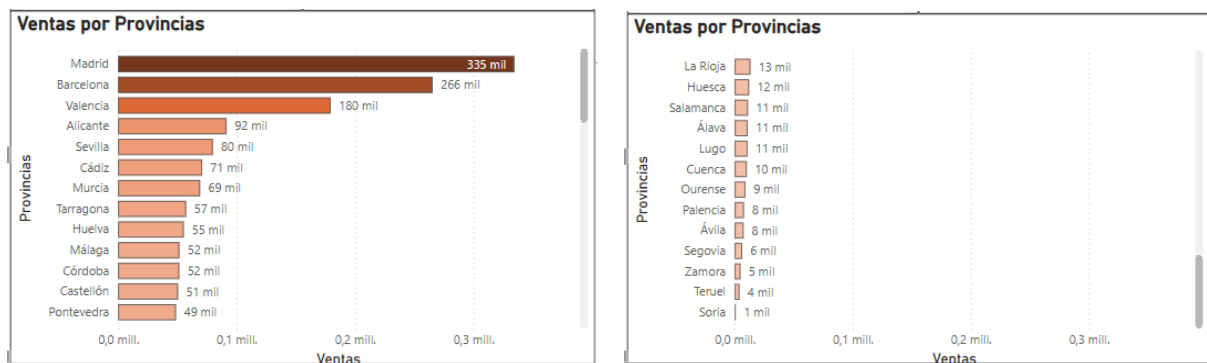
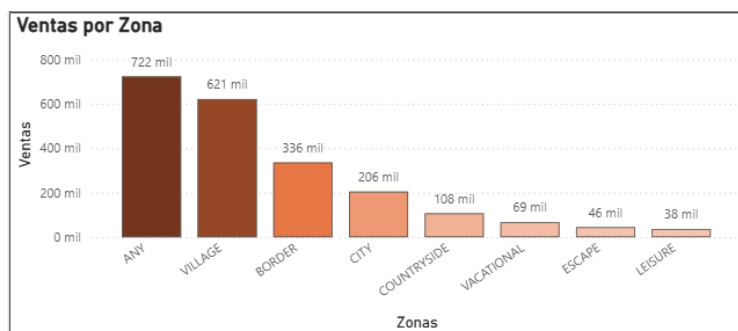


Ilustración 45: Gráficos de Ventas por Provincias

#### 6.3.2. Ventas por zona

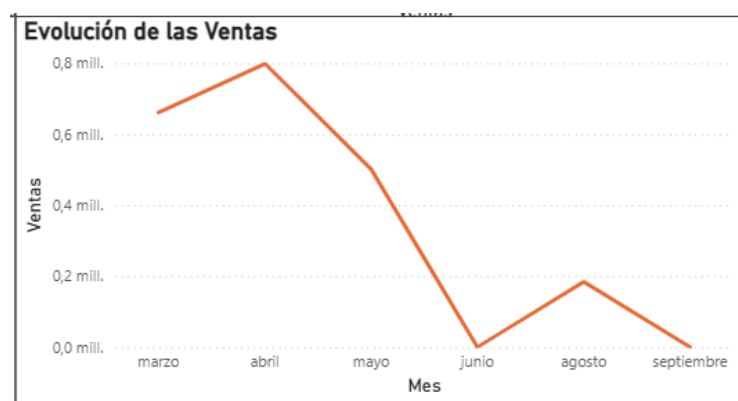
El segundo gráfico muestra la distribución de las ventas según el tipo de zona. En el primer lugar, se encuentra la zona ANY por lo que se entiende que no se ha podido clasificar la zona. Por ese motivo, se entiende que donde se han producido más ventas es en los pueblos (Village). Seguido por lo que se entiende como el extrarradio de las ciudades (Border) y las ciudades (City).



**Ilustración 46: Gráfico Ventas por Zona**

### 6.3.3. Evolución de las ventas

El gráfico de la evolución de ventas muestra anomalías en junio, agosto y septiembre. Después de un alto volumen de ventas en marzo, abril y mayo, es anómalo observar que en todo junio no haya apenas ventas, y en agosto y septiembre el volumen de ventas es muy reducido.



**Ilustración 47: Gráfico de Evolución de las Ventas**

### 6.3.4. Ventas por tipo de producto

Con la ayuda de este gráfico, se puede observar los tres tipos de producto que se está comercializando. ASL es la categoría que más se vende con más del doble de diferencia

respecto al segundo. En segundo lugar, tenemos los productos ETO. Por último, la categoría de producto con menos ventas es el ATA, con menos de la mitad de las ventas que el segundo producto.

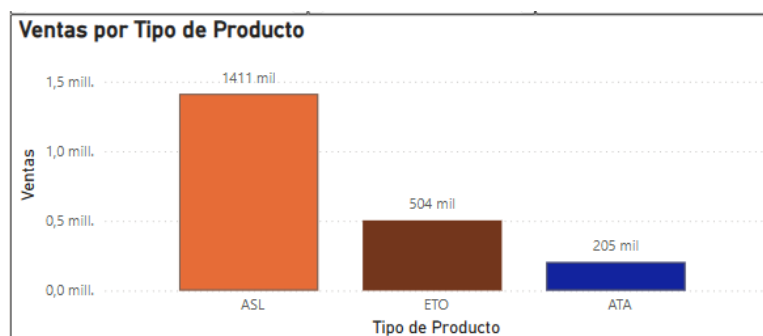


Ilustración 48: Gráfico Ventas por Tipo de Producto

### 6.3.5. Peso de tipo de producto por provincias

El gráfico de barras 100% apiladas, nos muestra el peso en porcentaje de cada producto por cada provincia. Como es de esperar en la gran mayoría de Provincias, los productos ASL son los más vendidos, seguidos por los ETO y, por último, los productos ATA. Soria es la excepción, ya que, según los datos, los productos ETO son los más vendidos. Aunque es importante recordar que, en el anterior gráfico, se ha observado que Soria es la provincia con apenas mil unidades vendidas.

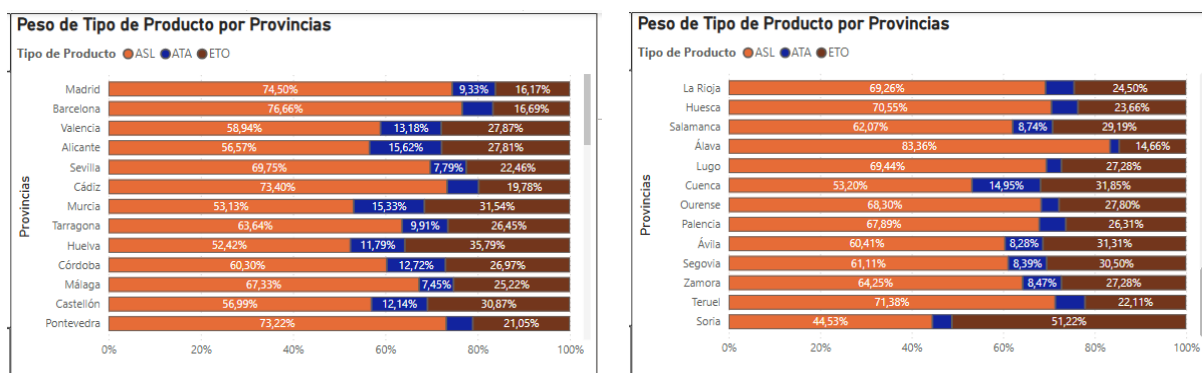


Ilustración 49: Gráfico de Tipo de Producto por Provincias

### 6.3.6. Peso de tipo de producto por zona

Si nos fijamos del peso que tienen las categorías de los productos en los distintos tipos de zona, se observa que los productos ASL son mayoritariamente los más vendidos. En Countryside y Escape son las zonas donde los productos ETO representan el 30% del volumen

de ventas. En el resto de los casos, el peso es menor del 30% en los productos ETO y del 10% o menor del 10% en los productos ATA.

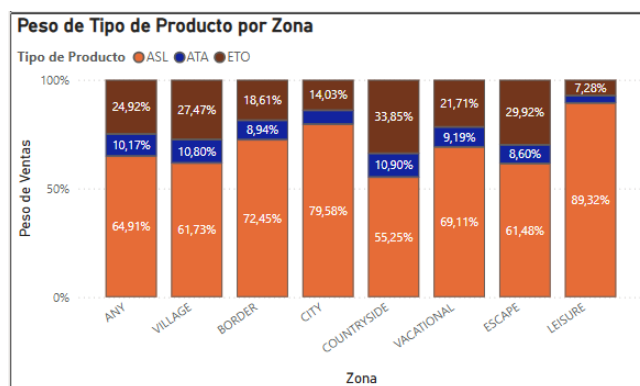


Ilustración 50: Gráfico Peso de Tipo de Producto por Zona

### 6.3.7. Evolución de las ventas por tipo de producto

Con el gráfico de líneas, se esperaba observar algún comportamiento estacionario de los productos. Al contrario de lo esperado, debido a la falta de información de los datos entre junio, agosto y septiembre, solo se puede observar la diferencia de volumen de ventas entre categorías como han mostrado los anteriores gráficos. Aunque ETO aparezca que no ha tenido ventas en junio, no es destacable porque ASL y ATA las ventas que ha habido son mínimas, por lo que no es información relevante.

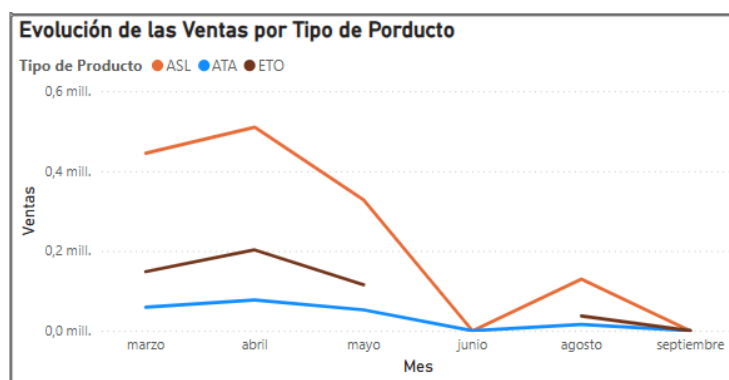


Ilustración 51: Gráfico Evolución de las Ventas por Tipo de Productos

## 7. MODELOS PREDICTIVOS CON RSTUDIO

En este apartado, se realizaron predicciones de ventas y entregas para la empresa Altadis mediante el uso del modelo ARIMA. Además, también se realizó una segmentación de los distintos tipos de tiendas mediante el uso del modelo K-Means.

### 7.1. MODELO PREDICTIVO DE VENTAS

Para la predicción de las ventas, primero se agruparon las mismas de manera semanal para poder así obtener la serie temporal de ventas:

```
## -----  
## MODELO PREDICTIVO ARIMA VENTAS  
## -----  
  
# Preparación de datos para series temporales  
library(forecast)  
library(tseries)  
library(mFilter)  
  
# Agregamos ventas por semana  
ventas_semanales <- ventas %>%  
mutate(Semana = floor_date(Sales_DAY, "week")) %>%  
group_by(Semana) %>%  
summarise(Ventas_Semana = sum(Sales_Uds))%>%  
complete(Semana = seq.Date(min(Semana), max(Semana), by="week"),  
fill = list(Ventas_Semana = 0))  
  
ggplot(ventas_semanales, aes(x = Semana, y = Ventas_Semana)) +  
geom_line() +  
scale_x_date(date_breaks = "1 month", date_labels = "%b %Y") +  
ggtitle("Ventas Semanales") +  
xlab("Fecha") +  
ylab("Unidades vendidas") +  
theme_minimal()
```

Ilustración 52: Código de ventas semanales

El gráfico de ventas semanales obtenido mediante este procedimiento fue el siguiente:

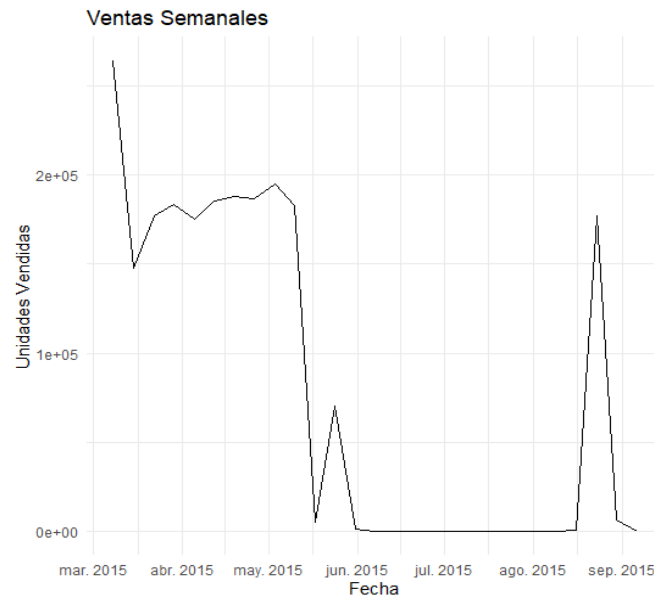


Ilustración 53: Gráfico de ventas semanales

En el gráfico generado se pudo observar una serie de ventas semanales que contaba con una elevada varianza, sobre todo desde inicios del mes de mayo de 2015, donde estas decrecieron mucho, no produciéndose a penas ventas durante verano y experimentándose un pico repentino final en el mes de setiembre. Dada la singularidad de la serie temporal, se procedió a realizar el entrenamiento y posterior validación del modelo ARIMA, pero teniendo en cuenta que sería complicado obtener resultados concluyentes dada las ya mencionadas características de los datos.

```
#Vamos a predecir los datos para 4 semanas aprox.
entrenamiento <- ventas_semanales[1:23,2]
validacion<- ventas_semanales[24:27,2]

entrenamiento_st <- ts(entrenamiento, frequency = 52, # 52 semanas = 1 año
  start = c(2015,1))

validacion_st <- ts(validacion,frequency = 52,
  start = c(2015,24))

# Modelo ARIMA semanal
modelo_arima <- auto.arima(entrenamiento_st)
prediccion <- forecast(modelo_arima, 4)
plot(prediccion)

prediccion$mean
predicciondatos<-prediccion$mean
plot(predicciondatos)

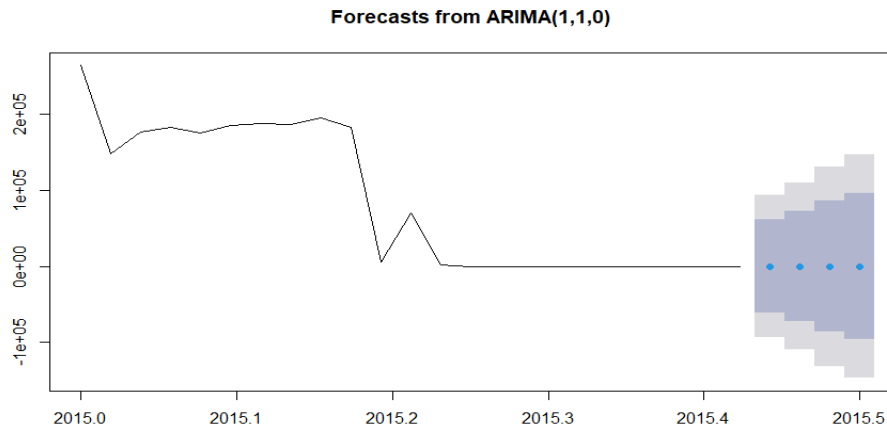
#Descomposición
descomposicion <- hpfiler(entrenamiento_st, freq=52)
tendenciaventasmensuales<-descomposicion$trend
plot(descomposicion$trend)

#Validación de los datos
plot(validacion_st)
lines(predicciondatos,col="blue")
accuracy(validacion_st,predicciondatos)
```

Ilustración 54: Predicción de ventas semanales



Para realizar esta primera predicción, se utilizaron 4 semanas para validar los datos, y los resultados obtenidos fueron los siguientes:

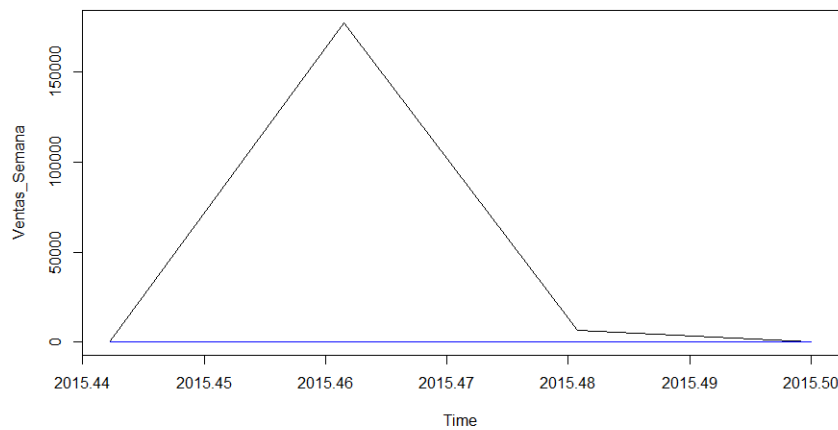


```
> predicción$mean  
Time Series:  
Start = c(2015, 24)  
End = c(2015, 27)  
Frequency = 52  
[1] 89.76462 111.42502 103.08200 106.29551
```

**Ilustración 55: Resultados de la predicción de ventas**

Como se había advertido previamente, la peculiaridad de la serie temporal ha propiciado que el modelo ARIMA no pueda predecir correctamente los datos para el último mes de la muestra.

Esto se puede observar mejor en el siguiente gráfico, dónde en color negro se aprecian las ventas semanales reales y en color azul la predicción del modelo ARIMA.



**Ilustración 56: Validación del modelo de ventas**

Si tenemos en cuenta las métricas del error entre los datos de entrenamiento y validación, podemos concluir que es tan elevado en todas ellas que no es correcto seleccionar a este modelo como válido.

```
> accuracy(validation_st, prediccionsdatos)
      ME      RMSE      MAE      MPE      MAPE      ACF1 Theil's U
Test set -45910.11 88428.2 45959.76 -41351.43 41398.14 -0.4060826 7737.666
```

**Ilustración 57: Métricas del error**

#### 7.1.1. Predicción de la serie posterior a mayo de 2015

Dado el ya mencionado volátil comportamiento de la serie posterior a inicios de mayo de 2015, se ha optado por predecir el resto de los datos temporales mediante los datos previos a esa fecha.

```
## -----
## MODELO PREDICTIVO ARIMA SERIE INCOMPLETA
## -----

# Empleamos 10 semanas
r <- ventas_semanales[1:10, 2]

r_serie <- ts(r$Ventas_Semana, frequency = 52, start = c(2015, 10)) # Semana 10 del año 2015
modelo_Rarima <- auto.arima(r_serie)
prediccionR <- forecast(modelo_Rarima, h = 17)

prediccionR$mean
prediccionRdatos <- prediccionR$mean
plot(prediccionRdatos)

plot(r_serie)
```

```
> prediccionR$mean
Time Series:
Start = c(2015, 20)
End = c(2015, 36)
Frequency = 52
[1] 179866.4 189538.6 179371.2 186785.1 182931.8 183872.8 184661.4 183243.2 184556.8 183683.0 184079.7 184037.9
[13] 183890.4 184088.7 183923.6 184023.5 183985.6
```

Ilustración 58: Código y resultados de la predicción de la serie incompleta

Los resultados obtenidos, muestran una serie semanal que tendría más sentido si la comparamos con los datos iniciales de la misma. Esto se puede observar mejor en el gráfico resultante de la predicción:

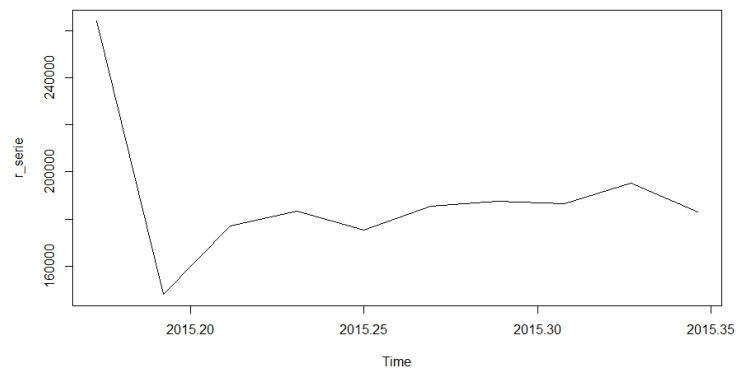


Ilustración 59: Gráfico de predicción de la serie incompleta

## 7.2. MODELO PREDICTIVO DE ENTREGAS

Para poder predecir las entregas, es decir, compras hechas por los estancos, se ha seguido el mismo proceso que en el caso de las ventas.

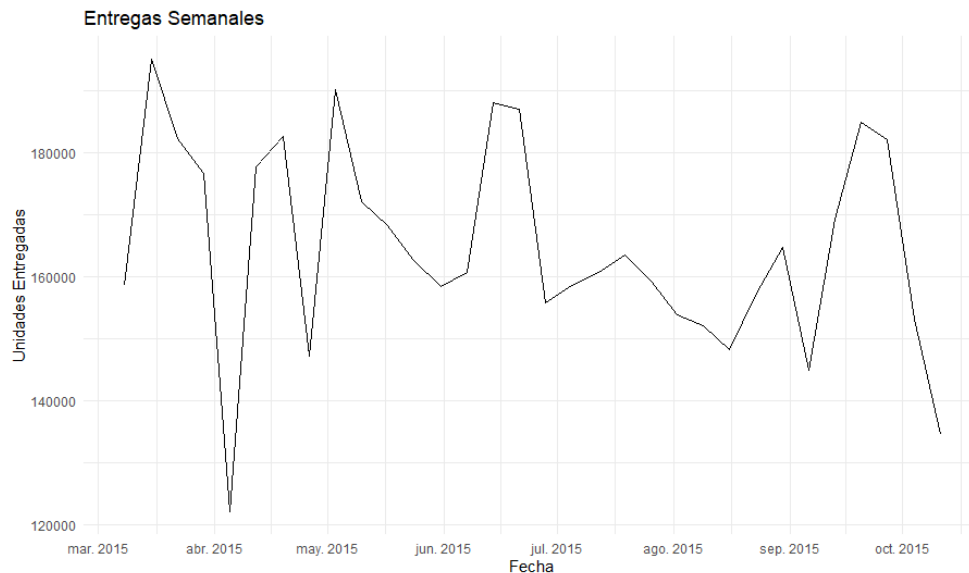
```
## -----
## MODELO PREDICTIVO ARIMA ENTREGAS
## -----

entregas_semanales <- reparto %>%
  mutate(Semana = floor_date(Delivery_DAY, "week")) %>%
  group_by(Semana) %>%
  summarise(Entregas_Semana = sum(Delivery_Uds))%>%
  complete(Semana = seq.Date(min(Semana), max(Semana), by="week"),
    fill = list(Entregas_Semana = 0))

ggplot(entregas_semanales, aes(x = Semana, y = Entregas_Semana)) +
  geom_line() +
  scale_x_date(date_breaks = "1 month", date_labels = "%b %Y") +
  ggtitle("Ventas Semanales") +
  xlab("Fecha") +
  ylab("Unidades vendidas") +
  theme_minimal()
```

Ilustración 60: Código de entregas semanales

Por lo tanto, se han agrupado las entregas por semanas, de donde se ha extraído la siguiente serie semanal:



Esta serie semanal parece tener una menor varianza, lo que nos lleva a la conclusión de que en este caso la predicción mediante ARIMA podría ajustarse mejor.

```
#Vamos a predecir los datos para 5 semanas aprox.
entrenamiento_entregas <- entregas_semanales[1:27,2]
validacion_entregas<- entregas_semanales[28:32,2]

entrenamiento_entregas_st <- ts(entrenamiento_entregas, frequency = 52, # 52 semanas = 1 año
start = c(2015,1))

validacion_entregas_st <- ts(validacion_entregas, frequency = 52,
start = c(2015,28))

# Modelo ARIMA semanal
modelo_arima_entregas <- auto.arima(entrenamiento_entregas_st)

prediccion_entregas <- forecast(modelo_arima_entregas, 5)
plot(prediccion_entregas)

prediccion_entregas$mean
predicciondatos_entregas<-prediccion_entregas$mean
plot(predicciondatos_entregas)

#Descomposición
descomposicion_entregas <- hpfilter(entrenamiento_entregas_st, freq=52)
tendenciaentregasmensuales<-descomposicion_entregas$trend
plot(descomposicion_entregas$trend)

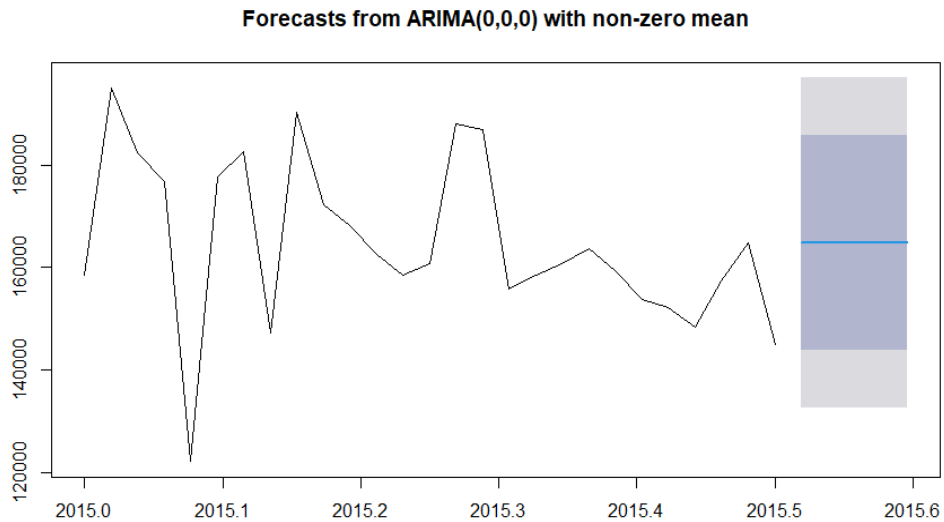
#Validación de los datos
plot(validacion_entregas_st)
lines(predicciondatos_entregas,col="blue")

autoplot(validacion_entregas_st, series = "Real") +
  autolayer(predicciondatos_entregas, series = "Predicción", color = "blue") +
  ggtitle("Validación vs Predicción - ARIMA") +
  ylab("Entregas Semanales") +
  xlab("Semana") +
  theme_minimal()

accuracy(validacion_entregas_st,predicciondatos_entregas)
```

Ilustración 61: Código de la predicción de entregas

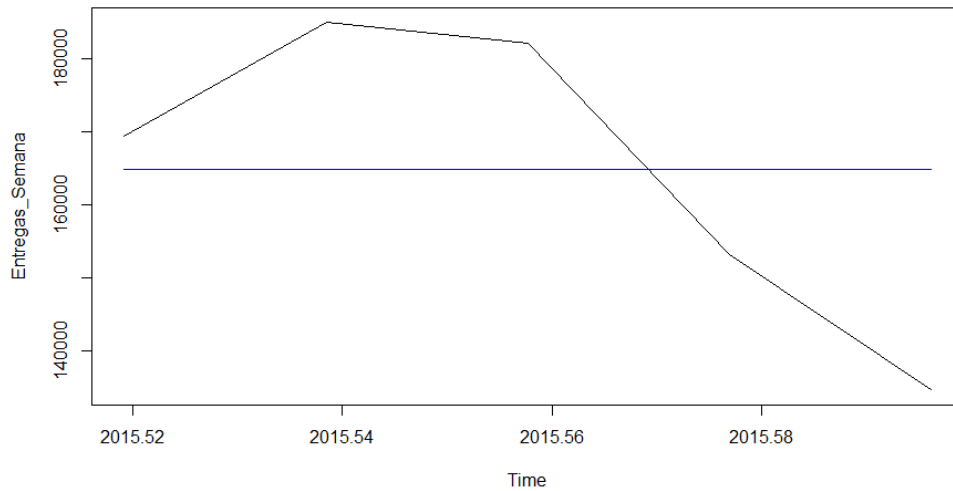
En este caso, la predicción ha dado el siguiente resultado para las últimas 5 semanas, las cuales tendremos que validar con los datos originales:



```
> prediccion_entregas$mean  
Time Series:  
Start = c(2015, 28)  
End = c(2015, 32)  
Frequency = 52  
[1] 164810.3 164810.3 164810.3 164810.3 164810.3
```

**Ilustración 62: Resultados de la predicción de entregas**

Por lo tanto, al comparar este resultado con los datos originales, se obtiene el siguiente resultado en donde la línea negra representa los datos reales de la serie y la línea azul los valores predichos:



**Ilustración 63: Validación del modelo de entregas**

De esta manera, al comparar los datos predichos con los reales, se obtienen las siguientes métricas de error:

```
> accuracy(validacion_entregas_st, prediccionsdatos_entregas)
      ME      RMSE      MAE      MPE      MAPE      ACF1 Theil's U
Test set -42.9037 18855.85 16787.54 -0.02603217 10.18598 0.3321158      Inf
```

**Ilustración 64: Métricas del error**

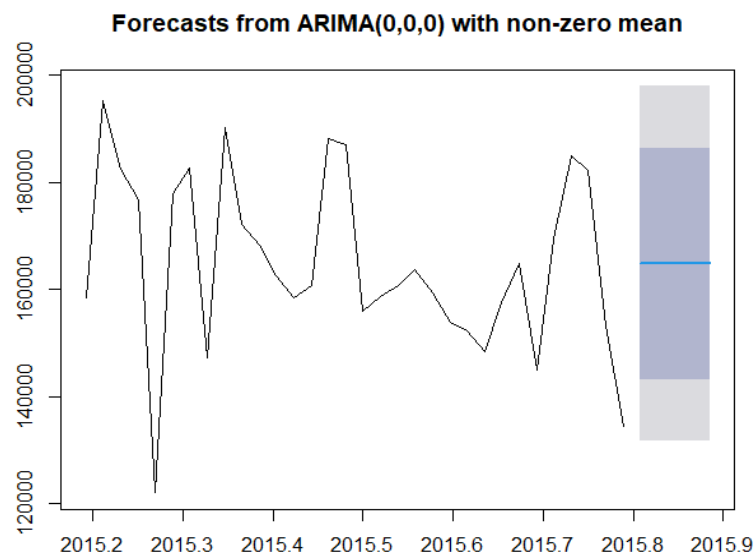
De ellas se puede deducir que, si bien son mejorables, especialmente a cuanto error absoluto medio (MAE) y a raíz del error cuadrático medio (RMSE), el modelo se puede considerar adecuado en lo referente a la captura de tendencias generales, como ocurre en el caso del porcentaje de error absoluto promedio (MAPE), el cual se encuentra entorno al 10,2%. Por tanto, se considera razonablemente útil dentro del contexto de esta investigación.

Dado este resultado, se procedió a realizar la predicción con el dataset al completo para poder predecir así las entregas de las próximas 5 semanas.

```
## -----  
## PREDICCIÓN PARA LAS PRÓXIMAS 5 SEMANAS  
## -----  
  
entregas_final <- ts(entregas_semanales$Entregas_Semana, frequency = 52, # 52 semanas = 1 año  
                     start = c(2015,11))  
  
# Modelo ARIMA semanal  
modelo_entregas_final <- auto.arima(entregas_final)  
  
prediccion_entregas_final <- forecast(modelo_entregas_final, 5)  
plot(prediccion_entregas_final)  
  
prediccion_entregas_final$mean  
predicciondatos_entregas <- prediccion_entregas_final$mean  
plot(prediccion_entregas_final)  
  
#Descomposición  
descomposicion_entregas_final <- hpfiler(entregas_final, freq=52)  
tendenciaentregasmensuales_final <- descomposicion_entregas_final$trend  
plot(descomposicion_entregas_final$trend)
```

Ilustración 65: Código de la predicción final

De esta predicción se pudieron extraer los siguientes resultados:



```
> prediccion_entregas_final$mean  
Time Series:  
Start = c(2015, 43)  
End = c(2015, 47)  
Frequency = 52  
[1] 164817 164817 164817 164817 164817
```

Ilustración 66: Resultados de la predicción final

En ellos se observa que el modelo predice que aproximadamente se realizarán alrededor de unas 164.817 entregas semanalmente durante las próximas 5 semanas.

## 8. MODELOS DE SEGMENTACIÓN CON RSTUDIO

### 8.1. SEGMENTACIÓN CON K-MEANS SIN NORMALIZAR

Para la realización de una correcta segmentación, se ha creado un dataset que une varios de los proporcionados por Altadis, con la intención de reunir la máxima información posible sobre cada una de las tiendas. Posteriormente se han eliminado las variables del código de afiliados, que ha sido utilizada para la realización de los joins y la variable del tamaño de la tienda:

```
## -----  
## SEGMENTACIÓN CON K-MEANS  
## -----  
  
# Preparación de datos para clustering  
library(cluster)  
library(factoextra)  
  
# Creamos un dataset con características de las tiendas  
tiendas_cluster <- ventas %>%  
  group_by(Affiliated_Code) %>%  
  summarise(  
    total_ventas = sum(Sales_Uds),  
    frecuencia_compra = n(),  
    productos_distintos = n_distinct(Product_Code)  
  ) %>%  
  left_join(  
    oos %>%  
      group_by(Affiliated_Code) %>%  
      summarise(roturas = n()),  
    by = "Affiliated_Code"  
  ) %>%  
  left_join(  
    reparto %>%  
      group_by(Affiliated_Code) %>%  
      summarise(  
        total_entregas = sum(Delivery_Uds),  
        frecuencia_entrega = n()  
      ),  
    by = "Affiliated_Code"  
  ) %>%  
  left_join(  
    tiendas %>% select(Affiliated_Code, Engage, Management_Cluster, Tam_m2),  
    by = "Affiliated_Code"  
  ) %>%  
  mutate_if(is.character, as.factor) %>%  
  mutate(roturas = ifelse(is.na(roturas), 0, roturas))  
  
tiendas_cluster <- tiendas_cluster %>%  
  select(-Affiliated_Code, -Tam_m2)
```

Ilustración 67: Creación del dataset a segmentar

Tras este proceso, se ha empleado el código que ha permitido la siguiente segmentación de los datos:



```
# Determinación del número óptimo de clusters
fviz_nbclust(tiendas_cluster, kmeans, method = "silhouette") +
  ggtitle("Método de silhouette para determinar número óptimo de clusters")

fviz_nbclust(tiendas_cluster, kmeans, method = "wss") +
  ggtitle("Método del codo para determinar el número óptimo de clusters")

# Aplicación de K-Means con k=3
kmeans_result <- kmeans(tiendas_cluster, 3)
kmeans(tiendas_cluster, 3)

# Calcular coeficiente de silhouette
sil <- silhouette(kmeans_result$cluster, dist(tiendas_cluster))

# Mostrar promedio del coeficiente de silhouette
sil_promedio <- mean(sil[, 3])
cat("Coeficiente promedio de silhouette:", round(sil_promedio, 3), "\n")

# Visualización de los clusters
summary(tiendas_cluster)
str(tiendas_cluster)
fviz_cluster(kmeans_result, data = tiendas_cluster,
  palette = c("#2E9FDF", "#00AFBB", "#E7B800"),
  geom = "point")

# Añadimos los clusters al dataset original
tiendas_cluster$cluster <- as.factor(kmeans_result$cluster)

# Visualización de características por cluster
ggplot(tiendas_cluster, aes(x = total_ventas, y = roturas, color = cluster)) +
  geom_point(alpha = 0.6) +
  ggtitle("Segmentación de Tiendas por Ventas y Roturas") +
  xlab("Total Ventas") +
  ylab("Total Roturas")
```

Ilustración 68: Código del modelo K-Means sin normalizar

Inicialmente, se determinó el número óptimo de clústeres, de dónde mediante el método del codo, se determinó que el valor adecuado para realizar la segmentación de los datos fue 3.

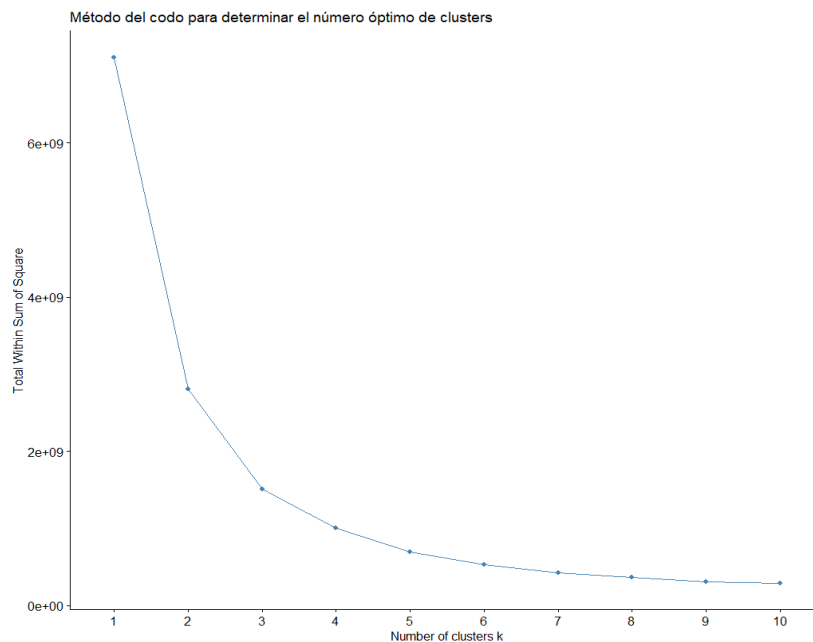


Ilustración 69: Gráfico del método del codo

Una vez seleccionado el número óptimo de clústeres, se procedió a crear el modelo K-Means, el cual generó el siguiente resultado:

```
> kmeans(tiendas_cluster, 3)
K-means clustering with 3 clusters of sizes 228, 1161, 2193

Cluster means:
  total_ventas frecuencia_compra productos_distintos roturas total_entregas frecuencia_entrega Engage Management_Cluster
1  2001.1623      728.3728      21.42105 71.08333      4960.719      413.3509 2.688596      2.057018
2   913.1688      439.0293      18.64255 81.80706      2221.740      256.8966 2.342808      2.061154
3   287.3256      169.9248      11.83539 68.89786       713.021      100.8637 1.845873      1.756042

Clustering vector:
[1] 3 3 1 3 2 2 2 3 2 3 3 3 3 2 3 3 2 3 2 2 3 2 3 3 3 3 2 3 3 3 3 3 2 2 3 2 2 3 3 3 2 3 2 2 3 3 3 1 3 3 3 3 3 3 2 3 3 3 2 3 2 3 3 3 3 3 3 3 2 3
[76] 3 3 2 3 2 3 3 3 2 2 2 3 2 3 3 3 3 2 3 1 3 3 2 2 3 3 3 3 1 2 2 3 3 3 3 3 3 2 2 3 3 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
[151] 3 3 3 1 2 2 2 3 2 3 3 2 2 1 2 3 3 3 3 3 2 3 2 3 3 3 1 3 2 2 1 2 2 3 3 3 3 3 1 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
[226] 2 3 2 3 3 3 3 3 3 2 3 3 3 1 3 2 3 2 3 3 1 3 3 3 3 3 3 3 3 3 3 3 3 2 3 3 2 3 3 1 3 2 3 3 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
[301] 3 2 3 2 3 2 3 3 2 1 2 3 2 2 2 3 3 3 3 3 2 1 3 2 3 2 3 3 3 3 2 3 3 3 3 2 2 2 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
[376] 3 3 3 1 2 1 3 2 3 3 3 2 3 3 3 3 3 2 3 3 3 3 3 2 3 3 3 3 3 2 3 3 3 3 3 3 3 3 3 1 3 2 1 3 3 2 3 3 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
[451] 2 2 3 3 3 3 3 3 3 3 3 2 3 3 3 2 2 1 3 2 3 3 3 3 2 3 1 3 3 3 3 3 3 3 3 3 3 3 3 2 3 1 3 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
[526] 1 2 2 3 3 3 3 3 2 2 2 3 3 3 3 3 3 3 3 3 3 3 3 2 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
[601] 1 3 3 1 3 1 3 2 3 2 2 3 2 2 3 2 2 3 2 3 3 3 3 3 3 3 3 2 1 3 2 2 3 3 3 3 3 3 3 3 3 1 2 3 3 2 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
[676] 2 3 2 2 3 2 3 3 3 3 3 3 3 3 3 2 3 3 3 3 2 3 3 3 2 3 3 3 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
[751] 2 3 3 3 2 2 3 2 3 3 3 3 3 2 3 3 3 3 3 2 2 3 1 3 3 3 3 2 2 1 3 2 3 2 1 2 2 3 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
[826] 3 2 2 2 3 3 2 3 3 3 3 3 3 3 2 3 2 2 3 3 3 3 3 3 3 3 3 3 3 2 2 2 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
[901] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
[976] 3 3 2 3 2 3 2 3 2 3 2 3 1 3 3 2 3 3 3 3 2 3 1 3 3 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
[ reached getoption("max.print") -- omitted 2582 entries ]

within cluster sum of squares by cluster:
[1] 637505588 453614679 417076566
(between_SS / total_SS = 78.8 %)

Available components:
[1] "cluster" "centers" "totss" "withinss" "tot.withinss" "betweenss" "size" "iter" "ifault"
```

#### Ilustración 70: Resultados del modelo K-Means sin normalizar

En él, se observan tres clústeres resultantes que no solo difieren en tamaño, sino también en sus características operativas y comerciales.

#### Clúster 1

Este grupo, compuesto por 228 tiendas, es el menor de los tres y representa los puntos de venta más dinámicos y con un mayor volumen de negocio:

- **Alto nivel de ventas:** Es la mayor de los tres grupos con 2001,16 unidades de media por tienda.
- **Alta frecuencia de compra:** Alrededor de 728,37 compras por tienda.
- **Gran diversidad de productos adquiridos:** En las tiendas de este clúster se adquirieron aproximadamente 21,42 productos distintos de media.
- **Alta presencia de incidencias en términos relativos:** En torno a 71,08, lo que se consideraría el grupo con un valor medio entre los otros dos, sin embargo, se ha de tener muy en cuenta que este grupo es con diferencia el menos numeroso, lo que indicaría que en términos relativos se producirían más roturas de stock que en los otros dos.

- **Alta cantidad de entregas:** Las entregas totales de este grupo son las más elevadas de los tres (4960,72).
- **Elevado nivel de entregas:** Dentro de este clúster se encuentran los mayores niveles de entregas promedio a los establecimientos afiliados, en torno a 413,35.
- **Gran compromiso con la marca:** Indicando una gran vinculación con la marca (2,69).
- **Segmento operativo:** No se aprecia una diferencia significativa con el clúster 2, situándose ambos en torno a 2,06.

Este clúster incluye por lo tanto las tiendas más activas, probablemente ubicadas en zonas turísticas o muy pobladas donde existe una gran demanda de estos productos. El nivel de roturas se encuentra relativamente elevado para el tamaño del grupo, pero se compensa por el elevado volumen de ventas de estos establecimientos.

## Clúster 2

Este clúster es el que se situaría entre el 1 y el 3 en la mayoría de los resultados, estando compuesto por 1.161 tiendas:

- **Medio nivel de ventas:** Este grupo obtiene aproximadamente 913,17 ventas de media.
- **Media frecuencia de compra:** Alrededor de 439,03 compras por tienda.
- **Moderada diversidad de productos adquiridos:** En las tiendas de este clúster se adquirieron aproximadamente 18,64 productos distintos de media.
- **Alta presencia de incidencias:** En torno a 81,81, lo que se consideraría el grupo con un mayor número de roturas de stock totales en términos absolutos, aunque no lo sería en términos relativos debido a su tamaño.
- **Media cantidad de entregas:** Las entregas totales de este grupo son de 2221,74.
- **Medio nivel de entregas:** Dentro de este clúster se encuentran los valores medios de entregas promedio a los establecimientos afiliados, siendo de en torno a 256,9.
- **Compromiso medio con la marca:** Indicando una buena vinculación con la marca (2,34) pero no siendo el grupo con una mayor vinculación.

- **Segmento operativo:** No se aprecia una diferencia significativa con el clúster 1, situándose ambos en torno a 2,06.

### Clúster 3

Este grupo, compuesto por 2.193 tiendas, es el más numeroso de los tres y representa los puntos de venta menos dinámicos y con menor volumen de negocio:

- **Bajo nivel de ventas:** Es la menor de los tres grupos con 287,33 unidades de media por tienda.
- **Baja frecuencia de compra:** Alrededor de 169,92 compras por tienda.
- **Baja diversidad de productos adquiridos:** En las tiendas de este clúster se adquirieron aproximadamente 11,83 productos distintos de media.
- **Baja presencia de incidencias:** En torno a 68,9, lo que se consideraría dentro de un margen aceptable, aunque debería reducirse más para no afectar demasiado a los beneficios de estos establecimientos debido al bajo volumen de ventas.
- **Baja cantidad de entregas:** Las entregas totales de este grupo son las más bajas de los tres (713,02).
- **Baja frecuencia de entregas:** Dentro de este clúster se encuentran los menores niveles de entregas promedio a los establecimientos afiliados, en torno a 100,86.
- **Bajo compromiso con la marca:** Indicando una baja vinculación con la marca (1,76).
- **Segmento operativo:** Se aprecia cierta diferencia con los otros dos clústeres (1,75 de media), lo que indicaría que muchas de las observaciones de este grupo estarían en un segmento operativo inferior.

Este clúster incluye por lo tanto las tiendas menos activas, probablemente ubicadas en zonas de baja demanda o con una baja capacidad operativa. A pesar de bajo su volumen, el nivel de roturas debería de disminuir para poder mejorar la rentabilidad de estos establecimientos asociados.

Para evaluar la validez del modelo de segmentación, se han considerado dos métricas clave: el coeficiente promedio de Silhouette y la proporción de varianza inter-clúster sobre la varianza total.

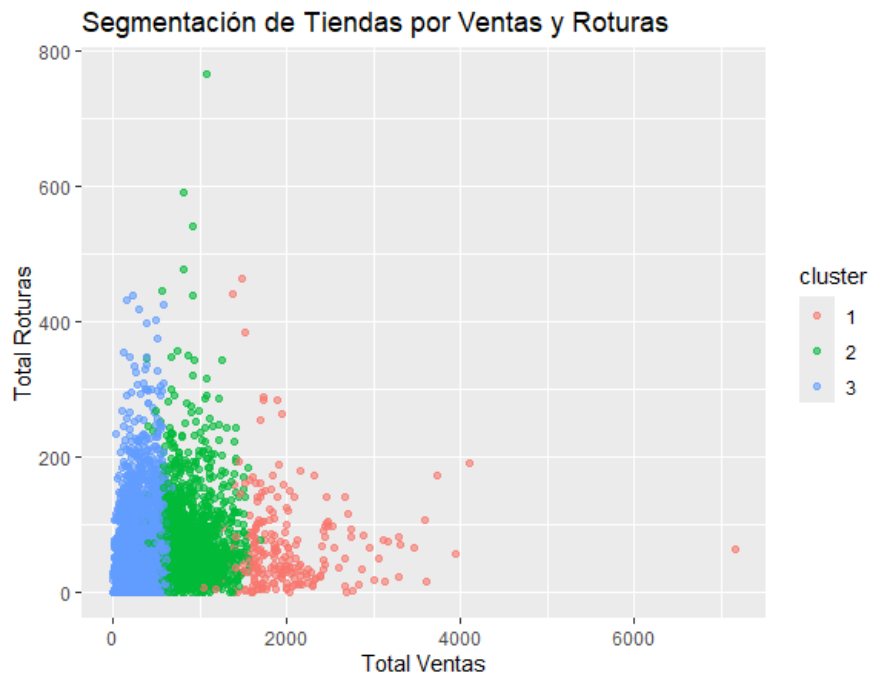
En primer lugar, el modelo obtuvo un coeficiente de Silhouette medio de 0,589, lo que indica una separación moderadamente buena entre los clústers. Este valor refleja que, en general, los puntos están más cerca de los centros de su propio clúster que de los de los demás, lo que justifica una estructura interna razonablemente compacta y diferenciada.

**Coeficiente promedio de silhouette: 0.589**

**Ilustración 71: Coeficiente de Silhouette del modelo sin normalizar**

En segundo lugar, se analizó la varianza interna y entre grupos. El modelo muestra una proporción de varianza inter-clúster del 78,8 % respecto a la varianza total ( $\text{between\_SS} / \text{total\_SS}$ ), lo cual es un resultado muy positivo. Este porcentaje indica que casi el 80 % de la variabilidad en los datos es explicada por las diferencias entre los grupos definidos por el modelo.

Además, las diferencias entre los distintos grupos también se pueden apreciar gráficamente, tal y como se muestra en la siguiente ilustración, donde se establece el eje Y como el número total de roturas de stock de los establecimientos y el eje X como el número total de ventas:



**Ilustración 72: Segmentación por ventas y roturas de stock**

## 8.2. SEGMENTACIÓN CON K-MEANS CON DATOS NORMALIZADOS

Tras haber realizado una primera segmentación sin normalizar los datos, se consideró conveniente repetir el proceso utilizando los mismos datos, pero previamente normalizados, es decir, transformando todas las variables a una escala común mediante estandarización (media 0 y desviación típica 1).

Para ello se ha realizado el mismo procedimiento, con la única diferencia de esta normalización como se muestra a continuación en el código:

```
# Determinación del número óptimo de clusters
fviz_nbclust(tiendas_cluster_normalizadas, kmeans)

fviz_nbclust(tiendas_cluster_normalizadas, kmeans, method = "silhouette") +
  ggtitle("Método de Silhouette para Determinar Número Óptimo de Clusters")

# Aplicación de K-Means normalizado con k=3
kmeans_norm <- kmeans(scale(tiendas_cluster_normalizadas), 3)
kmeans(scale(tiendas_cluster_normalizadas), 3)

# Calcular coeficiente de silhouette
sill <- silhouette(kmeans_norm$cluster, dist(tiendas_cluster_normalizadas))

# Mostrar promedio del coeficiente de silhouette
sil_promedio <- mean(sill[, 3])
cat("Coeficiente promedio de silhouette:", round(sil_promedio, 3), "\n")

# Visualización de los clusters
summary(tiendas_cluster_normalizadas)
str(tiendas_cluster_normalizadas)
fviz_cluster(kmeans_result, data = tiendas_cluster_normalizadas,
  palette = c("#2E9FDF", "#00AFBB", "#E7B800"),
  geom = "point")

# Añadimos los clusters al dataset original
tiendas_cluster_normalizadas$cluster <- as.factor(kmeans_norm$cluster)

# Visualización de características por cluster
ggplot(tiendas_cluster_normalizadas, aes(x = total_ventas, y = roturas, color = cluster)) +
  geom_point(alpha = 0.6) +
  ggtitle("Segmentación de Tiendas por Ventas y Roturas") +
  xlab("Total Ventas") +
  ylab("Total Roturas")
```

Ilustración 73: Código del modelo K-Means normalizado

Una vez generado el modelo K-means con los datos normalizados, se obtuvieron los siguientes resultados:

```
Coeficiente promedio de silhouette: 0.353
```

Ilustración 74: Coeficiente de Silhouette del modelo normalizado

```
K-means clustering with 3 clusters of sizes 1244, 700, 1638

Cluster means:
total_ventas frecuencia_compra productos_distintos roturas total_entregas frecuencia_entrega Engage Management_Cluster
1 -0.77014720 -0.9186174 -1.058517 -0.41003779 -0.76567480 -0.90781466 -0.7361518 -0.4202861
2 1.55917548 1.5535045 1.028626 0.03468205 1.54118440 1.50641597 0.9775168 0.1157730
3 -0.08141619 0.0337649 0.364320 0.29658704 -0.07712432 0.04568392 0.1413376 0.2697160

Clustering vector:
[1] 3 1 2 1 3 3 3 2 3 2 3 1 1 1 1 3 1 1 2 3 3 1 3 3 1 3 1 1 1 2 3 1 1 1 3 3 3 2 1 3 2 3 1 1 3 1 3 3 3 1 1 2 3 3 3 1 1 1 3 1 3 1 3 1 2 1 1 1 3 1 3 1 2 3
[76] 1 1 3 3 2 3 1 1 3 2 3 3 2 3 2 1 1 3 3 2 3 3 3 2 1 1 3 3 2 3 3 3 1 1 3 3 3 3 2 3 3 3 1 1 2 3 3 3 3 2 1 3 3 3 3 3 1 1 3 2 1 2 2 3 3 1 2 1 1
[151] 3 1 1 2 3 1 3 1 3 2 3 1 3 3 3 2 3 3 1 3 3 1 1 2 1 2 1 1 1 2 3 3 3 2 2 1 1 3 3 1 3 3 2 2 1 1 1 3 3 3 1 1 1 2 3 2 2 2 3 3 1 1 1 3 2 1 3 1 3 1 1
[226] 3 3 3 3 3 1 3 1 3 1 2 1 1 1 1 2 3 2 3 3 3 1 1 2 1 3 3 1 3 3 1 3 1 3 3 3 3 1 1 2 3 3 1 1 1 3 1 1 3 1 1 3 1 1 3 1 3 1 3 1 3 2 3 1 3 3
[301] 1 2 3 3 2 1 3 1 1 3 2 3 3 3 1 2 3 1 3 3 1 1 3 3 3 1 3 3 2 3 3 1 1 2 1 1 3 3 2 3 3 3 2 3 3 1 1 3 3 1 1 1 3 3 2 1 3 2 2 2 1 1 1 1 3 3 1 1 1 2 2 1 3 3
[376] 1 1 1 2 3 2 2 3 3 3 1 1 3 2 1 1 1 3 3 2 3 3 1 3 3 3 1 2 1 2 3 1 3 1 1 1 3 3 1 1 3 3 2 1 2 2 1 1 3 3 3 2 3 1 1 1 3 3 1 3 3 1 1 1 2 3 1 3 3
[451] 2 2 3 1 1 3 1 3 1 3 3 1 3 3 1 1 2 3 2 1 2 3 3 3 1 2 1 2 3 3 3 1 3 3 3 3 2 3 1 1 3 1 2 1 2 1 1 1 3 3 3 3 2 3 2 3 3 1 3 3 3 3 1 1 3 2 1 2 1 3 3
[526] 3 2 3 3 1 1 1 3 3 2 3 3 3 3 1 1 3 1 1 1 1 1 3 3 3 3 3 1 1 1 3 1 2 3 3 3 3 2 2 1 1 3 2 1 3 1 1 3 3 3 2 3 1 3 3 3 3 2 1 3 2 3 1 3 3 3 3 2 2 1 2 3
[601] 2 3 1 2 3 2 1 3 2 1 2 3 2 1 2 3 1 2 3 1 1 3 1 1 3 1 2 2 1 3 3 1 1 2 1 1 1 1 1 3 2 3 3 3 3 2 3 1 1 3 3 3 1 2 2 3 3 3 2 1 1 1 1 1 3 3 1 1 3 3
[676] 3 1 2 3 3 3 1 1 2 3 1 3 1 3 2 1 3 1 2 3 3 1 3 3 1 1 3 3 3 3 3 1 2 3 1 1 2 1 1 3 3 3 2 1 3 3 2 3 3 3 1 1 3 1 1 2 3 2 1 3 1 3 2 1 2 3 3 1 3 3 2 1 3 2
[751] 2 3 1 3 3 3 3 1 1 1 3 1 3 3 3 1 3 3 2 1 2 3 3 1 1 2 1 3 2 1 3 3 2 2 2 1 3 3 3 3 1 1 1 1 3 3 3 2 1 3 2 2 3 3 3 1 3 3 3 1 2 1 1 1 2 2 1 3 3 2 1 1
[826] 3 3 2 2 1 1 3 1 3 3 1 1 1 3 1 3 3 3 2 2 1 3 2 3 3 3 1 1 1 3 3 3 3 2 3 3 3 2 3 3 3 3 1 3 2 3 3 2 3 3 1 1 3 1 1 1 3 3 1 3 2 2 3 1 1 3 3 3 1 3 3 2 3 1
[901] 1 3 1 1 1 1 3 3 3 1 1 3 3 3 3 3 2 1 2 2 1 1 3 2 1 2 3 2 3 1 3 1 1 3 2 3 2 1 1 3 1 1 2 3 2 3 1 1 2 3 3 3 1 1 3 3 1 1 1 1 2 3 2 3 3 1 3 1 3 3 1 1 1 1
[976] 1 1 2 3 3 2 3 3 2 1 2 1 1 2 3 3 3 3 3 2 1 1
[ reached getOption("max.print") -- omitted 2582 entries ]

within cluster sum of squares by cluster:
[1] 3299.380 4577.958 6130.366
(between_SS / total_SS = 51.1 %)

Available components:
[1] "cluster" "centers" "totss" "withinss" "tot.withinss" "betweenss" "size" "iter" "ifault"
```

#### Ilustración 75: Resultados del modelo K-Means sin normalizar

En ellos se puede observar que, pese a que con los datos normalizados los grupos generados son algo más homogéneos, este modelo empeoraría los resultados sensiblemente respecto al anterior.

Concretamente, encontramos un coeficiente de Silhouette de 0,353 el cual sería muy inferior respecto al del modelo anterior (0,589) e indicaría una mala separación entre clústeres.

Ocurre de igual manera con el caso de la varianza inter-clúster del 51,1 % respecto a la varianza total ( $\text{between\_SS} / \text{total\_SS}$ ), lo cual es un resultado bastante negativo ya que indica que únicamente alrededor de la mitad de la variabilidad en los datos es explicada por las diferencias entre los grupos definidos por el modelo.

Dados estos resultados, se ha optado por descartar este modelo debido a su peor capacidad para segmentar los datos en comparación al de K-means sin normalización previa.



## 9. CONCLUSIONES

Finalmente, se ha conseguido diseñar un sistema de Business Intelligence para Imperial Brands, consiguiendo información valiosa para futuras tomas de decisiones. A continuación, se presentarán los resultados más destacados obtenidos a lo largo del trabajo.

Tanto las ventas (Sales\_Uds), como las entregas (Delivery\_Uds), presentan una alta dispersión de los datos. Aun así, la mediana de ventas diarias es de 1 y la media de 2,05. La mediana de las unidades repartidas al día es de 8 y la media de 8,55. En el caso de las ventas, rara vez los clientes compran más de 15 unidades de un mismo producto en un día. Por otro lado, será muy poco habitual encontrarse con 100 unidades entregadas al día. La media del tiempo medio de entregas por día es de 2,64 y una mediana de 1,42, lo que indica una frecuencia relativamente alta. Para cerrar el análisis descriptivo, las ventas y las entregas presentan una correlación muy alta (0.98) por tanto, se puede afirmar que ambas variables están estrechamente relacionadas, lo cual era esperable. Tanto las ventas como las entregas tienen una correlación moderada (0.55) y (0.56) respectivamente con Engage. Esta correlación podría indicar que los establecimientos con mayor nivel de compromiso tienden a registrar mejores ventas y, por ende, mejores entregas.

A la hora de representar los datos en un dashboard, también se ha podido obtener información relevante sobre el comportamiento de las ventas. Las provincias de Madrid y Barcelona son las que más ventas generan. Es de esperar ya que, se encuentran las dos ciudades más grandes de España y con más población. Por otro lado, Soria es la provincia con muchas menos ventas por diferencia. Respecto al tipo de zona, aunque la variable ANY aparezca como la zona con mayor concentración de ventas, se entiende que ANY hace referencia a que no se ha recogido correctamente la información de los datos. Las zonas con mayor volumen de ventas serían VILLAGE y BORDER, por lo que se entiende que en los pueblos y en el borde de las ciudades es donde más se compra. La categoría de producto más comprada ha sido ATL (más del 66%), seguida por ETO (24%) y ATA (10%). Al haber tanta diferencia de compras entre categoría de productos, en la gran mayoría de provincias, los hábitos de consumo son los mismos que en el general. Por último, gracias a los gráficos de las evoluciones de ventas a lo largo del tiempo, se puede observar cómo los datos presentan

anomalías a partir de mayo, lo que podría indicar que los datos no se han recogido correctamente a partir de esa fecha.

Se realizó una doble aproximación al análisis de datos comerciales de Altadis mediante técnicas de predicción y segmentación. En primer lugar, se aplicó el modelo ARIMA para realizar previsiones sobre las ventas y entregas semanales. En este sentido, se evidenció que la serie temporal de ventas presentaba una elevada varianza, especialmente a partir de mayo de 2015, lo cual dificultó una predicción fiable. La validación inicial del modelo, realizada con un periodo de cuatro semanas, confirmó esta limitación al mostrar errores significativos en todas las métricas evaluadas. Como consecuencia, se procedió a restringir el entrenamiento del modelo a los datos previos a la fecha mencionada, logrando así una predicción más coherente con la tendencia inicial de la serie.

En el caso de las entregas, el comportamiento temporal fue más estable, con una varianza considerablemente menor. Esto permitió que el modelo ARIMA ofreciera una mejor capacidad de ajuste y predicción, reflejado en un MAPE del 10,2 %, lo cual sugiere una aceptable capacidad para capturar tendencias generales, a pesar de que los valores de MAE y RMSE indiquen cierto margen de mejora. A partir de estos resultados, se decidió extender la predicción al conjunto completo de datos, estimando un promedio de 164.817 entregas semanales para las cinco semanas siguientes.

Por otro lado, se realizó una segmentación de las tiendas afiliadas utilizando el algoritmo K-Means. Para ello, se integraron varias fuentes de información con el objetivo de construir una base de datos robusta y representativa. El análisis de codo permitió establecer que el número óptimo de clústeres era tres. Los resultados del modelo evidenciaron una diferenciación clara entre los grupos formados: el clúster 1, compuesto por tiendas altamente dinámicas con altos niveles de ventas, entregas y compromiso con la marca; el clúster 2, con características intermedias tanto en volumen comercial como en comportamiento operativo; y el clúster 3, que aglutina a los puntos de venta menos activos, con bajos indicadores de rendimiento y operatividad.

El coeficiente medio de Silhouette alcanzó un valor de 0,589, lo que indica una adecuada separación entre los clústeres. Asimismo, la proporción de varianza inter-clúster sobre la varianza total fue del 78,8 %, reforzando la idea de que la segmentación explica de forma eficiente la variabilidad del conjunto de datos. En comparación, la repetición del proceso con datos normalizados resultó ser considerablemente peores, con un coeficiente de Silhouette de 0,353 y una varianza explicada del 51,1 %. Por tanto, se optó por mantener el modelo sin normalización, al presentar una estructura más clara y operativamente útil.

Se concluye que, aunque la predicción de ventas se vio limitada por la inestabilidad de la serie temporal, la predicción de entregas mediante ARIMA fue adecuada en términos generales. Además, la segmentación de tiendas aporta un valor estratégico significativo al identificar tipologías diferenciadas de establecimientos, lo que puede facilitar la toma de decisiones personalizadas para mejorar el rendimiento comercial de cada grupo.

## 10. RECOMENDACIONES

Para concluir, se van a proporcionar un seguido de recomendaciones a Imperial Brands a partir del modelo Business Intelligence diseñado. Las recomendaciones provienen de los resultados obtenidos a partir de los datos proporcionados.

### **Optimización del modelo de distribución de entregas:**

Como primera recomendación, dada la alta correlación entre ventas y entregas, y la mayor estabilidad de la serie temporal de estas últimas, se recomienda utilizar las entregas como proxy para planificar la reposición de productos. La predicción obtenida mediante el modelo ARIMA puede servir como base para ajustar los volúmenes de distribución semanal y evitar tanto roturas de stock como sobre aprovisionamiento.

### **Profundización en el análisis de comportamiento territorial:**

Como el análisis geográfico indicó diferencias entre provincias, se aconseja analizar en mayor detalle los factores que contribuyen al bajo rendimiento de ciertas zonas (como Soria), así como las prácticas que impulsan las ventas en áreas destacadas como Madrid y Barcelona. Ello permitiría replicar buenas prácticas o ajustar estrategias de marketing territorial.

### **Diversificación del portafolio de productos según patrones de consumo:**

Dado el claro predominio de la categoría ATL en las ventas, se recomienda, por ejemplo, revisar la composición del surtido ofrecido en tienda. Una estrategia de promoción cruzada o de introducción progresiva de las categorías ATA y ETO podría contribuir a equilibrar el portafolio y reducir riesgos asociados a la concentración en un único tipo de producto.

### **Segmentación de puntos de venta y personalización de estrategias:**

La segmentación en tres clústeres diferenciados permite adaptar las estrategias comerciales a las características de cada grupo. Para el clúster 1, se recomendaría focalizar los recursos comerciales y promocionales en las tiendas, con el objetivo de consolidar su rendimiento. Para los establecimientos que pertenecen al clúster 2 con potencial de crecimiento, deberían implementar acciones de fidelización y formación. Evaluar la viabilidad comercial de las

tiendas del clúster 3, desarrollando intervenciones específicas para mejorar su desempeño o, si procede, replantear su continuidad en la red.

**Revisión y mejora en la calidad de los datos:**

Finalmente, las anomalías detectadas a partir de mayo y la presencia de valores atípicos relevantes ponen de manifiesto la necesidad de mejorar los procesos de captura y validación de datos. Resulta fundamental asegurar la integridad y precisión de las bases de datos para garantizar la fiabilidad de las decisiones futuras.

## 11. REFERENCIAS

- Figueroa Rivera, M. F., & Reyes Canales, S. A. (2023). Gestión de inventarios a través del Business Intelligence en una empresa del sector Retail: Caso Mumuso. Recuperado de <https://tesis.pucp.edu.pe/server/api/core/bitstreams/92a6179f-a4d6-4a99-8efd-36e125f43a62/content>
- Hyndman, R. J., & Athanasopoulos, G. (2021). *Forecasting: Principles and practice* (3rd ed.). OTexts. Recuperado de <https://otexts.com/fpp3>
- Imperial Brands. (2024). *Principales conceptos del negocio*. Recuperado de <https://iseazy.com/dl/c9398f1ce9c44c958b61c3252a305f4d#/slide/VZ0qDyvXQF>
- Microsoft Corporation. (2023). *Power BI documentation*. Microsoft. Recuperado de <https://learn.microsoft.com/en-us/power-bi/fundamentals/power-bi-overview>
- Pérez, M., Ortega, D. & Bastidas, D. (2023). Inteligencia de Negocios para PYMES: Optimiza tus decisiones con PowerBI. Recuperado de [https://itq.edu.ec/wp-content/uploads/2023/10/2023-09-29\\_inteligencia\\_de\\_negocios\\_para\\_pymes.pdf](https://itq.edu.ec/wp-content/uploads/2023/10/2023-09-29_inteligencia_de_negocios_para_pymes.pdf)
- PricewaterhouseCoopers. (2015). Retailing 2015: New Frontiers. Recuperado de <https://www.pwc.com/cl/es/publicaciones/assets/retailing2015.pdf>
- Solano, L. E. S. (2018). Business Intelligence: un balance para su implementación. *Innovag*, (3), 27-36. Recuperado de [https://scholar.google.es/scholar?q=Silva,+L.+\(2018\).+Business+Intelligence:+Un+balance+para+su+implementaci%C3%B3n.&hl=es&as\\_sdt=0&as\\_vis=1&oi=scholar](https://scholar.google.es/scholar?q=Silva,+L.+(2018).+Business+Intelligence:+Un+balance+para+su+implementaci%C3%B3n.&hl=es&as_sdt=0&as_vis=1&oi=scholar)
- Uzcátegui-Sánchez, C., & Camino-Mogro, S. (2017). Estructura de la competencia del sector tabacalero en España: cigarrillos y tabaco de liar. *Revista Ciencia UNEMI*, 10(22), 20-28. Recuperado de <https://www.redalyc.org/journal/5826/582661263002/582661263002.pdf>