Análisis de ventas BigMart

Carlos Pérez Manzano

Tabla de contenidos

Introducción

En este proyecto vamos a realizar un análisis gráfico de un problema de ventas. Contamos con un dataset de 14204 observaciones y 12 variables que recogen las ventas de 1559 productos en 10 establecimientos de la empresa BigMart en el año 2013. Se puede encontrar el conjunto de datos en https://zenodo.org/records/6509955.

Descripción de las variables:

- **Item_Identifier**: variable cualitativa nominal indicando el código del artículo.
- **Item_Weight**: variable numérica indicando el peso del artículo. No se nos informa la unidad de medida en la que se han recogido los datos.
- **Item_Fat_Content**: variable cualitativa ordinal indicando el nivel de grasa del artículo. Los posibles valores son `Low Fat`, `Regular`, `low fat`, `LF`, `reg`.
- **Item_Visibility**: valor numérico que indica cómo de visible es un artículo.
- **Item_Type**: variable categórica indicando el tipo de producto: `Dairy`, `Soft Drinks`, `Meat`, `Fruits and Vegetables`, `Household`, `Baking Goods`, `Snack Foods`, `Frozen Foods`, `Breakfast`, `Health and Hygiene`, `Hard Drinks`, `Canned`, `Breads`, `Starchy Foods`, `Others`, `Seafood`.
- **Item_MRP**: variable numérica indicando el MRP (Maximum Retail Price) del producto.
- **Outlet_Identifier**: variable categórica que contiene el identificador del establecimiento.

- **Outlet_Establishment_Year**: variable cuantitativa discreta indicando el año de inauguración del establecimiento.
- **Outlet_Size**: variable cualtitativa ordinal que muestra el tamaño del establecimiento. Toma los valores `Medium` , `High` y `Small`.
- **Outlet_Location_Type**: variable categórica para indicar la localización del establecimiento: `Tier 1`, `Tier 2`, `Tier 3`.
- **Outlet_Type**: variable categórica que indica el tipo de establecimiento: `Supermarket Type1`, `Supermarket Type2`, `Supermarket Type3`, `Grocery Store`.
- **Item_Outlet_Sales**: variable cuantitativa discreta que indica el número de productos vendidos.

El objetivo es realizar un análisis exploratorio gráfico para comprender las propiedades de los productos y los establecimientos que pueden desempeñar un papel clave en el aumento de las ventas.

Análisis Gráfico BigMart

Carga y limpieza de datos

Cargamos en primer lugar los paquetes necesarios para lo que sigue.

```
library(naniar)
library(dplyr)
library(ggplot2)
library(gridExtra)
```

Cargamos el dataset y veamos cómo están codificadas las variables.

```
data = read.csv("data.csv")
str(data)
```

```
'data.frame':   14204 obs. of  12 variables:
 $ Item_Identifier          : chr  "FDA15" "DRC01" "FDN15" "FDX07" ...
 $ Item_Weight              : num  9.3 5.92 17.5 19.2 8.93 ...
 $ Item_Fat_Content         : chr  "Low Fat" "Regular" "Low Fat" "Regular"
 ...
 $ Item_Visibility          : num  0.016 0.0193 0.0168 0 0 ...
 $ Item_Type                : chr  "Dairy" "Soft Drinks" "Meat" "Fruits and
 Vegetables" ...
 $ Item_MRP                 : num  249.8 48.3 141.6 182.1 53.9 ...
```

```
 $ Outlet_Identifier        : chr  "OUT049" "OUT018" "OUT049" "OUT010" ...
 $ Outlet_Establishment_Year: int  1999 2009 1999 1998 1987 2009 1987 1985
 2002 2007 ...
 $ Outlet_Size              : chr  "Medium" "Medium" "Medium" "" ...
 $ Outlet_Location_Type     : chr  "Tier 1" "Tier 3" "Tier 1" "Tier 3" ...
 $ Outlet_Type              : chr  "Supermarket Type1" "Supermarket Type2"
 "Supermarket Type1" "Grocery Store" ...
 $ Item_Outlet_Sales        : num  3735 443 2097 732 995 ...
```

La variable `Item_Fat_Content` cuenta con los niveles `Low Fat`, `Regular`, `low fat`, `LF`, `reg`. Redefinimos los niveles estableciendo `Low Fat` y `Regular` como las dos únicas categorías.

```
unique(data$Item_Fat_Content)
```

```
[1] "Low Fat" "Regular" "low fat" "LF"      "reg"
```

```
data <- data %>%
  mutate(Item_Fat_Content = recode(Item_Fat_Content,
                                   "low fat" = "Low Fat",
                                   "LF" = "Low Fat",
                                   "Regular" = "Regular",
                                   "reg" = "Regular"))
```

Veamos la consistencia de los datos en el resto de variables que procedan.

```
unique(data$Item_Type)
```

```
 [1] "Dairy"                 "Soft Drinks"           "Meat"
 [4] "Fruits and Vegetables" "Household"             "Baking Goods"
 [7] "Snack Foods"           "Frozen Foods"          "Breakfast"
[10] "Health and Hygiene"    "Hard Drinks"           "Canned"
[13] "Breads"                "Starchy Foods"         "Others"
[16] "Seafood"
```

```
unique(data$Outlet_Identifier)
```

```
 [1] "OUT049" "OUT018" "OUT010" "OUT013" "OUT027" "OUT045" "OUT017" "OUT046"
 [9] "OUT035" "OUT019"
```

```
unique(data$Outlet_Establishment_Year)
```

```
[1] 1999 2009 1998 1987 1985 2002 2007 1997 2004
```

```
unique(data$Outlet_Size)
```

```
[1] "Medium" ""       "High"   "Small"
```

```
unique(data$Outlet_Location_Type)
```

```
[1] "Tier 1" "Tier 3" "Tier 2"
```

```
unique(data$Outlet_Type)
```

```
[1] "Supermarket Type1" "Supermarket Type2" "Grocery Store"
[4] "Supermarket Type3"
```

```
data <- data %>%
  mutate(Outlet_Size = ifelse(!Outlet_Size %in% c("Medium", "High",
   ↪  "Small"), NA, Outlet_Size))
```

Vemos el número de valores nulos en cada variable.

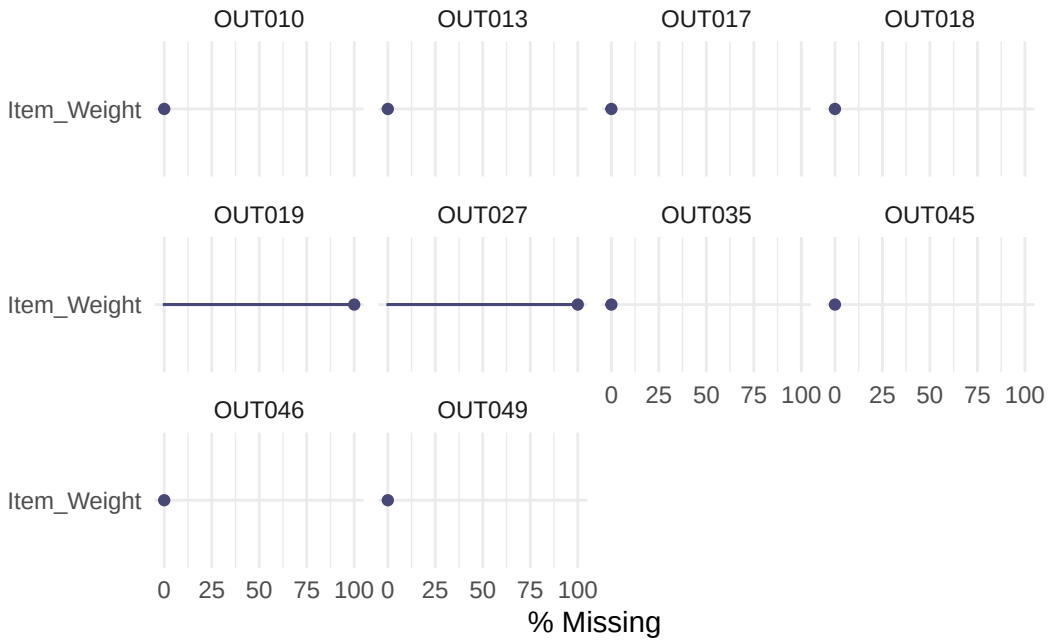
```
null_values = data %>%
  summarise(across(everything(), ~ sum(is.na(.))))

t(null_values)
```

```
                           [,1]
Item_Identifier               0
Item_Weight                2439
Item_Fat_Content              0
Item_Visibility               0
Item_Type                     0
Item_MRP                      0
Outlet_Identifier             0
Outlet_Establishment_Year     0
Outlet_Size                4016
Outlet_Location_Type          0
Outlet_Type                   0
Item_Outlet_Sales             0
```

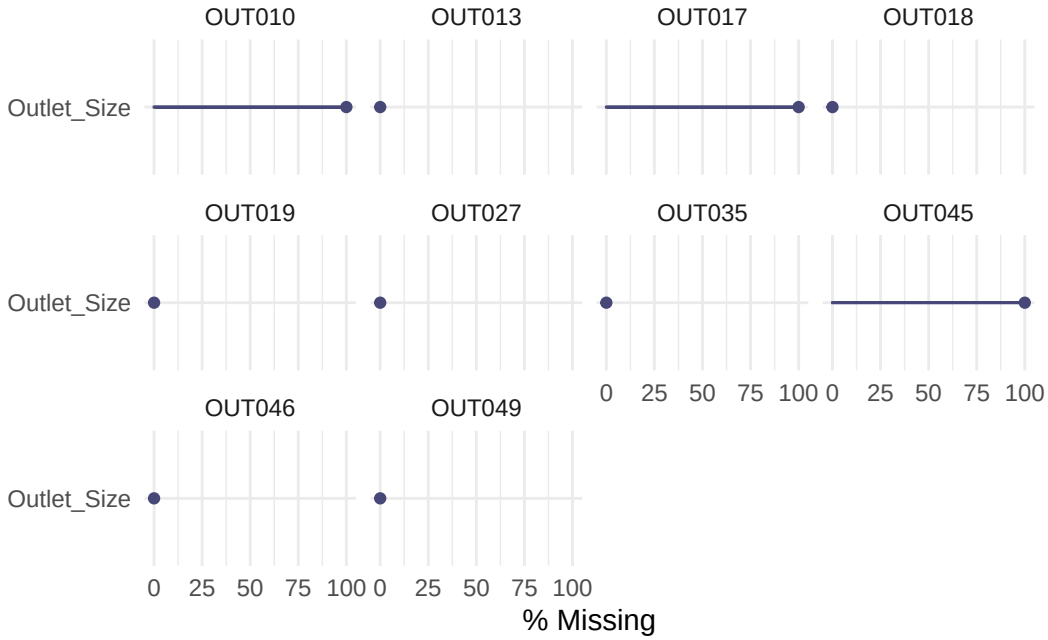
Solo tenemos valores nulos en la variable `Item_Weight` y `Outlet_Size`. La fuente de los datos nos dice que ciertos establecimientos no han informado de algunas características debido a problemas técnicos.

Analicemos el origen de los valores nulos de ambas variables.

```
gg_miss_var(data[,c(2,7)], facet = Outlet_Identifier, show_pct = TRUE) +
  theme(axis.title.y = element_blank()) +
  scale_fill_manual(values = c("#1f77b4"))
```




```
gg_miss_var(data[,c(7,9)], facet = Outlet_Identifier, show_pct = TRUE) +
  theme(axis.title.y = element_blank()) +
  scale_fill_manual(values = c("#1f77b4"))
```



Observamos como los establecimientos OUT019 y OUT027 no han informado del peso del producto en ningún caso, mientras que los demás sí lo han hecho en todos ellos. Por otro lado, los establecimientos OUT010, OUT017 y OUT045 no han informado del tamaño del establecimiento.

Sin embargo, los productos cuyo peso no han sido informados son también vendidos por otros establecimientos. Por tanto, vamos a imputar los valores nulos extrayendo el peso real de los productos del resto de establecimientos.

```
data <- data %>%
  group_by(Item_Identifier) %>%
  mutate(Item_Weight = coalesce(Item_Weight,
   ↪  first(Item_Weight[!is.na(Item_Weight)]))) %>%
  ungroup()

data %>% filter(is.na(Item_Weight)) %>% count(Item_Weight)
```

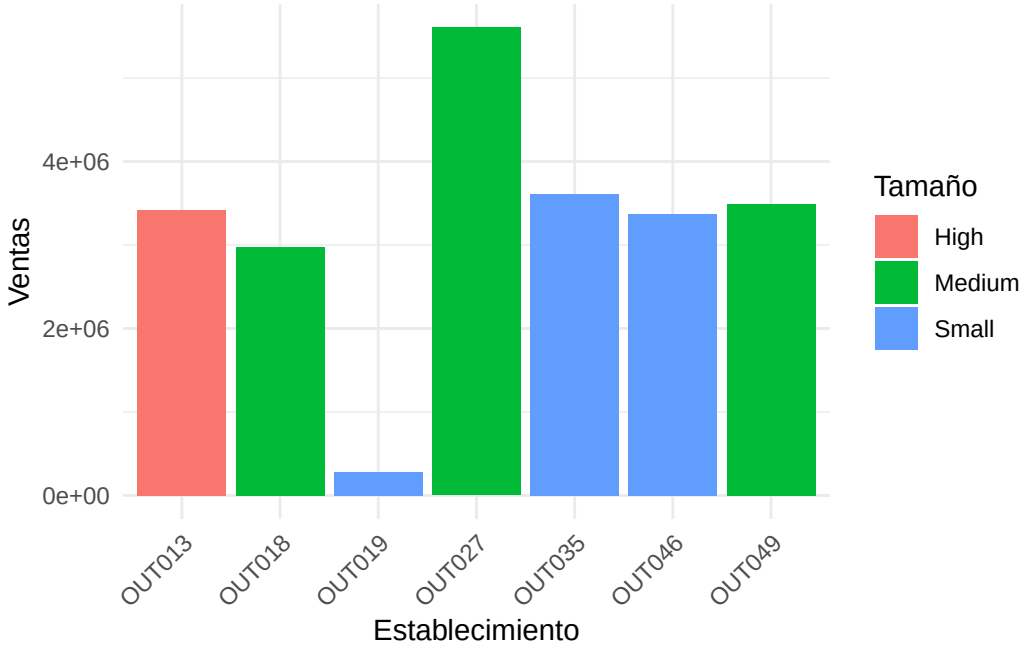
```
# A tibble: 0 x 2
# i 2 variables: Item_Weight <dbl>, n <int>
```

Observamos que ya no tenemos valores nulos en `Item_Weight`.

En cuanto a la variable `Outlet_Size`, vamos a realizar una predicción de manera exploratoria del tamaño del establecimiento en función del número de ventas. Es aceptable argumentar que

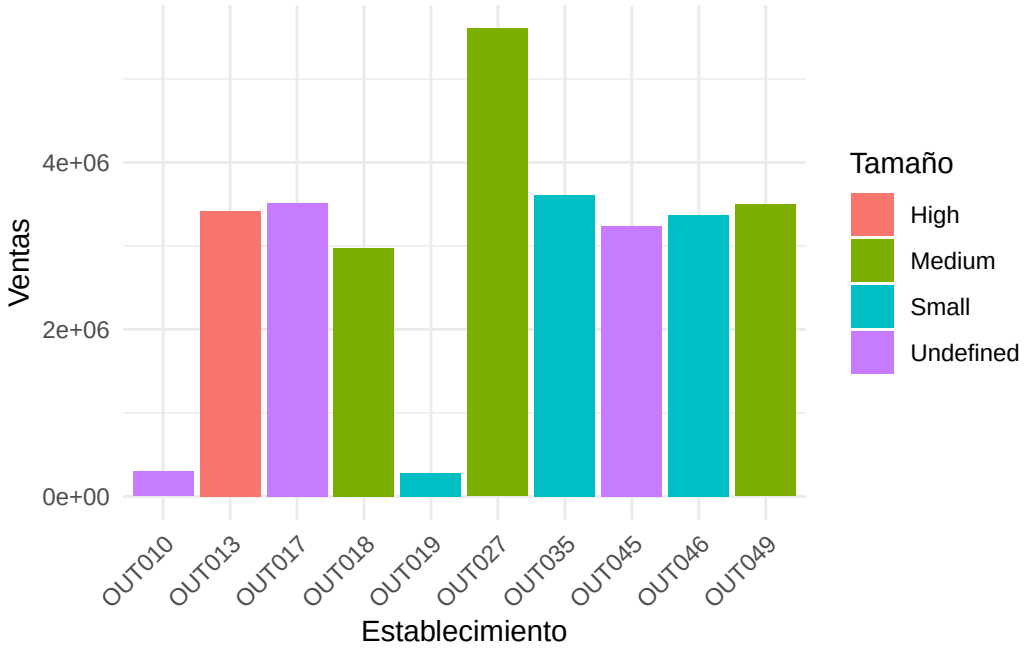
los establecimientos con un mayor número de ventas se deben corresponder con aquellos de un mayor tamaño.

```
ggplot(data %>%
         filter(!is.na(Outlet_Size)) %>%
         group_by(Outlet_Identifier, Outlet_Size) %>%
         summarise(ventas = sum(Item_Outlet_Sales), .groups = "drop"),
       aes(x = Outlet_Identifier, y = ventas, fill =
       ↪  as.factor(Outlet_Size))) +
  geom_col() +
  labs(x = "Establecimiento", y = "Ventas", fill = "Tamaño") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```




```
ggplot(data %>%
         group_by(Outlet_Identifier, Outlet_Size) %>%
         summarise(ventas = sum(Item_Outlet_Sales), .groups = "drop") %>%
         mutate(Outlet_Size = ifelse(!Outlet_Size %in% c("Medium",
          ↪  "High", "Small"), "Undefined", Outlet_Size)),
       aes(x = Outlet_Identifier, y = ventas, fill =
       ↪  as.factor(Outlet_Size))) +
```

```
  geom_col() +
  labs(x = "Establecimiento", y = "Ventas", fill = "Tamaño") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



Teniendo en cuenta los gráficos anteriores, en el establecimiento OUT010 imputamos `Outlet_Size` = "Small", y el resto por "Medium".

```
data <- data %>%
  mutate(Outlet_Size = ifelse(Outlet_Identifier == "OUT010", "Small",
   ↪  Outlet_Size))

data <- data %>%
  mutate(Outlet_Size = ifelse(Outlet_Identifier %in%
   ↪  c("OUT017","OUT045"), "Medium", Outlet_Size))
```

Comprobamos como no tenemos ya valores nulos.

```
null_values = data %>%
  summarise(across(everything(), ~ sum(is.na(.))))
```

```
t(null_values)
```

```
                           [,1]
Item_Identifier               0
Item_Weight                   0
Item_Fat_Content              0
Item_Visibility               0
Item_Type                     0
Item_MRP                      0
Outlet_Identifier             0
Outlet_Establishment_Year     0
Outlet_Size                   0
Outlet_Location_Type          0
Outlet_Type                   0
Item_Outlet_Sales             0
```

También tenemos el problema de que existen productos con visibilidad nula pero cuentan con ventas. Sustituiremos por la media de la visibilidad de los productos del mismo tipo.

```
data <- data %>%
  group_by(Item_Type) %>%
  mutate(Item_Visibility = ifelse(Item_Visibility == 0,
                                  mean(Item_Visibility[Item_Visibility >
                                   ↪  0]), Item_Visibility)) %>%
  ungroup()
```

Vamos a crear dos dataframes que emplearemos a lo largo de lo que sigue del proyecto.

El primero, referente a las características de los productos. Notar que tanto el MRP como la visibilidad y obviamente el número de ventas dependen del establecimiento. Tomaremos la media en para las dos primeras variables y la suma total para las ventas. Veamos un ejemplo para el producto FDA15.

```
data %>% filter(Item_Identifier == "FDA15") %>%
  select(Item_Identifier,Item_MRP,Item_Visibility) %>%
  slice_head(n = 5)
```

```
# A tibble: 5 x 3
  Item_Identifier Item_MRP Item_Visibility
  <chr>              <dbl>           <dbl>
1 FDA15               250.          0.0160
```

```
2 FDA15                250.          0.0161
3 FDA15                249.          0.0160
4 FDA15                250.          0.0161
5 FDA15                249.          0.0268
```

Efectivamente, estas variables dependen también del establecimiento.

```
items = data %>% group_by(Item_Identifier) %>%
  summarise(Item_Weight = first(Item_Weight),
            Item_Fat_Content = first(Item_Fat_Content),
            Item_Visibility = mean(Item_Visibility),
            Item_Type = first(Item_Type),
            Item_MRP = mean(Item_MRP),
            Item_Sales = sum(Item_Outlet_Sales)) %>%
  arrange(desc(Item_Sales))
```

Por otro lado un dataframe referente a las características de los establecimientos. Para las ventas, tomaremos la suma.

```
outlets = data %>% group_by(Outlet_Identifier) %>%
  summarise(Outlet_Establishment_Year =
   ↪  first(Outlet_Establishment_Year),
            Outlet_Size = first(Outlet_Size),
            Outlet_Location_Type = first(Outlet_Location_Type),
            Outlet_Type = first(Outlet_Type),
            Outlet_Sales = sum(Item_Outlet_Sales),
            Outlet_Items = n()) %>%
  arrange(desc(Outlet_Sales))
```

Análisis gráfico

Estudio univariante

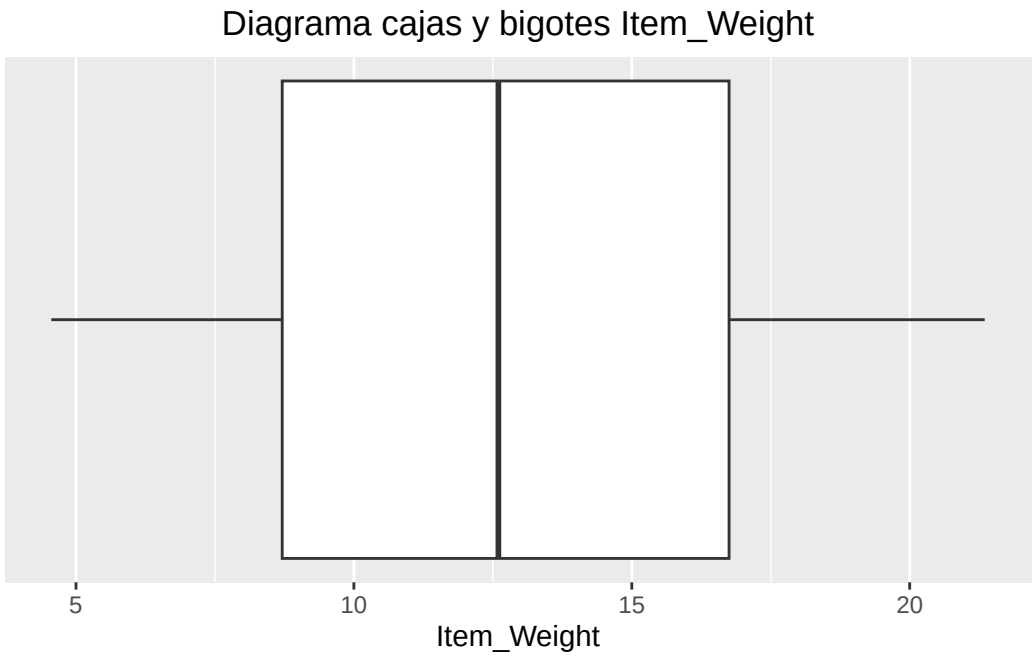
Analicemos en primer lugar las variables individualmente.

Comencemos con las variables continuas.

Variable `Item_Weight`:

```
ggplot(items, aes(Item_Weight)) +
  geom_boxplot() +
  scale_y_continuous(NULL, breaks = NULL) +
  theme(plot.title = element_text(hjust = 0.5)) +
```

```
  labs(title = "Diagrama cajas y bigotes Item_Weight")
```




```
median(items$Item_Weight) - IQR(items$Item_Weight)/2
```

```
[1] 8.58
```

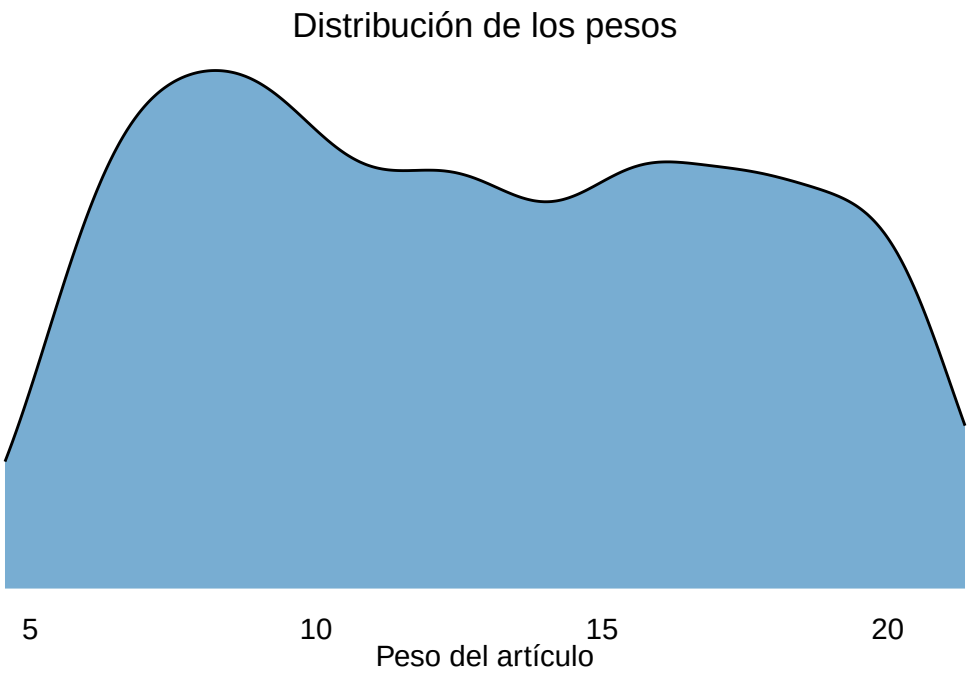
```
median(items$Item_Weight) + IQR(items$Item_Weight)/2
```

```
[1] 16.62
```

La mayor parte de los pesos de los artículos se encuentran entre 8.58 y 16.62.

```
ggplot(items, aes(Item_Weight)) +
  geom_density(fill = "#1f77b4", alpha = 0.6) +
  ylab("") +
  scale_y_continuous(NULL, breaks = NULL) +
  theme_void() +  # Elimina el fondo y los grids
  theme(axis.text = element_text(color = "black"),
        axis.title = element_text(color = "black"),
         plot.title = element_text(hjust = 0.5)) +
```

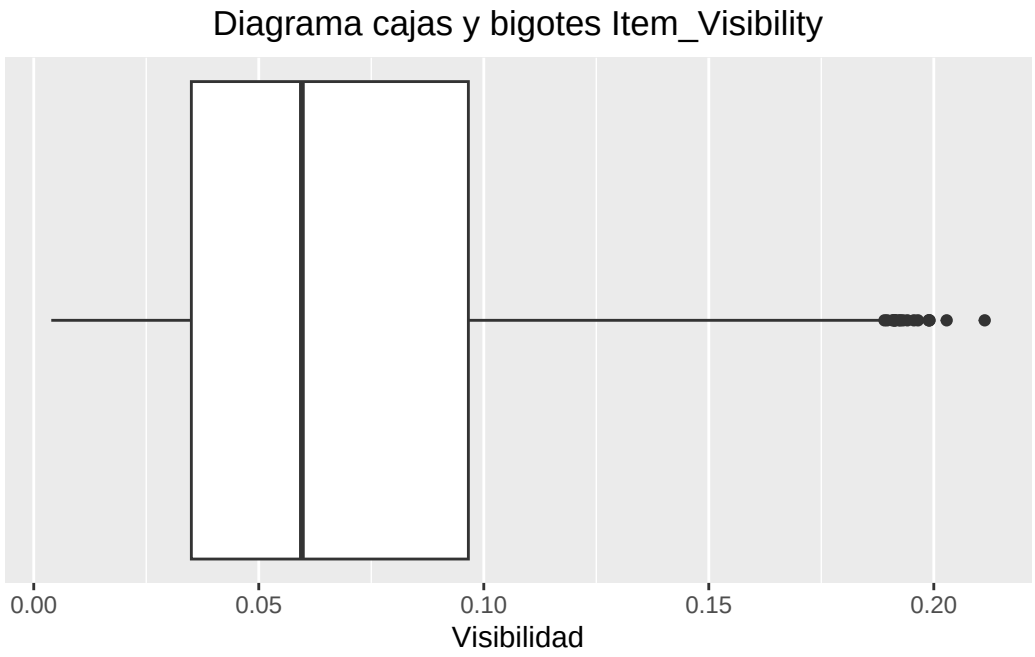
```
  labs(title =  "Distribución de los pesos",
       x = "Peso del artículo")
```



La mayor parte de los pesos de los productos están comprendidos entre 8.58 y 16.62. No tenemos presencia de outliers. Observamos además una distribución bastante simétrica.

Variable `Item_Visibility`:

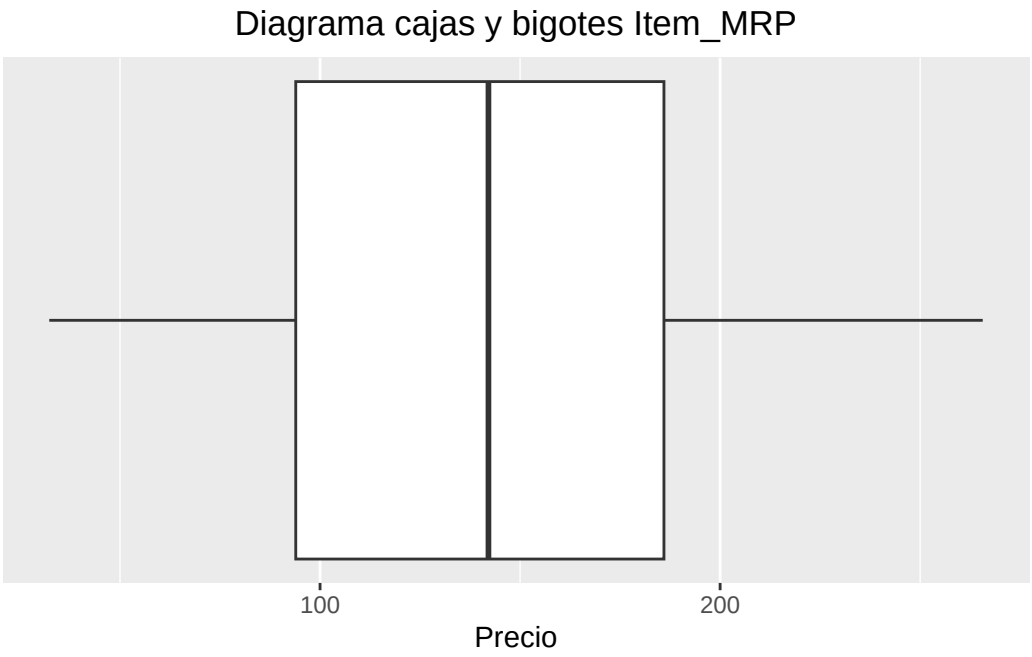
```
ggplot(items, aes(Item_Visibility)) +
  geom_boxplot() +
  scale_y_continuous(NULL, breaks = NULL) +
  theme(plot.title = element_text(hjust = 0.5)) +
  labs(title = "Diagrama cajas y bigotes Item_Visibility",
       x = "Visibilidad")
```



Observamos como la mayoría de los productos tienen una visibilidad por debajo de 0.1 y tenemos un cierto número de productos con una visibilidad considerablemente por encima del resto.

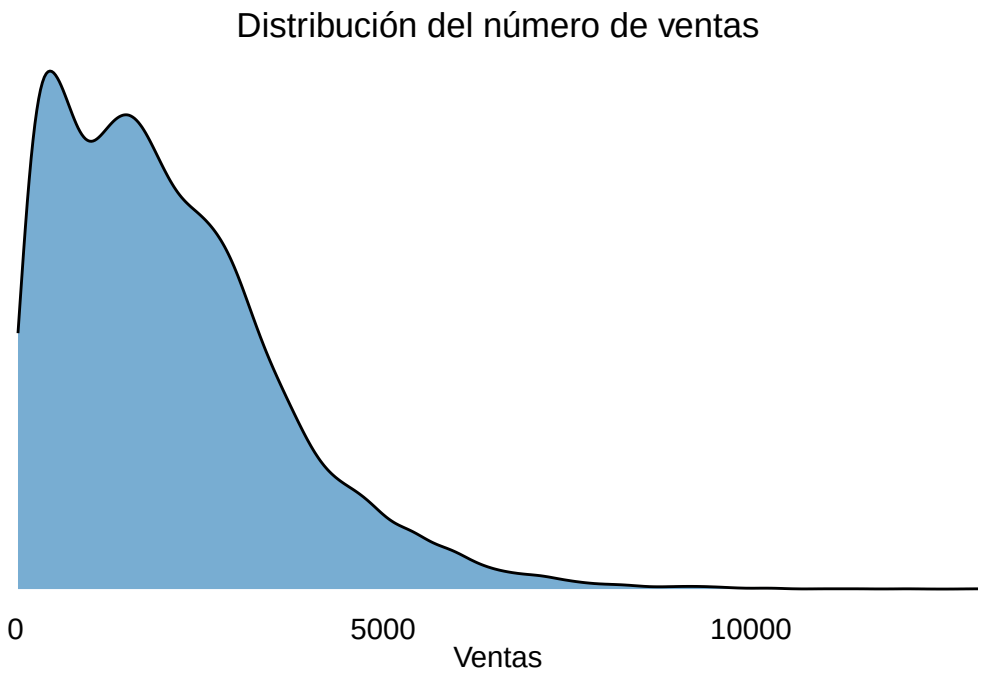
Variable `Item_MRP`:

```
ggplot(items, aes(Item_MRP)) +
  geom_boxplot() +
  scale_y_continuous(NULL, breaks = NULL) +
  theme(plot.title = element_text(hjust = 0.5)) +
  labs(title = "Diagrama cajas y bigotes Item_MRP",
       x = "Precio")
```



Variable `Item_Outlet_Sales`.

```
ggplot(data, aes(Item_Outlet_Sales)) +
  geom_density(fill = "#1f77b4", alpha = 0.6) +
  ylab("") +
  scale_y_continuous(NULL, breaks = NULL) +
  theme_void() +  # Elimina el fondo y los grids
  theme(axis.text = element_text(color = "black"),
        axis.title = element_text(color = "black"),
        plot.title = element_text(hjust = 0.5)) +
  labs(title =  "Distribución del número de ventas",
       x = "Ventas")
```



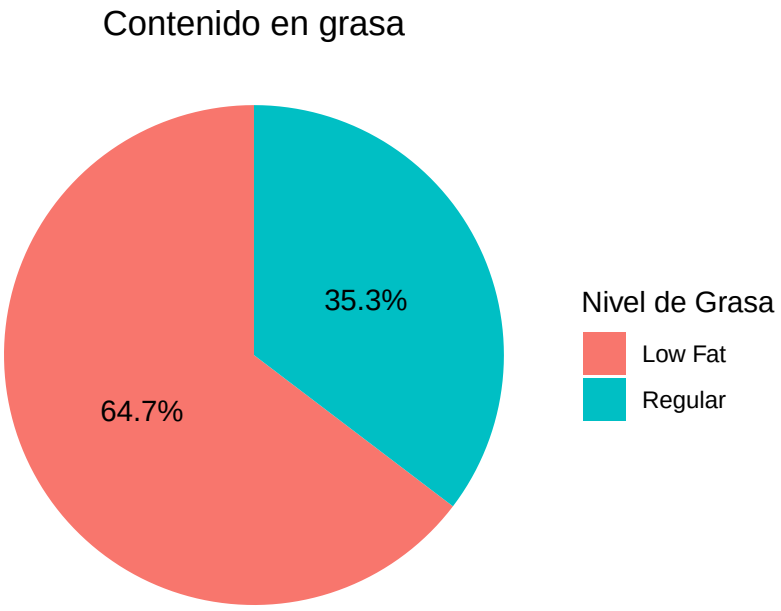
Observamos una distribución similar a una de grado en ley de potencia. El número de ventas de los productos en los establecimientos suele ser parecido, mientras que hay ciertos productos que tienen un número de ventas mucho mayor, o más bien productos que logran un mayor número de ventas en ciertos establecimientos.

Pasemos ahora a las variables categóricas.

Variable `Item_Fat_Content`:

```
fatContent = data %>% group_by(Item_Identifier) %>% summarise(fc =
 ↪  first(Item_Fat_Content))

df = as.data.frame(table(fatContent$fc))
colnames(df) <- c("categoria", "frecuencia")

df$porcentaje <- round(100 * df$frecuencia / sum(df$frecuencia), 1)

ggplot(df, aes(x = "", y = frecuencia, fill = categoria)) +
  geom_bar(stat = "identity", width = 1) +
  coord_polar("y", start = 0) +
  theme_void() +
  geom_text(aes(label = paste0(porcentaje, "%")), position =
   ↪  position_stack(vjust = 0.5)) +
```

```
  theme(plot.title = element_text(hjust = 0.5)) +
  labs(fill = "Nivel de Grasa", title = "Contenido en grasa")
```

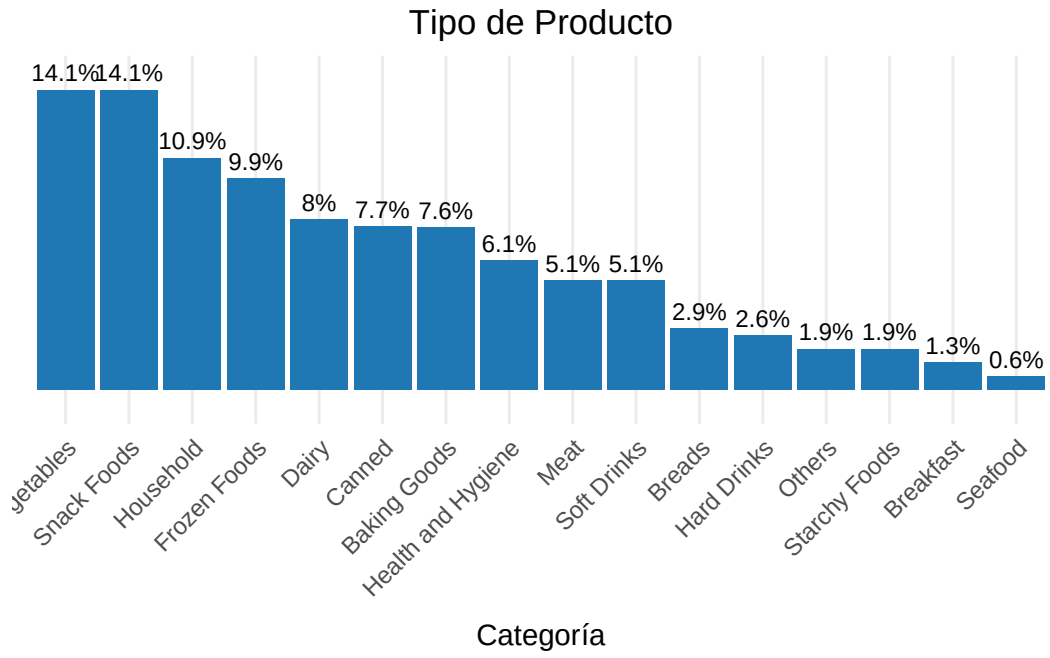


Vemos como hay un mayor número de productos con un bajo nivel de grasa.

Variable `Item_Type`:

```
itemType <- data %>% group_by(Item_Identifier) %>% summarise(it =
 ↪  first(Item_Type))
df <- as.data.frame(table(itemType$it))
colnames(df) <- c("categoria", "frecuencia")
df$porcentaje <- round(100 * df$frecuencia / sum(df$frecuencia), 1)

ggplot(df, aes(x = reorder(categoria, -frecuencia), y = frecuencia)) +
  geom_col(fill = "#1f77b4") +
  geom_text(aes(label = paste0(porcentaje, "%")), vjust = -0.5, size =
   ↪  3) +
  theme_minimal() +
  labs(x = "Categoría", y = "", title = "Tipo de Producto") +
  scale_y_continuous(NULL, breaks = NULL, expand = c(0, 25)) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1),
```

```
        plot.title = element_text(hjust = 0.5))
```

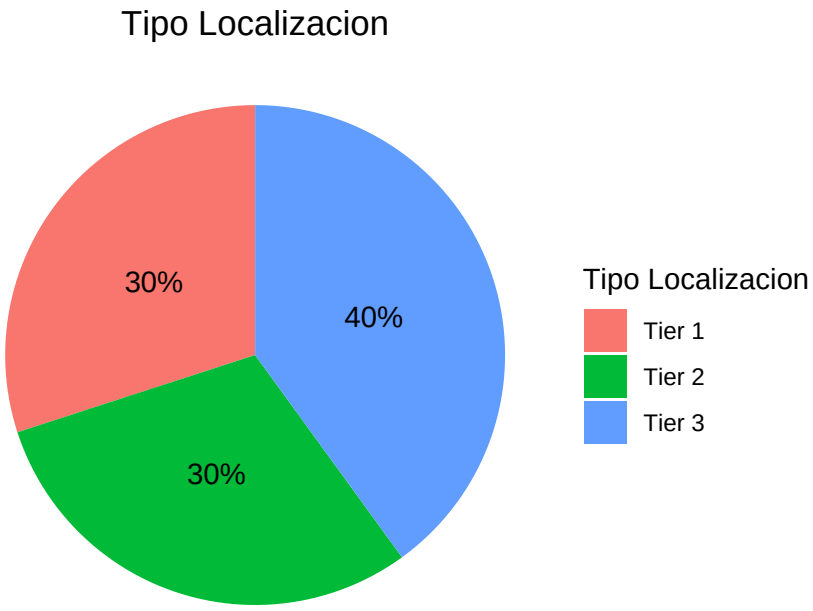


Los vegetales y los snacks son los tipos de comida con un mayor porcentaje de presencia entre los productos.

Variable `Outlet_Location_Type`:

```
locationType = data %>% group_by(Outlet_Identifier) %>% summarise(lt =
 ↪  first(Outlet_Location_Type))

df = as.data.frame(table(locationType$lt))
colnames(df) <- c("categoria", "frecuencia")

df$porcentaje <- round(100 * df$frecuencia / sum(df$frecuencia), 1)

ggplot(df, aes(x = "", y = frecuencia, fill = categoria)) +
  geom_bar(stat = "identity", width = 1) +
  coord_polar("y", start = 0) +
  theme_void() +
   geom_text(aes(label = paste0(porcentaje, "%")), position =
    ↪  position_stack(vjust = 0.5)) +
```

```
  labs(fill = "Tipo Localizacion", title = "Tipo Localizacion") +
  theme(plot.title = element_text(hjust = 0.5))
```

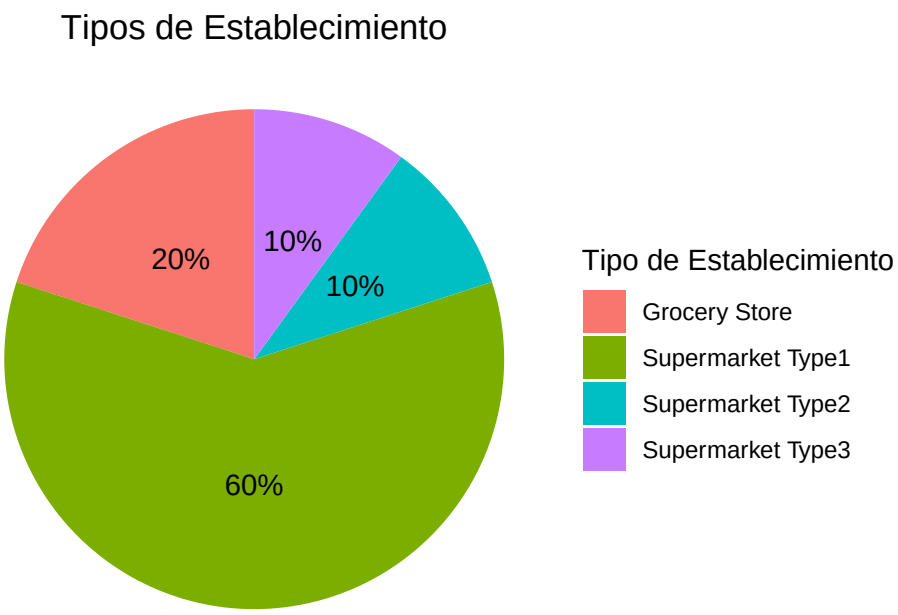


Están bastante balanceados los tipos de localización.

Variable `Outlet_Type`:

```
outletType = data %>% group_by(Outlet_Identifier) %>% summarise(et =
 ↪  first(Outlet_Type))

df = as.data.frame(table(outletType$et))
colnames(df) <- c("categoria", "frecuencia")

df$porcentaje <- round(100 * df$frecuencia / sum(df$frecuencia), 1)

ggplot(df, aes(x = "", y = frecuencia, fill = categoria)) +
  geom_bar(stat = "identity", width = 1) +
  coord_polar("y", start = 0) +
  theme_void() +
   geom_text(aes(label = paste0(porcentaje, "%")), position =
    ↪  position_stack(vjust = 0.5)) +
```

```
  labs(fill = "Tipo de Establecimiento", title = "Tipos de
   ↪  Establecimiento") +
  theme(plot.title = element_text(hjust = 0.5))
```



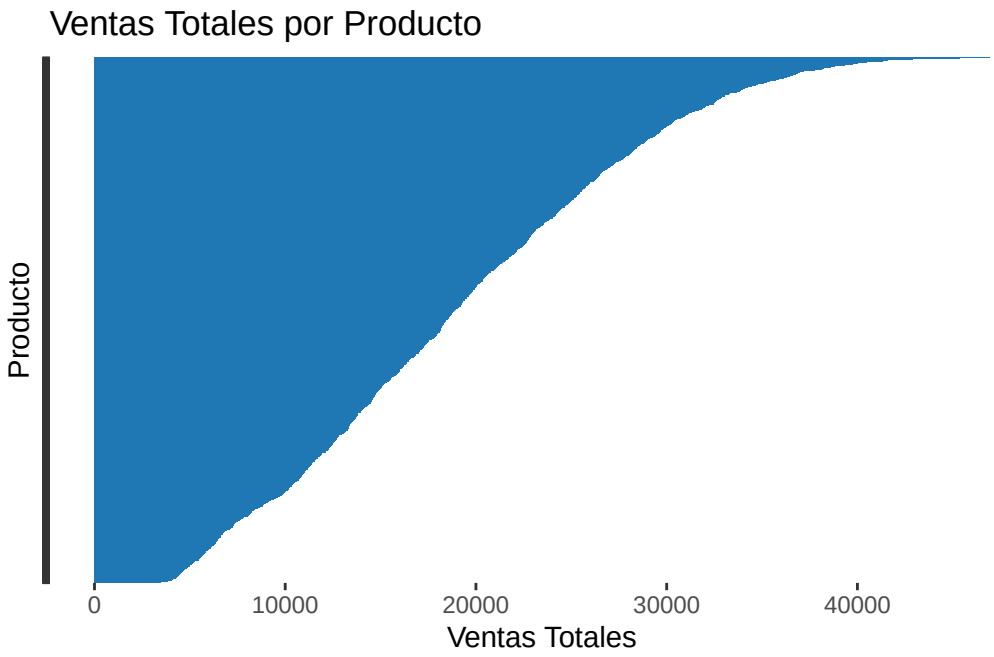
El tipo más común es el supermercado tipo 1.

Relación entre Variables

Analicemos en primer lugar las variables correspondientes a los productos.

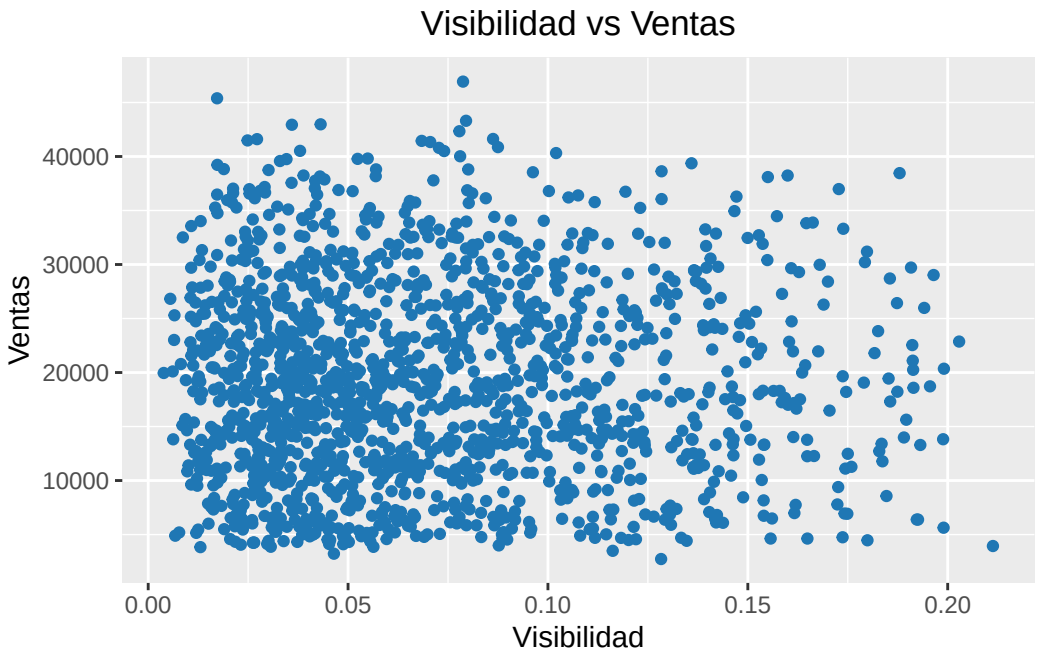
Veamos cuáles son los productos más vendidos.

```
ggplot(items, aes(Item_Sales, forcats::fct_reorder(Item_Identifier,
 ↪  Item_Sales))) +
  geom_col(fill = "#1f77b4") +
  theme(axis.text.y = element_blank()) +
  labs(x = "Ventas Totales" ,
       y = "Producto",
       title = "Ventas Totales por Producto")
```



Hay grandes diferencias entre el número de ventas de los distintos productos. Esto es algo muy a tener en cuenta para la logística del stock. Veamos la relación del número de ventas de los productos y la visibilidad.

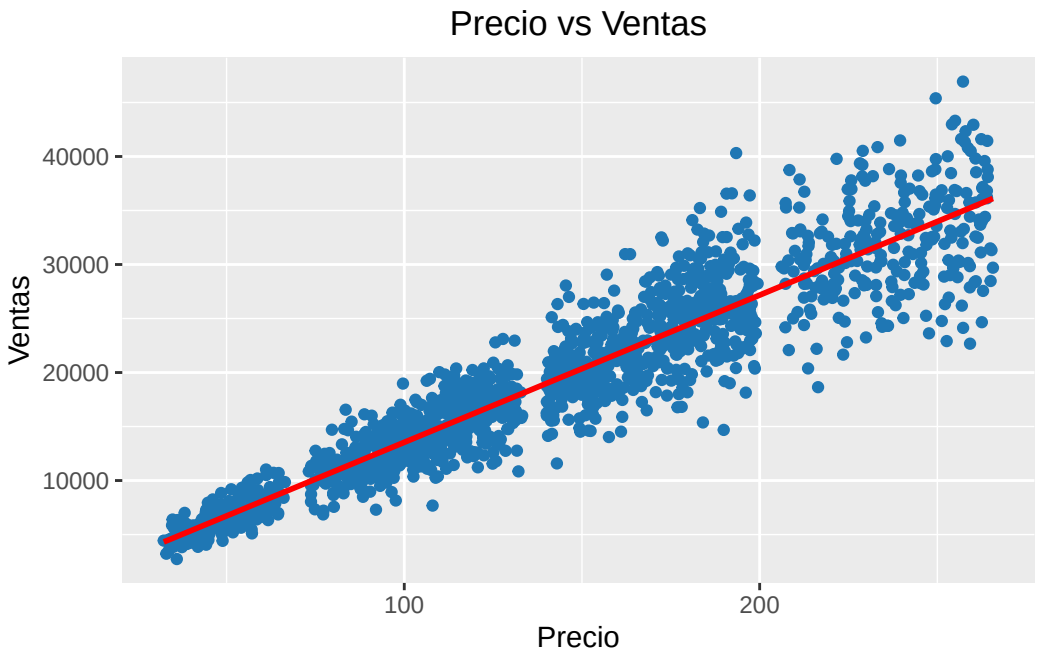
```
ggplot(items, aes(x = Item_Visibility, y = Item_Sales)) +
  geom_point(color = "#1f77b4") +
  labs(x = "Visibilidad", y = "Ventas",
       title = "Visibilidad vs Ventas") +
  theme(plot.title = element_text(hjust = 0.5))
```



Observamos como los productos con un mayor número de ventas no tienen una gran visibilidad, mientras que hay artículos con una mayor visibilidad que no necesariamente tienen un gran número de ventas. Esto se puede deber a que los productos que consiguen un mayor número de ventas se corresponden con productos con una gran demanda, lo que provoca que no sea necesario darles gran visibilidad. Por otro lado, se intenta potenciar a los productos que tienen un menor número de ventas (quizás por una baja demanda) dándoles mayor visibilidad.

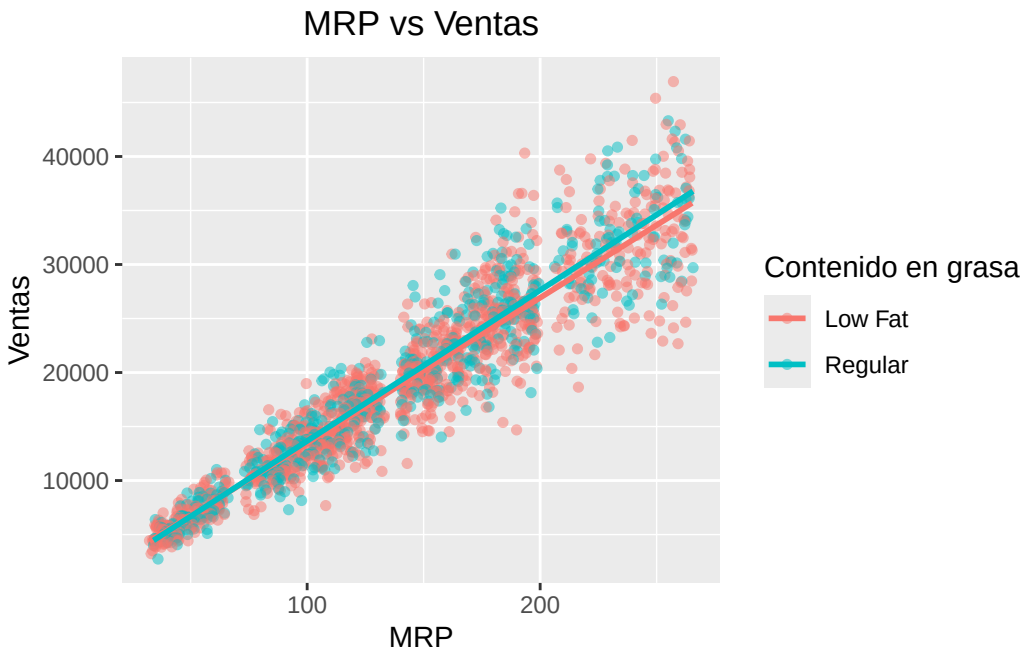
Ahora, analicemos como afecta el precio del producto (MRP) en las ventas.

```
ggplot(items, aes(x = Item_MRP, y = Item_Sales)) +
  geom_point(color = "#1f77b4") +
  geom_smooth(method = "lm", color = "red", se = FALSE) +
  labs(x = "Precio", y = "Ventas",
       title = "Precio vs Ventas") +
  theme(plot.title = element_text(hjust = 0.5))
```



Contamos con una clara relación lineal entre las ventas y el precio. Las ventas son sustancialmente mayores conforme el precio de los productos aumenta. Notamos también como la fluctuación de las ventas es mayor a medida que se va aumentado el precio del producto. Añadimos en el gráfico anterior un indicador del nivel de grasa para ver qué impacto tiene el contenido en grasa en los precios y las ventas.

```
ggplot(items, aes(x = Item_MRP, y = Item_Sales, color =
 ↪  factor(Item_Fat_Content))) +
  geom_point(size = 1.3, alpha = 0.5) +
  geom_smooth(method = "lm",se = FALSE) +
  labs(x = "MRP", y = "Ventas",
       title = "MRP vs Ventas",
       color = "Contenido en grasa") +
  theme(plot.title = element_text(hjust = 0.5))
```

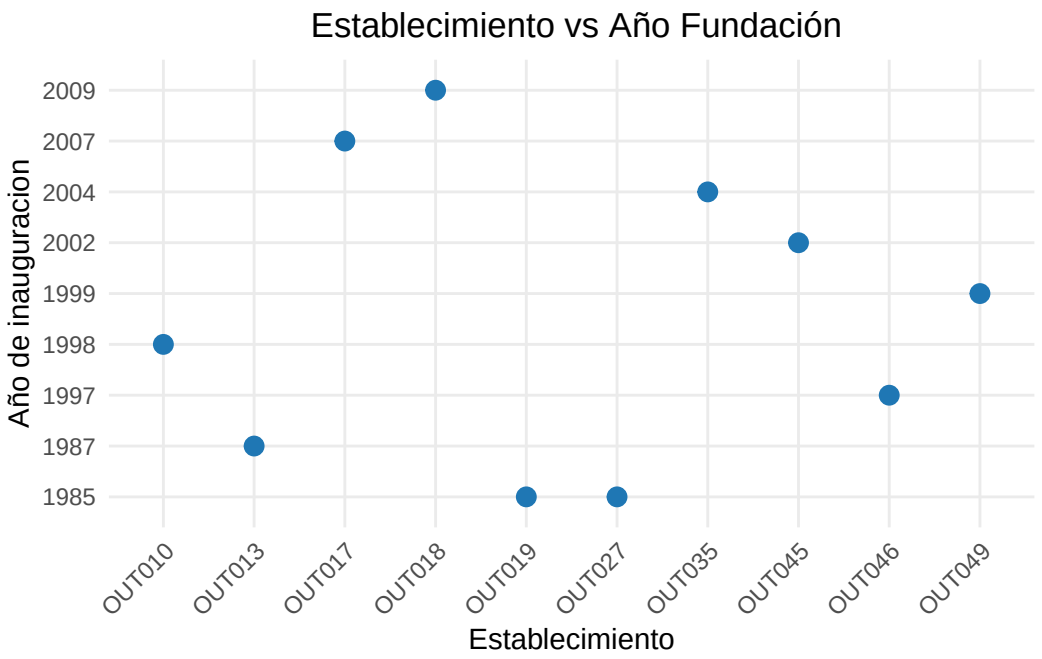


No se aprecia relación entre el contenido en grasa y el precio o las ventas de los productos.

Pasemos a ver ahora las características de los establecimientos

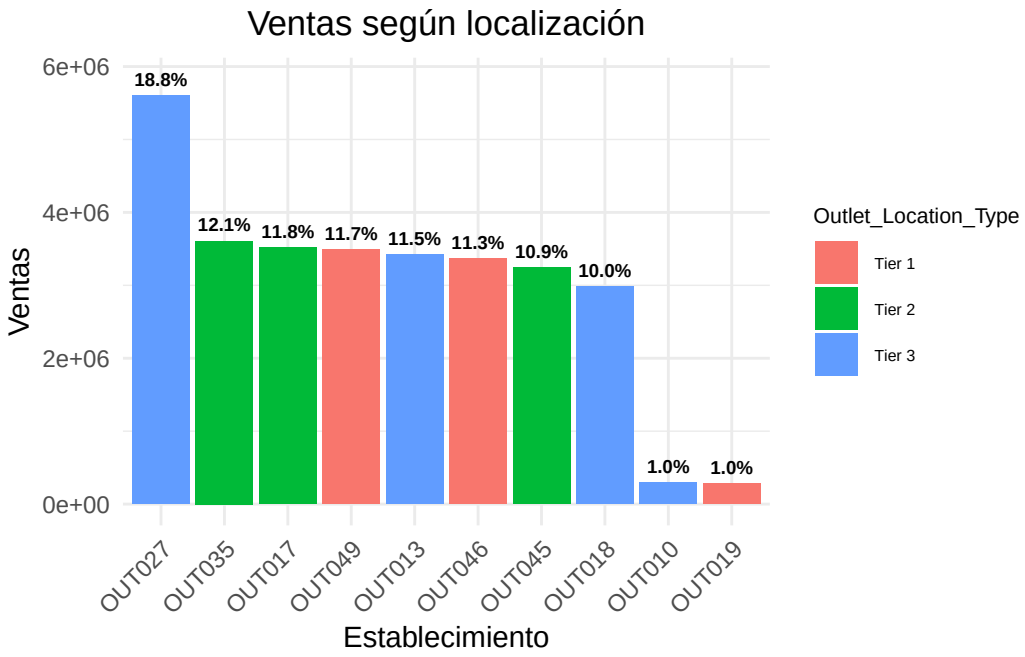
Veamos en primer como se distribuyen los años de establecimiento de los establecimientos.

```
ggplot(outlets, aes(x = Outlet_Identifier, y =
↪  factor(Outlet_Establishment_Year))) +
  geom_point(size = 3, color = "#1f77b4") +
  theme_minimal() +
  labs(x = "Establecimiento", y = "Año de inauguracion",
       title = " Establecimiento vs Año Fundación") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1),
        plot.title = element_text(hjust = 0.5))
```



Veamos las ventas por establecimiento, distinguiendo la ubicación

```
df <- outlets %>%
  group_by(Outlet_Identifier, Outlet_Location_Type) %>%
  summarise(ventas = sum(Outlet_Sales), .groups = "drop") %>%
  mutate(porcentaje = ventas / sum(ventas) * 100)

ggplot(df, aes(x = reorder(Outlet_Identifier, -ventas), y = ventas, fill
 ↪  = Outlet_Location_Type)) +
  geom_col() +
  geom_text(aes(label = sprintf("%.1f%%", porcentaje),
                y = ventas + max(ventas) * 0.04), size = 2.5, fontface =
                 ↪  "bold") +
  theme_minimal() +
  labs(x = "Establecimiento", y = "Ventas", title = "Ventas según
   ↪  localización") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1),
        plot.title = element_text(hjust = 0.5),
        legend.title = element_text(size = 8),
        legend.text = element_text(size = 6))
```

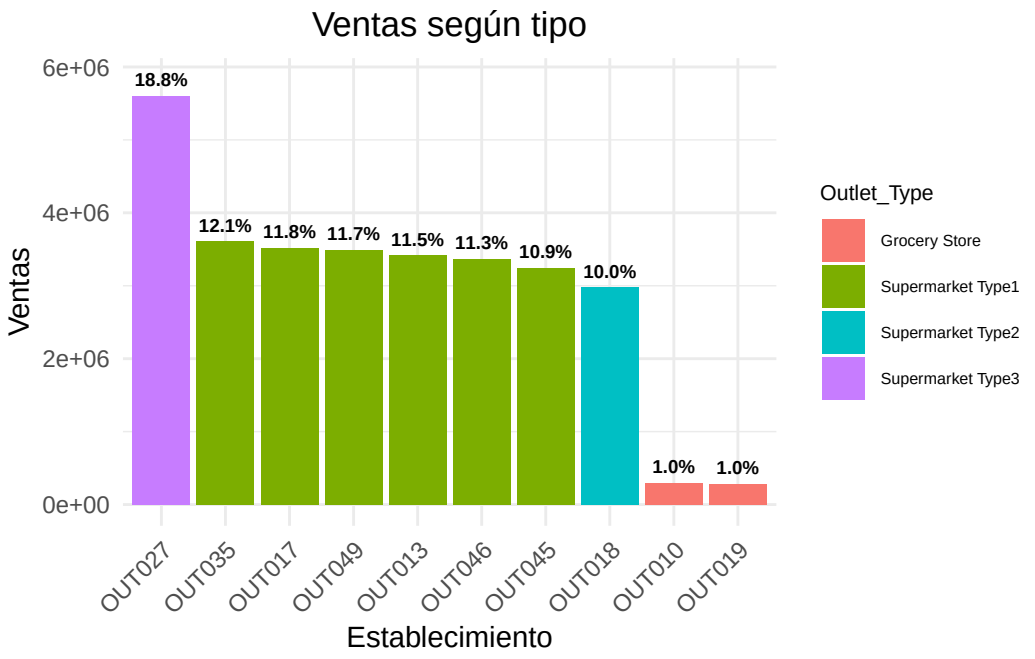


El establecimiento OUT027 es claramente el que más ventas realiza, siendo el porcentaje sobre el total de 18.8. Destacan también los establecimientos OUT010 y OUT019 pues realizan unicamente un 1% de las ventas totales. El resto de tiendas tienen unas ventas bastante similares. No contemplamos sin embargo una relación entre la ubicación y el número de ventas.

Análogamente para el tipo de establecimiento.

```
df <- outlets %>%
  group_by(Outlet_Identifier, Outlet_Type) %>%
  summarise(ventas = sum(Outlet_Sales), .groups = "drop") %>%
  mutate(porcentaje = ventas / sum(ventas) * 100)

ggplot(df, aes(x = reorder(Outlet_Identifier, -ventas), y = ventas, fill
 ↪  = Outlet_Type)) +
  geom_col() +
  geom_text(aes(label = sprintf("%.1f%%", porcentaje),
                y = ventas + max(ventas) * 0.04), size = 2.5, fontface =
                 ↪  "bold") +
  theme_minimal() +
  labs(x = "Establecimiento", y = "Ventas", title = "Ventas según tipo")
   ↪  +
```

```
  theme(axis.text.x = element_text(angle = 45, hjust = 1),
        plot.title = element_text(hjust = 0.5),
        legend.title = element_text(size = 8),
        legend.text = element_text(size = 6))
```



Para esta variable si vemos una clara diferenciación. Los establecimientos con un menor número de ventas se corresponden con tiendas de comestibles, lo cual es lógico. El establecimiento con más ventas es el único que es del tipo 3.

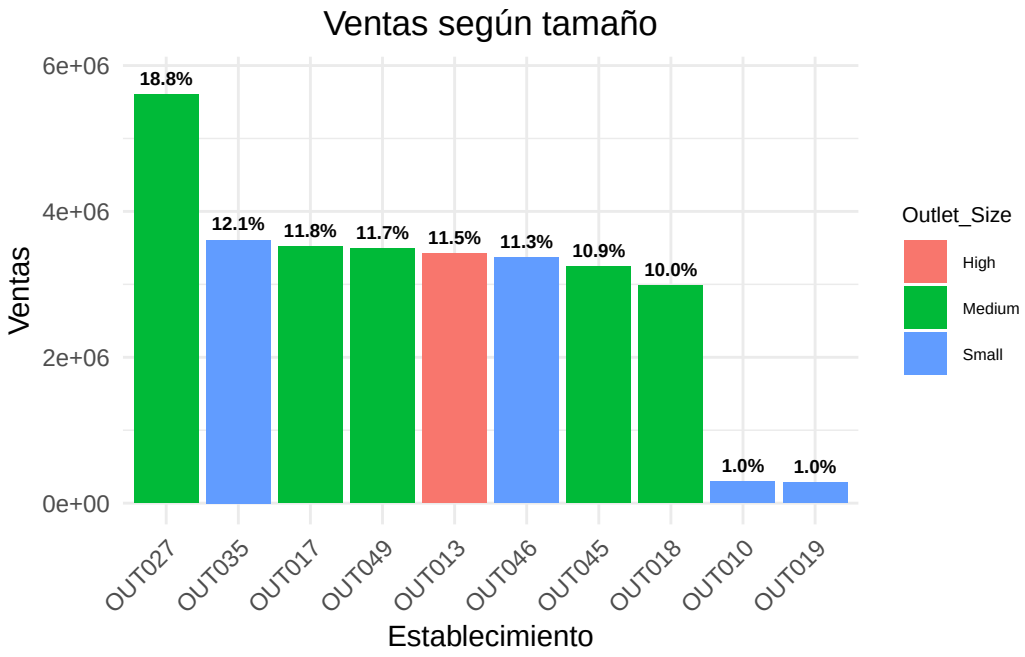
Podemos dar una ordenación de los tipos de supermercados según sus ventas:

`Supermarket Type3` > `Supermarket Type1` > `Supermarket Type2` > `Grocery Store`

Análogamente para el tamaño de establecimiento

```
df <- outlets %>%
  group_by(Outlet_Identifier, Outlet_Size) %>%
  summarise(ventas = sum(Outlet_Sales), .groups = "drop") %>%
  mutate(porcentaje = ventas / sum(ventas) * 100)

ggplot(df, aes(x = reorder(Outlet_Identifier, -ventas), y = ventas, fill
 ↪  = Outlet_Size)) +
```

```
geom_col() +
geom_text(aes(label = sprintf("%.1f%%", porcentaje),
              y = ventas + max(ventas) * 0.04), size = 2.5, fontface =
               ↪  "bold") +
theme_minimal() +
labs(x = "Establecimiento", y = "Ventas", title = "Ventas según
 ↪  tamaño") +
theme(axis.text.x = element_text(angle = 45, hjust = 1),
      plot.title = element_text(hjust = 0.5),
      legend.title = element_text(size = 8),
      legend.text = element_text(size = 6))
```

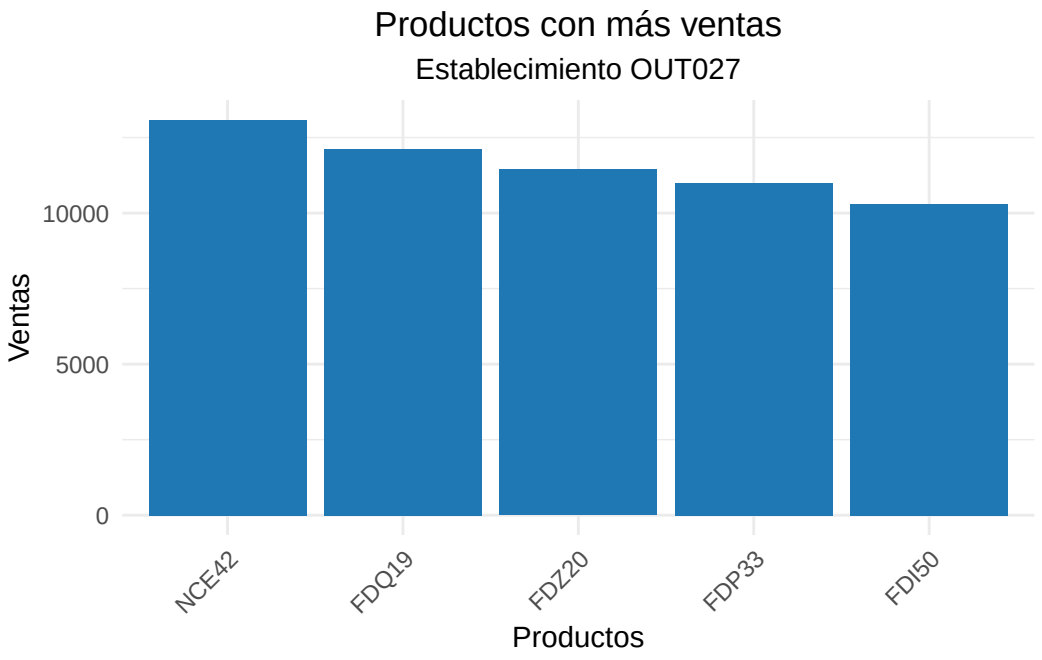


Podemos considerar que uno de los mayores factores de éxito de OUT027 es la oferta de ciertos items que producen gran interés en los clientes. Además, estos productos pueden no venderse de igual forma en el resto de establecimientos debido a una mala gestión de marketing o que la calidad de los mismos difiere.

Vamos a tomar los 2 prductos más vendidos para el OUT027 y para ellos veremos la posición que ocupan en el ránking de artículos más vendidos para el resto de establecimientos

```r
df = data %>% filter (Outlet_Identifier == "OUT027") %>%
            group_by(Outlet_Identifier, Item_Identifier) %>%
            summarise(Item_Sales = sum(Item_Outlet_Sales), .groups =
             ↪  "drop") %>%
            arrange(desc(Item_Sales)) %>%
            slice_head(n = 5)  %>%
            select(Item_Identifier, Item_Sales)

products=df$Item_Identifier[1:2]
```

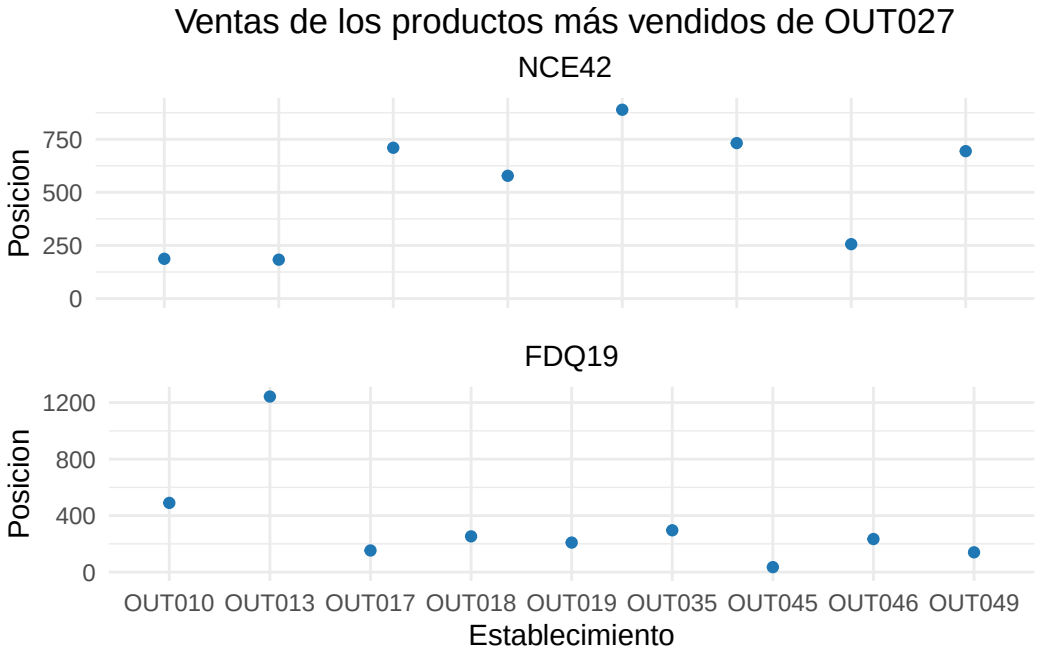
```r
ggplot(df, aes(x = reorder(Item_Identifier, -Item_Sales), y =
 ↪  Item_Sales)) +
  geom_col(fill = "#1f77b4") +
  theme_minimal() +
  labs(title = "Productos con más ventas", subtitle = "Establecimiento
   ↪  OUT027",
       x = "Productos", y = "Ventas" ) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1),
        plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5))
```



Vemos como los ítems NCE42 y FDQ19 son los que más ventas realizan.

```
df = data %>%
  filter(Outlet_Identifier != "OUT027") %>%
  group_by(Outlet_Identifier, Item_Identifier) %>%
  summarise(Item_Sales = sum(Item_Outlet_Sales), .groups = "drop") %>%
  arrange(Outlet_Identifier, desc(Item_Sales)) %>%
  group_by(Outlet_Identifier) %>%
  mutate(Item_Outlet_Ranking = row_number()) %>%
  ungroup() %>%
  filter(Item_Identifier %in% products)
```

```
p1 = ggplot(df %>% filter(Item_Identifier == "NCE42"),
 ↪  aes(Outlet_Identifier, Item_Outlet_Ranking)) +
      geom_point(color = "#1f77b4") +
      labs(title = "Ventas de los productos más vendidos de
       ↪  OUT027",subtitle="NCE42",
           x = "Establecimiento", y = "Posicion") +
      theme_minimal() +
      theme(axis.title.x = element_blank(), axis.text.x =
       ↪  element_blank(),
            plot.title = element_text(hjust = 0.5),
            plot.subtitle =  element_text(hjust = 0.5)) +
      scale_y_continuous(limits = c(0, 900))

p2 = ggplot(df %>% filter(Item_Identifier == "FDQ19"),
 ↪  aes(Outlet_Identifier, Item_Outlet_Ranking)) +
      geom_point(color = "#1f77b4") +
      labs(subtitle="FDQ19",
           x = "Establecimiento", y = "Posicion") +
      theme_minimal() +
      theme(plot.title = element_text(hjust = 0.5),
            plot.subtitle = element_text(hjust = 0.5) ) +
      scale_y_continuous(limits = c(0, 1250))


grid.arrange(p1, p2, ncol = 1)
```



Hay establecimientos como el OUT019, OUT045 o OUT017 que deberían potenciar más la venta del producto NCE42, mientras que el OUT013 es el que más destaca en los problemas para vender el producto FDQ19.

Como conclusión, hemos visto que las ventas difieren en gran medida según el establecimiento y el producto. En cuanto a los establecimientos, el factor más determinante es el tipo, siendo el Supermarket Type3 el que más ventas obtiene. En relación con los productos, destacamos la relación prácticamente lineal entre el precio del producto y las ventas.

Además, consideramos la decisión de potenciar las ventas de los productos que mayor éxito tienen en el establecimiento OUT027, que es con diferencia el que más renombre tiene entre los estudiados.