EDA Dataset Vegetales

Estephanie Gomez Ramirez

2025-04-13

```
knitr::opts_chunk$set(echo = TRUE)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(readr)
library(ggplot2)
library(summarytools)
```

```
## Warning: package 'summarytools' was built under R version 4.4.3
```

```
##
## Attaching package: 'summarytools'
##
## The following object is masked from 'package:tibble':
##
##     view
```

```
library(nortest)
library(DT)
```

```
## Warning: package 'DT' was built under R version 4.4.3
```

Parte 1: Lectura de datos

En esta etapa el objetivo es cargar los datos tal cual y leer a nivel alto que tipos de datos tienen, columnas, filas, dimensiones del data set, asegurarse que podemos leeer correctamente la funte de datos, en este caso un csv de kaggle

```
df <- read_csv("vegetables Dataset.csv")
```

```
## Rows: 150 Columns: 15
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (13): Name, Scientific Name, Category, Color, Season, Origin, Nutritiona...
## dbl  (2): Vegetable ID, Shelf Life (days)
```

```
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
glimpse (df) # N[umero de filas, columnas, tipos de datos y primeros valores
```

```
## Rows: 150
## Columns: 15
## $ `Vegetable ID`                 <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, ~
## $ Name                           <chr> "Carrot", "Spinach", "Tomato", "Broccol~
## $ `Scientific Name`              <chr> "Daucus carota", "Spinacia oleracea", "~
## $ Category                       <chr> "Root", "Leafy", "Fruit", "Flower", "Tu~
## $ Color                          <chr> "Orange", "Green", "Red", "Green", "Bro~
## $ Season                         <chr> "Winter", "Spring/Fall", "Summer", "Fal~
## $ Origin                         <chr> "Middle East", "Central Asia", "South A~
## $ `Nutritional Value (per 100g)` <chr> "41 kcal, 0.9g protein, 2.8g fiber", "2~
## $ `Price (per kg)`               <chr> "$1.50", "$2.00", "$3.00", "$2.50", "$1~
## $ Availability                   <chr> "Year-round", "Year-round", "Seasonal",~
## $ `Shelf Life (days)`            <dbl> 30, 7, 7, 10, 60, 10, 7, 7, 5, 10, 15, ~
## $ `Storage Requirements`         <chr> "Refrigeration", "Refrigeration", "Cool~
## $ `Growing Conditions`           <chr> "Well-drained soil, full sunlight", "Mo~
## $ `Health Benefits`              <chr> "Improves vision, rich in Vitamin A", "~
## $ `Common Varieties`             <chr> "Nantes, Imperator, Chantenay", "Savoy,~
```

```
head(df)     # Primeras filas
```

```
## # A tibble: 6 x 15
##   `Vegetable ID` Name        `Scientific Name`    Category Color   Season Origin
##            <dbl> <chr>       <chr>                <chr>    <chr>   <chr>  <chr>
## 1              1 Carrot      Daucus carota        Root     Orange  Winter Middl~
## 2              2 Spinach     Spinacia oleracea    Leafy    Green   Sprin~ Centr~
## 3              3 Tomato      Solanum lycopersicum Fruit    Red     Summer South~
## 4              4 Broccoli    Brassica oleracea    Flower   Green   Fall/~ Medit~
## 5              5 Potato      Solanum tuberosum    Tuber    Brown   Fall   South~
## 6              6 Bell Pepper Capsicum annuum      Fruit    Red, G~ Summer Centr~
## # i 8 more variables: `Nutritional Value (per 100g)` <chr>,
## #   `Price (per kg)` <chr>, Availability <chr>, `Shelf Life (days)` <dbl>,
## #   `Storage Requirements` <chr>, `Growing Conditions` <chr>,
## #   `Health Benefits` <chr>, `Common Varieties` <chr>
```

```
dim(df)      # Dimensiones del dataset
```

```
## [1] 150  15
```

```
colnames(df) # Nombres de columnas
```

```
##  [1] "Vegetable ID"                 "Name"
##  [3] "Scientific Name"              "Category"
##  [5] "Color"                        "Season"
##  [7] "Origin"                       "Nutritional Value (per 100g)"
##  [9] "Price (per kg)"               "Availability"
## [11] "Shelf Life (days)"            "Storage Requirements"
## [13] "Growing Conditions"           "Health Benefits"
## [15] "Common Varieties"
```

```
summary(df)  # Resumen estadístico básico
```

```
##   Vegetable ID        Name           Scientific Name      Category
```

```
##  Min.   :  1.00   Length:150         Length:150         Length:150
##  1st Qu.: 37.25   Class :character   Class :character   Class :character
##  Median : 73.50   Mode  :character   Mode  :character   Mode  :character
##  Mean   : 74.17
##  3rd Qu.:111.75
##  Max.   :150.00
##     Color              Season            Origin
##  Length:150         Length:150         Length:150
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##  Nutritional Value (per 100g) Price (per kg)     Availability
##  Length:150                   Length:150         Length:150
##  Class :character             Class :character   Class :character
##  Mode  :character             Mode  :character   Mode  :character
##
##
##
##  Shelf Life (days) Storage Requirements Growing Conditions Health Benefits
##  Min.   :  5.00    Length:150           Length:150         Length:150
##  1st Qu.:  7.00    Class :character     Class :character   Class :character
##  Median :  7.00    Mode  :character     Mode  :character   Mode  :character
##  Mean   : 13.01
##  3rd Qu.: 14.00
##  Max.   :120.00
##  Common Varieties
##  Length:150
##  Class :character
##  Mode  :character
##
##
##
```

Parte 2: Calidad de datos

Queremos conocer valores faltantes y en cuales columnas para decidir si requiere eliminación de los mismos o rellenarlos con la media, mediana, moda, según corresponda, en este dataset en específico no hay datos faltantes.

```
colSums(is.na(df))
```

```
##                Vegetable ID                        Name
##                           0                           0
##             Scientific Name                    Category
##                           0                           0
##                       Color                      Season
##                           0                           0
##                      Origin Nutritional Value (per 100g)
##                           0                           0
##               Price (per kg)                Availability
##                           0                           0
##           Shelf Life (days)         Storage Requirements
##                           0                           0
```

```
##           Growing Conditions              Health Benefits
##                            0                            0
##             Common Varieties
##                            0
```

Parte 3: Análisis univariado.

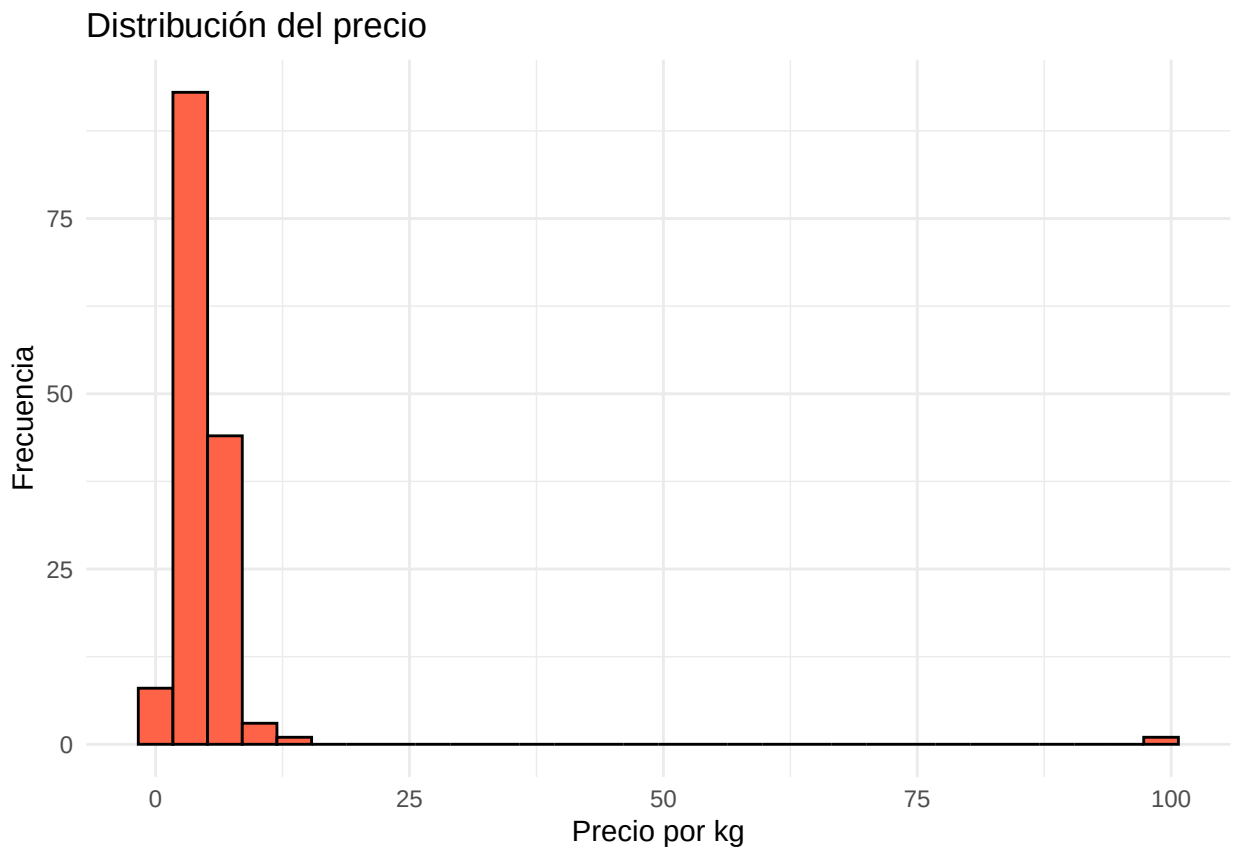
En esta sección se van a mostrar los histogramas de las variables númericas y son dos el precio y la vida útil.

Al analizar los precios por categoría de vegetal, se observa que la mayoría de categorías mantienen un precio relativamente bajo y homogéneo. Sin embargo, destaca el caso del *wasabi*, el cual se presenta como un **outlier significativo con un precio por kilogramo superior a 100 unidades**, lo que lo posiciona como el vegetal más caro del conjunto.

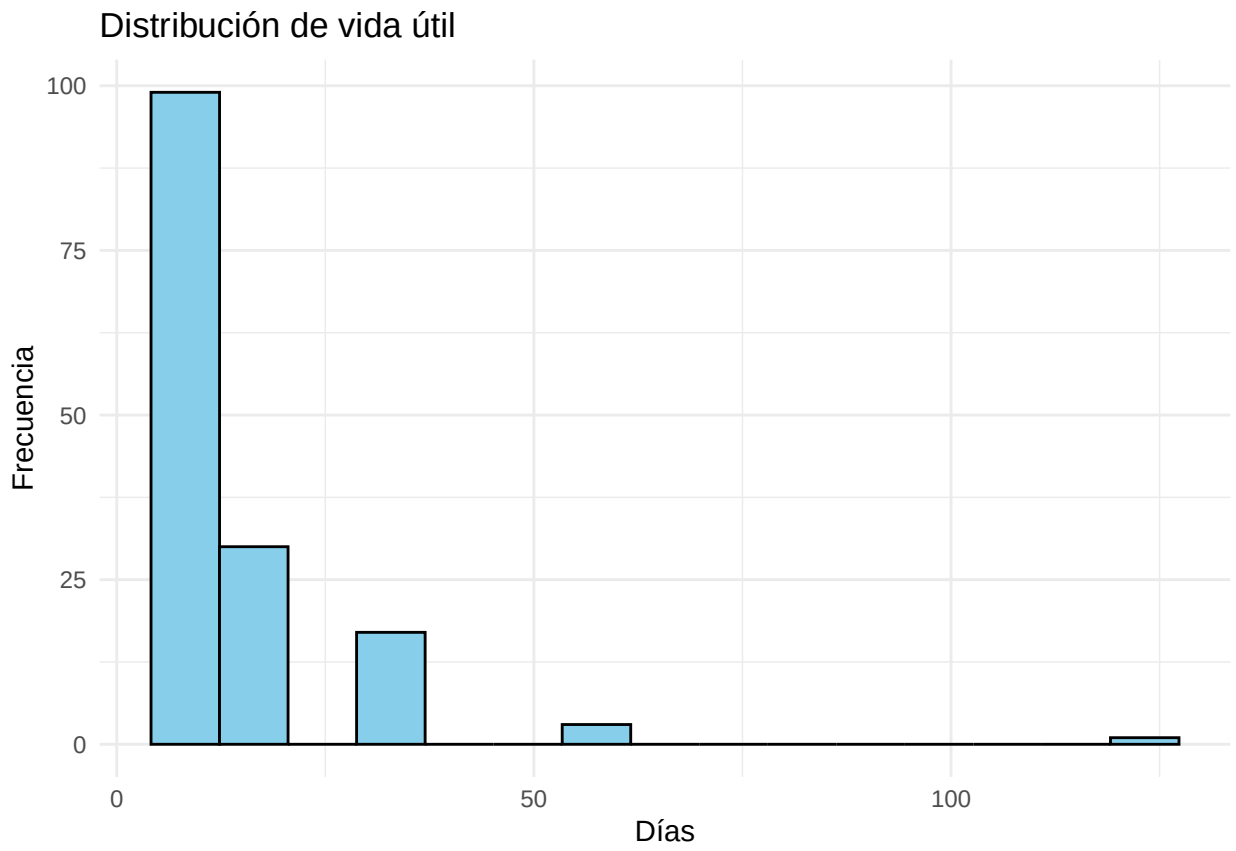
En cuanto a la vida útil, se detecta que la mayoría de categorías tienen productos con duración promedio entre 5 y 30 días. No obstante, dentro de la categoría **"Bulb"** aparece un vegetal con una vida útil atípicamente alta de **120 días**, convirtiéndose también en un *outlier*. Estos datos extremos aportan información relevante para la gestión logística y de almacenamiento de vegetales según su tipo.

Este análisis permite identificar qué categorías pueden requerir condiciones especiales de conservación o precio, siendo útil tanto en la cadena de suministro como en decisiones de consumo o venta.

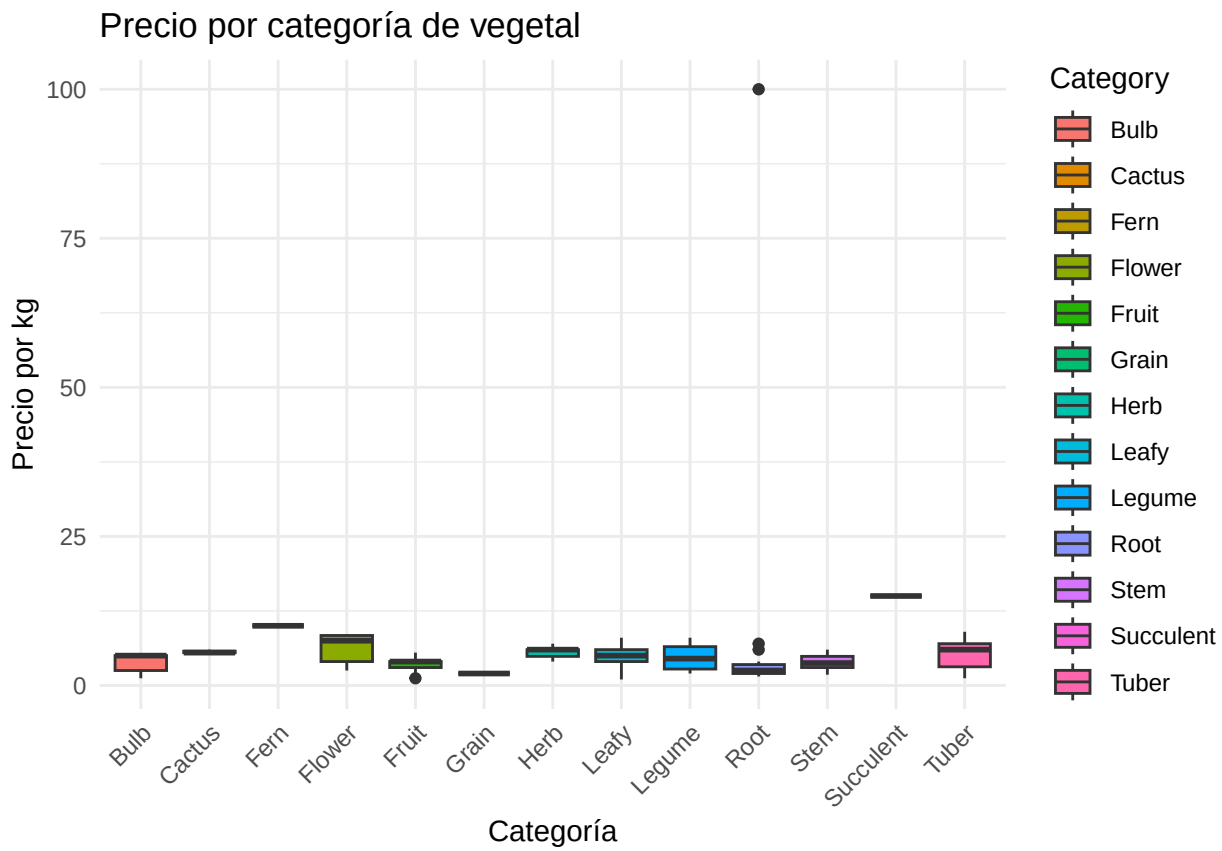
```r
df$`Price (per kg)` <- parse_number(as.character(df$`Price (per kg)`))


ggplot(df, aes(x = `Price (per kg)`)) +
  geom_histogram(bins = 30, fill = "tomato", color = "black") +
  labs(title = "Distribución del precio", x = "Precio por kg", y = "Frecuencia") +
  theme_minimal()
```



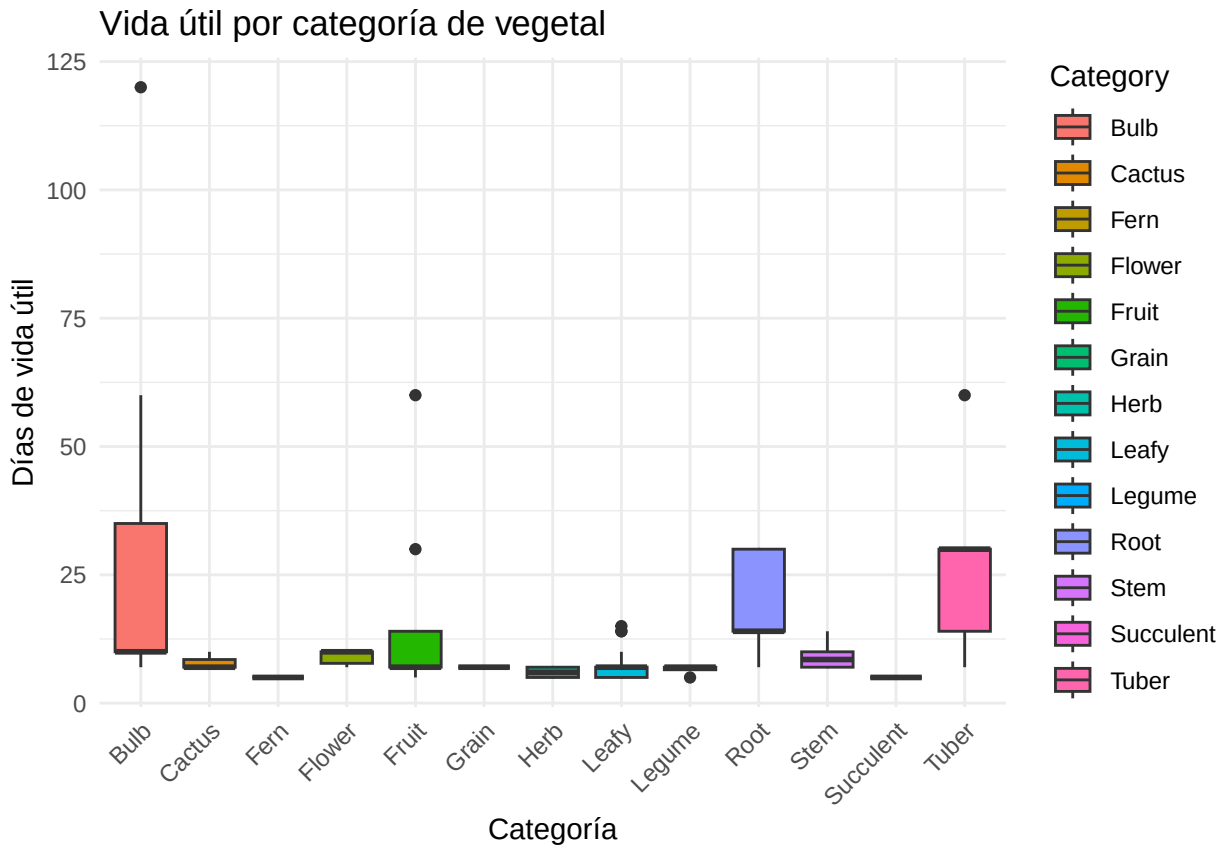

```
ggplot(df, aes(x = `Shelf Life (days)`)) +
  geom_histogram(bins = 15, fill = "skyblue", color = "black") +
  labs(title = "Distribución de vida útil", x = "Días", y = "Frecuencia") +
  theme_minimal()
```




```
ggplot(df, aes(x = Category, y = `Price (per kg)`, fill = Category)) +
  geom_boxplot() +
  labs(
    title = "Precio por categoría de vegetal",
    x = "Categoría",
    y = "Precio por kg"
  ) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```




```r
ggplot(df, aes(x = Category, y = `Shelf Life (days)`, fill = Category)) +
  geom_boxplot() +
  labs(
    title = "Vida útil por categoría de vegetal",
    x = "Categoría",
    y = "Días de vida útil"
  ) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



Parte 4: Análisis de Normalidad

La normalidad se puede analizar con el test de Shapiro Wilk o Anderson Darling, lo que hace la diferencia es la cantidad de datos, es una muestra relativamente pequeña de 150 filas, por lo tanto, es mejor utilizar Anderson Darling por que es más sensible a los extremos.

Se aplicó la prueba de normalidad de Anderson-Darling a las variables `Price (per kg)` y `Shelf Life (days)`. En ambos casos, el valor del estadístico A fue elevado (30.139 y 20.231 respectivamente) y el p-valor resultó ser menor a 2.2e-16.

Esto indica que **ambas variables presentan una desviación significativa respecto a una distribución normal**. Por lo tanto, **no se puede asumir normalidad** en estas variables, lo cual debe tenerse en cuenta al momento de aplicar técnicas estadísticas que requieren esta condición (por ejemplo, pruebas paramétricas o modelos de regresión lineal sin transformaciones previas).

```
ad.test(df$`Price (per kg)`)
```

```
##
##  Anderson-Darling normality test
##
## data:  df$`Price (per kg)`
## A = 30.139, p-value < 2.2e-16
```

```
ad.test(df$`Shelf Life (days)`)
```

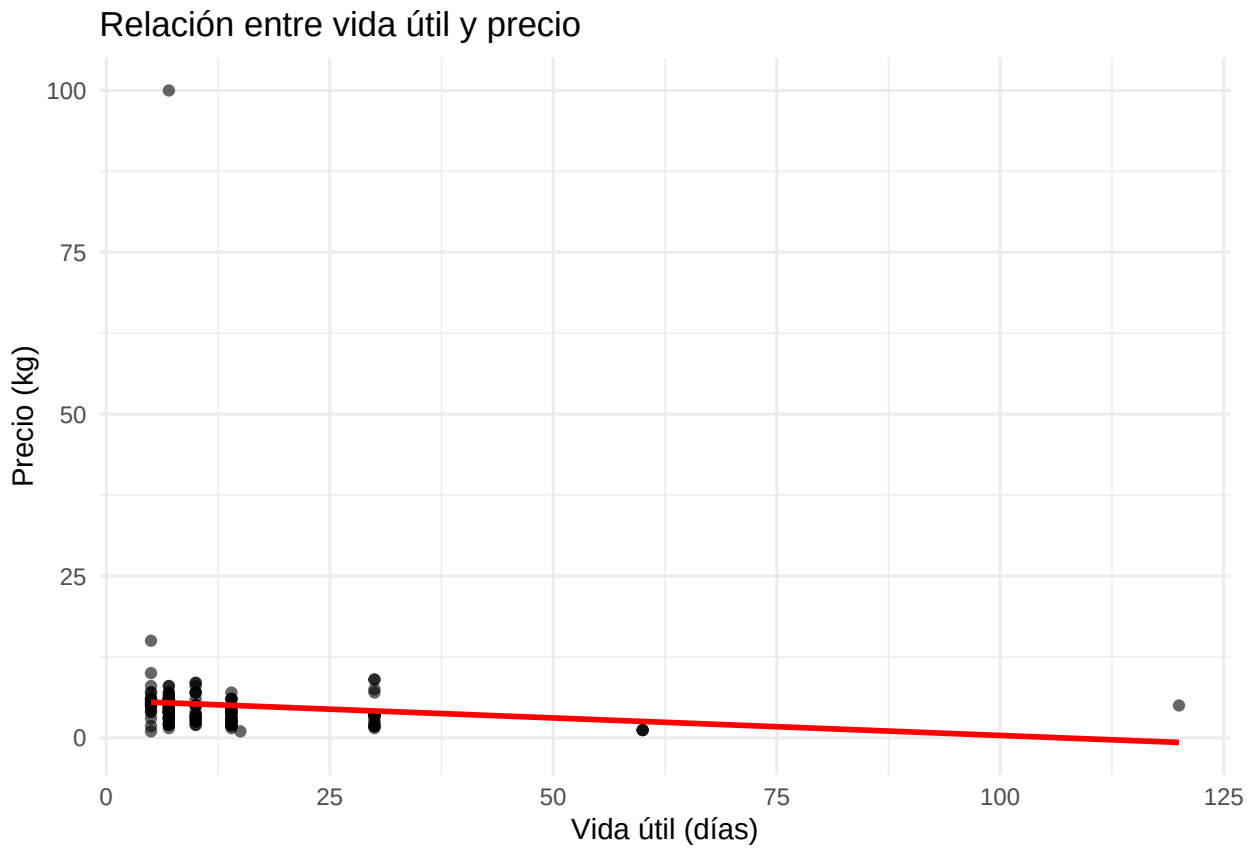
```
##
##  Anderson-Darling normality test
##
## data:  df$`Shelf Life (days)`
## A = 20.231, p-value < 2.2e-16
```

```r
cor(df$`Price (per kg)`, df$`Shelf Life (days)`, use = "complete.obs")
```

```
## [1] -0.0893221
```

```r
ggplot(df, aes(x = `Shelf Life (days)`, y = `Price (per kg)`)) +
  geom_point(alpha = 0.6) +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(title = "Relación entre vida útil y precio", x = "Vida útil (días)", y = "Precio (kg)") +
  theme_minimal()
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

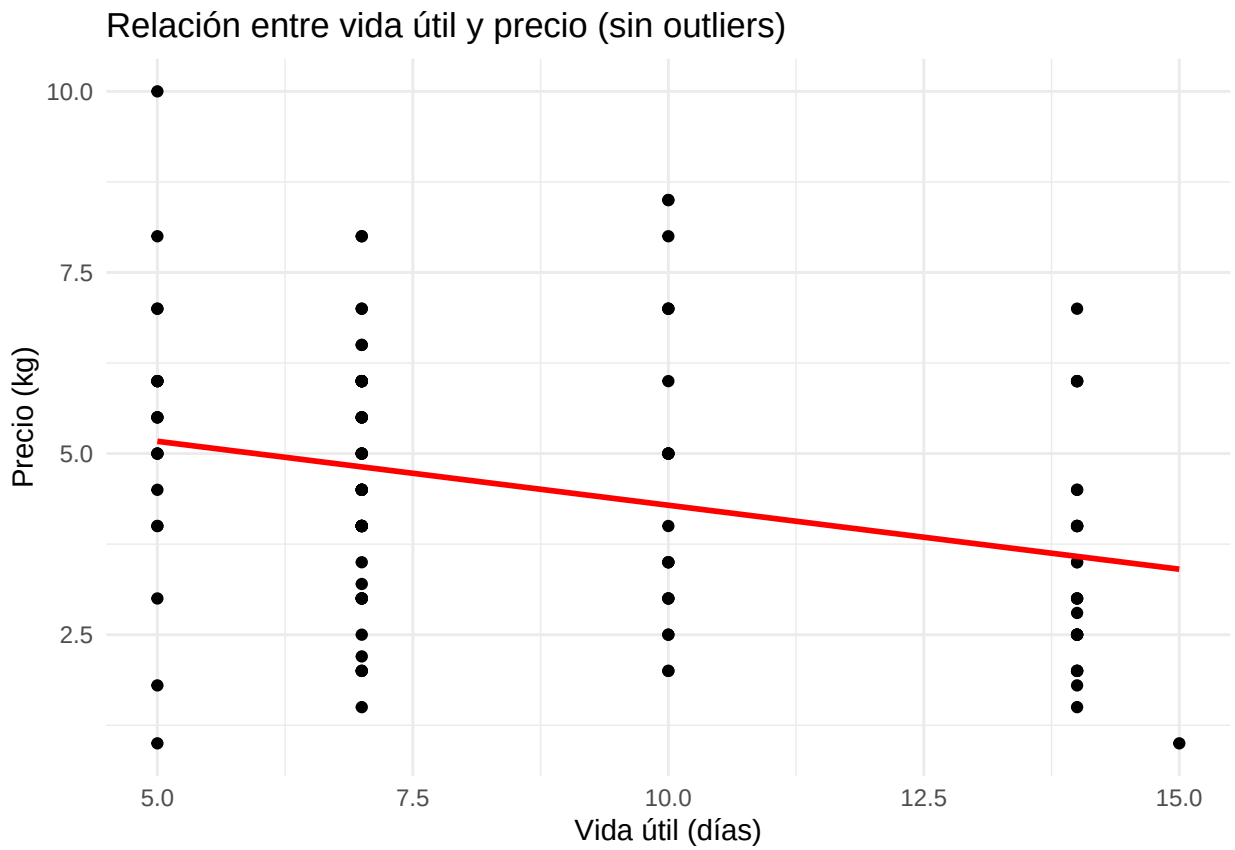


Nota: Si Este bloque elimina los outliers con el propósito de comprender si mejora la relación lineal

```r
remove_outliers <- function(x) {
  q1 <- quantile(x, 0.25, na.rm = TRUE)
  q3 <- quantile(x, 0.75, na.rm = TRUE)
  iqr <- q3 - q1
  lower <- q1 - 1.5 * iqr
  upper <- q3 + 1.5 * iqr
  return(x >= lower & x <= upper)
}

# Filtrar dataframe sin outliers en ambas variables
df_no_outliers <- df[
  remove_outliers(df$`Price (per kg)`) &
  remove_outliers(df$`Shelf Life (days)`),
]
```

```r
ggplot(df_no_outliers, aes(x = `Shelf Life (days)`, y = `Price (per kg)`)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(
    title = "Relación entre vida útil y precio (sin outliers)",
    x = "Vida útil (días)",
    y = "Precio (kg)"
  ) +
  theme_minimal()
```

```
## `geom_smooth()` using formula = 'y ~ x'
```




```r
modelo <- lm(`Price (per kg)` ~ `Shelf Life (days)`, data = df_no_outliers)
r2 <- summary(modelo)$r.squared
paste("El valor de R² es:", round(r2, 4))
```

```
## [1] "El valor de R² es: 0.0993"
```

Conclusiones:

El análisis exploratorio permitió obtener una visión clara de la estructura y comportamiento de los datos del conjunto de vegetales. Se identificaron diferencias marcadas en el precio y la vida útil entre distintas categorías, así como la presencia de valores atípicos como el *wasabi* en precio o un vegetal tipo *Bulb* con una vida útil de 123 días. Mediante pruebas de normalidad, se comprobó que las variables `Price (per kg)` y `Shelf Life (days)` no siguen una distribución normal, lo cual es relevante para elegir las herramientas estadísticas adecuadas. Aunque se exploró una posible relación lineal entre precio y vida útil, el bajo valor

de R^2 indica que esta relación es débil. En conjunto, el análisis demuestra el valor de la visualización y el procesamiento de datos para extraer conclusiones útiles desde una perspectiva exploratoria, utilizando RStudio como herramienta central.

Al tratarse de un dataset diverso, que incluye tanto alimentos comunes como vegetales más exóticos, es natural encontrar una mayor presencia de valores atípicos. Clasificaciones amplias como "Root" agrupan productos muy distintos en precio y popularidad: por ejemplo, el *wasabi* con un precio elevado, y la *zanahoria* con un costo mucho más accesible (alrededor de $1.50). Este contraste refleja cómo una categoría genérica puede enmascarar variaciones importantes dentro del dataset.

Se comparó el coeficiente de determinación R^2 al ajustar un modelo de regresión lineal entre la vida útil y el precio, tanto con outliers como sin ellos. Con los datos originales, el valor de R^2 fue negativo (-0.08), indicando que el modelo ajustado no explica la variabilidad del precio, y de hecho es peor que asumir una media constante.

Al remover los outliers, el valor de R^2 aumentó ligeramente a 0.09. Aunque este valor sigue siendo bajo y no indica una relación lineal fuerte, **sí muestra una mejora en la capacidad explicativa del modelo** al eliminar valores extremos. Esto sugiere que los outliers pueden distorsionar las relaciones estadísticas, y que la vida útil por sí sola no es un buen predictor del precio en este conjunto de datos.