Análisis Exploratorio de Datos

Erick Jair Becerra Acosta

2025-03-12

Contents

1. Análisis Descriptivo del Dataset Iris

1.1 Carga y Exploración de los Datos

Instalación y Activación de los Paquetes Necesarios

```
#install.packages("tidyverse")
#install.packages("ggplot2")
#install.packages("janitor")
#install.packages("dplyr")

library(tidyverse)
```

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

## Warning: package 'ggplot2' was built under R version 4.4.3

## Warning: package 'tidyr' was built under R version 4.4.3

## Warning: package 'readr' was built under R version 4.4.3

## Warning: package 'forcats' was built under R version 4.4.3

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

```
## -- 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     1.0.4
## -- 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(ggplot2)
library(janitor)
```

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

```
##
## Adjuntando el paquete: 'janitor'
##
## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test
```

```
library(dplyr)
```

Importar el Dataset Iris en R

```
data(iris)
```

Visualizar la estructura de los datos

Para la estructura del dataset:

```
str(iris)
```

```
## 'data.frame':    150 obs. of  5 variables:
##  $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
##  $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
##  $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
##  $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
##  $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...
```

Para observar las primeras filas:

```
head(iris)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
```

```
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
## 4          4.6         3.1          1.5         0.2  setosa
## 5          5.0         3.6          1.4         0.2  setosa
## 6          5.4         3.9          1.7         0.4  setosa
```

Para obtener un resumen estadístico basico:

```
summary(iris)
```

```
##   Sepal.Length    Sepal.Width     Petal.Length    Petal.Width
##  Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100
##  1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300
##  Median :5.800   Median :3.000   Median :4.350   Median :1.300
##  Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199
##  3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800
##  Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500
##        Species
##  setosa    :50
##  versicolor:50
##  virginica :50
##
##
##
```

Verificar dimensiones, tipos de datos y valores faltantes

Dimensiones:

```
dim(iris)
```

```
## [1] 150   5
```

Tipos de Datos:

```
sapply(iris,class)
```

```
## Sepal.Length  Sepal.Width Petal.Length  Petal.Width      Species
##    "numeric"    "numeric"    "numeric"    "numeric"     "factor"
```

Valores Faltantes:

```
colSums(is.na(iris))
```

```
## Sepal.Length  Sepal.Width Petal.Length  Petal.Width      Species
##            0            0            0            0            0
```

1.2 Cálculo de Medidas de Tendencia Central y Dispersión

Calcule Media, Mediana, Moda, Varianza y Desviación Estándar por Especie.

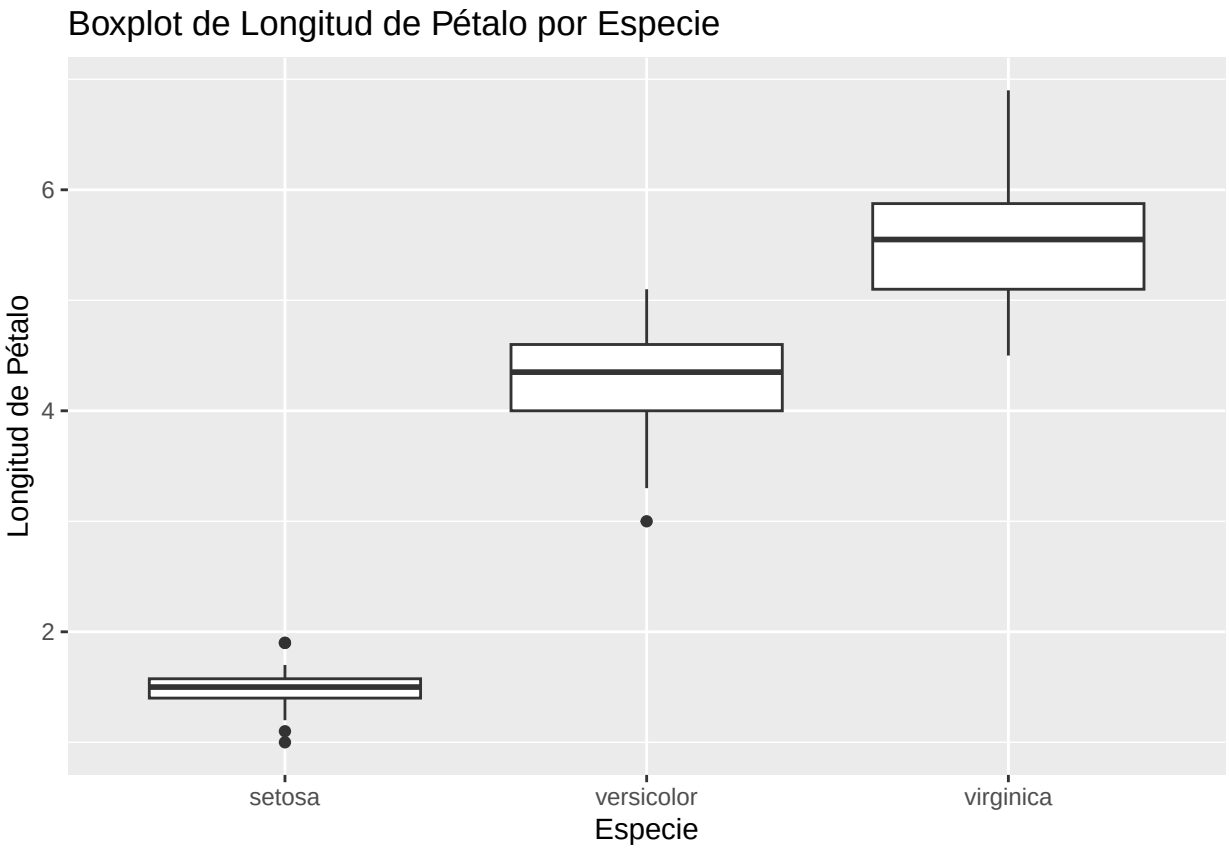
Para lograr esto, creamos una variable a la cuál le vamos a asignar todos los elementos calculados por especie.

```
iris_summary <- iris %>%
  group_by(Species) %>%
  summarise(
    Media = mean(Petal.Length),
    Mediana = median(Petal.Length),
    Moda = as.numeric(names(sort(table(Petal.Length), decreasing = TRUE)[1])),
    Varianza = var(Petal.Length),
    DesviacionEstandar = sd(Petal.Length)
  )
print(iris_summary)
```

```
## # A tibble: 3 x 6
##   Species    Media Mediana  Moda Varianza DesviacionEstandar
##   <fct>      <dbl>   <dbl> <dbl>    <dbl>              <dbl>
## 1 setosa      1.46    1.5    1.4   0.0302              0.174
## 2 versicolor  4.26    4.35   4.5   0.221               0.470
## 3 virginica   5.55    5.55   5.1   0.305               0.552
```

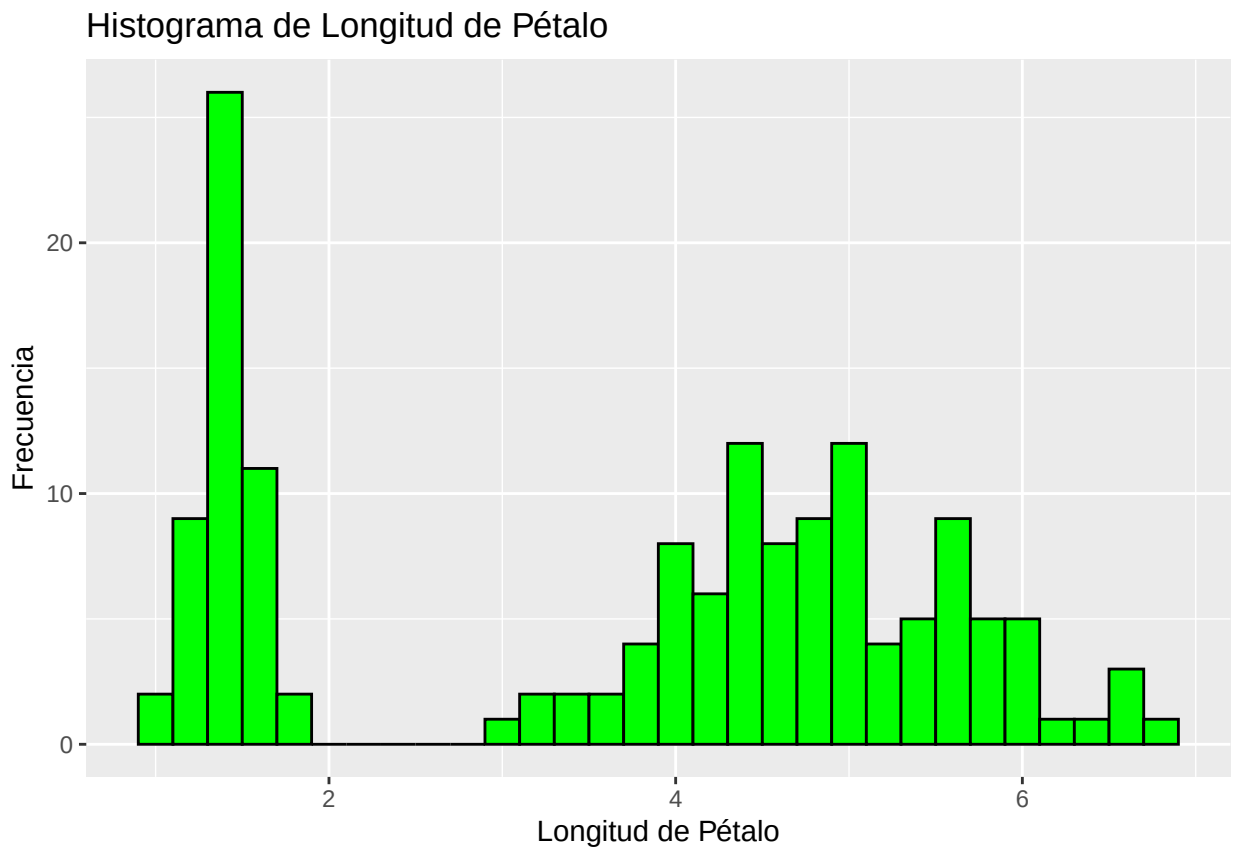
Genere una Visualización con Diagramas de Cajas.

```
ggplot(iris, aes(x = Species, y = Petal.Length)) +
  geom_boxplot() +
  labs(title = "Boxplot de Longitud de Pétalo por Especie", x = "Especie", y = "Longitud de Pétalo")
```



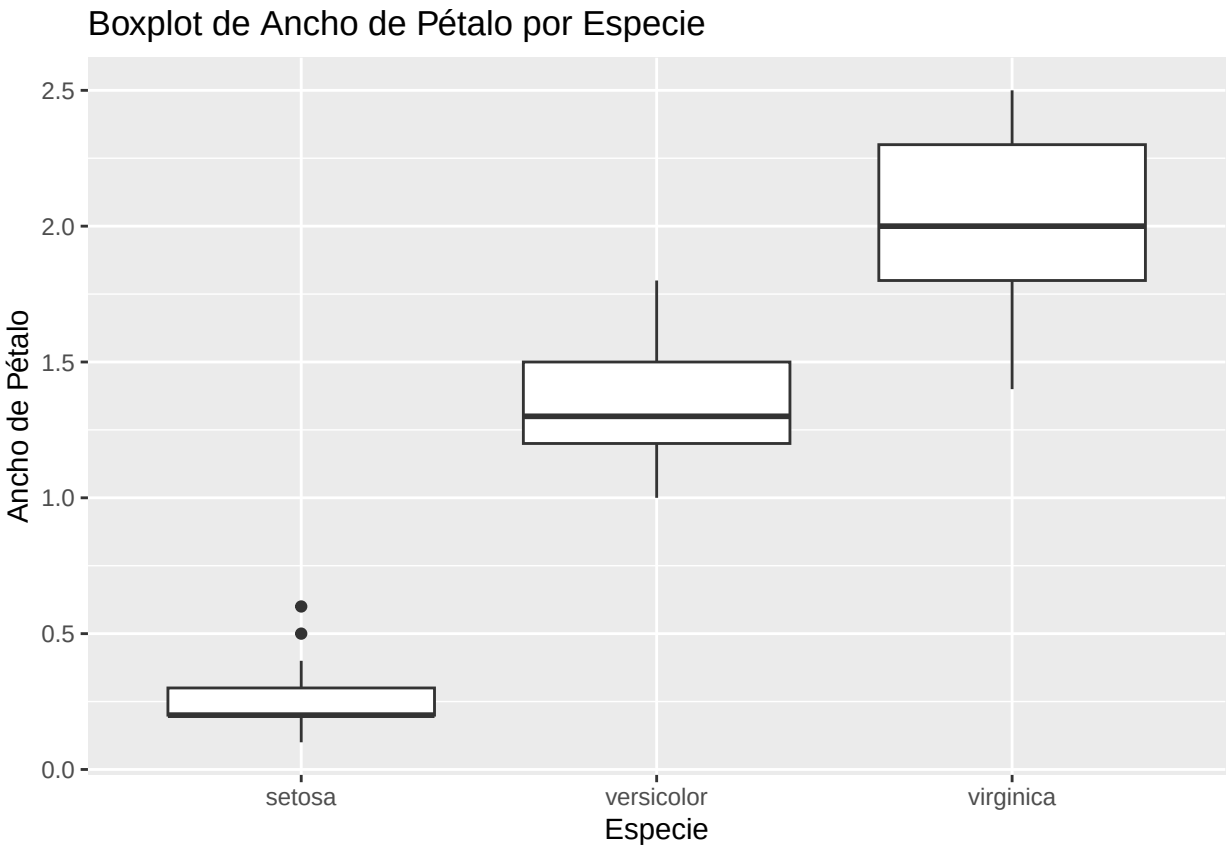
Cree un Histograma con la Variable Petal.Length.

```
ggplot(iris, aes(x = Petal.Length)) +
  geom_histogram(binwidth = 0.2, fill = "green", color = "black") +
  labs(title = "Histograma de Longitud de Pétalo", x = "Longitud de Pétalo", y = "Frecuencia")
```



Genere Boxplots de Petal.Width por Cada Tipo de Especie.

```
ggplot(iris, aes(x = Species, y = Petal.Width)) +
  geom_boxplot() +
  labs(title = "Boxplot de Ancho de Pétalo por Especie", x = "Especie", y = "Ancho de Pétalo")
```



Interprete los Resultados Obtenidos.

En base a lo observado en las pruebas anteriores, hay Varios puntos que podemos resltar:

- La especie Setosa tiene la desviación estpandar más pequeña, lo que significa que los valores de la longitud de sus pétalos están muy cercanos a la media.
- La especie Virgínica, es la que mayor desviación estándar posee, lo que quiere decir que de las tres especies, esta es la que tiene mayor variabilidad en la longitud de susu petalos.
- Las especies con mayor cantidad de Outliers son Setosa y Versicolor. ESto lo podemos comprobar visualmente con los boxplots.
- La longitud de los pétalos tiene un alza notable en los valores menores a 2, sin embargo muestra una distribución normal en los valores entre 3 y 7.

2. Tablas de Frecuencia y Visualización de Datos.

Se le pidió a un grupo de personas que marque la imagen de su bebida preferida y los resultados fueron los siguientes:

- Duff: 4
- Pepsi: 5
- Coca-Cola: 6
- Sprite: 5

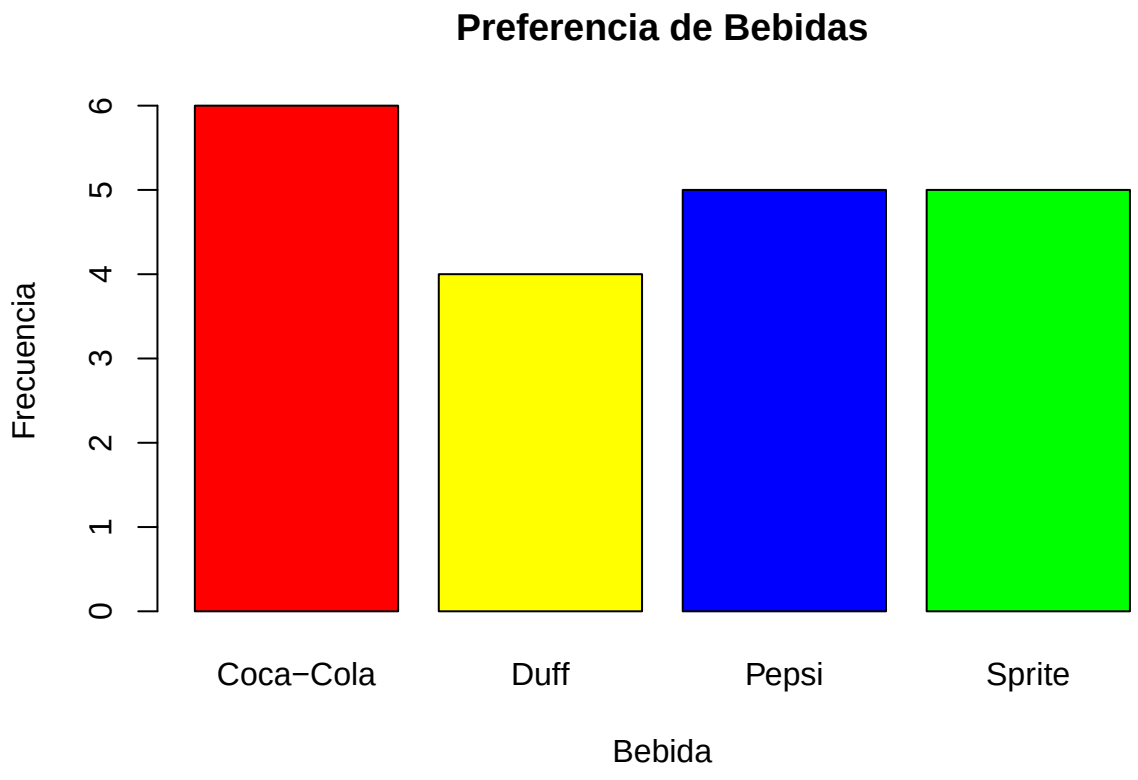
Tabla de Frecuencias

```
Bebidas <- c(rep("Duff", 4), rep("Pepsi", 5), rep("Coca-Cola", 6), rep("Sprite", 5))

tabla_frecuencias <- table(Bebidas)
print(tabla_frecuencias)
```

```
## Bebidas
## Coca-Cola      Duff     Pepsi    Sprite
##         6         4         5         5
```

Gráfico para Representar los Resultados

```
barplot(tabla_frecuencias,
        main = "Preferencia de Bebidas",
        xlab = "Bebida",
        ylab = "Frecuencia",
        col = c("red", "yellow", "blue", "green"),
        legend = FALSE)
```



Interpretación de los Resultados Obtenidos

Con esta información podemos concluir que:

- Coca-Cola es la bebida más preferida (parece que el estudio fue hecho en México, pues allá no hay persona o familia que no consuma Coca-Cola)
- Duff es la bebida menos preferida por las personas sobre quienes se realizó el estudio.
- Pepsi y Sprite son preferido por la misma cantidad de Personas.

3. Análisis del Dataset SWISS

Cargar el Dataset Swiss en R

```
data(swiss)
```

Verifique los Tipos de Variables Contenidas en la Base de Datos

```
str(swiss)
```

```
## 'data.frame':    47 obs. of  6 variables:
##  $ Fertility       : num  80.2 83.1 92.5 85.8 76.9 76.1 83.8 92.4 82.4 82.9 ...
##  $ Agriculture     : num  17 45.1 39.7 36.5 43.5 35.3 70.2 67.8 53.3 45.2 ...
##  $ Examination     : int  15 6 5 12 17 9 16 14 12 16 ...
##  $ Education       : int  12 9 5 7 15 7 7 8 7 13 ...
##  $ Catholic        : num  9.96 84.84 93.4 33.77 5.16 ...
##  $ Infant.Mortality: num  22.2 22.2 20.2 20.3 20.6 26.6 23.6 24.9 21 24.4 ...
```

Calcule los Principales Indicadores Estadísticos de las Variables Fertility e Infant.Mortality

```
swiss_summary <- swiss %>%
  summarise(
    Fertility_Mean = mean(Fertility),
    Fertility_Median = median(Fertility),
    Fertility_Var = var(Fertility),
    Fertility_SD = sd(Fertility),
    Fertility_Min = min(Fertility),
    Fertility_Max = max(Fertility),
    Infant_Mortality_Mean = mean(Infant.Mortality),
    Infant_Mortality_Median = median(Infant.Mortality),
    Infant_Mortality_Var = var(Infant.Mortality),
    Infant_Mortality_SD = sd(Infant.Mortality),
    Infant_Mortality_Min = min(Infant.Mortality),
    Infant_Mortality_Max = max(Infant.Mortality)
  )
print(swiss_summary)
```

```
##   Fertility_Mean Fertility_Median Fertility_Var Fertility_SD Fertility_Min
## 1       70.14255             70.4      156.0425      12.4917            35
##   Fertility_Max Infant_Mortality_Mean Infant_Mortality_Median
```

```
## 1          92.5              19.94255                       20
##   Infant_Mortality_Var Infant_Mortality_SD Infant_Mortality_Min
## 1             8.483802            2.912697                 10.8
##   Infant_Mortality_Max
## 1                 26.6
```

4. Notas de Estudiantes y Análisis de Aprobación

La siguiente información corresponde a las notas obtenidas en una prueba de competencias computacionales realizada a un grupo de estudiantes.

```
nf <- c(4.1, 2.7, 3.1, 3.2, 3.0, 3.2, 2.0, 2.4, 1.6, 3.2, 3.1, 2.6, 2.0, 2.4, 2.8, 3.3, 4.0, 3.4, 3.0, 3
```

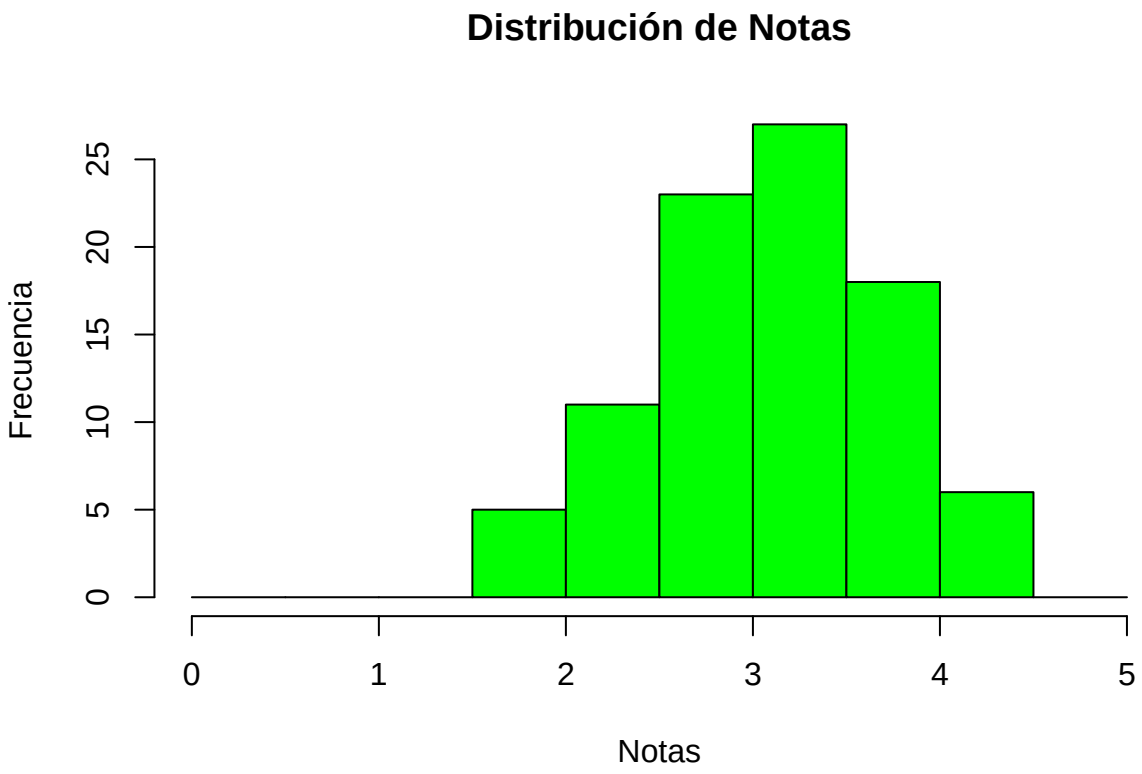
Tabla de Distribución de Frecuencias

```
intervalos <- seq(0, 5, by = 0.5)

tabla_frecuencias <- table(cut(nf, breaks = intervalos))

print(tabla_frecuencias)
```

```
##
## (0,0.5] (0.5,1] (1,1.5] (1.5,2] (2,2.5] (2.5,3] (3,3.5] (3.5,4] (4,4.5] (4.5,5]
##       0       0       0       5      11      23      27      18       6       0
```

Gráfico de Barras

```
hist(nf,
     breaks = intervalos,
     main = "Distribución de Notas",
     xlab = "Notas",
     ylab = "Frecuencia",
     col = "green")
```



Calcular Indicadores Estadísticos

```
media <- mean(nf)
mediana <- median(nf)
moda <- as.numeric(names(sort(table(nf), decreasing = TRUE)[1]))
varianza <- var(nf)
desviacion_estandar <- sd(nf)
minimo <- min(nf)
maximo <- max(nf)

cat("Media:", media, "\n")
```

```
## Media: 3.136667
```

```
cat("Mediana:", mediana, "\n")
```

```
## Mediana: 3.1
```

```
cat("Moda:", moda, "\n")
```

```
## Moda: 3
```

```r
cat("Varianza:", varianza, "\n")
```

```
## Varianza: 0.3529101
```

```r
cat("Desviación Estándar:", desviacion_estandar, "\n")
```

```
## Desviación Estándar: 0.5940624
```

```r
cat("Mínimo:", minimo, "\n")
```

```
## Mínimo: 1.6
```

```r
cat("Máximo:", maximo, "\n")
```

```
## Máximo: 4.3
```

Porcentaje de Estudiantes que Reprobaron la Evaluación

Para calcular el porcentaje de estudiantes que reprobaron, o sea que su nota es menor a 3, haremos los siguientes cálculos;

```r
reprobados <- sum(nf < 3.0)
total_estudiantes <- length(nf)
porcentaje_reprobados <- (reprobados / total_estudiantes) * 100

cat("Porcentaje de estudiantes que reprobaron:", round(porcentaje_reprobados, 2), "%\n")
```

```
## Porcentaje de estudiantes que reprobaron: 27.78 %
```

5. Distribución de Cargos en una Empresa por Género

Primero crearemos un dataframe que contenga los datos proporcionados

	Mujer	Hombre
Administrativo	32	21
Operativo	62	140
Vendedor	132	55

Figure 1: Distribución de Cargos de la Empresa

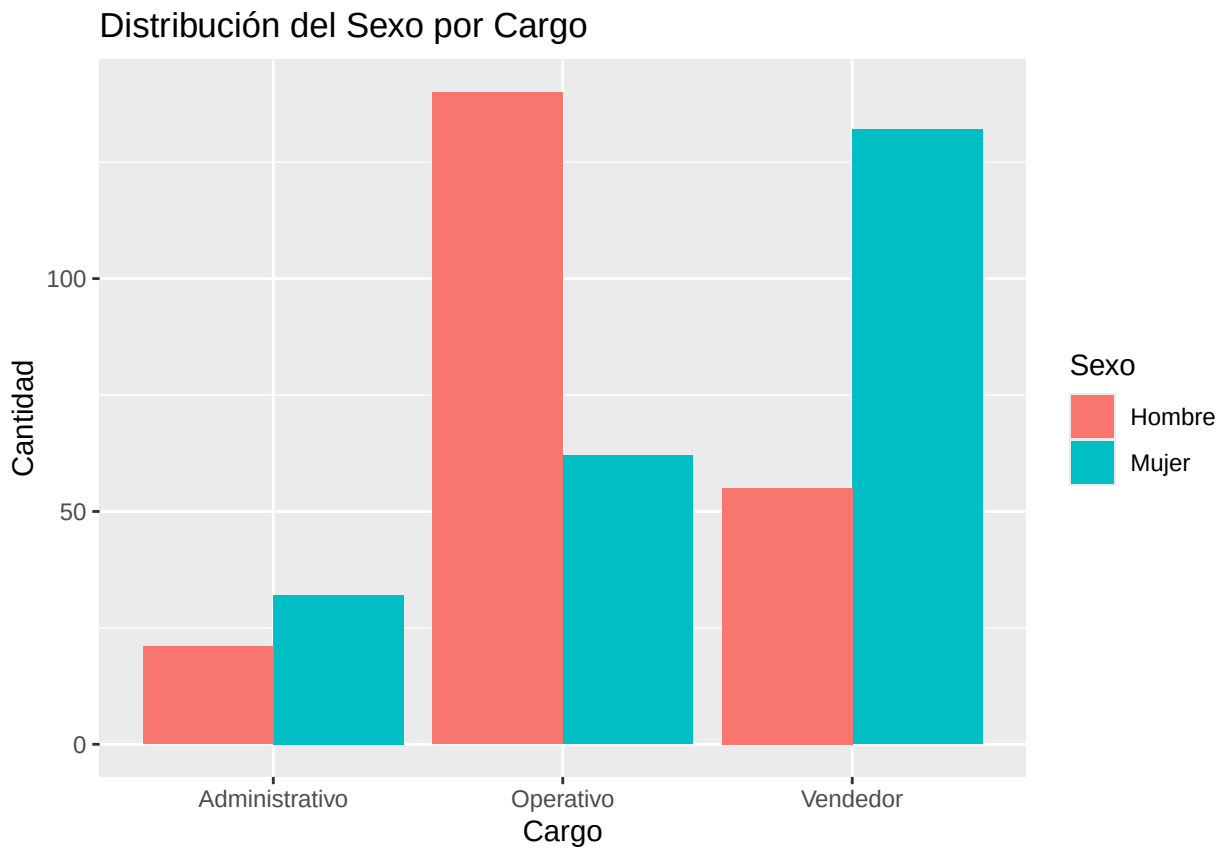
Dataframe de los Datos

```r
datos <- data.frame(
  Cargo = c("Administrativo", "Operativo", "Vendedor"),
  Mujer = c(32, 62, 132),
  Hombre = c(21, 140, 55)
)

print(datos)
```

```
##            Cargo Mujer Hombre
## 1 Administrativo    32     21
## 2      Operativo    62    140
## 3       Vendedor   132     55
```

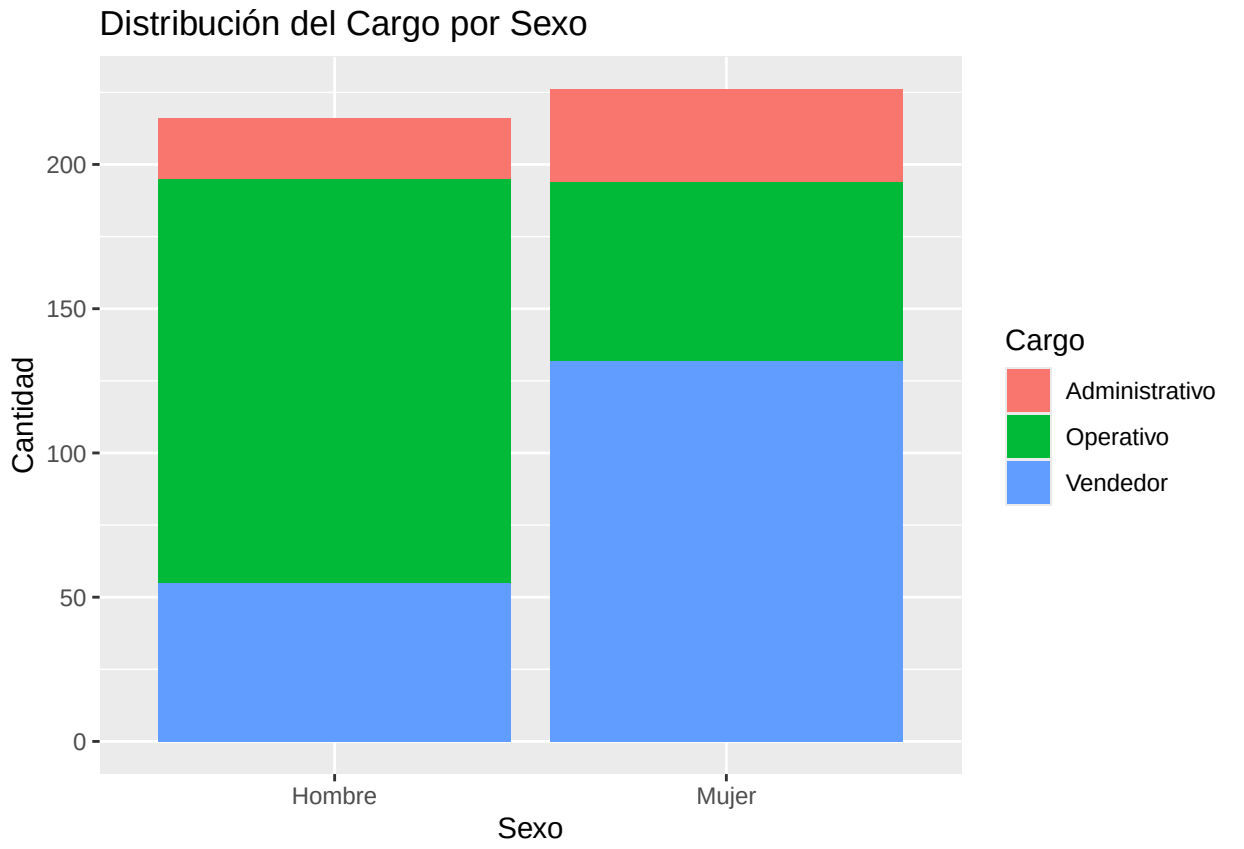
Distribución del Sexo por Cargo

```r
datos_long <- tidyr::pivot_longer(datos, cols = c(Mujer, Hombre), names_to = "Sexo", values_to = "Canti

ggplot(datos_long, aes(x = Cargo, y = Cantidad, fill = Sexo)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Distribución del Sexo por Cargo",
       x = "Cargo",
       y = "Cantidad",
       fill = "Sexo")
```



Distribución del Cargo por Sexo

```
ggplot(datos_long, aes(x = Sexo, y = Cantidad, fill = Cargo)) +
  geom_bar(stat = "identity", position = "stack") +
  labs(title = "Distribución del Cargo por Sexo",
       x = "Sexo",
       y = "Cantidad",
       fill = "Cargo")
```



Interpretación de los Resultados

Con los datos visualizados anteriormente podemos llegar a las siguientes conclusiones:

- En el cargo de Administrativo hay más mujeres que hombres
- En el cargo Operativo hay una diferencia bastante significativa en cuanto al número de hombres comparado con el de mujeres.
- El cargo de Vendedor son las mujeres las que predominan en número.
- En total, hay más mujeres que hombres en la empresa.

6. Generacion de Gráficos con Herramientas de IA

El proceso que seguiremos para poder trabajar con Power BI será el siguiente:

Exportar Datos a un Archivo CSV

Exportaremos el dataset Iris y los datos de los empleados por cargo y género a un archivo CSV

```
write.csv(iris, "iris_data.csv", row.names = FALSE)
write.csv(datos, "empleados_data.csv", row.names = FALSE)
```

Importar Datos y Generar Gráficos Automáticos

Después de crear nuestra cuenta gratuita en Power BI, creamos dos nuevos informes, en los cuáles cargamos los archivos CSV creados anteriormente y con eso generamos los siguientes diagramas:

Histograma de Longitud del Pétalo

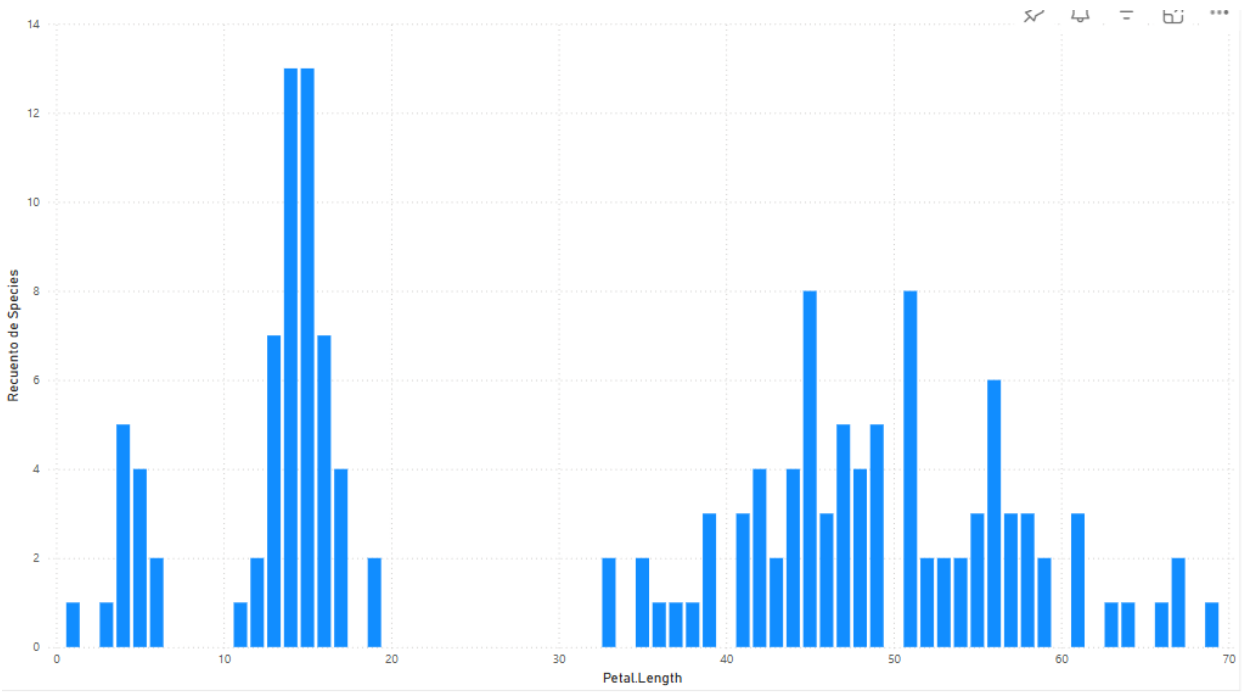


Figure 2: Longitud del Pétalo generado con Power BI

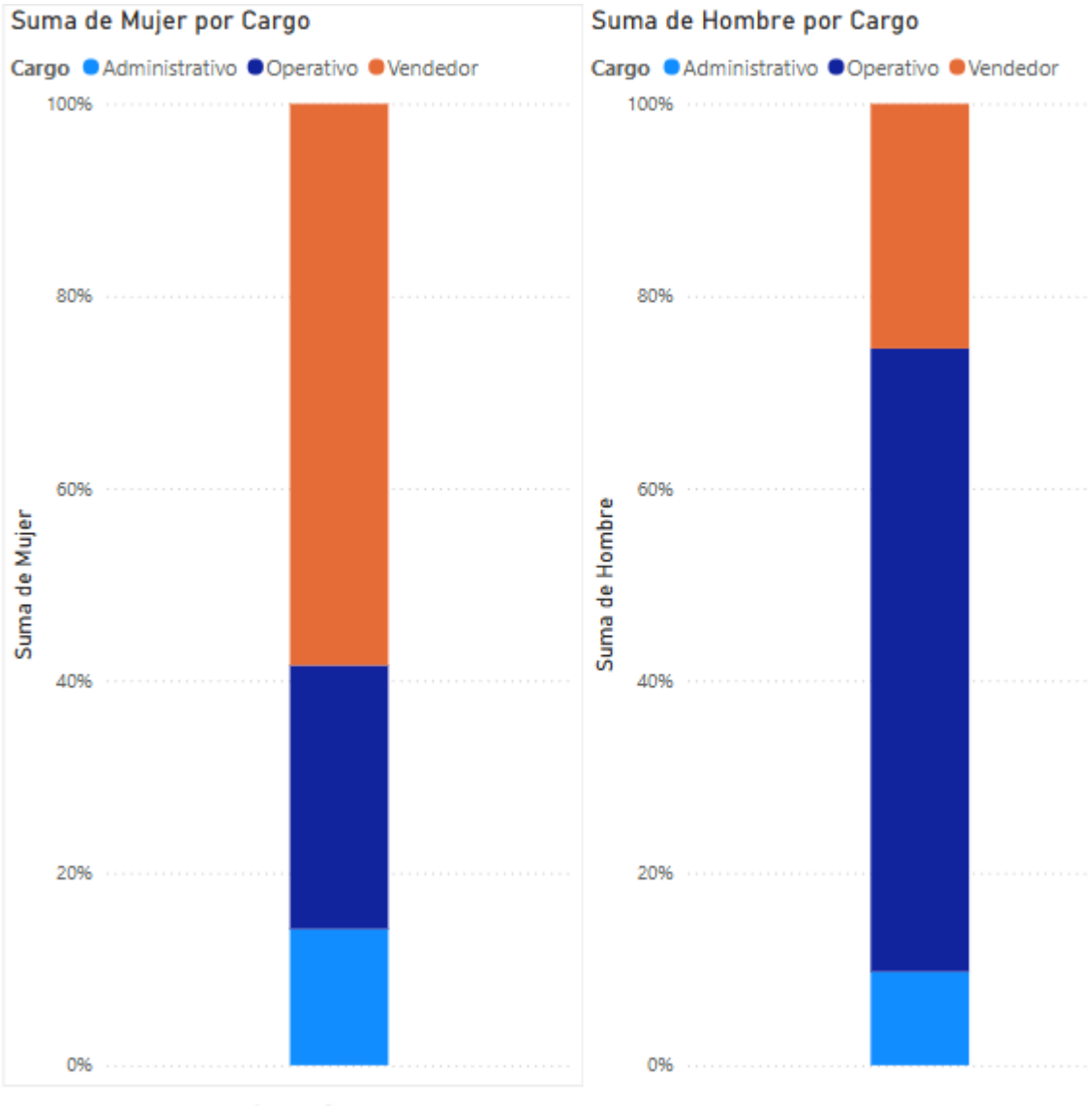
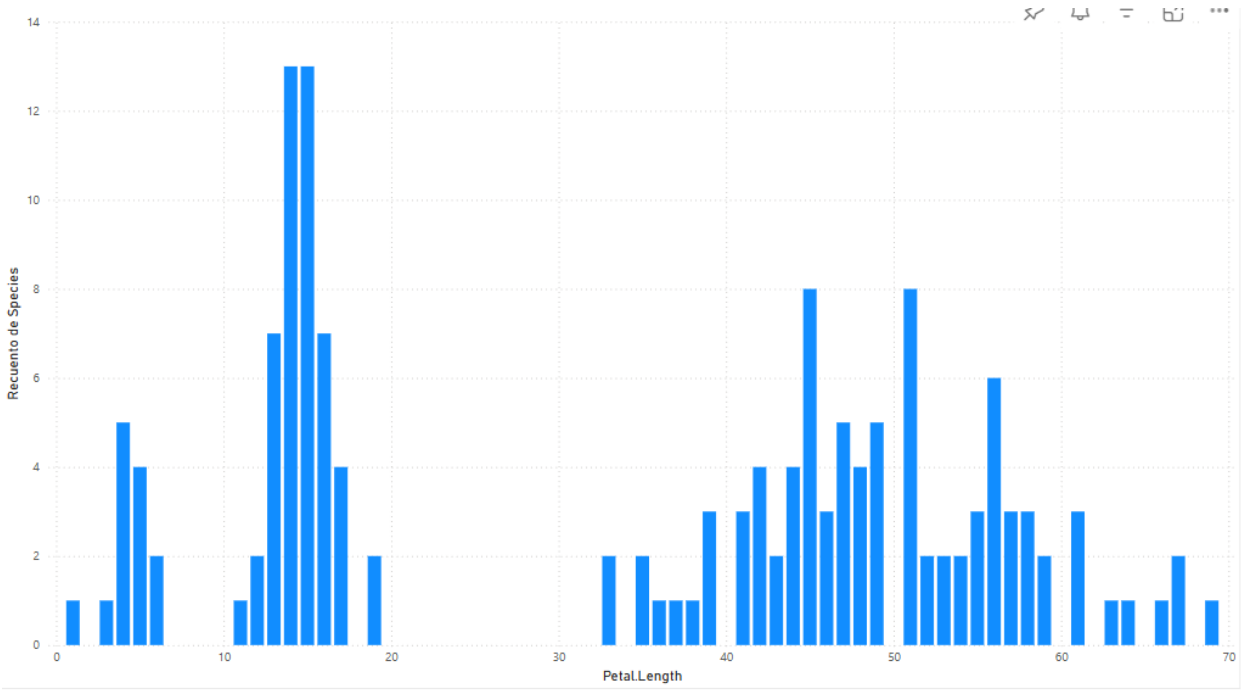


Figure 3: Cargos por Género con Power BI

Gráfico de Columnas Apiladas de Cargos por Género

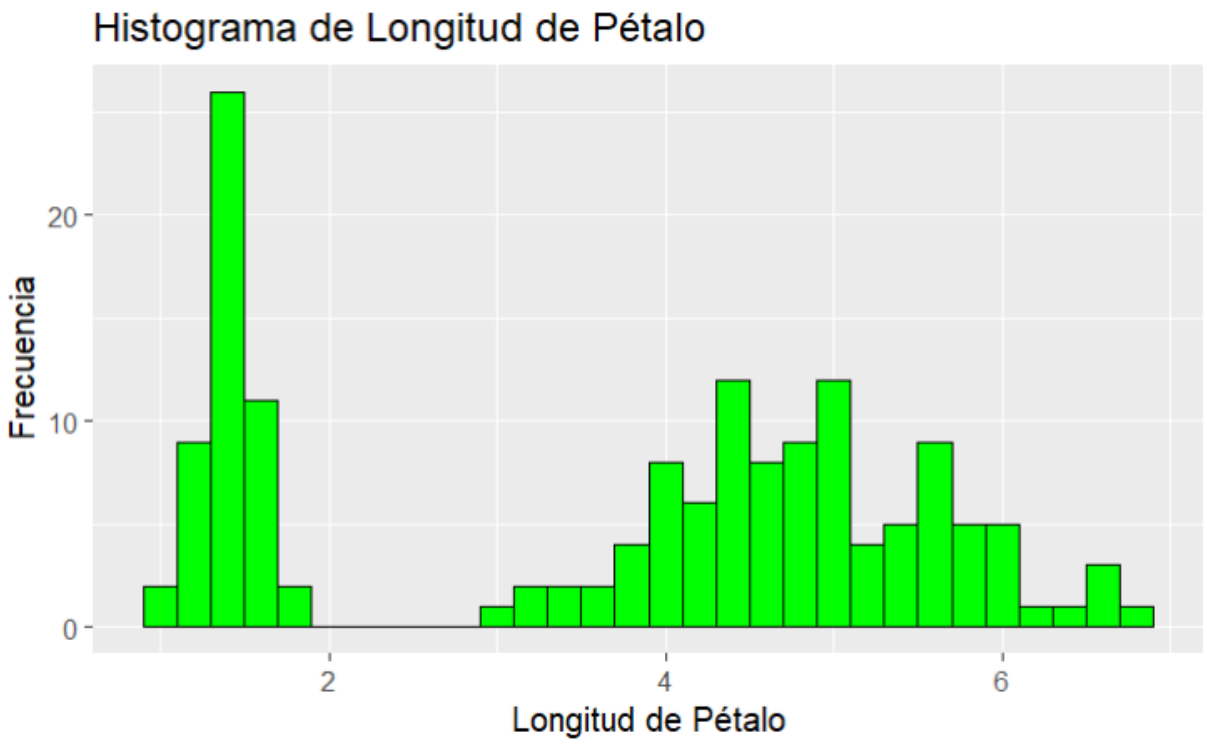
Comparación de Gráficos

Histograma de Longitud del Pétalo



{withd

= 50%}

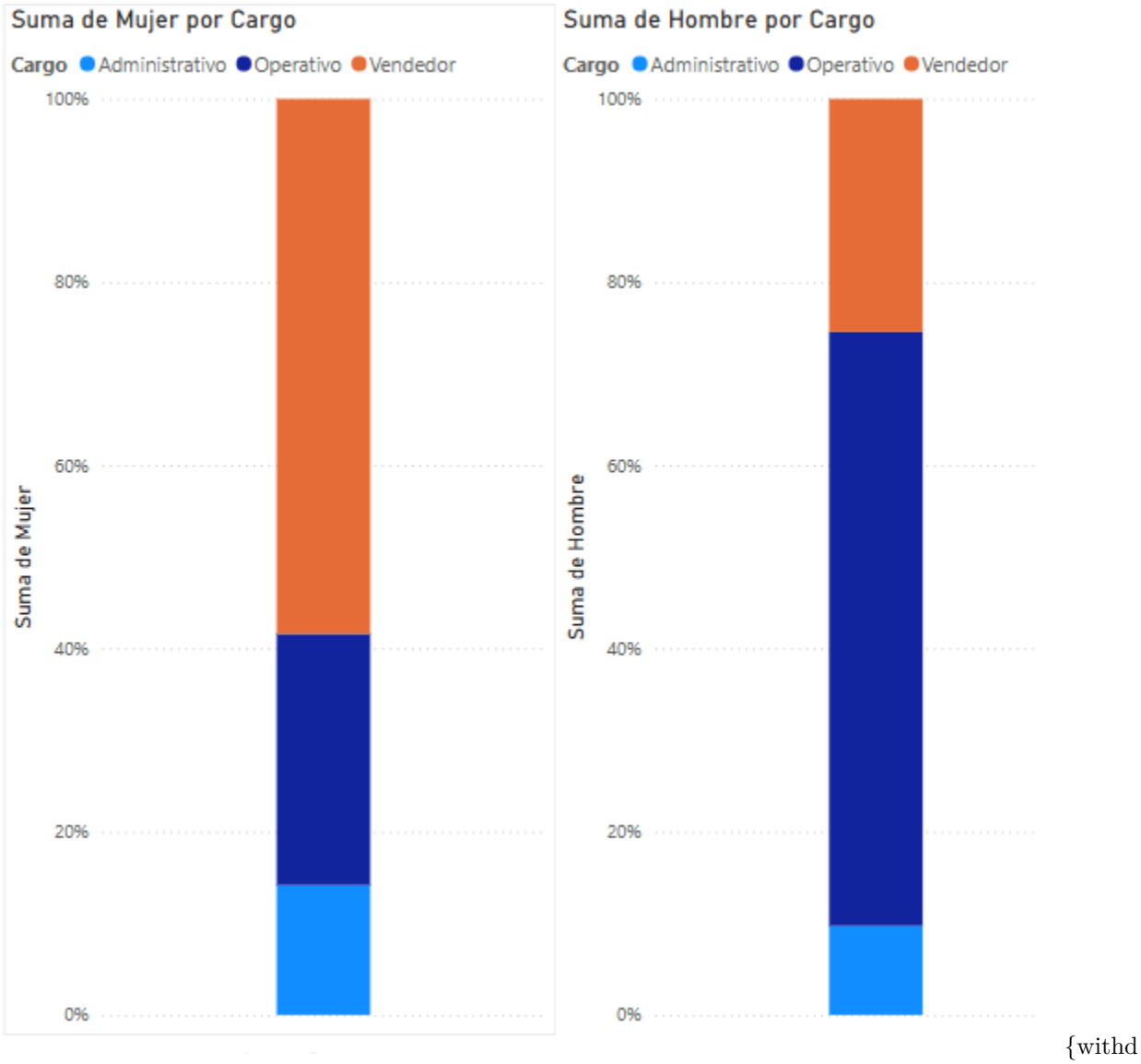


{withd

= 50%}

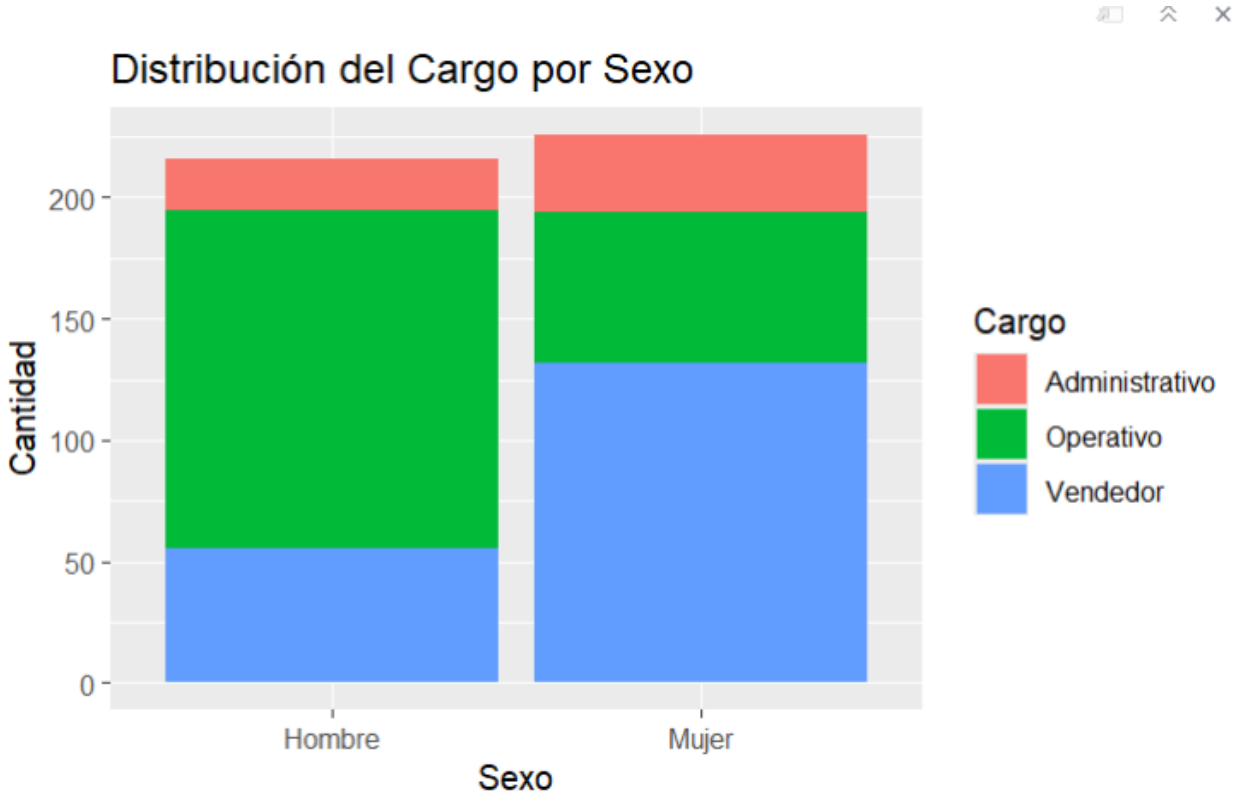
El diagrama generado por Power Bi tiene más detalles, en el sentido de tener otros intérvalos y al parecer muestra más datos que el generado por R. Claro, también entra la posibilidad de que el generado por Power BI esté tomando otros parámetros o que el autor (o sea yo) no lo haya formulado bien a la hora de crearlo. Sin embargo por la simplicidad de creación, me es más factible crearlo directamente en R.

Gráfico de Columnas Apiladas de Cargos por Género



{withd

= 50%}



{withd = 50%}

Tras analizar y comparar el resultado de estos gráficos, creo que es muy evidente que la falta de experiencia que tiene el autor en el manejo de Power BI afecta demasiado la generación de los diagramas solicitados. De nuevo, tanto por su simplicidad en la creación, como en la visualización final, es recomendable utilizar R para la generación de los gráficos.

Conclusión

Tras la realización de este trabajo, se lograron aplicar técnicas de análisis exploratorio de datos y visualización utilizando herramientas como R y Power BI. Este proceso me permitió obtener una comprensión muy profunda tanto de los conjuntos de datos analizados, como de los usos y funciones de RStudio y RMarkDown de las que puedo sacar provecho para generar mi informe mientras codifico mi análisis.

Me encontré con gran cantidad de retos durante el proceso, pues antes de este trabajo mi experiencia trabajando con R era de prácticamente cero, así que tuve que investigar mucho, sobre todo ver videos que me ayudaran a hacer desde lo más básico, como crear un nuevo archivo en RMarkDown, hasta lo más complejo, como la creación de todos los gráficos y la organización de los datos para su correcto manejo.

Gracias a esto he desarrollado habilidades prácticas en ciencia de datos, los cuales me serán de gran ayuda en proyectos y trabajos futuros, tanto dentro de la escuela como en el ámbito profesional.

Bibliografia

- [1] J. Doe, “Análisis Exploratorio de Datos con R”, Mi Canal de Ciencia de Datos, 2023. [Video en línea]. Disponible: https://youtu.be/2sdYmbqVDTY?si=WEbkx0YrTlXwA6-2 . [Fecha de acceso: 15 de octubre de 2023].

- [2] J. Smith, “Visualización de Datos con Power BI”, Data Visualization Hub, 2023. [Video en línea]. Disponible: https://youtu.be/9QFr63HiWCw?si=WIhgjOJcbWrSJav3 . [Fecha de acceso: 15 de octubre de 2023].
- [3] M. Johnson, “Introducción a Power Query en Power BI”, Power BI Tutorials, 2023. [Video en línea]. Disponible: https://youtu.be/_ei3eTTg8tU?si=3LDVK_cci_S01_X6 . [Fecha de acceso: 15 de octubre de 2023].
- [4] L. Brown, “Creación de Gráficos Avanzados en R”, Advanced R Programming, 2023. [Video en línea]. Disponible: https://youtu.be/U3bC--Zm3pw?si=4UeQ5_1NH6PiZfaL . [Fecha de acceso: 15 de octubre de 2023].
- [5] K. Davis, “Comparación entre R y Power BI para Análisis de Datos”, Data Science Insights, 2023. [Video en línea]. Disponible: https://youtu.be/72PUOyn33OY?si=DMis3bGD5_FfYXdl . [Fecha de acceso: 15 de octubre de 2023].