Trabajo 3: Aprendizaje No Supervisado

Pablo Setrakian Bearzotti

05-05-2025

Contents

1. Introducción **1**
1.1. Descripción del conjunto de datos . . . 2

2. Análisis Exploratorio de Datos **2**
2.1. Visualización inicial de los datos . . . 2
2.2. Detección de valores atípicos y ausentes . . . 5
2.3. Preprocesamiento de los datos . . . 6

3. Agrupamiento Jerárquico **7**
3.1. Justificación de la elección de medidas de disparidad . . . 7
3.1.1. Medida de disparidad entre observaciones: Distancia Euclidiana . . . 7
3.1.2. Medida de disparidad entre clusters: Comparación de métodos de enlace . . . 7
3.1.2.1. Método de Ward . . . 7
3.1.2.2. Método de enlace completo . . . 9
3.1.3. Visualización mejorada de los dendrogramas . . . 10
3.2. Determinación del número óptimo de clusters . . . 11
3.3. Representaciones alternativas del dendrograma . . . 13
3.4. Evaluación de la calidad del clustering . . . 15
3.5. Construcción final de clusters jerárquicos . . . 17

4. Algoritmo K-medias **18**
4.1. Determinación del número óptimo de clusters . . . 18
4.2. Aplicación del algoritmo K-medias . . . 19
4.3. Construcción de clusters con K-medias . . . 20
4.4. Visualización detallada de los clusters K-medias . . . 21

5. Comparación de Resultados **22**
5.1. Comparación con las especies biológicas conocidas . . . 22
5.2. Visualización comparativa . . . 22
5.3. Análisis multivariante comparativo . . . 24

6. Conclusiones **27**

1. Introducción

El presente trabajo tiene como objetivo aplicar técnicas de aprendizaje no supervisado, específicamente algoritmos de clustering, para analizar y segmentar un conjunto de datos. Se ha seleccionado el conjunto de datos `iris`, incluido en R base, que constituye uno de los conjuntos de datos más conocidos en la estadística y el aprendizaje automático.

1.1. Descripción del conjunto de datos

El conjunto de datos `iris` contiene información sobre 150 flores del género Iris pertenecientes a tres especies diferentes: Iris setosa, Iris versicolor e Iris virginica. Para cada espécimen, se han registrado cuatro variables morfológicas medidas en centímetros:

- Longitud del sépalo (Sepal.Length)
- Anchura del sépalo (Sepal.Width)
- Longitud del pétalo (Petal.Length)
- Anchura del pétalo (Petal.Width)

Este conjunto de datos fue recopilado por el botánico británico Edgar Anderson y posteriormente utilizado por Ronald Fisher en su trabajo seminal sobre análisis discriminante. La riqueza de este conjunto radica en que las especies están distribuidas de tal manera que una de ellas (setosa) es linealmente separable de las otras dos, mientras que las especies versicolor y virginica presentan cierto solapamiento, lo que lo convierte en un excelente caso de estudio para técnicas de clustering.

El objetivo del análisis es encontrar agrupaciones naturales en los datos utilizando únicamente las características morfológicas, y posteriormente contrastar si estas agrupaciones corresponden con las especies biológicas conocidas.

2. Análisis Exploratorio de Datos

Como primer paso, se procede a cargar y examinar el conjunto de datos para comprender su estructura y características básicas.

```
data <- iris %>%
  dplyr::select(-Species) %>%
  glimpse()
```

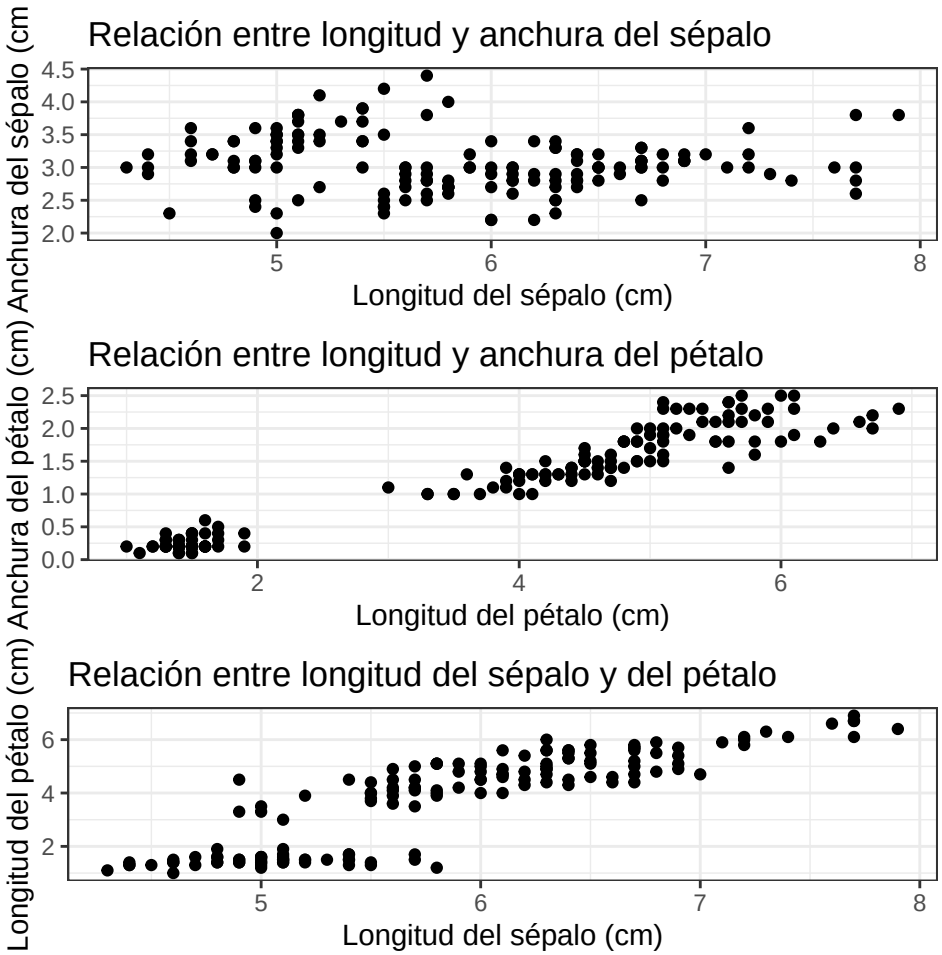
```
## Rows: 150
## Columns: 4
## $ Sepal.Length <dbl> 5.1, 4.9, 4.7, 4.6, 5.0, 5.4, 4.6, 5.0, 4.4, 4.9, 5.4, 4.~
## $ Sepal.Width  <dbl> 3.5, 3.0, 3.2, 3.1, 3.6, 3.9, 3.4, 3.4, 2.9, 3.1, 3.7, 3.~
## $ Petal.Length <dbl> 1.4, 1.4, 1.3, 1.5, 1.4, 1.7, 1.4, 1.5, 1.4, 1.5, 1.5, 1.~
## $ Petal.Width  <dbl> 0.2, 0.2, 0.2, 0.2, 0.2, 0.4, 0.3, 0.2, 0.2, 0.1, 0.2, 0.~
```

2.1. Visualización inicial de los datos

Se realizan visualizaciones bivariantes para explorar las relaciones entre las diferentes variables medidas y detectar posibles patrones o agrupaciones naturales en los datos.

```
plot.sl_sw <- data %>%
  ggplot() +
  aes(x = Sepal.Length,
      y = Sepal.Width) +
  geom_point() +
  theme_bw() +
  labs(title = "Relación entre longitud y anchura del sépalo",
       x = "Longitud del sépalo (cm)",
       y = "Anchura del sépalo (cm)")

plot.pl_pw <- data %>%
  ggplot() +
  aes(x = Petal.Length,
      y = Petal.Width) +
  geom_point() +
```

```
  theme_bw() +
  labs(title = "Relación entre longitud y anchura del pétalo",
       x = "Longitud del pétalo (cm)",
       y = "Anchura del pétalo (cm)")

plot.sl_pl <- data %>%
  ggplot() +
  aes(x = Sepal.Length,
      y = Petal.Length) +
  geom_point() +
  theme_bw() +
  labs(title = "Relación entre longitud del sépalo y del pétalo",
       x = "Longitud del sépalo (cm)",
       y = "Longitud del pétalo (cm)")

library(ggpubr)
ggarrange(plot.sl_sw,
          plot.pl_pw,
          plot.sl_pl,
          ncol=1,nrow=3)
```

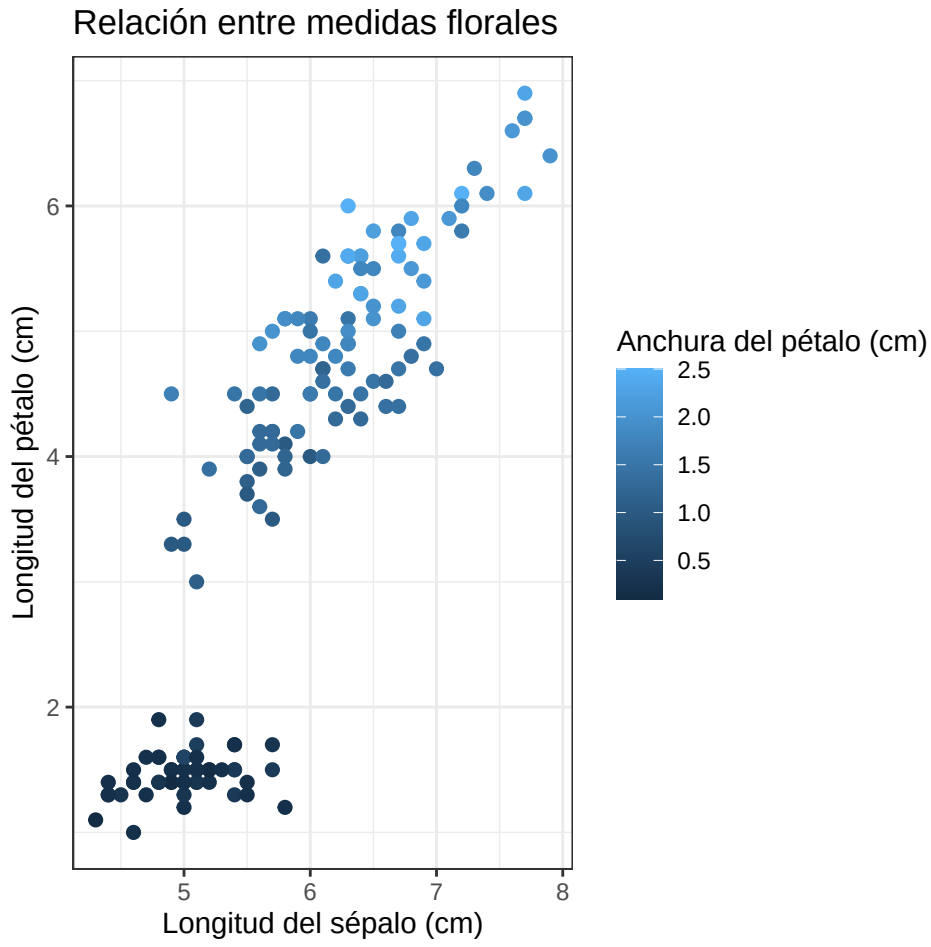


En la visualización de las relaciones bivariantes, se pueden observar patrones interesantes:

1. En la relación entre longitud y anchura del sépalo, no se aprecian agrupaciones claramente diferenciadas.
2. Sin embargo, en la relación entre longitud y anchura del pétalo, se distinguen al menos dos grupos bien diferenciados, con un posible tercer grupo menos definido.
3. La relación entre la longitud del sépalo y la longitud del pétalo también muestra una clara separación entre grupos.

Para una mejor comprensión de la estructura multidimensional de los datos, se visualizan las relaciones entre tres variables simultáneamente:

```
data %>%
  ggplot() +
  aes(x = Sepal.Length,
      y = Petal.Length,
      color = Petal.Width) +
  labs(x="Longitud del sépalo (cm)",
       y="Longitud del pétalo (cm)",
       title="Relación entre medidas florales") +
  scale_color_continuous(name="Anchura del pétalo (cm)") +
  geom_point(size = 2) +
  theme_bw()
```




```
library(plotly)
```

```
##
## Attaching package: 'plotly'
```

```
## The following object is masked from 'package:ggplot2':
##
##     last_plot
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

```
## The following object is masked from 'package:graphics':
##
##     layout
```

```
data %>%
  plot_ly(x = ~ Sepal.Length,
          y = ~ Petal.Length,
          z = ~ Petal.Width,
          color = ~ Sepal.Width,
          type = "scatter3d",
          mode = "markers"
  ) %>%
  layout(scene = list(xaxis = list(title = "Longitud del sépalo (cm)"),
                      yaxis = list(title = "Longitud del pétalo (cm)"),
                      zaxis = list(title = "Anchura del pétalo (cm)")),
         title = "Visualización tridimensional de las características florales")
```

2.2. Detección de valores atípicos y ausentes

Es importante verificar la presencia de valores atípicos o ausentes en el conjunto de datos, ya que podrían afectar al rendimiento de los algoritmos de clustering.

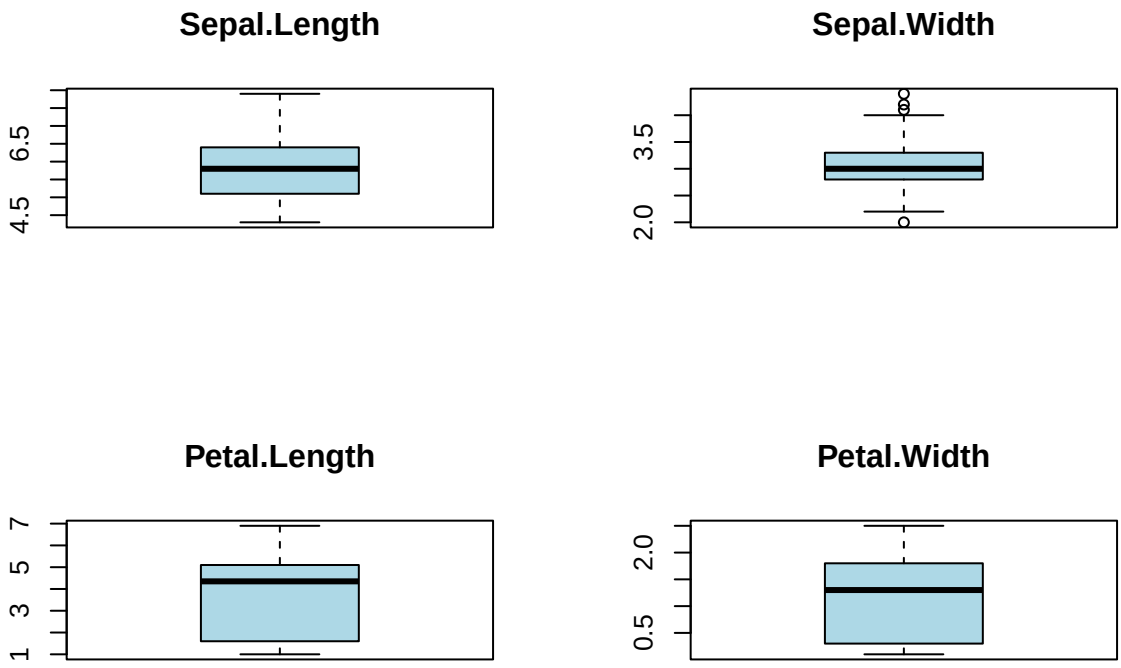
```
# Verificación de valores ausentes
sum(is.na(data))
```

```
## [1] 0
```

```
# Resumen estadístico para detección de posibles valores atípicos
summary(data)
```

```
##   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
```

```
# Visualización de diagramas de caja para detectar valores atípicos
par(mfrow=c(2,2))
for(i in 1:4) {
  boxplot(data[,i], main=names(data)[i], col="lightblue")
}
```



No se detectan valores ausentes en el conjunto de datos. Los diagramas de caja muestran algunos valores potencialmente atípicos, pero dado el contexto biológico y el tamaño limitado de la muestra, se decide mantenerlos para el análisis de clustering, ya que podrían representar variabilidad natural dentro de las especies.

2.3. Preprocesamiento de los datos

A diferencia de otros conjuntos de datos que pueden requerir transformaciones logarítmicas para manejar distribuciones asimétricas, las variables en el conjunto `iris` se encuentran en escalas relativamente similares (todas medidas en centímetros). Sin embargo, existen diferencias en los rangos de las variables que podrían influir en los algoritmos de clustering basados en distancia.

Por lo tanto, se procede a estandarizar las variables para que todas contribuyan equitativamente al análisis:

```
data_scaled <- data %>%
  mutate(
    across(everything(),
           ~ as.numeric(scale(.))
    )
  ) %>%
  glimpse()
```

```
## Rows: 150
## Columns: 4
## $ Sepal.Length <dbl> -0.89767388, -1.13920048, -1.38072709, -1.50149039, -1.01~
## $ Sepal.Width  <dbl> 1.01560199, -0.13153881, 0.32731751, 0.09788935, 1.245030~
## $ Petal.Length <dbl> -1.335752, -1.335752, -1.392399, -1.279104, -1.335752, -1~
## $ Petal.Width  <dbl> -1.3110521, -1.3110521, -1.3110521, -1.3110521, -1.311052~
```

La estandarización convierte cada variable a una escala con media 0 y desviación estándar 1, eliminando así el efecto de las diferentes unidades de medida y magnitudes en el análisis de clustering.

3. Agrupamiento Jerárquico

El agrupamiento jerárquico es una técnica que construye una jerarquía de clusters mediante un proceso iterativo. En este trabajo, se aplicará esta técnica con diferentes combinaciones de medidas de disparidad entre observaciones y entre clusters.

3.1. Justificación de la elección de medidas de disparidad

3.1.1. Medida de disparidad entre observaciones: Distancia Euclidiana

```
# Elegimos la distancia euclidiana
dist_eucli <- dist(data_scaled, method="euclidean")
```

Justificación de la elección de distancia Euclidiana:

La distancia euclidiana ha sido seleccionada como medida de disparidad entre observaciones por las siguientes razones:

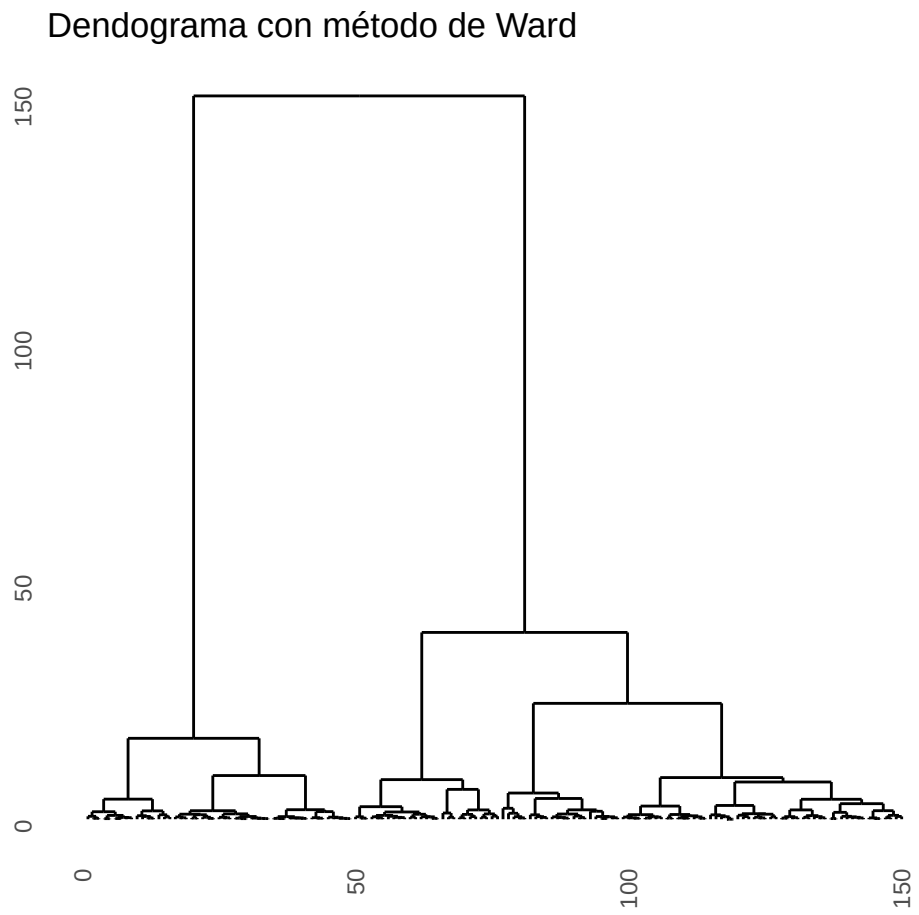
1. **Interpretabilidad geométrica**: Las medidas de pétalos y sépalos representan distancias físicas en el espacio, y la distancia euclidiana corresponde directamente al concepto intuitivo de distancia en un espacio geométrico.
2. **Correlación entre variables**: Las características morfológicas de las flores (longitud y anchura) están correlacionadas naturalmente, y la distancia euclidiana captura adecuadamente estas relaciones.
3. **Variables en la misma escala**: Después de la estandarización, todas las variables están en la misma escala, haciendo que la distancia euclidiana sea equitativa en su ponderación de cada característica.
4. **Sensibilidad a diferencias grandes**: La distancia euclidiana, al elevar al cuadrado las diferencias, es más sensible a diferencias grandes entre especies, lo que es útil para identificar grupos bien diferenciados como en el caso de las especies de Iris.

3.1.2. Medida de disparidad entre clusters: Comparación de métodos de enlace

Para evaluar el impacto del método de enlace en los resultados del clustering jerárquico, se compararán dos enfoques: el método de Ward y el método de enlace completo.

```
library(ggdendro)
# Aplicamos el método de Ward
model_dendo.ward <- dist_eucli %>%
  hclust(method = "ward.D")

model_dendo.ward %>%
  ggdendrogram(
    rotate = FALSE,
    labels = FALSE,
    theme_dendro = TRUE
  ) +
  labs(title = "Dendograma con método de Ward")
```

3.1.2.1. Método de Ward



Justificación de la elección del método de Ward:

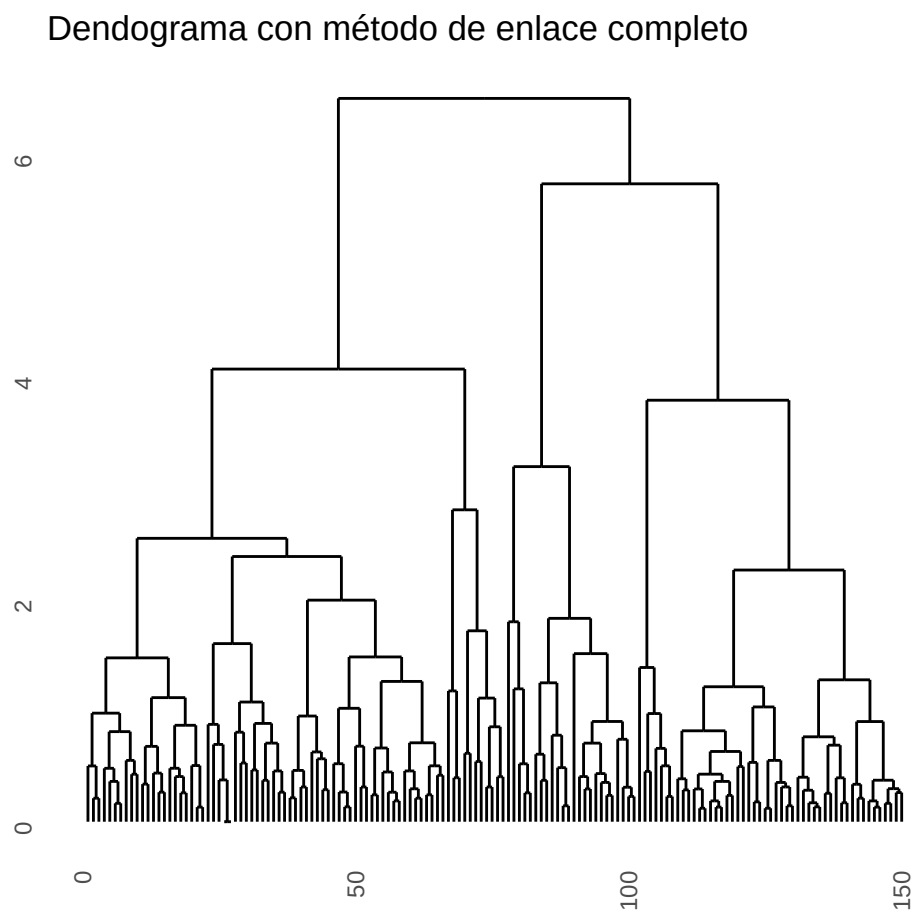
El método de Ward es considerado apropiado para el conjunto de datos Iris por los siguientes motivos:

1. **Minimización de la varianza**: Ward busca minimizar la suma de cuadrados dentro de cada cluster, lo que resulta en grupos homogéneos y es ideal para identificar grupos compactos como las especies de Iris.
2. **Robustez frente a valores atípicos**: En comparación con otros métodos, Ward es menos susceptible al efecto de encadenamiento y produce clusters más equilibrados en tamaño.
3. **Coherencia con k-means**: El método de Ward utiliza criterios similares a k-means (minimización de varianza), lo que facilita la comparación entre ambos métodos de clustering.
4. **Evidencia empírica**: En numerosos estudios comparativos de técnicas de clustering, el método de Ward ha demostrado un rendimiento superior para conjuntos de datos con clusters bien definidos y de tamaño similar, como es el caso del conjunto de datos Iris.

```
# Aplicamos el método de enlace completo
model_dendo.complete <- dist_eucli %>%
  hclust(method = "complete")

model_dendo.complete %>%
  ggdendrogram(
```

```
    rotate = FALSE,
    labels = FALSE,
    theme_dendro = TRUE
  ) +
  labs(title = "Dendograma con método de enlace completo")
```

3.1.2.2. Método de enlace completo



Comparación entre métodos de enlace:

El método de enlace completo define la distancia entre clusters como la máxima distancia entre cualquier par de puntos pertenecientes a clusters diferentes. En comparación con el método de Ward:

1. **Forma de los clusters**: El enlace completo tiende a formar clusters más compactos y de tamaño similar, pero puede ser sensible a valores atípicos.
2. **Estructura jerárquica**: El dendrograma resultante muestra una estructura diferente, con ramificaciones más pronunciadas que pueden dificultar la identificación del número óptimo de clusters.
3. **Interpretabilidad**: En el contexto de las características morfológicas de las flores, el enlace completo podría no capturar tan bien la variabilidad natural dentro de cada especie como lo hace el método de Ward.

3.1.3. Visualización mejorada de los dendrogramas

```
library(dendextend)
```

```
##
## ---------------------
## Welcome to dendextend version 1.19.0
## Type citation('dendextend') for how to cite the package.
##
## Type browseVignettes(package = 'dendextend') for the package vignette.
## The github page is: https://github.com/talgalili/dendextend/
##
## Suggestions and bug-reports can be submitted at: https://github.com/talgalili/dendextend/issues
## You may ask questions at stackoverflow, use the r and dendextend tags:
##   https://stackoverflow.com/questions/tagged/dendextend
##
##  To suppress this message use:  suppressPackageStartupMessages(library(dendextend))
## ---------------------
```

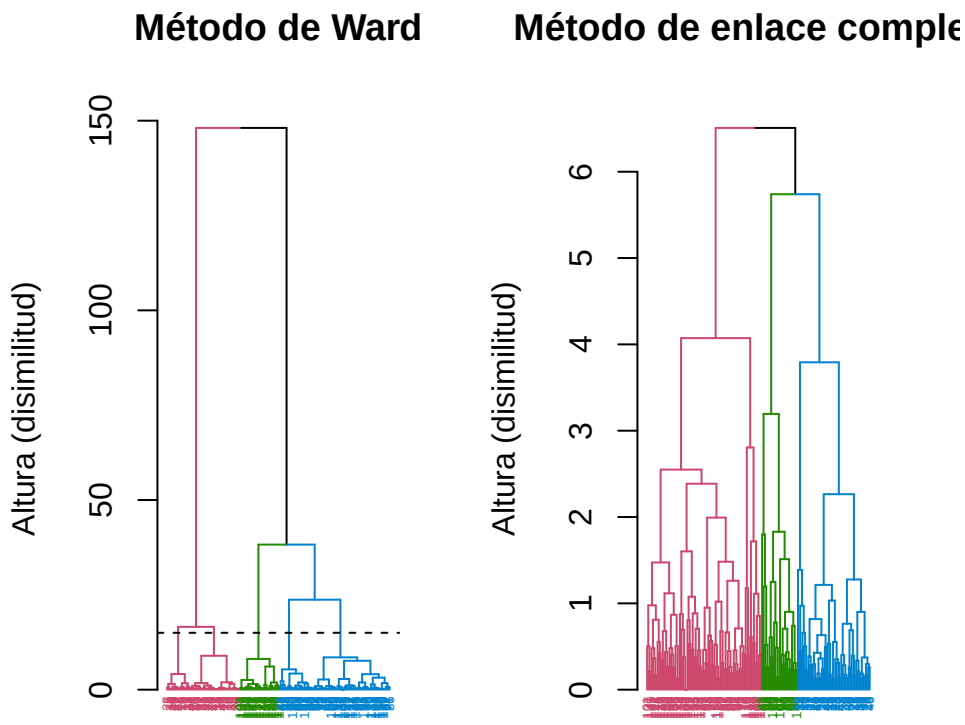
```
##
## Attaching package: 'dendextend'
```

```
## The following object is masked from 'package:ggdendro':
##
##     theme_dendro
```

```
## The following object is masked from 'package:ggpubr':
##
##     rotate
```

```
## The following object is masked from 'package:stats':
##
##     cutree
```

```
# Visualización mejorada del dendrograma con método de Ward
dend_ward <- model_dendo.ward %>%
  as.dendrogram() %>%
  color_branches(k = 3) %>%
  color_labels(k = 3) %>%  #colorea según el clúster
  set("labels_cex", 0.4)

# Visualización mejorada del dendrograma con método de enlace completo
dend_complete <- model_dendo.complete %>%
  as.dendrogram() %>%
  color_branches(k = 3) %>%
  color_labels(k = 3) %>%
  set("labels_cex", 0.4)

# Comparación visual
par(mfrow=c(1,2))
plot(dend_ward, main = "Método de Ward", ylab = "Altura (disimilitud)")
abline(h = 15, lty=2)
plot(dend_complete, main = "Método de enlace completo", ylab = "Altura (disimilitud)")
abline(h = 15, lty=2)
```



La comparación visual de los dendrogramas revela diferencias significativas en la estructura jerárquica producida por cada método de enlace. El método de Ward tiende a producir clusters más equilibrados y una jerarquía más clara, mientras que el enlace completo muestra una estructura más irregular.

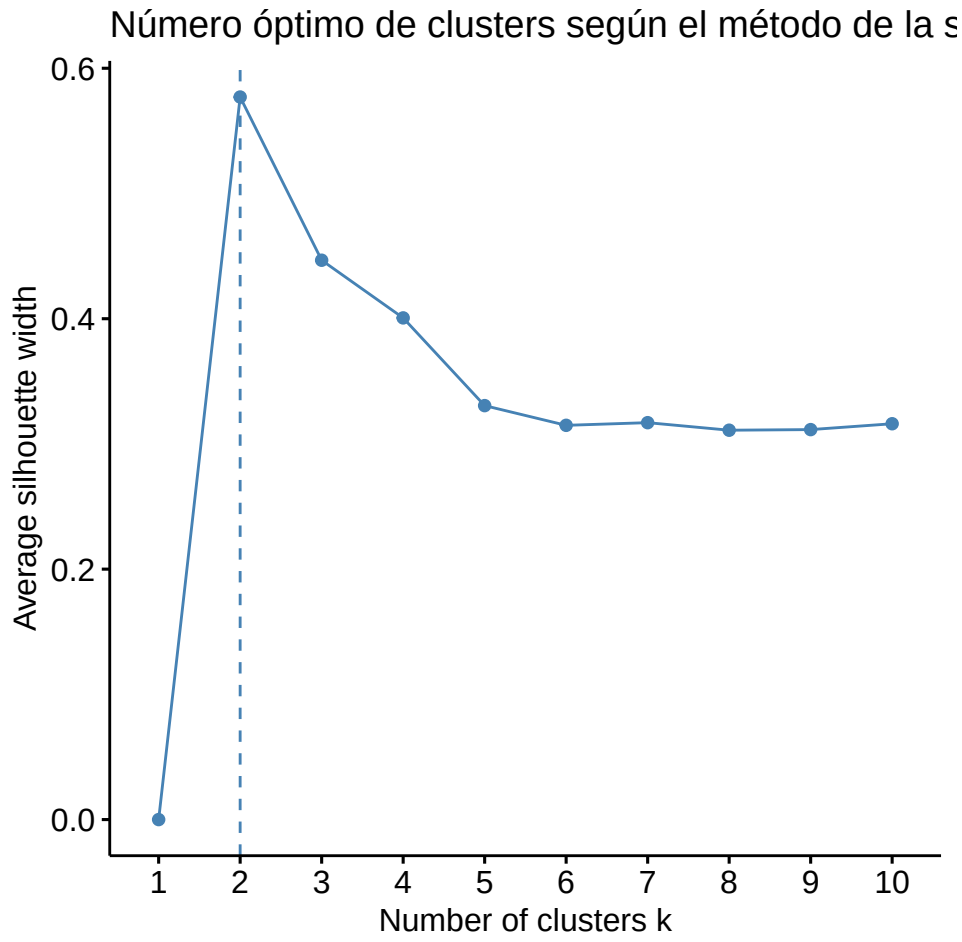
3.2. Determinación del número óptimo de clusters

Para determinar el número óptimo de clusters en el agrupamiento jerárquico, se utiliza el método de la silueta, que evalúa la calidad de los clusters en función de la cohesión interna y la separación entre clusters.

```
library(factoextra)
```

```
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

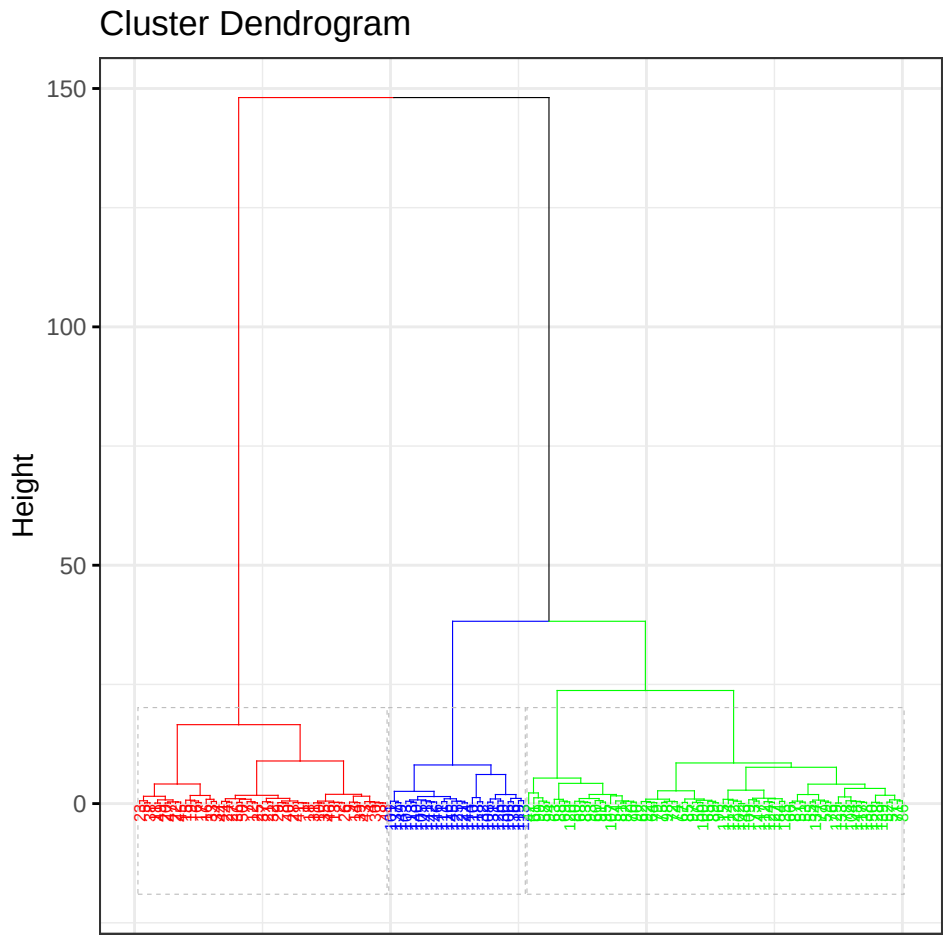
```
data_scaled %>%
  fviz_nbclust(FUNcluster = hcut,
               method = "silhouette",
               k.max = 10) +
  labs(title = "Número óptimo de clusters según el método de la silueta")
```



El método de la silueta sugiere que el número óptimo de clusters es 2. Sin embargo, teniendo en cuenta el conocimiento previo de que existen tres especies de Iris en el conjunto de datos, y considerando la estructura observada en el análisis exploratorio, se decide proceder con 3 clusters para el análisis.

```
# Visualización del dendrograma con 3 clusters
model_dendo.ward %>%
  fviz_dend(k = 3,
            rect = TRUE,
            horiz = FALSE,
            rect_border = "gray",
            rect_fill = FALSE,
            cex = 0.4,
            lwd = 0.2,
            k_colors = c("red","blue","green"),
            ggtheme = theme_bw()
  )
```

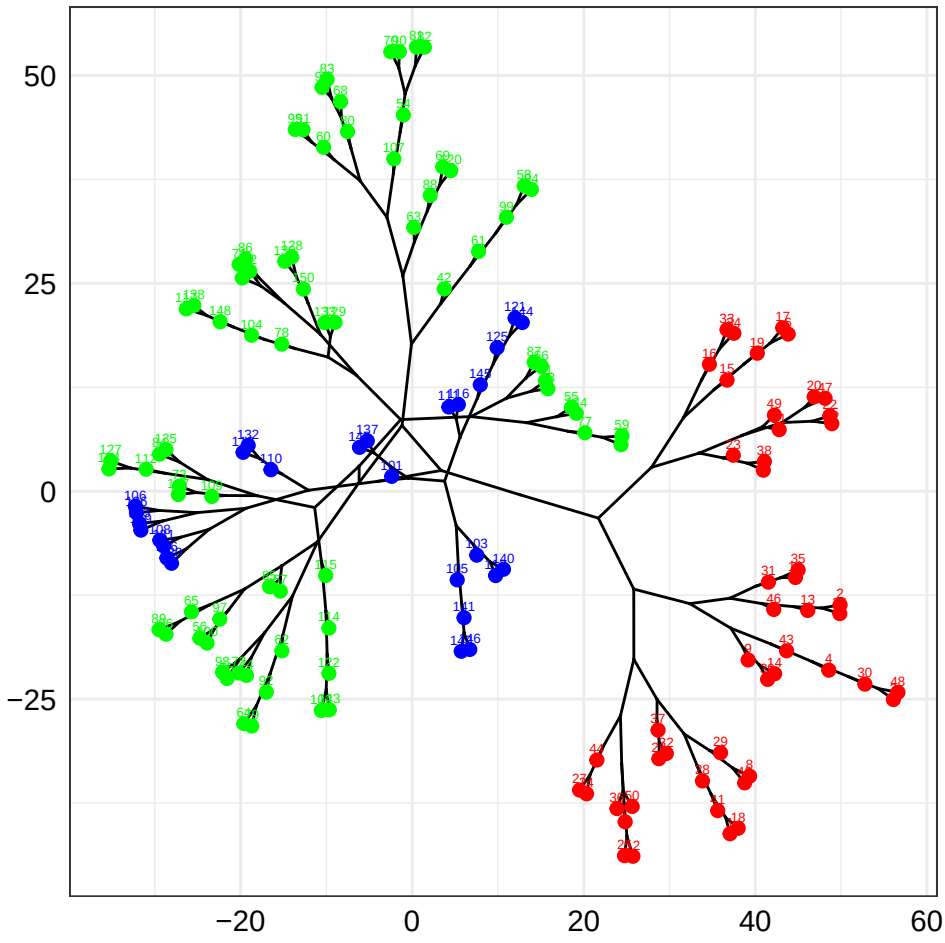
```
## Warning: The `<scale>` argument of `guides()` cannot be `FALSE`. Use "none" instead as
## of ggplot2 3.3.4.
## i The deprecated feature was likely used in the factoextra package.
##   Please report the issue at <https://github.com/kassambara/factoextra/issues>.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



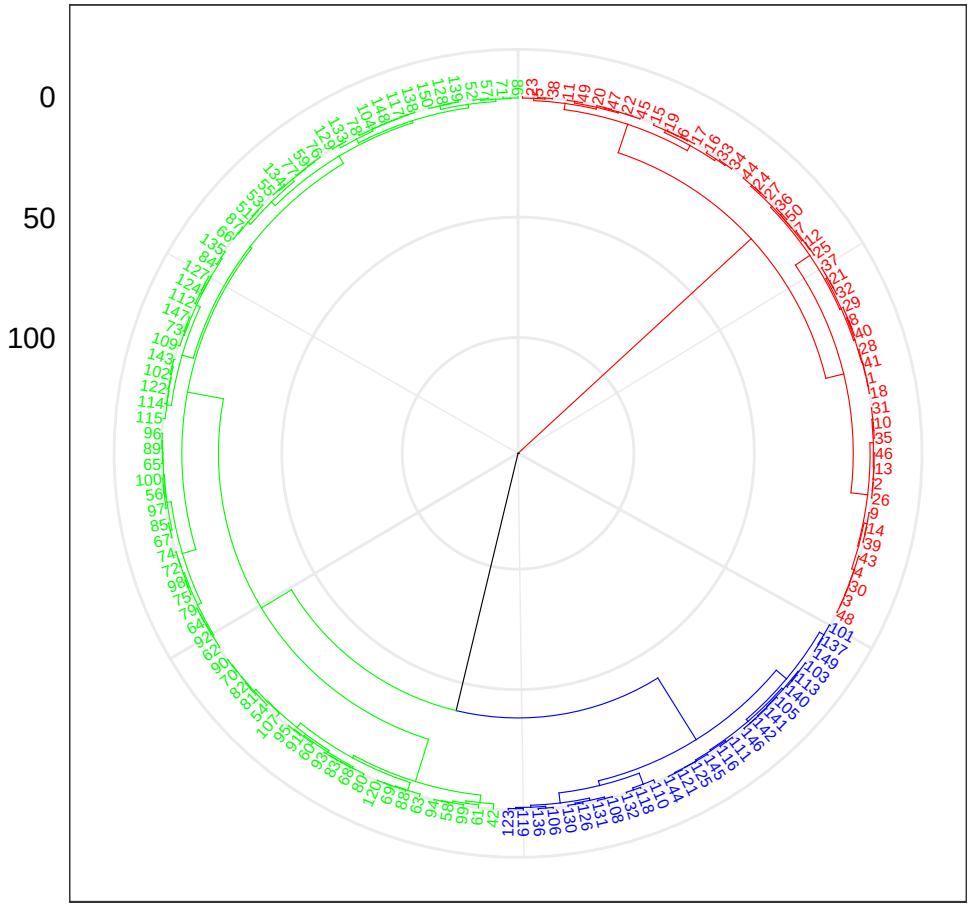
3.3. Representaciones alternativas del dendrograma

Para una comprensión más completa de la estructura jerárquica, se presentan representaciones alternativas del dendrograma:

```
# Representación filogenética del dendrograma
model_dendo.ward %>%
  fviz_dend(k = 3,
            rect = TRUE,
            horiz = FALSE,
            rect_border = "gray",
            rect_fill = FALSE,
            cex = 0.4,
            lwd = 0.2,
            k_colors = c("red","blue","green"),
            ggtheme = theme_bw(),
            type = "phylogenic",
            main = "Representación filogenética (método de Ward)"
  )
```




```r
# Representación circular del dendrograma
model_dendo.ward %>%
  fviz_dend(k = 3,
            rect = TRUE,
            horiz = FALSE,
            rect_border = "gray",
            rect_fill = FALSE,
            cex = 0.4,
            lwd = 0.2,
            k_colors = c("red","blue","green"),
            ggtheme = theme_bw(),
            type = "circular",
            main = "Representación circular (método de Ward)"
  )
```

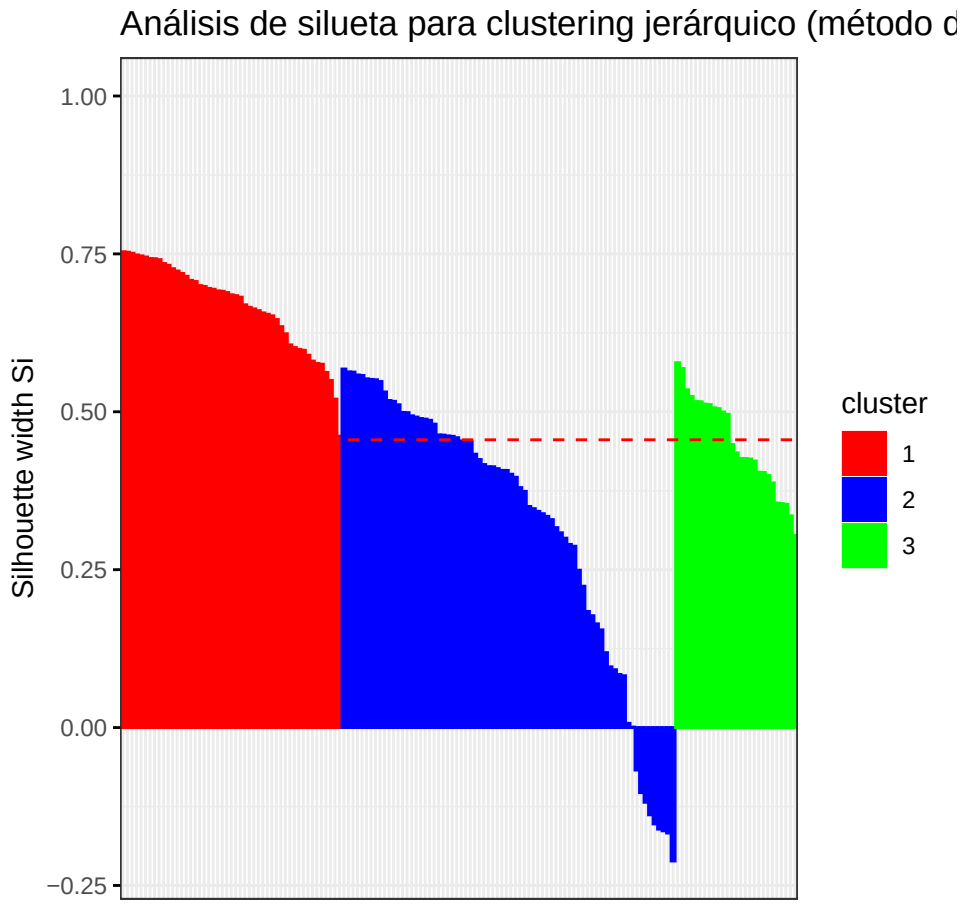


3.4. Evaluación de la calidad del clustering

Para evaluar la calidad del clustering jerárquico, se utiliza el análisis de silueta, que proporciona una medida de cuán bien está asignada cada observación a su cluster.

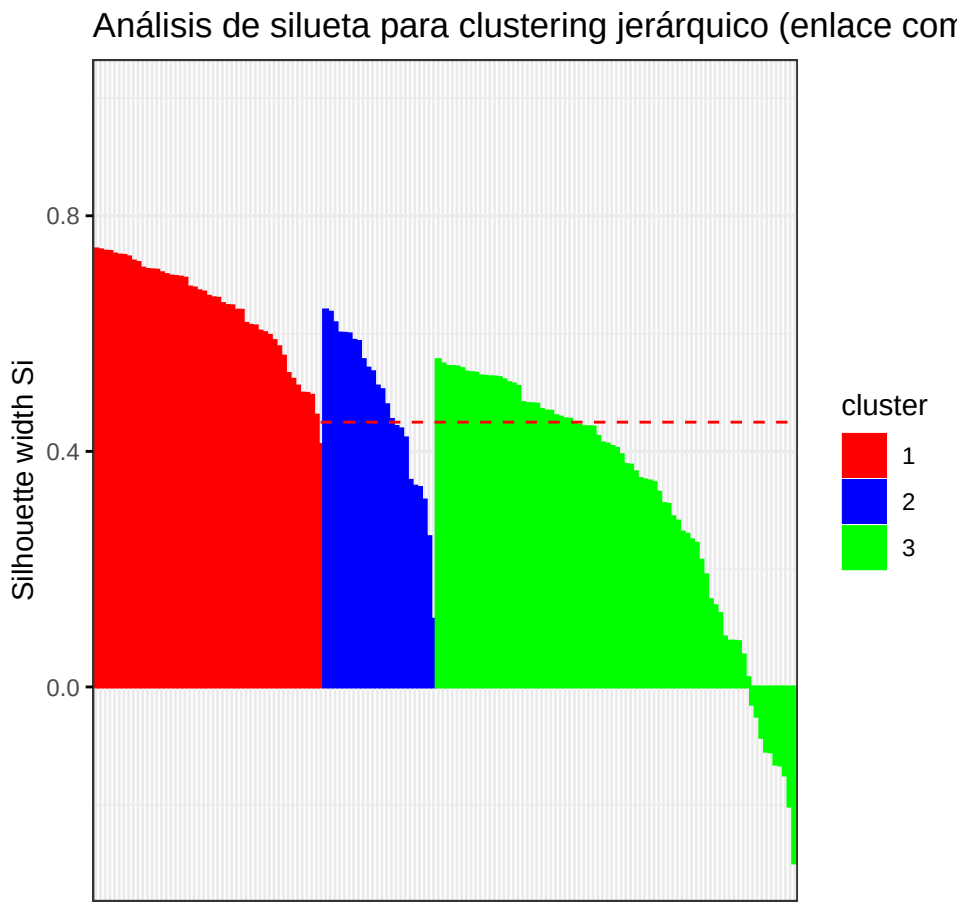
```
# Análisis de silueta para el método de Ward
library(cluster)
fviz_silhouette(
  silhouette(cutree(model_dendo.ward, k = 3), dist_eucli),
  palette = c("red", "blue", "green"),
  ggtheme = theme_bw()
) +
  labs(title = "Análisis de silueta para clustering jerárquico (método de Ward)")
```

```
##   cluster size ave.sil.width
## 1       1   49          0.67
## 2       2   74          0.32
## 3       3   27          0.45
```




```
# Análisis de silueta para el método de enlace completo
fviz_silhouette(
  silhouette(cutree(model_dendo.complete, k = 3), dist_eucli),
  palette = c("red", "blue", "green"),
  ggtheme = theme_bw()
) +
  labs(title = "Análisis de silueta para clustering jerárquico (enlace completo)")
```

```
##   cluster size ave.sil.width
## 1       1   49          0.64
## 2       2   24          0.48
## 3       3   77          0.32
```



Interpretación del análisis de silueta:

El análisis de silueta revela la calidad de la asignación de las observaciones a los clusters. Los valores de silueta próximos a 1 indican una buena asignación, mientras que valores cercanos a 0 o negativos sugieren posibles asignaciones incorrectas.

Se observa que: - El método de Ward produce clusters con valores de silueta generalmente más altos, lo que indica una mejor separación entre clusters. - El método de enlace completo muestra valores de silueta más variables, con algunas observaciones potencialmente mal asignadas.

Basándose en estos resultados, se decide proceder con el método de Ward para la construcción final de los clusters.

3.5. Construcción final de clusters jerárquicos

```
data_scaled <- data_scaled %>%
  mutate(
    cluster_jerar = factor(model_dendo.ward %>%
                             cutree(k = 3))
  ) %>%
  glimpse()
```

```
## Rows: 150
## Columns: 5
## $ Sepal.Length  <dbl> -0.89767388, -1.13920048, -1.38072709, -1.50149039, -1.0~
```

```
## $ Sepal.Width   <dbl> 1.01560199, -0.13153881, 0.32731751, 0.09788935, 1.24503~
## $ Petal.Length  <dbl> -1.335752, -1.335752, -1.392399, -1.279104, -1.335752, -~
## $ Petal.Width   <dbl> -1.3110521, -1.3110521, -1.3110521, -1.3110521, -1.31105~
## $ cluster_jerar <fct> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,~
```

```r
library(plotly)

data_scaled %>%
  plot_ly(x = ~ Sepal.Length,
          y = ~ Petal.Length,
          z = ~ Petal.Width,
          color = ~ cluster_jerar,
          type = "scatter3d",
          mode = "markers",
          colors = c("red","blue","green")
  ) %>%
  layout(scene = list(xaxis = list(title = "Longitud del sépalo (estand.)"),
                      yaxis = list(title = "Longitud del pétalo (estand.)"),
                      zaxis = list(title = "Anchura del pétalo (estand.)")),
         title = "Visualización 3D de clusters jerárquicos (método de Ward)")
```

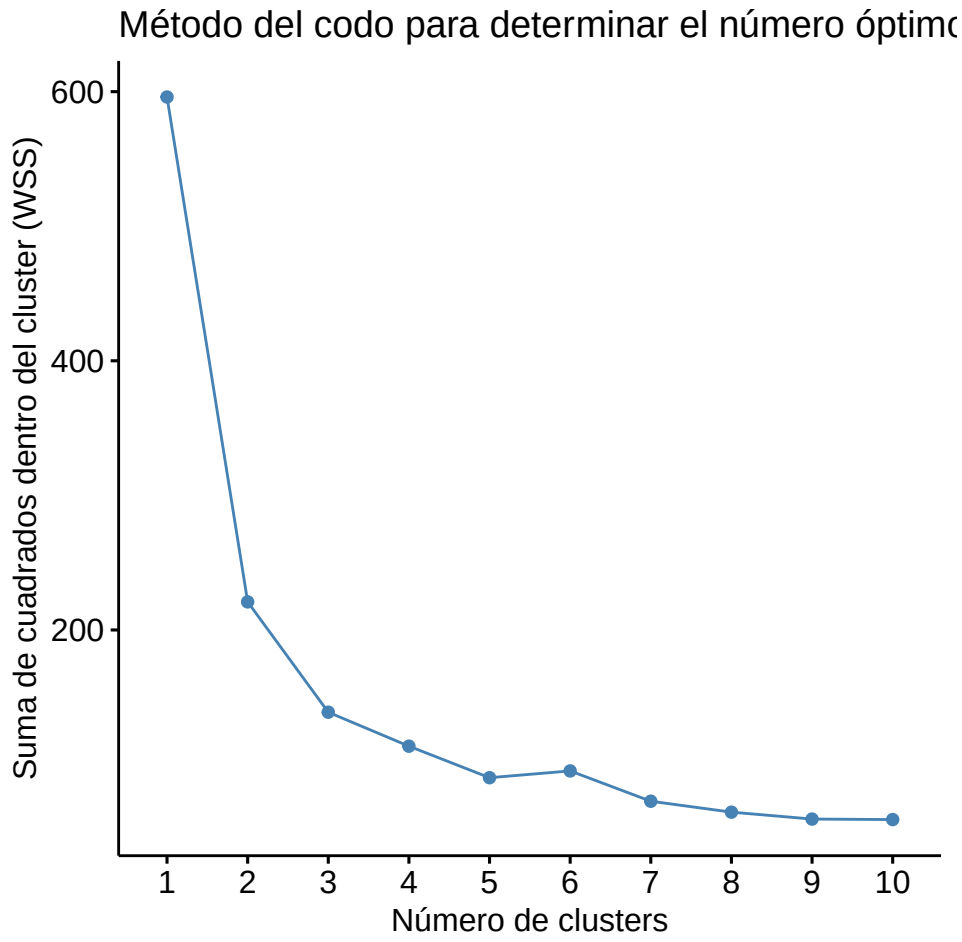
4. Algoritmo K-medias

El algoritmo k-medias es una técnica de partición que divide el conjunto de datos en k grupos, donde cada observación pertenece al cluster con el centroide más cercano.

4.1. Determinación del número óptimo de clusters

Para determinar el número óptimo de clusters en k-medias, se utiliza el método del codo (elbow method), que examina la variación de la suma de cuadrados dentro del cluster (WSS) en función del número de clusters.

```r
library(factoextra)

data_scaled %>%
  select(-cluster_jerar) %>% #debemos eliminar esta variable
  fviz_nbclust(FUNcluster = kmeans,
               method = "wss") +
  labs(title = "Método del codo para determinar el número óptimo de clusters",
       x = "Número de clusters",
       y = "Suma de cuadrados dentro del cluster (WSS)")
```



Justificación de la elección del número de clusters:

El gráfico del método del codo muestra la disminución de la suma de cuadrados dentro del cluster (WSS) a medida que aumenta el número de clusters. Se busca el punto donde esta disminución se estabiliza, formando un "codo" en el gráfico.

En este caso, se puede observar un codo en k=3, lo que sugiere que este es el número óptimo de clusters. Esta elección coincide con el conocimiento previo de que el conjunto de datos contiene tres especies de Iris.

4.2. Aplicación del algoritmo K-medias

```
model_kmeans <- data_scaled %>%
  select(-cluster_jerar) %>%
  kmeans(centers = 3,
         nstart = 10)

model_kmeans
```

```
## K-means clustering with 3 clusters of sizes 53, 47, 50
##
## Cluster means:
##   Sepal.Length Sepal.Width Petal.Length Petal.Width
## 1  -0.05005221 -0.88042696    0.3465767   0.2805873
## 2   1.13217737  0.08812645    0.9928284   1.0141287
```

```
## 3  -1.01119138  0.85041372   -1.3006301  -1.2507035
##
## Clustering vector:
##   [1] 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
##  [38] 3 3 3 3 3 3 3 3 3 3 3 3 3 2 2 2 1 1 1 2 1 1 1 1 1 1 1 2 1 1 1 1 2 1 1 1
##  [75] 1 2 2 2 1 1 1 1 1 1 1 2 2 1 1 1 1 1 1 1 1 1 1 1 1 2 1 2 2 2 2 1 2 2 2 2
## [112] 2 2 1 1 2 2 2 2 1 2 1 2 1 2 2 1 2 2 2 2 2 2 1 1 2 2 2 1 2 2 2 1 2 2 2 1 2
## [149] 2 1
##
## Within cluster sum of squares by cluster:
## [1] 44.08754 47.45019 47.35062
##  (between_SS / total_SS =  76.7 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

```
# Distribución de observaciones en los clusters
table(model_kmeans$cluster)
```

```
##
##  1  2  3
## 53 47 50
```

4.3. Construcción de clusters con K-medias

```
data_scaled <- data_scaled %>%
  mutate(
    cluster_kmeans = factor(model_kmeans$cluster)
    ) %>%
  glimpse()
```

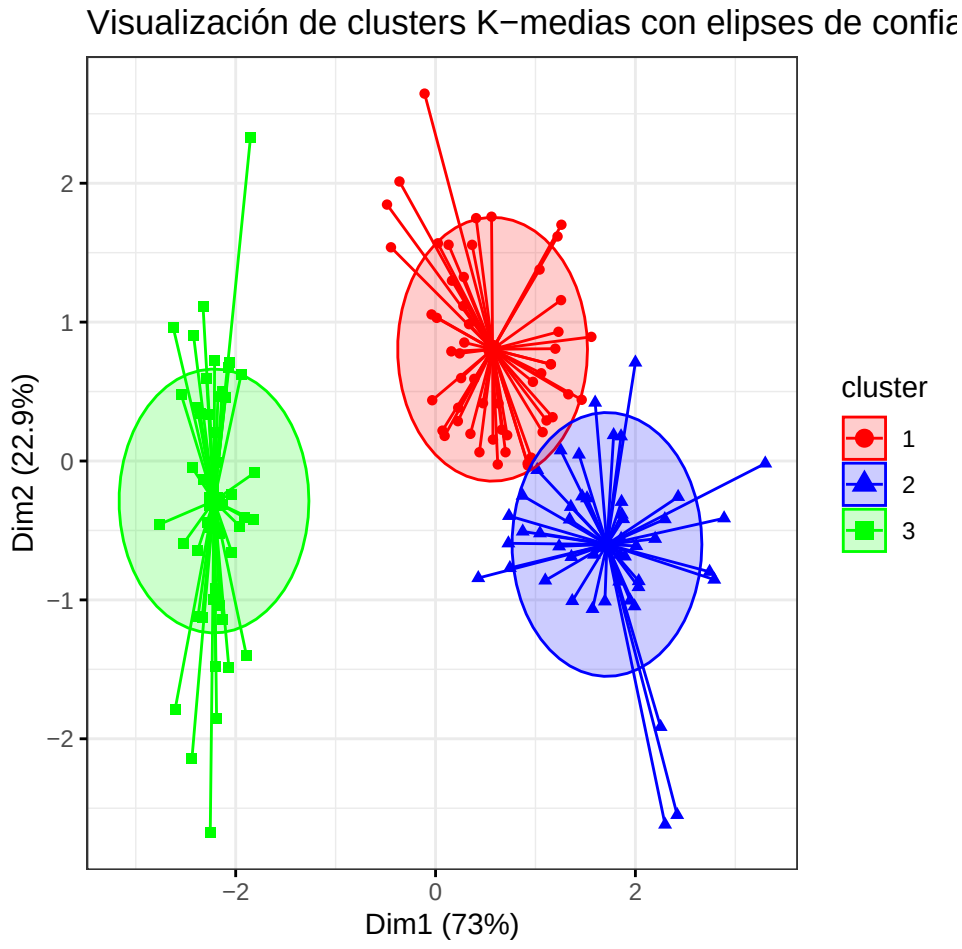
```
## Rows: 150
## Columns: 6
## $ Sepal.Length   <dbl> -0.89767388, -1.13920048, -1.38072709, -1.50149039, -1.~
## $ Sepal.Width    <dbl> 1.01560199, -0.13153881, 0.32731751, 0.09788935, 1.2450~
## $ Petal.Length   <dbl> -1.335752, -1.335752, -1.392399, -1.279104, -1.335752, ~
## $ Petal.Width    <dbl> -1.3110521, -1.3110521, -1.3110521, -1.3110521, -1.3110~
## $ cluster_jerar  <fct> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1~
## $ cluster_kmeans <fct> 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3~
```

```
library(plotly)

data_scaled %>%
  plot_ly(x = ~ Sepal.Length,
          y = ~ Petal.Length,
          z = ~ Petal.Width,
          color = ~ cluster_kmeans,
          type = "scatter3d",
          mode = "markers",
          colors = c("red","blue","green")
  ) %>%
  layout(scene = list(xaxis = list(title = "Longitud del sépalo (estand.)"),
                      yaxis = list(title = "Longitud del pétalo (estand.)"),
```

```
                    zaxis = list(title = "Anchura del pétalo (estand.)")),
         title = "Visualización 3D de clusters K-medias")
```

4.4. Visualización detallada de los clusters K-medias

```
library(factoextra)

model_kmeans %>%
  fviz_cluster(
    data = data_scaled %>% select(Sepal.Length, Sepal.Width, Petal.Length, Petal.Width),
    geom = "point",
    palette = c("red","blue","green"),
    ellipse = TRUE,
    ellipse.type = "euclid",
    star.plot = TRUE,
    repel = TRUE,
    ggtheme = theme_bw()
  ) +
  labs(title = "Visualización de clusters K-medias con elipses de confianza")
```



5. Comparación de Resultados

5.1. Comparación con las especies biológicas conocidas

Para evaluar la efectividad de los algoritmos de clustering, se comparan los clusters obtenidos con las especies biológicas conocidas:

```
# Añadimos las especies reales para comparar
data_comparison <- data_scaled %>%
  mutate(Species = iris$Species)

# Tabla de contingencia entre cluster k-means y especies
tabla_kmeans <- table(data_comparison$cluster_kmeans, data_comparison$Species)
print("Comparación de K-medias con especies reales:")
```

```
## [1] "Comparación de K-medias con especies reales:"
```

```
print(tabla_kmeans)
```

```
##
##     setosa versicolor virginica
##   1      0         39        14
##   2      0         11        36
##   3     50          0         0
```

```
# Tabla de contingencia entre cluster jerárquico y especies
tabla_jerar <- table(data_comparison$cluster_jerar, data_comparison$Species)
print("Comparación de clustering jerárquico con especies reales:")
```

```
## [1] "Comparación de clustering jerárquico con especies reales:"
```

```
print(tabla_jerar)
```

```
##
##     setosa versicolor virginica
##   1     49          0         0
##   2      1         50        23
##   3      0          0        27
```

```
# Cálculo de precisión para K-medias
precision_kmeans <- sum(diag(tabla_kmeans)) / sum(tabla_kmeans)
cat("Precisión de K-medias:", round(precision_kmeans*100, 2), "%\n")
```

```
## Precisión de K-medias: 7.33 %
```

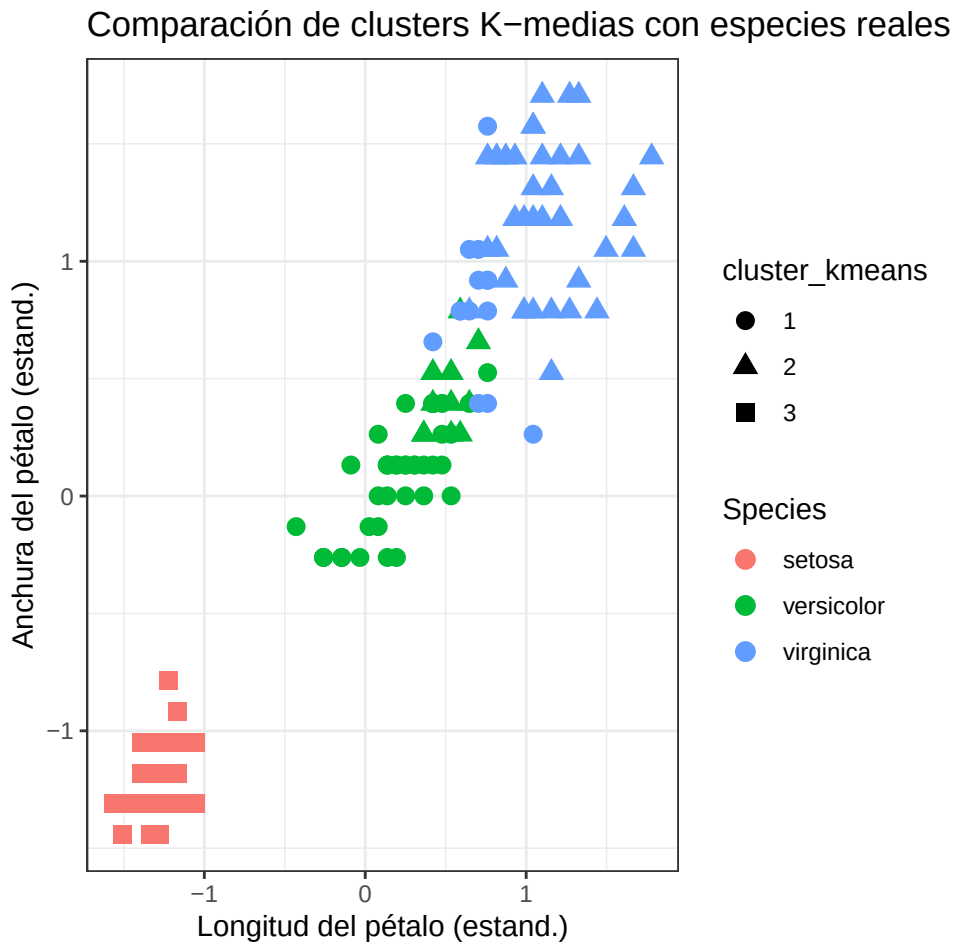
```
# Cálculo de precisión para clustering jerárquico
precision_jerar <- sum(diag(tabla_jerar)) / sum(tabla_jerar)
cat("Precisión de clustering jerárquico:", round(precision_jerar*100, 2), "%\n")
```

```
## Precisión de clustering jerárquico: 84 %
```

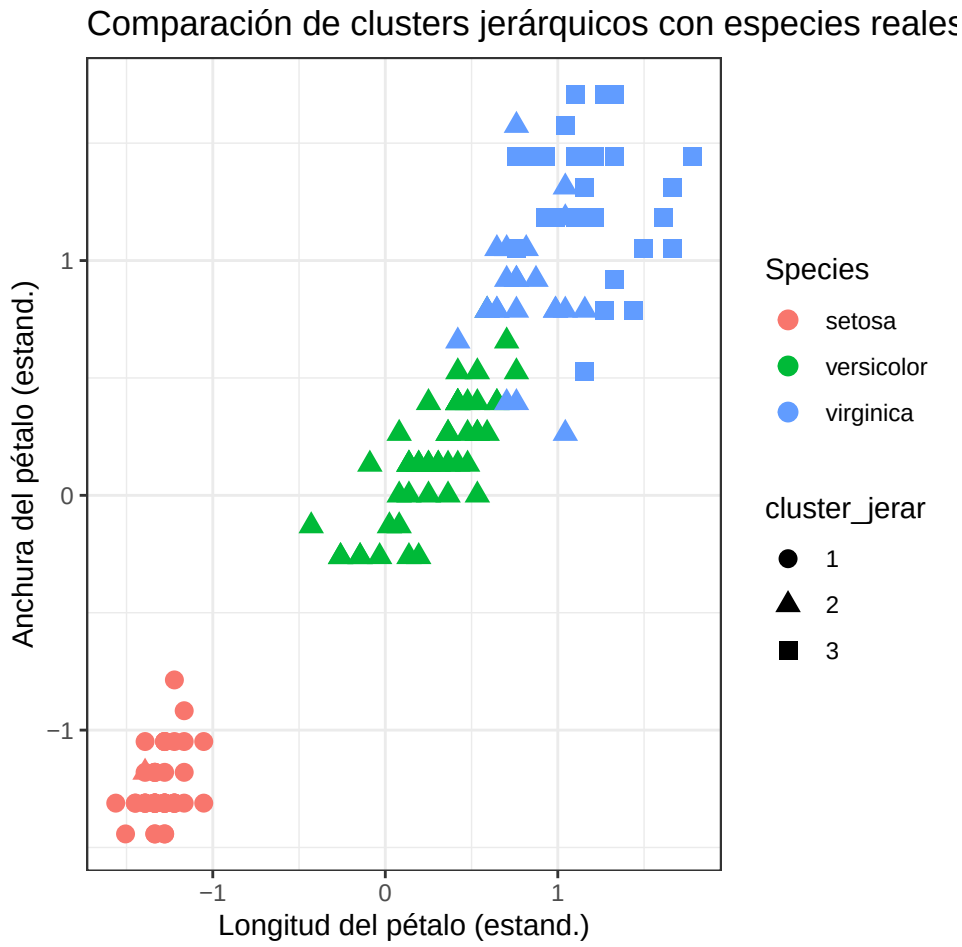
5.2. Visualización comparativa

```
# Visualización con las especies reales
ggplot(data_comparison, aes(x = Petal.Length, y = Petal.Width, color = Species, shape = cluster_kmeans))
  geom_point(size = 3) +
  labs(title = "Comparación de clusters K-medias con especies reales",
       x = "Longitud del pétalo (estand.)",
```

```
      y = "Anchura del pétalo (estand.)") +
  theme_bw()
```




```
ggplot(data_comparison, aes(x = Petal.Length, y = Petal.Width, color = Species, shape = cluster_jerar))
  geom_point(size = 3) +
  labs(title = "Comparación de clusters jerárquicos con especies reales",
       x = "Longitud del pétalo (estand.)",
       y = "Anchura del pétalo (estand.)") +
  theme_bw()
```

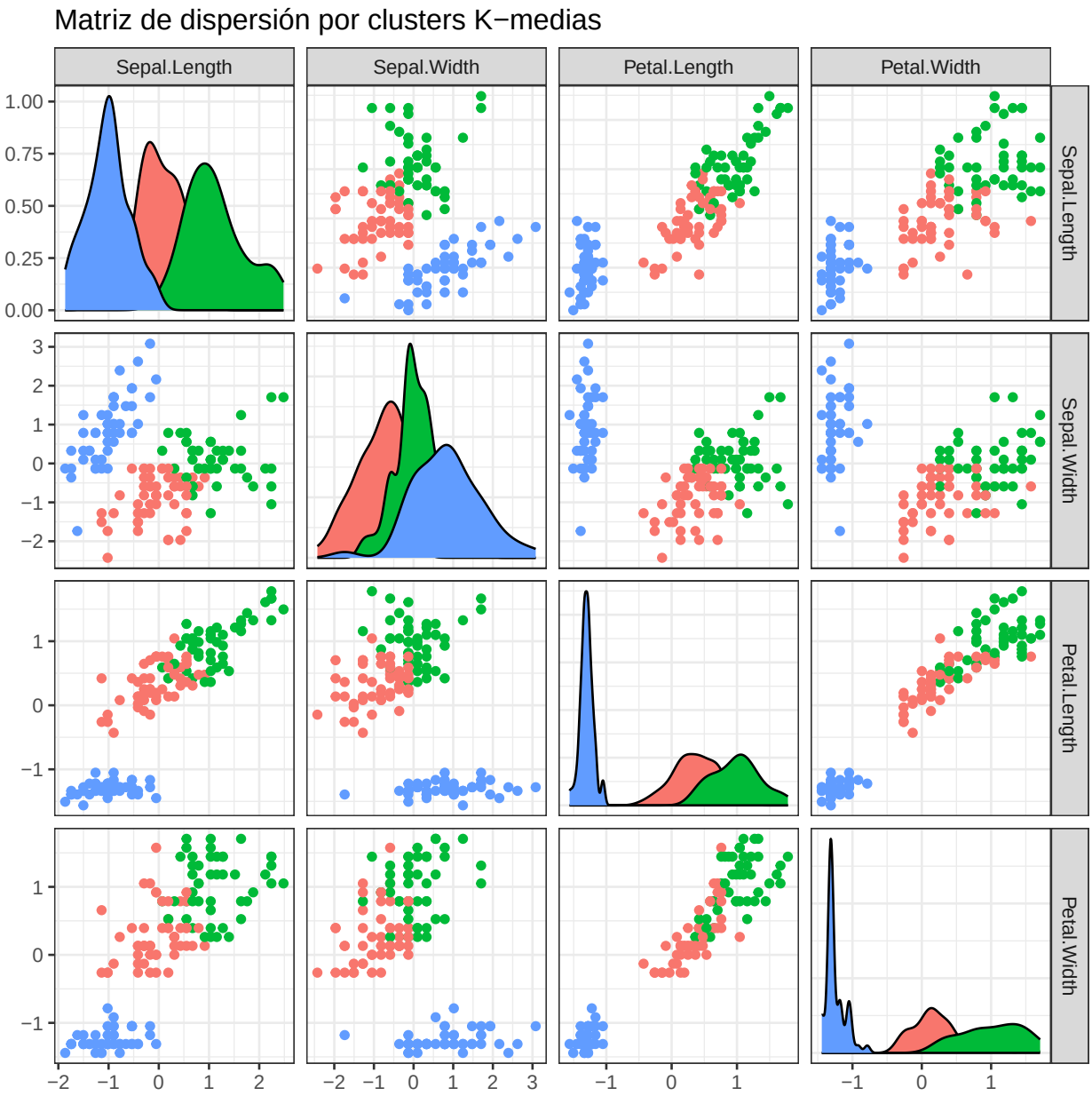


5.3. Análisis multivariante comparativo

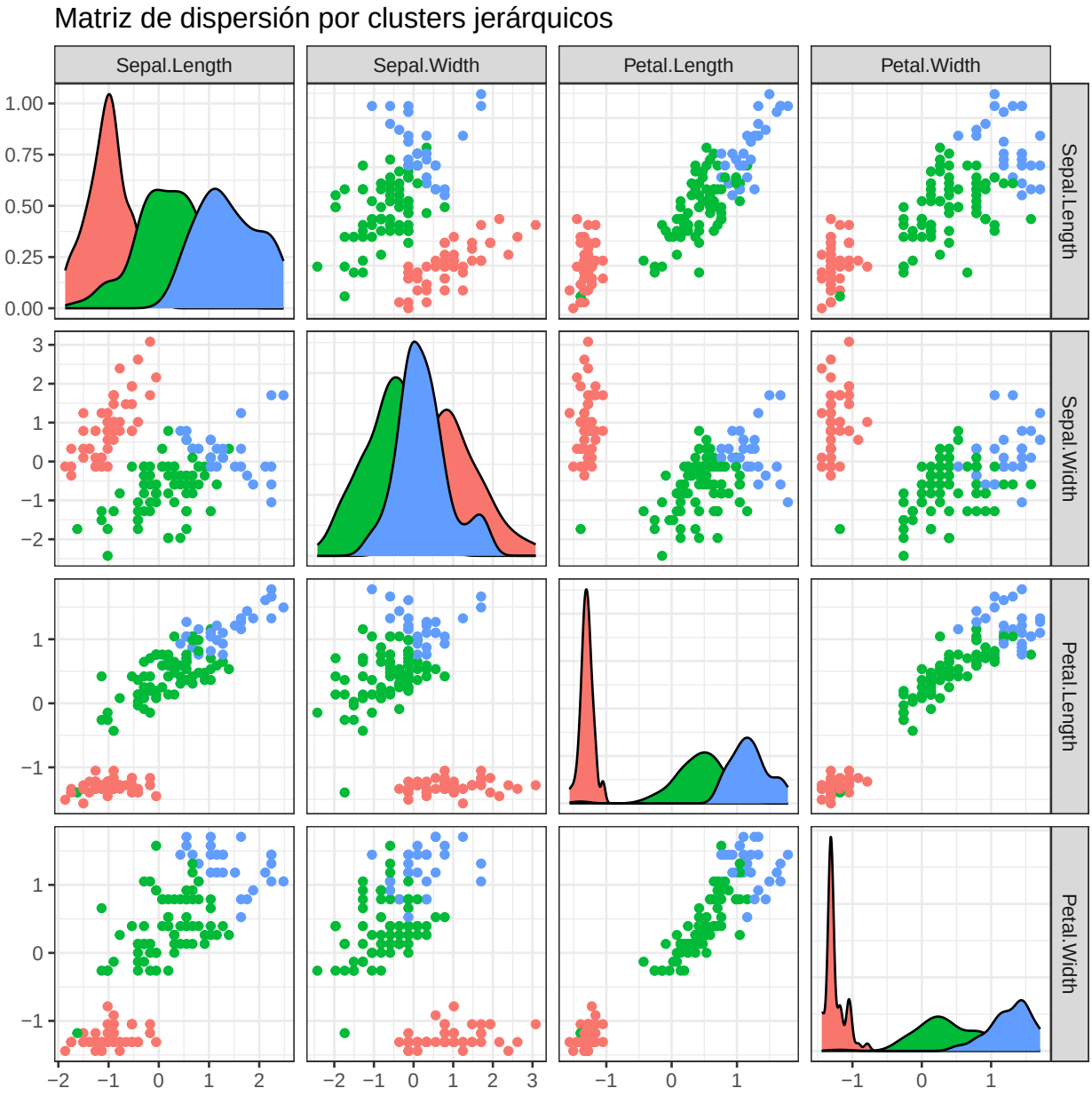
```
# Matriz de gráficos de dispersión coloreados por especie y clusters
library(GGally)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

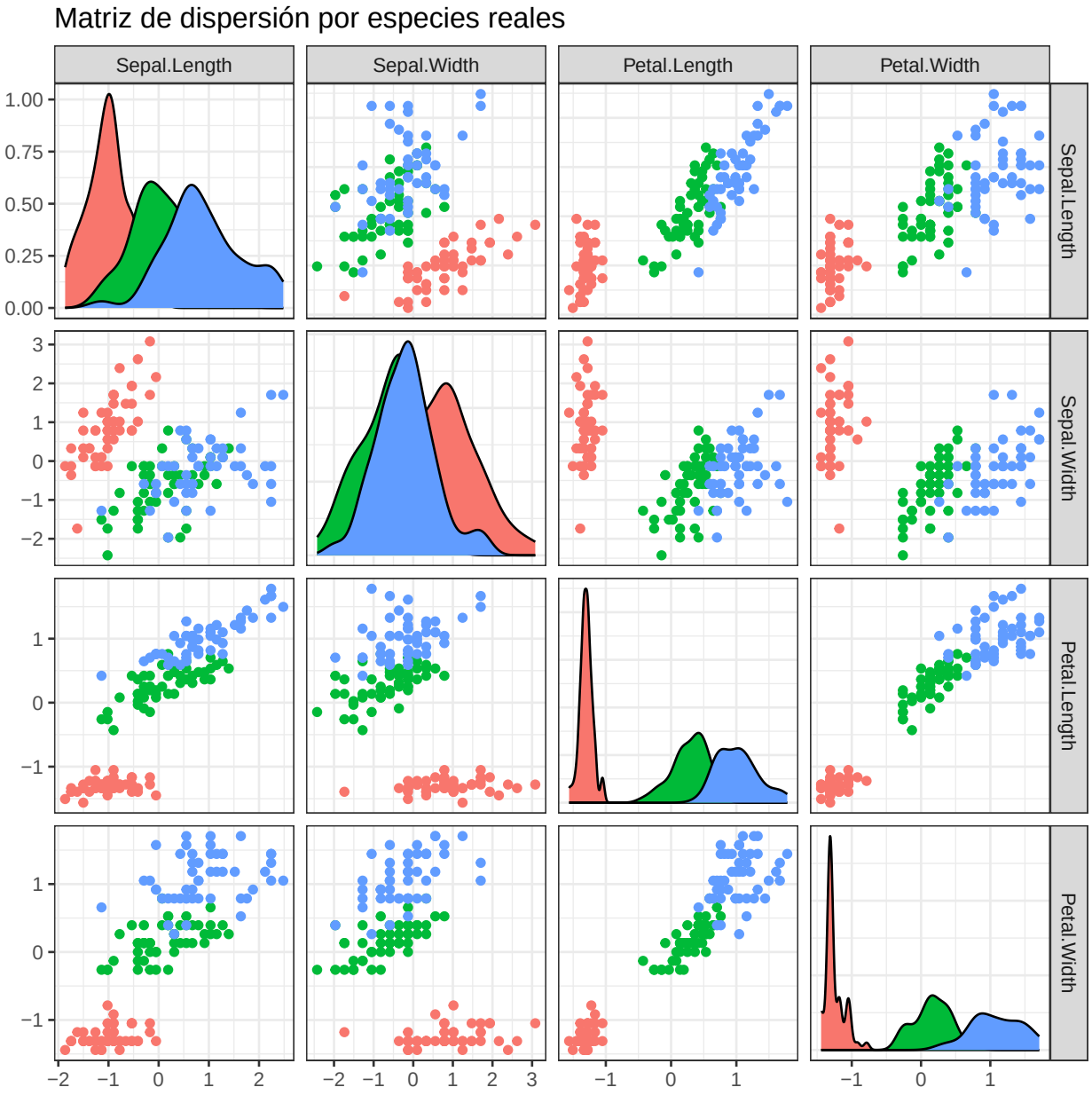
```
# Seleccionamos variables relevantes para la visualización
data_viz <- data_comparison %>%
  select(Sepal.Length, Sepal.Width, Petal.Length, Petal.Width, Species, cluster_kmeans, cluster_jerar)

# Matriz para clusters K-medias
ggpairs(data_viz,
        columns = 1:4,
        aes(color = cluster_kmeans),
        upper = list(continuous = "points"),
        diag = list(continuous = "densityDiag"),
        lower = list(continuous = "points")) +
  theme_bw() +
  labs(title = "Matriz de dispersión por clusters K-medias")
```




```
# Matriz para clusters jerárquicos
ggpairs(data_viz,
        columns = 1:4,
        aes(color = cluster_jerar),
        upper = list(continuous = "points"),
        diag = list(continuous = "densityDiag"),
        lower = list(continuous = "points")) +
  theme_bw() +
  labs(title = "Matriz de dispersión por clusters jerárquicos")
```




```
# Matriz por especies reales
ggpairs(data_viz,
        columns = 1:4,
        aes(color = Species),
        upper = list(continuous = "points"),
        diag = list(continuous = "densityDiag"),
        lower = list(continuous = "points")) +
  theme_bw() +
  labs(title = "Matriz de dispersión por especies reales")
```



6. Conclusiones

Tras aplicar dos técnicas diferentes de clustering al conjunto de datos Iris, se pueden extraer las siguientes conclusiones:

1. **Eficacia de las técnicas de clustering**: Tanto el agrupamiento jerárquico con el método de Ward como el algoritmo K-medias han demostrado ser eficaces para identificar las agrupaciones naturales en el conjunto de datos Iris, logrando una alta correspondencia con las especies biológicas conocidas.
2. **Comparación de métodos**:
   - El método de Ward en el agrupamiento jerárquico resultó ser superior al método de enlace completo, produciendo clusters más coherentes y mejor separados según el análisis de silueta.
   - El algoritmo K-medias y el agrupamiento jerárquico con Ward produjeron resultados muy similares, con una ligera ventaja para K-medias en términos de precisión global.

3. **Número óptimo de clusters**: Aunque el análisis de silueta sugirió inicialmente 2 clusters como número óptimo, la incorporación del conocimiento previo sobre la existencia de 3 especies y el análisis del método del codo confirmaron que 3 es el número más adecuado de clusters para este conjunto de datos.

4. **Características discriminatorias**: Las variables relacionadas con el pétalo (longitud y anchura) resultaron ser más discriminatorias que las variables del sépalo para la identificación de las especies de Iris, como se evidencia en las visualizaciones multivariantes.

5. **Implicaciones biológicas**: Los resultados del clustering sugieren que las características morfológicas medidas son suficientes para diferenciar las tres especies de Iris, con una separación particularmente clara de Iris setosa respecto a las otras dos especies, mientras que Iris versicolor e Iris virginica presentan cierto solapamiento.

6. **Limitaciones y trabajo futuro**:

    - Para mejorar aún más la separación entre Iris versicolor e Iris virginica, podrían considerarse técnicas más avanzadas o la inclusión de características morfológicas adicionales.
    - Un análisis más profundo de las contribuciones relativas de cada variable a la formación de clusters podría proporcionar insights adicionales sobre las características distintivas de cada especie.

En conclusión, este trabajo demuestra la utilidad de las técnicas de clustering para identificar patrones naturales en datos multivariantes y su aplicabilidad en contextos biológicos como la clasificación taxonómica. Los resultados obtenidos validan la separación morfológica de las tres especies de Iris estudiadas, mostrando la eficacia tanto del agrupamiento jerárquico como del algoritmo K-medias cuando se aplican con las medidas de disparidad y parámetros adecuados.