Laboratorio 8, Tópicos en análisis datos 1

Joshua Cervantes Artavia - Moisés Monge Cordonero

2023-11-03

```
knitr::opts_chunk$set(warning = FALSE, message = FALSE)
```

```
tryCatch(
    {
        # Directorio donde se ubica el qmd
        directory <- dirname(rstudioapi::getSourceEditorContext()$path)
        setwd(directory) # Establecer el directorio del archivo como la raiz
    },
    error = function(e) {
        message("")
        print("")
    }
)
```

```
[1] ""
```

```
source("cod/set_up.R")
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.2     v readr     2.1.4
v forcats   1.0.0     v stringr   1.5.0
v ggplot2   3.4.4     v tibble    3.2.1
v lubridate 1.9.2     v tidyr     1.3.0
v purrr     1.0.1
```

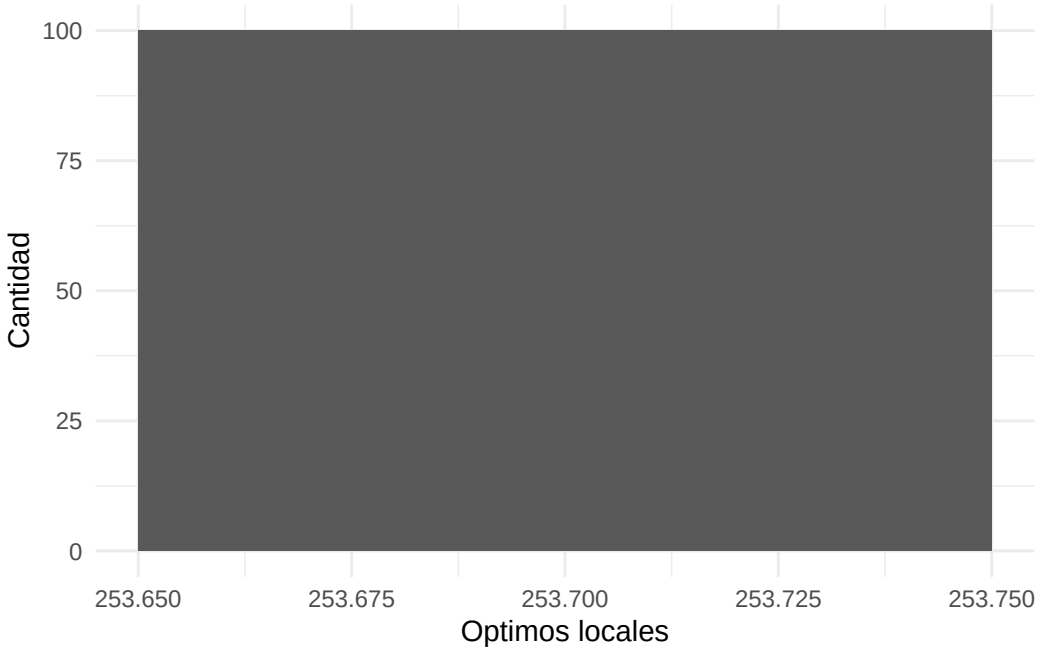
```
-- Conflicts --------------------------------------- tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
Loading required package: tictoc
```

1 Aplicaciones del método de k-means

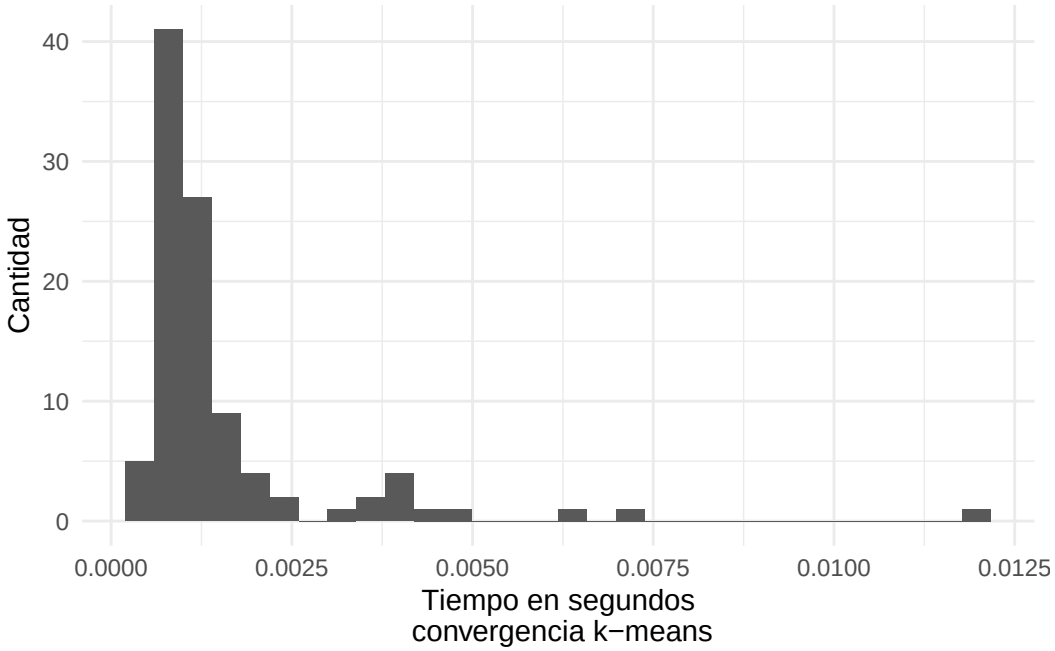
1.1 Notas escolares

```
# We read the excel with the data
df_notas_escolares <- read.xlsx("./data/Ejercicios-Cap3.xlsx", "9.NotasFrancesas")

# We make the name of rows the name of the studentes
rownames(df_notas_escolares) <- df_notas_escolares[, 1]

# We delete the firs column
df_notas_escolares <- df_notas_escolares[, -1]


# We estimate some of the point asked
notas_k_2 <- fn_punto_1(df = df_notas_escolares, k = 2)
notas_k_3 <- fn_punto_1(df = df_notas_escolares, k = 3)
notas_k_4 <- fn_punto_1(df = df_notas_escolares, k = 4)


# We print the summary asked for the point
notas_k_2$resumen
```

```
$plot_optimos
```


100
75
50
25
0
Cantidad
253.650
253.675
253.700
253.725
253.750
Optimos locales

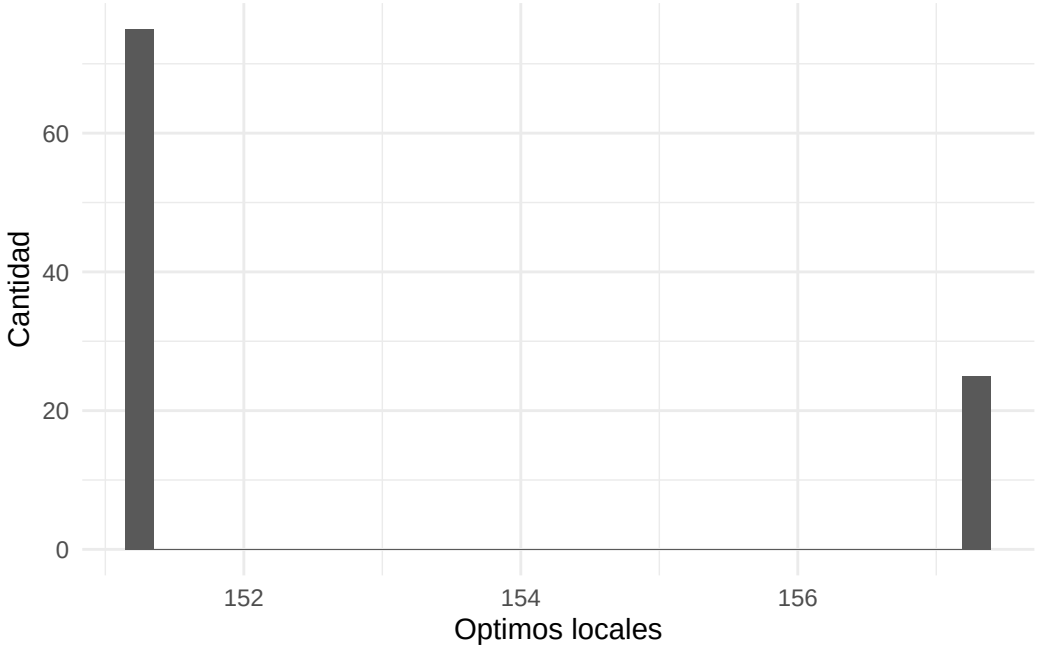
```
$plot_tiempo
```


40
30
20
10
0
Cantidad
0.0000
0.0025
0.0050
0.0075
0.0100
0.0125
Tiempo en segundos
convergencia k−means

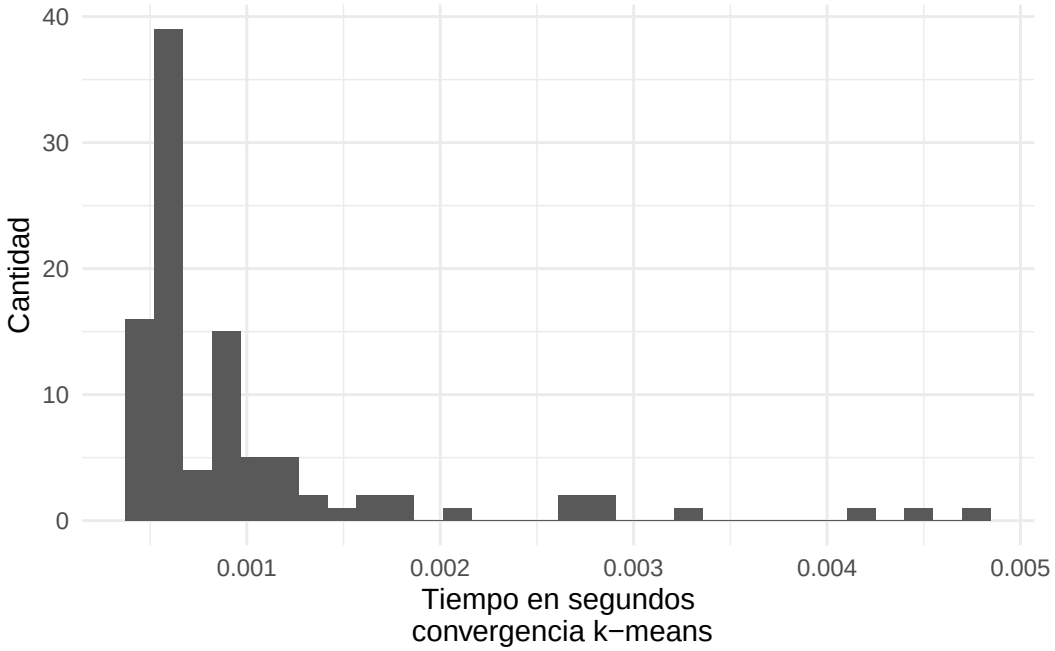
```
$optimo_promedio
[1] 253.7125

$mejor_optimo
[1] 253.7125

$atraccion_mejor_optimo
[1] 100
```

```
notas_k_3$resumen
```

```
$plot_optimos
```



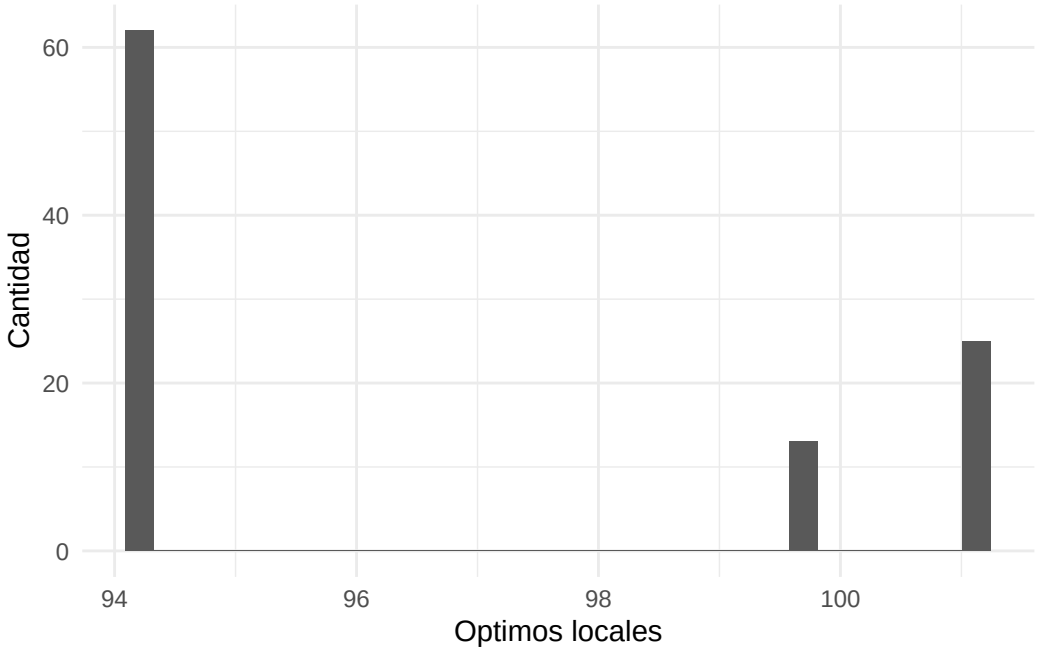

```
$plot_tiempo
```



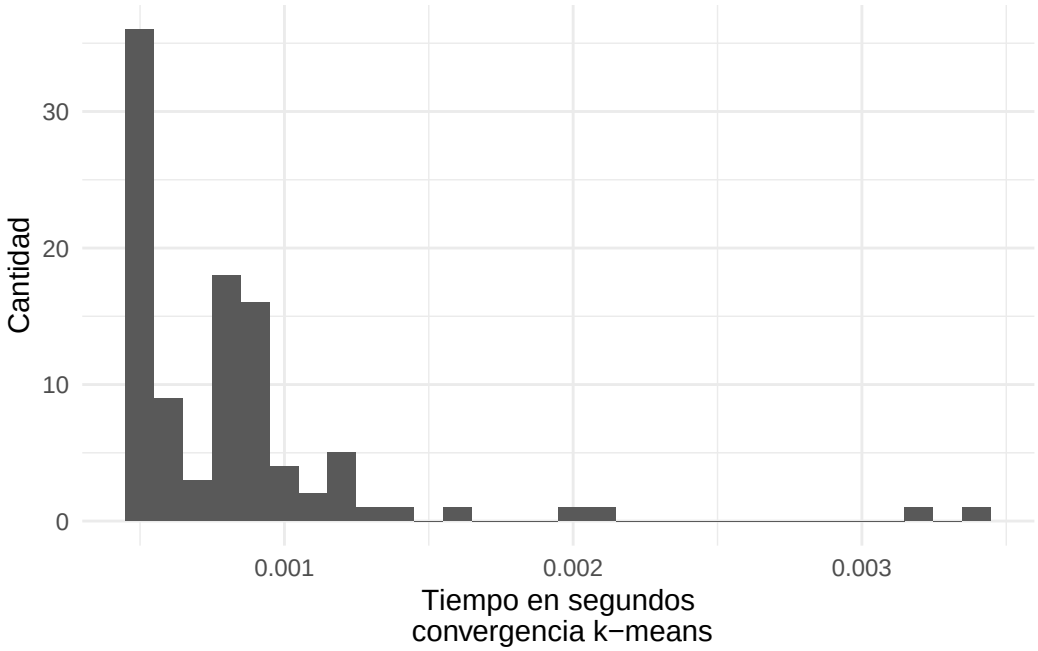

```
$optimo_promedio
[1] 152.8438

$mejor_optimo
[1] 151.3333

$atraccion_mejor_optimo
[1] 75
```

```
notas_k_4$resumen
```

```
$plot_optimos
```


60
40
20
0
Cantidad
94
96
98
100
Optimos locales

```
$plot_tiempo
```


30
20
10
0
Cantidad
0.001
0.002
0.003
Tiempo en segundos
convergencia k−means

```
$optimo_promedio
[1] 96.64167

$mejor_optimo
[1] 94.20833

$atraccion_mejor_optimo
[1] 62
```

En este caso como es de esperarse así como en los que siguen el mejor resultado se obtiene con 4 clusters. Entonces se reportan los resultados de este método de K means

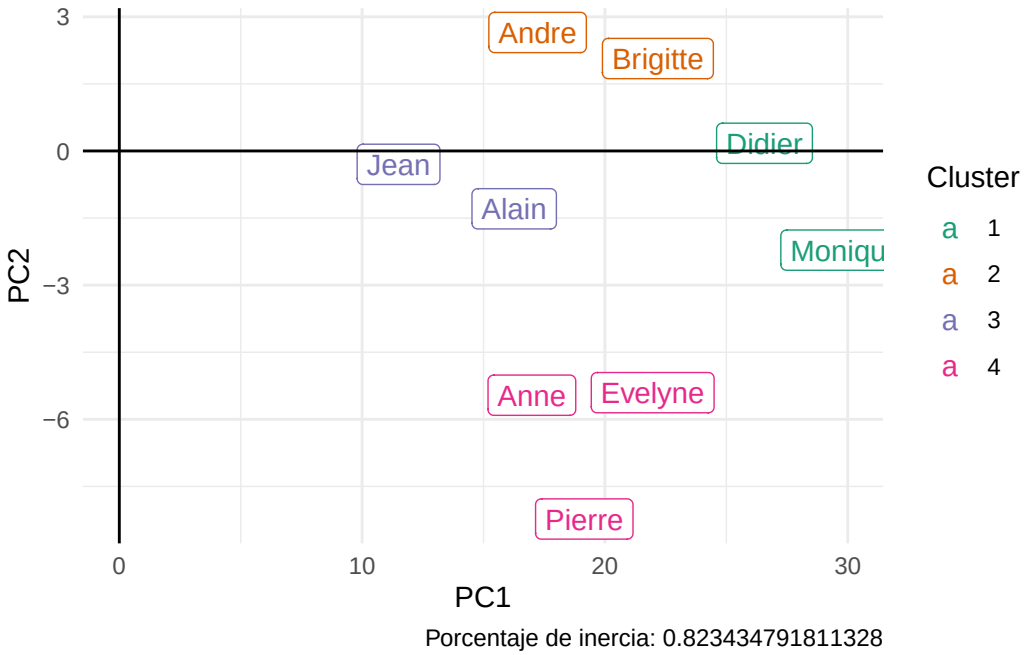
```
notas_k_4$informacion_general$mejor_km
```

```
K-means clustering with 4 clusters of sizes 2, 2, 2, 3

Cluster means:
       Mate    Fisica Frances Latin Deportes
1 14.250000 14.250000   13.75 13.75      9.0
2 12.000000 11.250000    7.00  8.25     12.5
3  7.000000  7.000000    6.50  6.75      8.5
4  6.833333  7.833333   12.50 11.00     13.0

Clustering vector:
    Jean    Alain     Anne  Monique   Didier    Andre   Pierre Brigitte
       3        3        4        1        1        2        4        2
 Evelyne
       4

Within cluster sum of squares by cluster:
[1] 11.50000 13.25000 12.12500 57.33333
 (between_SS / total_SS =  78.6 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
[6] "betweenss"    "size"         "iter"         "ifault"
```

En este caso se puede observar que el primer cluster es el presenta las nota más altas en general. Mientras que el segundo presenta las mejores notas para idiomas. El tercer cluster presenta buenas notas para mate y física, pero no para el resto, y el último cluster presenta las notas más bajas en términos generales. Además, se tiene que un 78.6% de la inercia es producto de las inercia interclase.

```
fn_clusters_km(notas_k_4, etiquetas = rownames(df_notas_escolares))
```

```
$plot_clusters
```



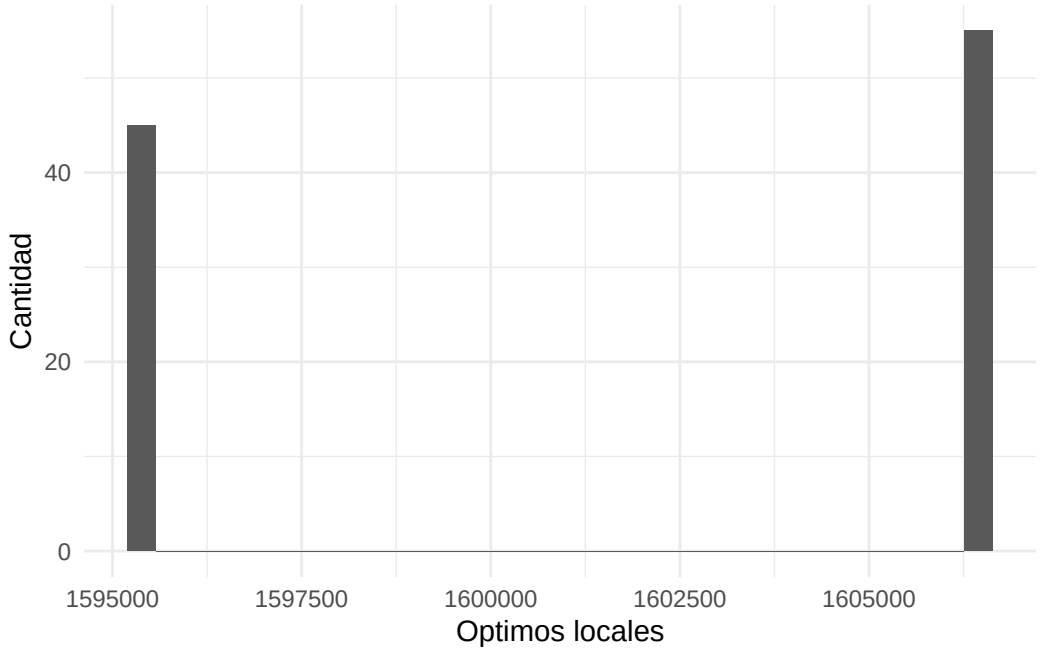
Mediante el ACP se puede observar una clara separación en la proyección sobre el plano principal.

1.2 Notas Amiard

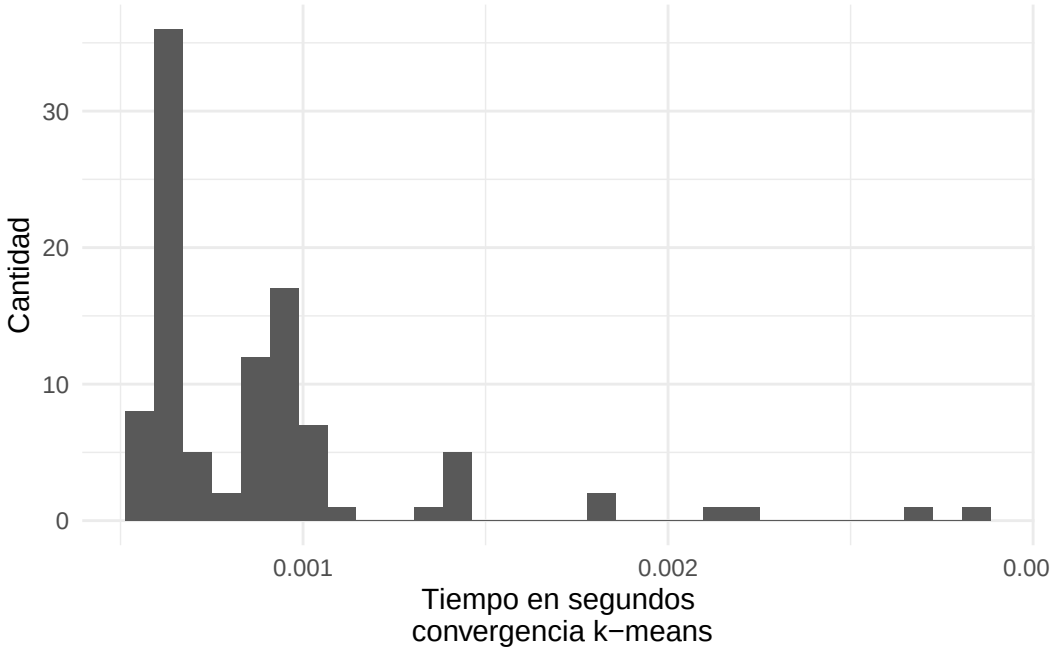
```
# We read the excel with the data
df_amiard <- read.xlsx("./data/Ejercicios-Cap3.xlsx", "10.Amiard")

# We make the name of rows the name of the studentes
rownames(df_amiard) <- df_amiard[, 1]

# We delete the firs column
df_amiard <- df_amiard[, -1]


# We estimate some of the point asked
amiard_k_2 <- fn_punto_1(df = df_amiard, k = 2)
```

```r
amiard_k_3 <- fn_punto_1(df = df_amiard, k = 3)
amiard_k_4 <- fn_punto_1(df = df_amiard, k = 4)

# We print the summary asked for the point
amiard_k_2$resumen
```

$plot_optimos



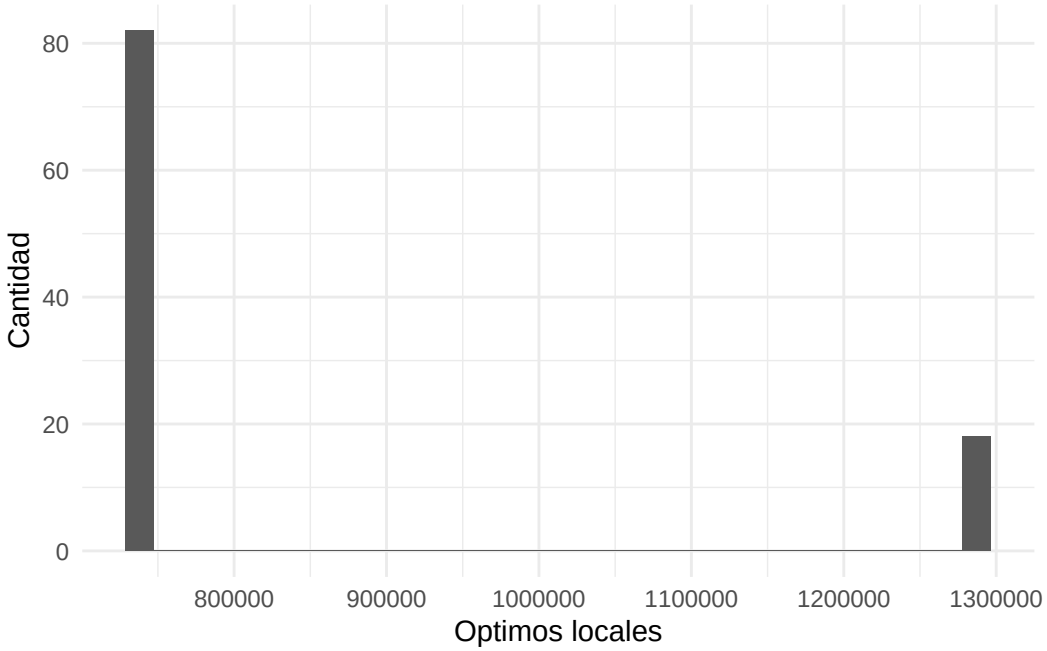
$plot_tiempo



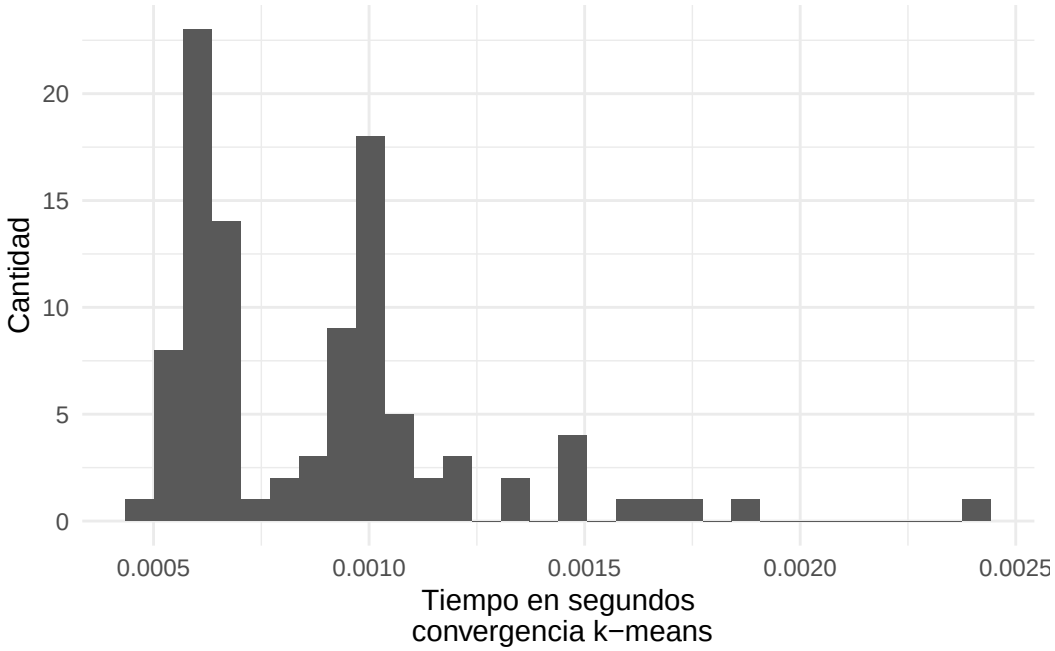

```
$optimo_promedio
[1] 1601554

$mejor_optimo
[1] 1595470

$atraccion_mejor_optimo
[1] 45
```

```
amiard_k_3$resumen
```

```
$plot_optimos
```


80
60
40
20
0
Cantidad
800000
900000
1000000
1100000
1200000
1300000
Optimos locales

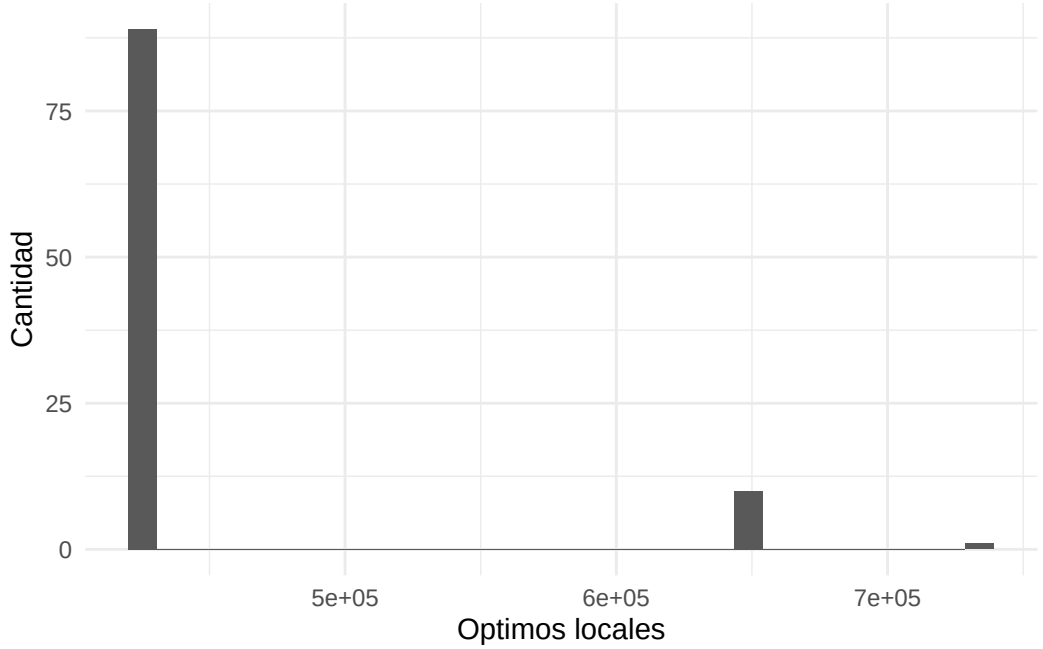
$plot_tiempo


20
15
10
5
0
Cantidad
0.0005
0.0010
0.0015
0.0020
0.0025
Tiempo en segundos
convergencia k−means

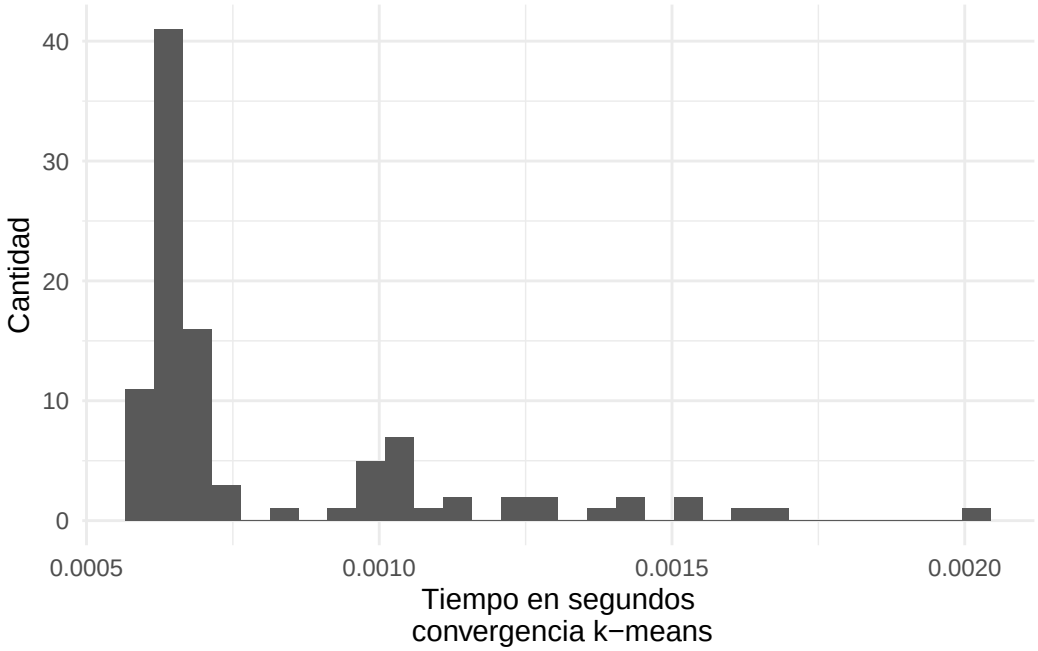
```
$optimo_promedio
[1] 839718.6

$mejor_optimo
[1] 740907.8

$atraccion_mejor_optimo
[1] 82
```

```
amiard_k_4$resumen
```

```
$plot_optimos
```




```
$plot_tiempo
```




```
$optimo_promedio
[1] 449560.5

$mejor_optimo
[1] 420471.9

$atraccion_mejor_optimo
[1] 27
```

```
amiard_k_4$informacion_general$mejor_km
```

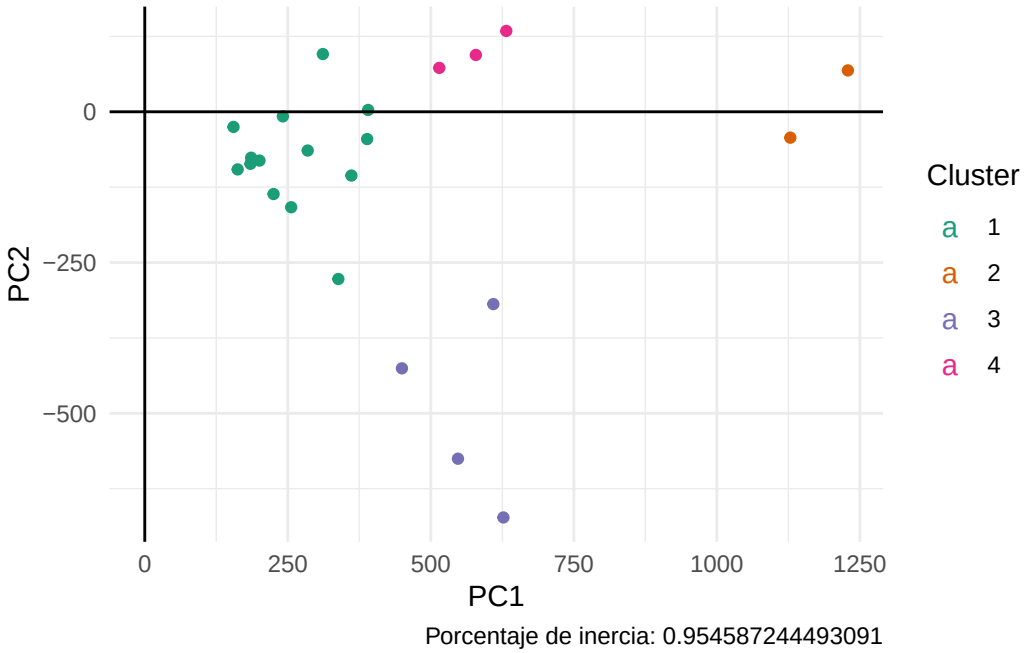
```
K-means clustering with 4 clusters of sizes 14, 2, 4, 3

Cluster means:
    RadOjo    RadBra    RadOpe   RadAle   RadHig   RadDig    RadRin   RadEsc
1 11.50000  76.28571  72.21429 120.7143 23.92857 168.7857  8.714286 178.2143
2 23.00000 161.00000 224.50000 313.0000 22.00000 957.5000 11.000000 574.0000
3 24.75000 174.75000 200.00000 267.7500 35.50000 156.7500  9.500000 644.2500
4 16.33333 108.33333  83.33333 135.0000 35.00000 524.0000  9.000000 209.3333
  RadMus Peso     Long   LonEst   AncCab    Ancho   AchHoc    DiaOjo
1    2.5 90.0 197.0000 176.2143 44.35714 40.42857 13.71429  9.928571
```

```
2    2.0 72.5 184.5000 164.0000 40.00000 37.00000 13.00000  9.500000
3    5.5 69.0 179.0000 163.0000 40.50000 38.00000 13.00000  9.000000
4    5.0 69.0 179.3333 159.6667 40.33333 38.33333 14.00000 10.000000

Clustering vector:
 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 18 19 20 21 22 23 24
 1  1  1  1  1  1  1  1  1  1  4  3  4  4  1  1  3  2  3  1  3  2  1

Within cluster sum of squares by cluster:
[1] 240448.36  11956.50 155851.75  12215.33
 (between_SS / total_SS =  85.2 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
[6] "betweenss"    "size"         "iter"         "ifault"
```

Se puede observar que entre más alto mayor es la media de algunas variables en este cluster. En este caso los clusters 2 y 3 presentan el RadHig y RadDis más altos. En el caso de RadEsc el más alto se tiene en el cluster 4.

```
fn_clusters_km(amiard_k_4)
```

```
$plot_clusters
```



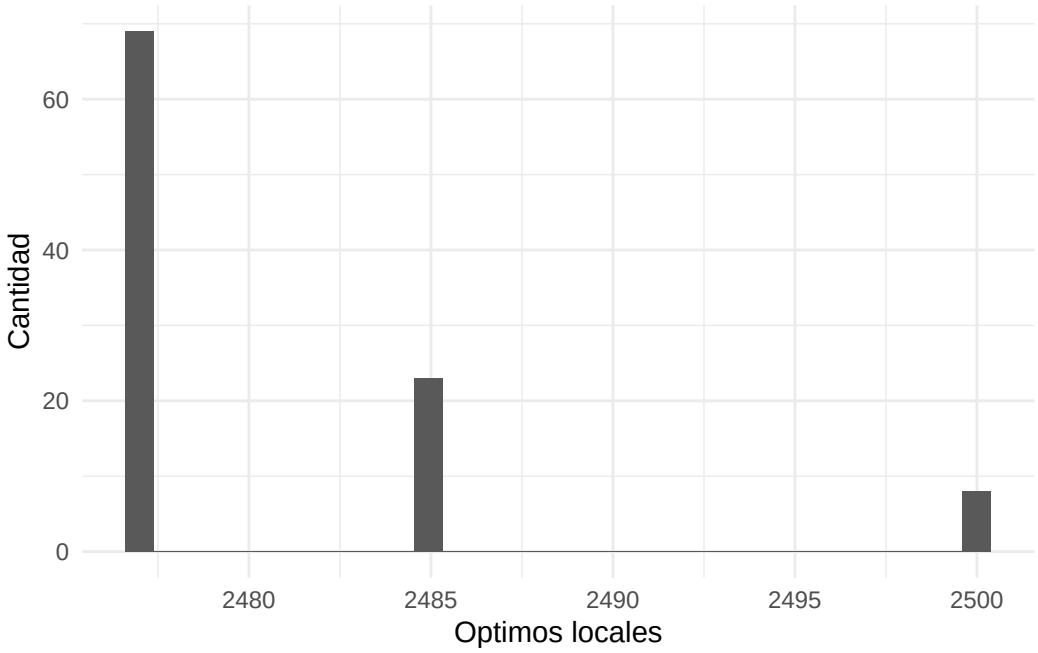
En este caso sí se puede observar una clara separación de los clusters, exceptuando elc aso de los clusters 2 y 3. Que estos presentaban promedios similares según el método de kmeans.

1.3 Notas proteinas

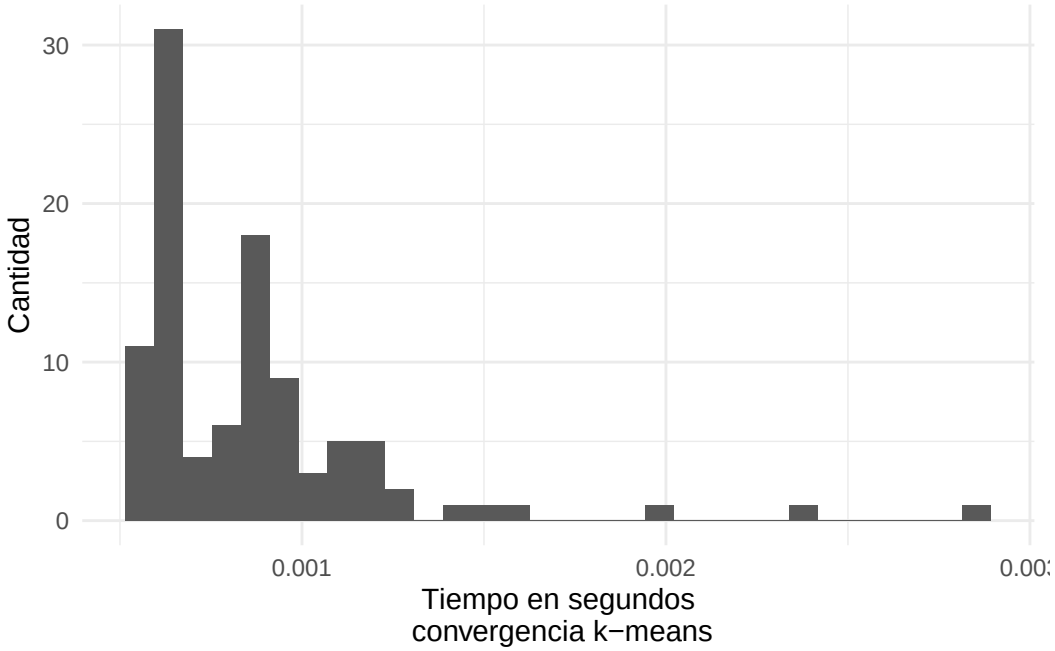
```
# We read the excel with the data
df_proteinas <- read.xlsx("./data/Ejercicios-Cap3.xlsx", "12.Proteinas")

# We make the name of rows the name of the studentes
rownames(df_proteinas) <- df_proteinas[, 1]

# We delete the firs column
df_proteinas <- df_proteinas[, -1]


# We estimate some of the point asked
proteinas_k_2 <- fn_punto_1(df = df_proteinas, k = 2)
proteinas_k_3 <- fn_punto_1(df = df_proteinas, k = 3)
proteinas_k_4 <- fn_punto_1(df = df_proteinas, k = 4)
```

```
# We print the summary asked for the point
proteinas_k_2$resumen
```

```
$plot_optimos
```



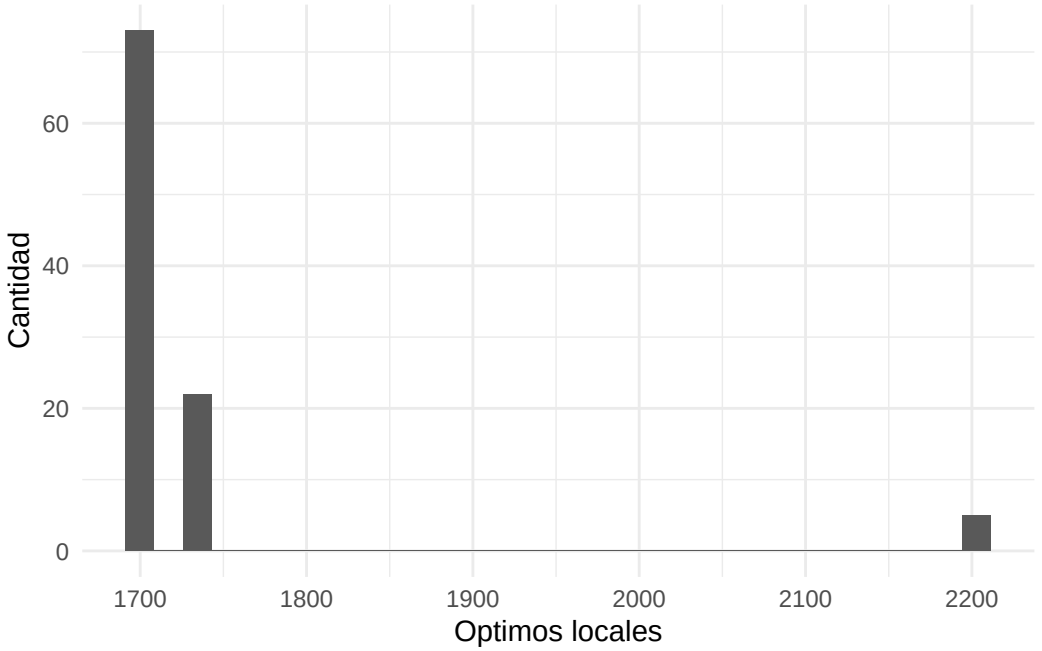

```
$plot_tiempo
```



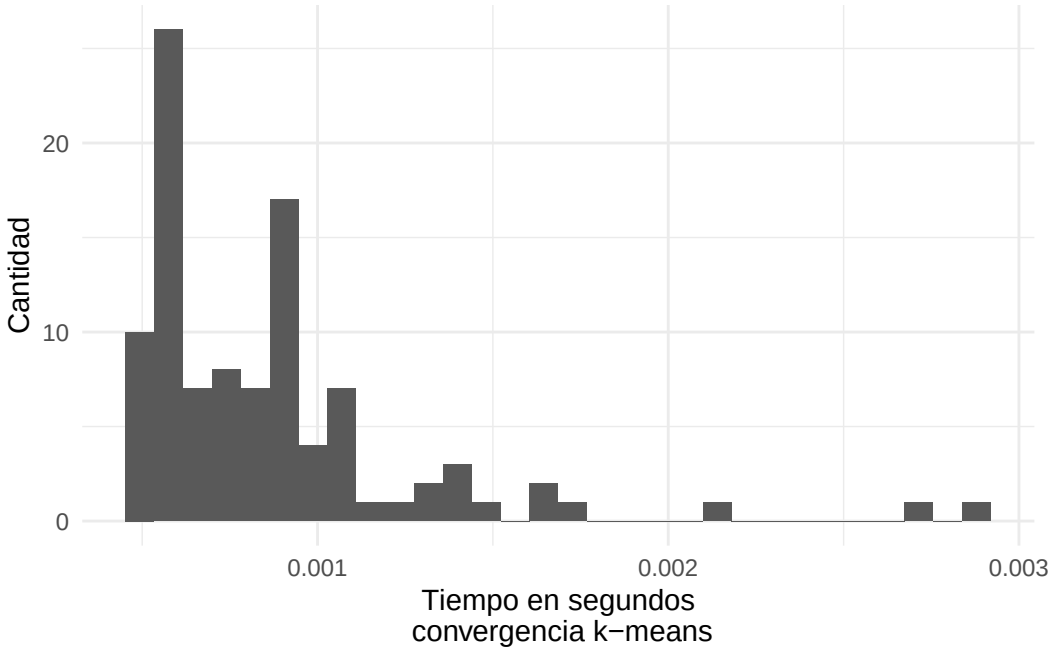

```
$optimo_promedio
[1] 2480.474

$mejor_optimo
[1] 2476.749

$atraccion_mejor_optimo
[1] 69
```

```
proteinas_k_3$resumen
```

```
$plot_optimos
```


60
40
20
0
Cantidad
1700
1800
1900
2000
2100
2200
Optimos locales

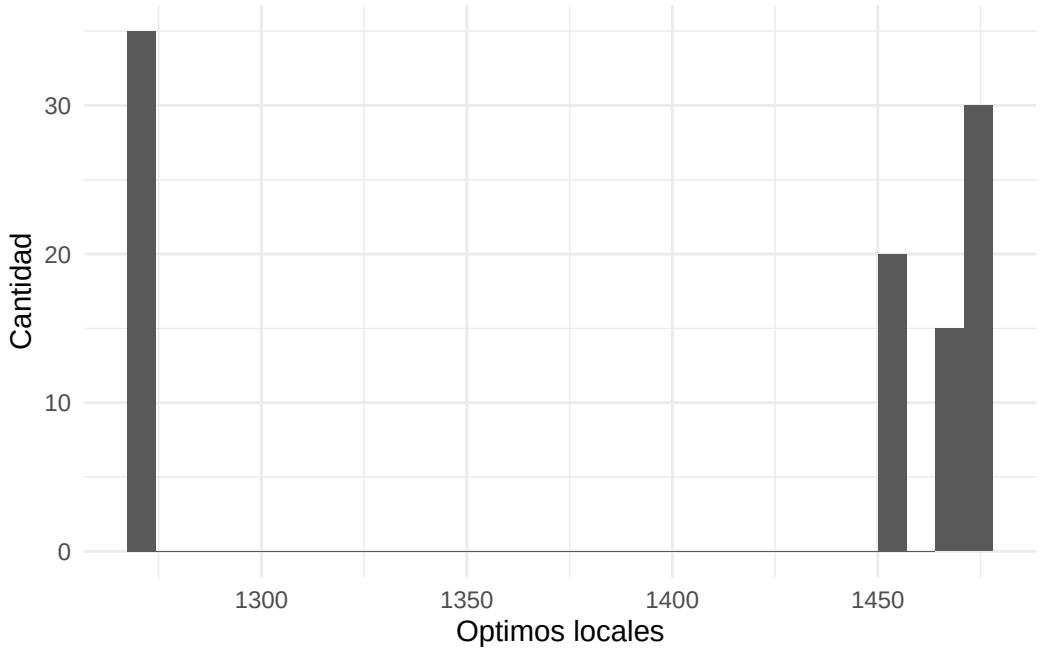
```
$plot_tiempo
```


20
10
0
Cantidad
0.001
0.002
0.003
Tiempo en segundos
convergencia k−means

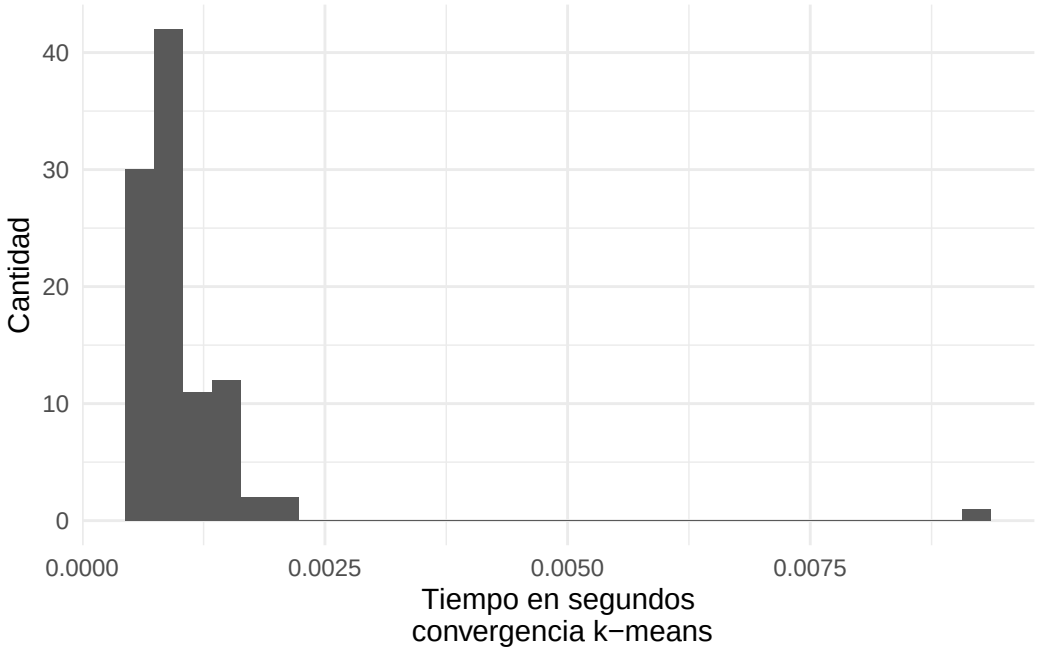
```
$optimo_promedio
[1] 1738.134

$mejor_optimo
[1] 1707.05

$atraccion_mejor_optimo
[1] 73
```

```
proteinas_k_4$resumen
```

```
$plot_optimos
```




```
$plot_tiempo
```




```
$optimo_promedio
[1] 1396.199

$mejor_optimo
[1] 1269.05

$atraccion_mejor_optimo
[1] 35
```

```
proteinas_k_4$informacion_general$mejor_km
```

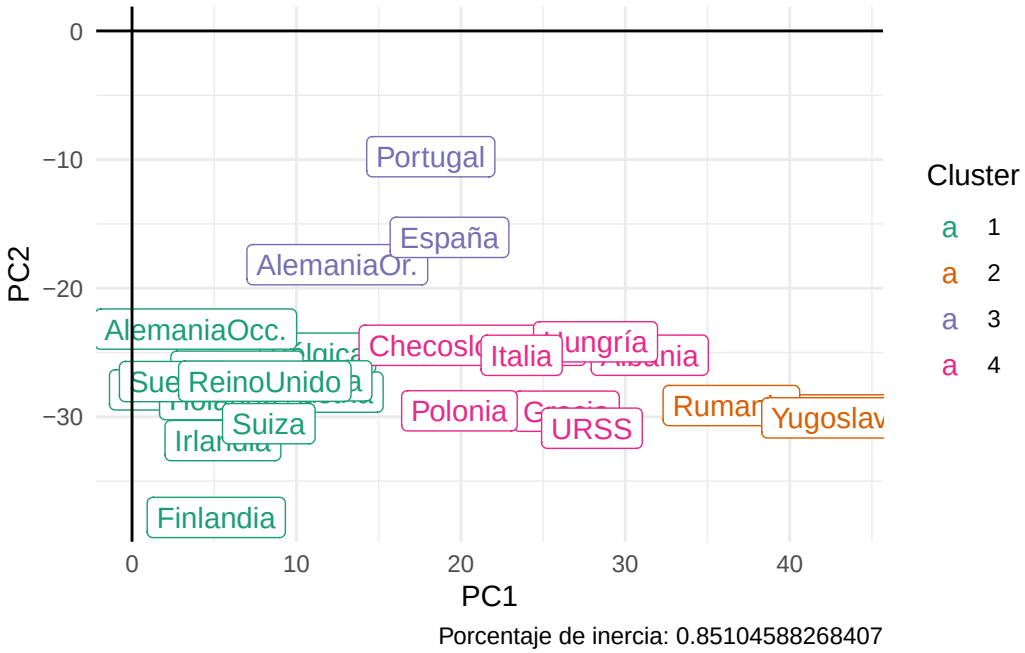
```
K-means clustering with 4 clusters of sizes 12, 3, 3, 7

Cluster means:
       RUMI     AVES     HUEV      LECH      PESC     CERE     ALMI     LEGU
1 12.091667 9.441667 3.708333 23.000000 4.9916667 24.02500 4.616667 1.766667
2  6.133333 5.766667 1.433333  9.633333 0.9333333 54.06667 2.400000 4.900000
3  7.233333 6.233333 2.633333  8.200000 8.8666667 26.93333 6.033333 3.800000
4  8.642857 6.871429 2.385714 14.042857 2.5428571 39.27143 3.742857 4.214286
      VERD
1 3.491667
```

```
2 3.400000
3 6.233333
4 4.657143

Clustering vector:
       Albania        Austria        Bélgica       Bulgaria Checoslovaquia
             4              1              1              2              4
     Dinamarca    AlemaniaOr.      Finlandia        Francia         Grecia
             1              3              1              1              4
       Hungría       Irlandia         Italia        Holanda        Noruega
             4              1              4              1              1
       Polonia       Portugal        Rumania         España         Suecia
             4              3              2              3              1
         Suiza     ReinoUnido           URSS   AlemaniaOcc.     Yugoslavia
             1              1              4              1              2

Within cluster sum of squares by cluster:
[1] 656.4517  47.0000 148.6067 416.9914
 (between_SS / total_SS =  75.8 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
[6] "betweenss"    "size"         "iter"         "ifault"
```

En este caso es interesante observar que la mayoría de los clusters presentan un promedio alto de cerde, pero el que presenta un valor más alto es del segundo cluster. El segundo valor más alto para el primer cluster es pesc y destaca sobre todos los demás clusters por este valor. En el caso del segundo cluster se encuentra que el más alto es cere notablemente por encima de todos los demás, seguido de lech y bajo en pesc. En el caso del tecer cluster este presenta valores superior a 1 para todas las clases por lo que es un cluster más balanceado en conjunto con el 4 que sí destacada en legu que no lo hace este anterior. Se puede observar que en cierta forma siempre se busca compensar las proteínas al no encontrarse en ningún caso un cluster de 0s.

```
fn_clusters_km(proteinas_k_4, etiquetas = rownames(df_proteinas))
```

```
$plot_clusters
```

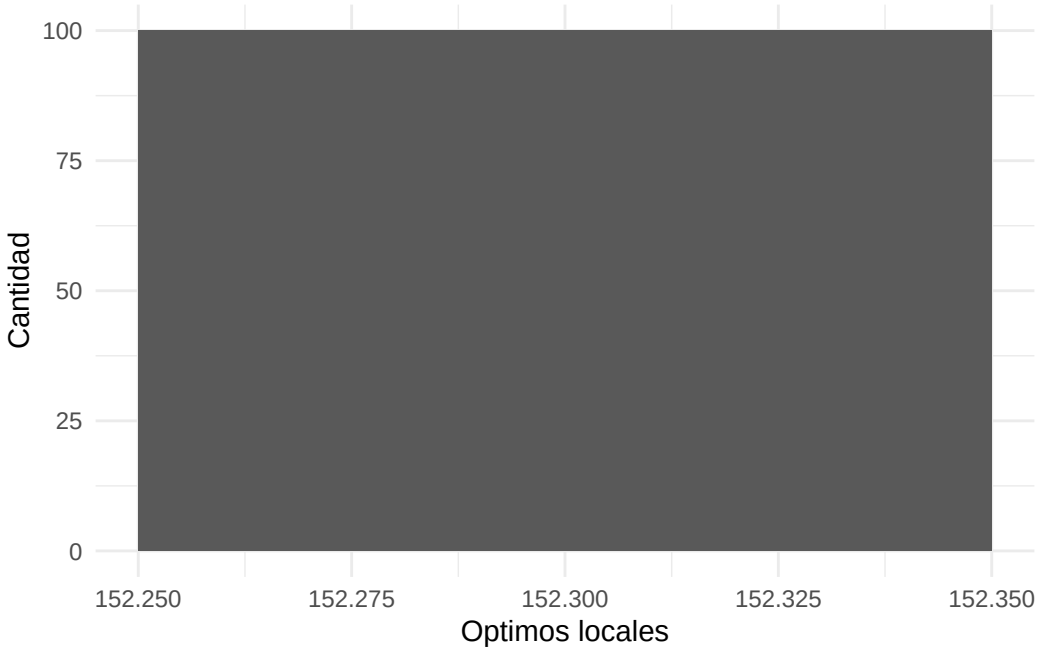


En este caso se observa una clara separación de los clusters.

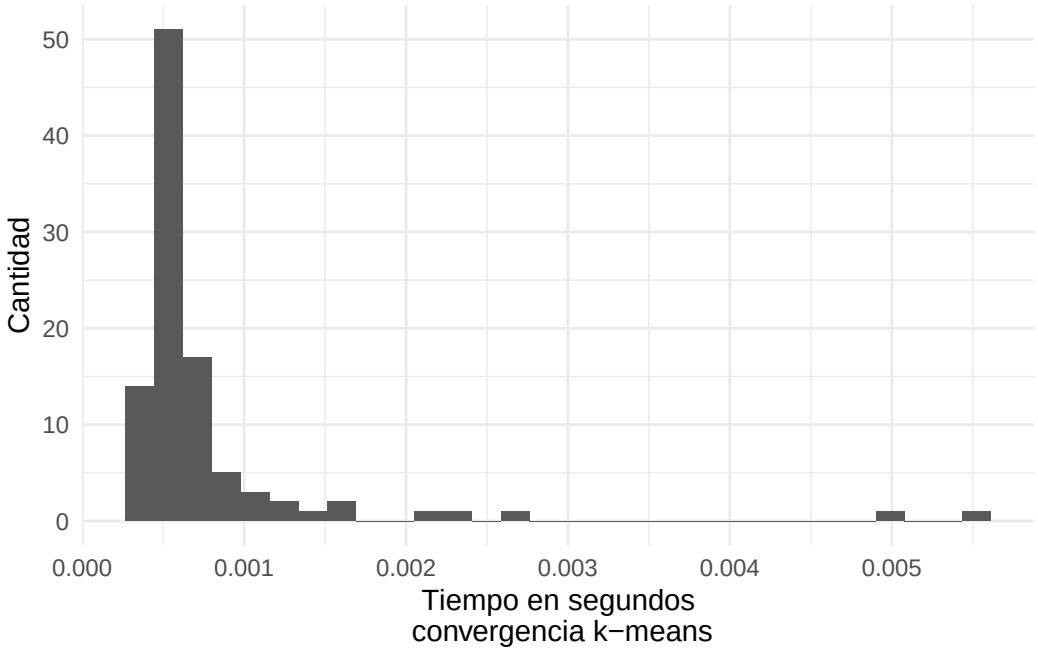
1.4 Iris

```
# We estimate some of the point asked
iris_k_2 <- fn_punto_1(df = iris[, -5], k = 2)
iris_k_3 <- fn_punto_1(df = iris[, -5], k = 3)
iris_k_4 <- fn_punto_1(df = iris[, -5], k = 4)


# We print the summary asked for the point
iris_k_2$resumen
```

```
$plot_optimos
```


100
75
50
25
0
Cantidad
152.250
152.275
152.300
152.325
152.350
Optimos locales

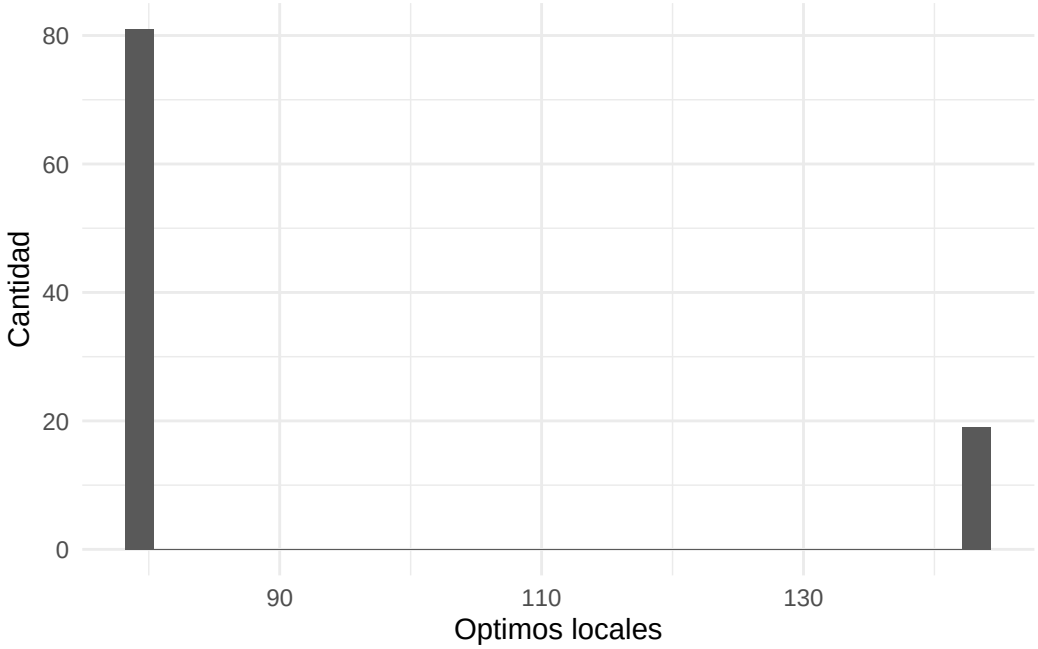
```
$plot_tiempo
```


50
40
30
20
10
0
Cantidad
0.000
0.001
0.002
0.003
0.004
0.005
Tiempo en segundos
convergencia k−means

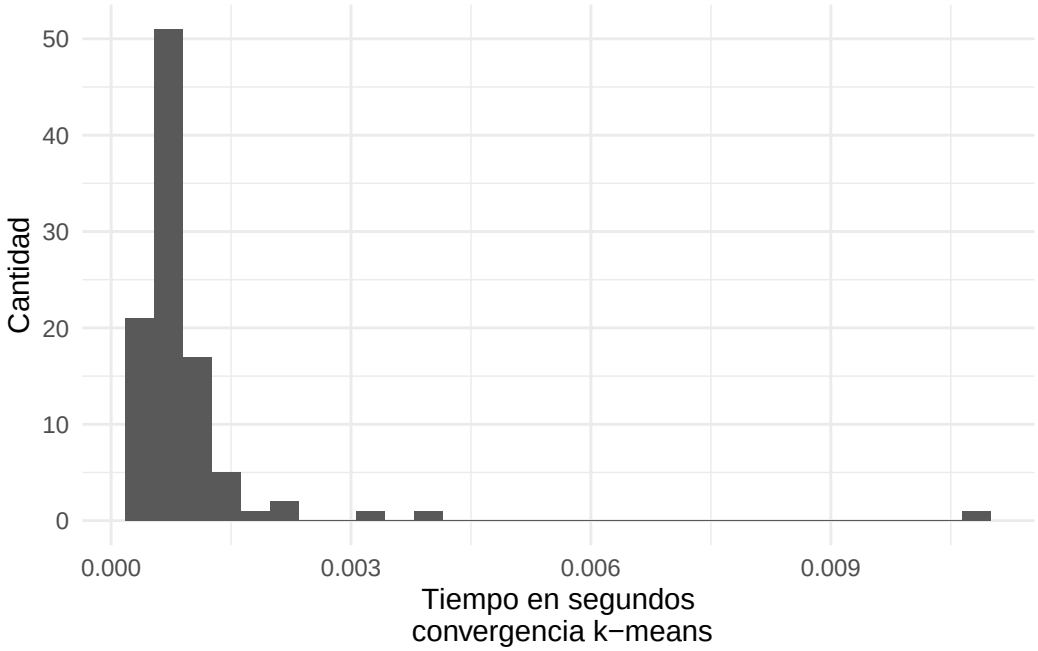
```
$optimo_promedio
[1] 152.348

$mejor_optimo
[1] 152.348

$atraccion_mejor_optimo
[1] 100
```

```
iris_k_3$resumen
```

```
$plot_optimos
```



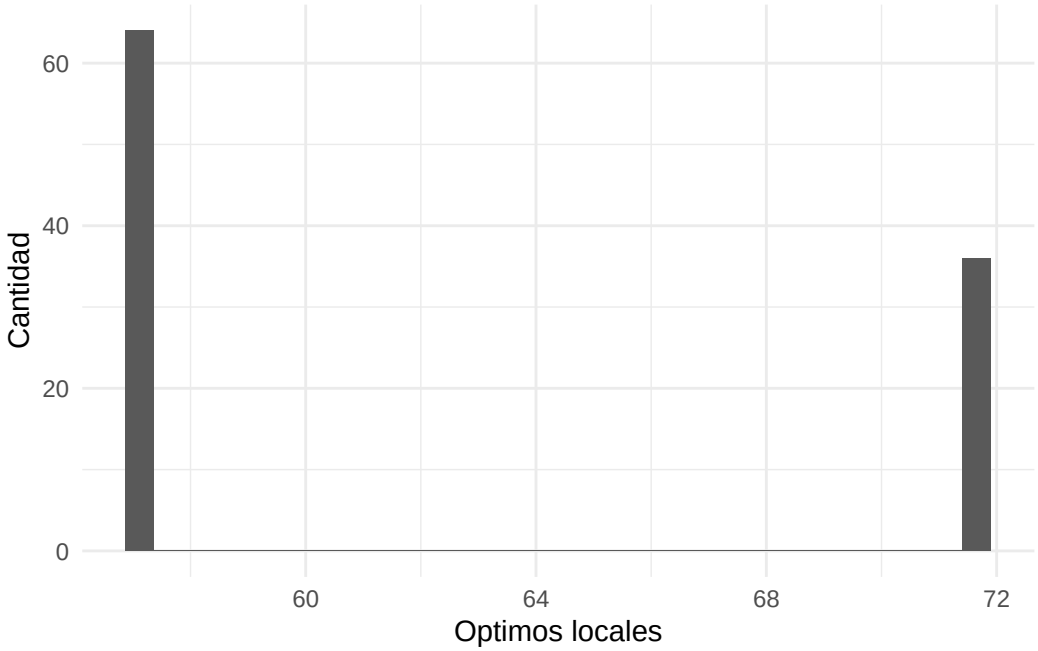

```
$plot_tiempo
```



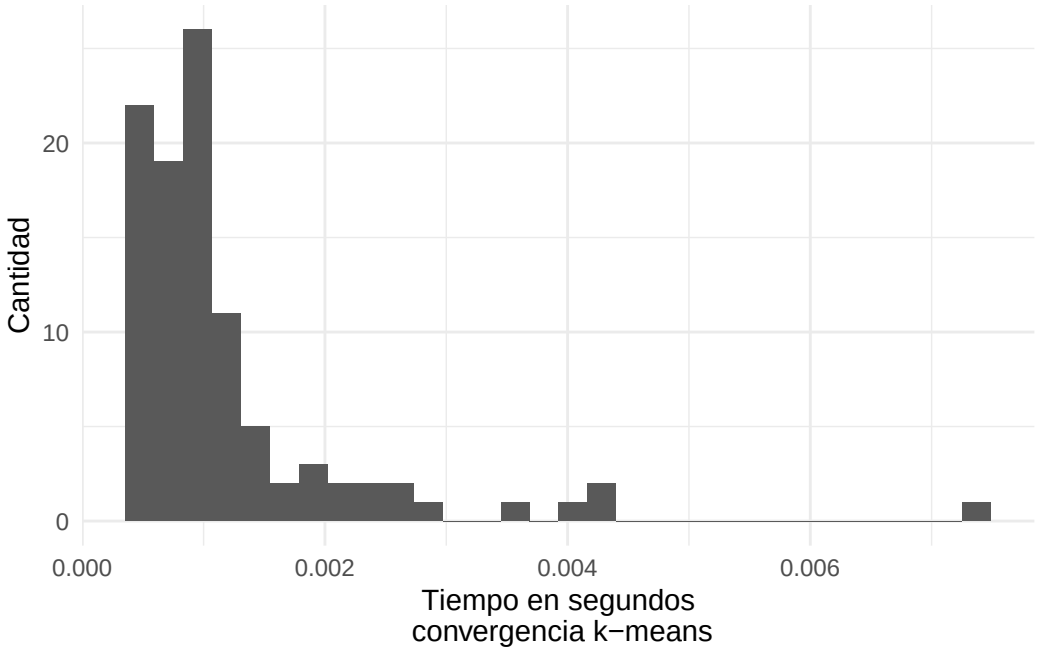

```
$optimo_promedio
[1] 90.99284

$mejor_optimo
[1] 78.85144

$atraccion_mejor_optimo
[1] 81
```

```
iris_k_4$resumen
```

```
$plot_optimos
```


60
40
20
0
Cantidad
60
64
68
72
Optimos locales

```
$plot_tiempo
```


20
10
0
Cantidad
0.000
0.002
0.004
0.006
Tiempo en segundos
convergencia k−means

```
$optimo_promedio
[1] 62.39483

$mejor_optimo
[1] 57.22847

$atraccion_mejor_optimo
[1] 27
```

```
iris_k_4$informacion_general$mejor_km
```

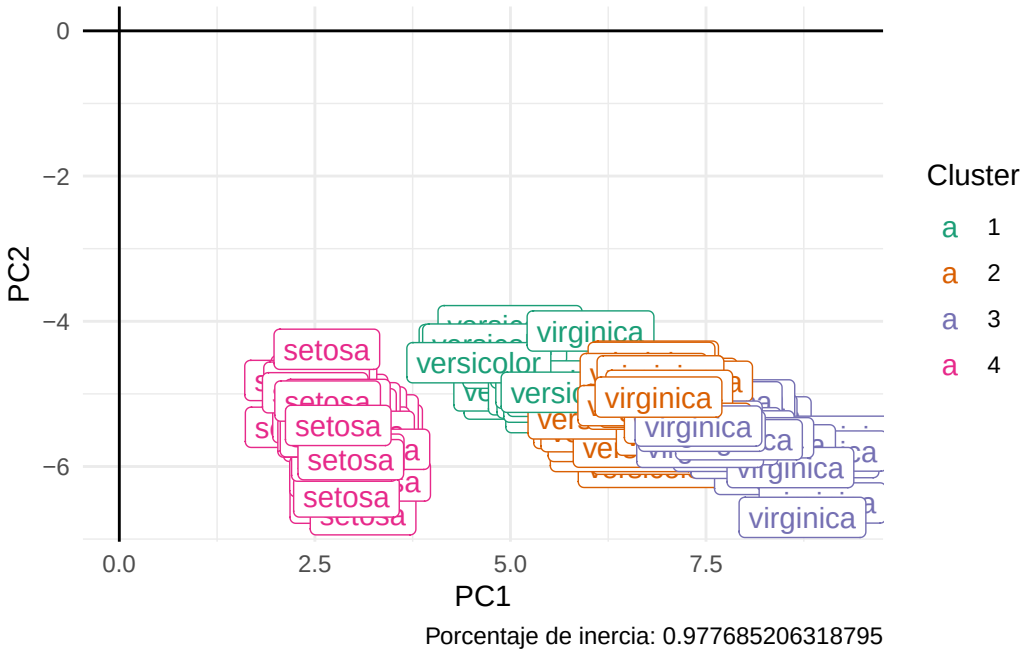
```
K-means clustering with 4 clusters of sizes 28, 40, 32, 50

Cluster means:
  Sepal.Length Sepal.Width Petal.Length Petal.Width
1     5.532143    2.635714     3.960714    1.228571
2     6.252500    2.855000     4.815000    1.625000
3     6.912500    3.100000     5.846875    2.131250
4     5.006000    3.428000     1.462000    0.246000

Clustering vector:
  [1] 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
 [38] 4 4 4 4 4 4 4 4 4 4 4 4 4 2 2 2 1 2 1 2 1 2 1 1 1 1 2 1 2 1 1 2 1 2 1 2 2
 [75] 2 2 2 2 2 1 1 1 1 2 1 2 2 2 1 1 1 2 1 1 1 1 1 2 1 1 3 2 3 3 3 3 1 3 3 3 2
[112] 2 3 2 2 3 3 3 3 2 3 2 3 2 3 3 2 2 3 3 3 3 3 2 2 3 3 3 2 3 3 3 2 3 3 3 2 2
[149] 3 2

Within cluster sum of squares by cluster:
[1]  9.749286 13.624750 18.703437 15.151000
 (between_SS / total_SS =  91.6 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
[6] "betweenss"    "size"         "iter"         "ifault"
```

Se puede observar que todos los clusters presentan un largo de sépalo alto. En el caso del cluster 1 se puede obsevar como el ancho del pétalos es realmente bajo. Mientras que en el tercer cluster se destaca un largo de pétalo mayor que en los demás clusters. En el caso del tercer cluster se pueden destacar valores más altos en general. En el caso del segundo se destacan valores más altos que los del

primer cluster en términos generales, pero menores que en el cuarto. El cuarto se encuentra detras del tercer cluster en términos generales.

```
fn_clusters_km(iris_k_4, etiquetas = iris[, 5])
```

```
$plot_clusters
```



Se puede observar una clara separación de los clusters, pero es interesante que en la proyección todos se encuentran relamente cercanos.