clustering

Samuel de Paúl

9/11/2021

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.4     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.0.2     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
## Linking to ImageMagick 6.9.12.3
## Enabled features: cairo, freetype, fftw, ghostscript, heic, lcms, pango, raw, rsvg, webp
## Disabled features: fontconfig, x11
```

```
##
## Attaching package: 'scales'
```

```
## The following object is masked from 'package:purrr':
##
##     discard
```

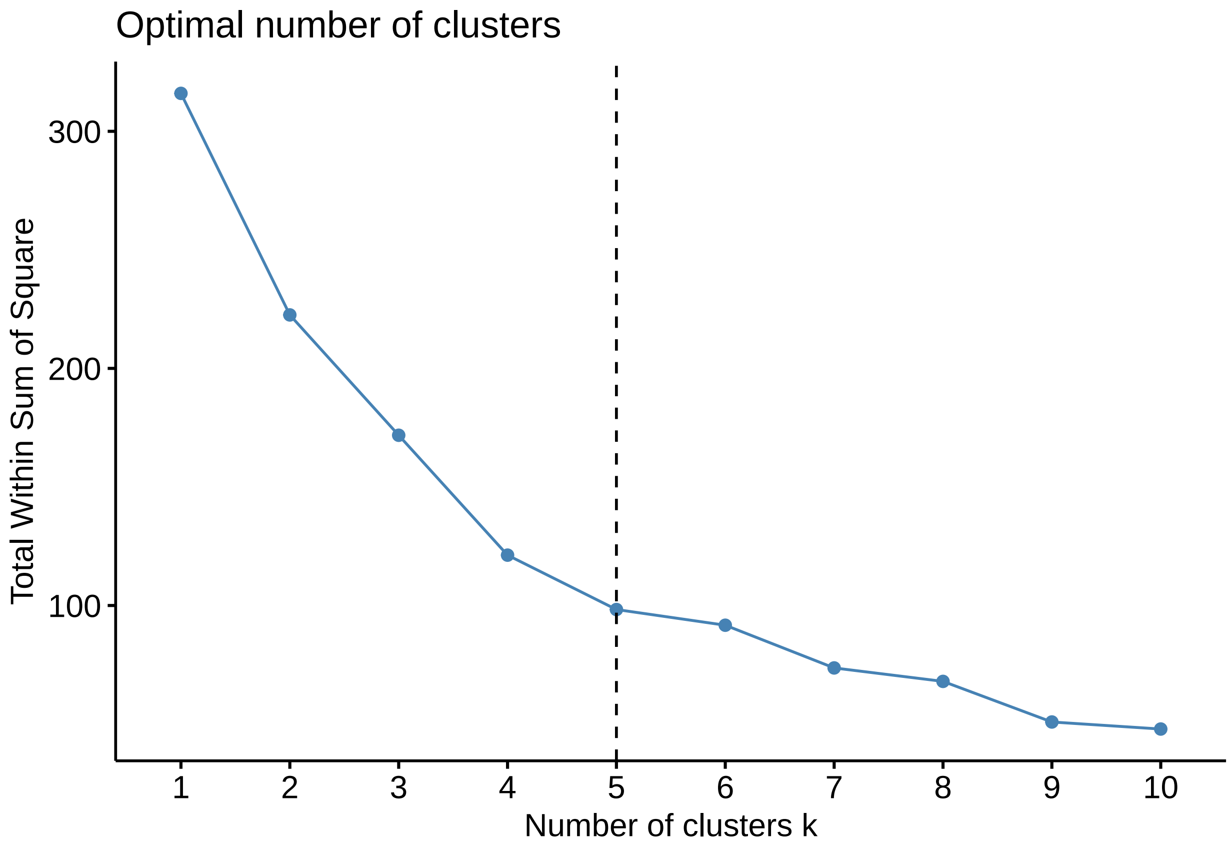
```
## The following object is masked from 'package:readr':
##
##     col_factor
```

En el dataset original las variables tienen nombres muy largos y engorrosos por lo que primero las renombraremos, para poder trabajar con ellas con mas facilidad, y a continuación detallaremos la información que aporta cada una.

```r
#Clustering

set.seed(101)
#Omitimos el uso de la variable levodopa porque solo toma el valor 0
datos = data[c("onset", "duration", "clonazepam", "UPDRSIII")]
datos <- datos[1:80,]

datos$onset = as.numeric(datos$onset)
datos$duration = as.numeric(datos$duration)
datos$clonazepam = as.numeric(datos$clonazepam)
datos$UPDRSIII = as.numeric(datos$UPDRSIII)

datos1 <- datos
datos <- scale(datos)

#install.packages("factoextra")
library(factoextra)
```

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

```
fviz_nbclust(x = datos, FUNcluster = kmeans, method = "wss",
 diss = dist(datos, method = "euclidean")) +
 geom_vline(xintercept = 5, linetype = 2)
```

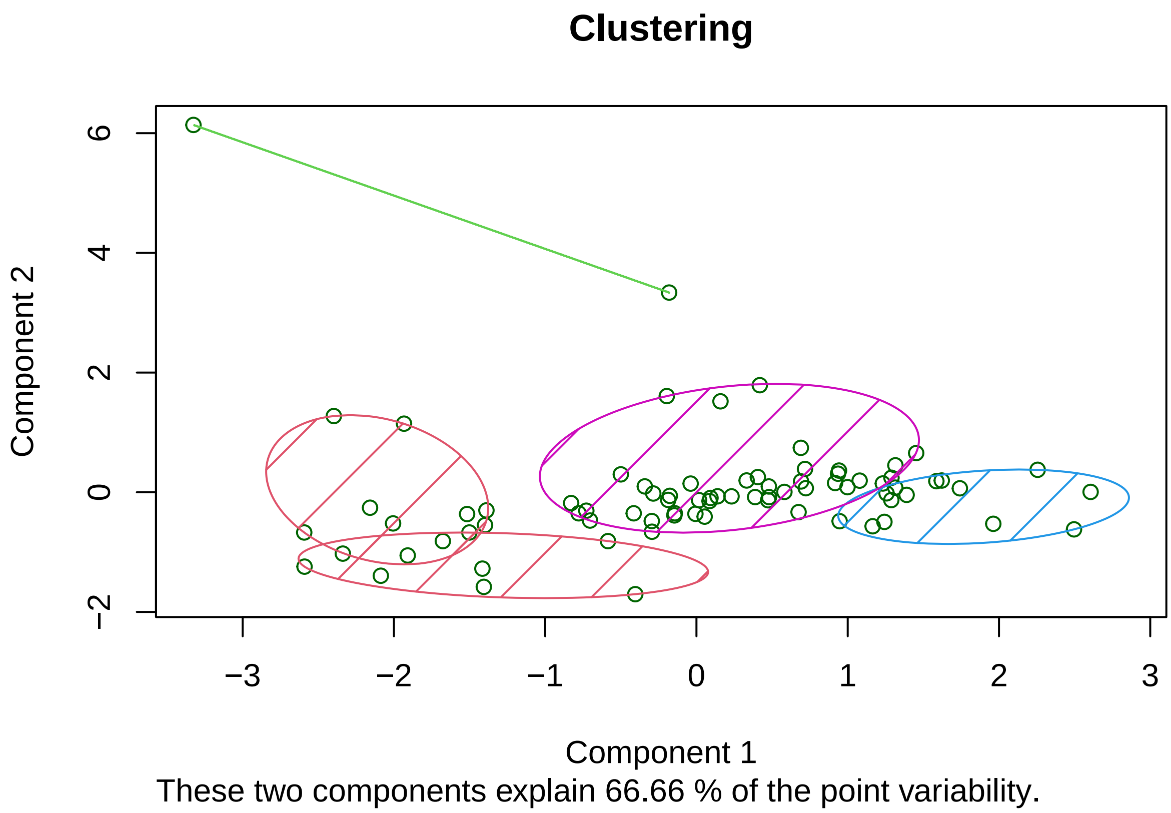


```
km_clusters <- kmeans(x = datos, centers = 5, nstart = 50)
km_clusters
```

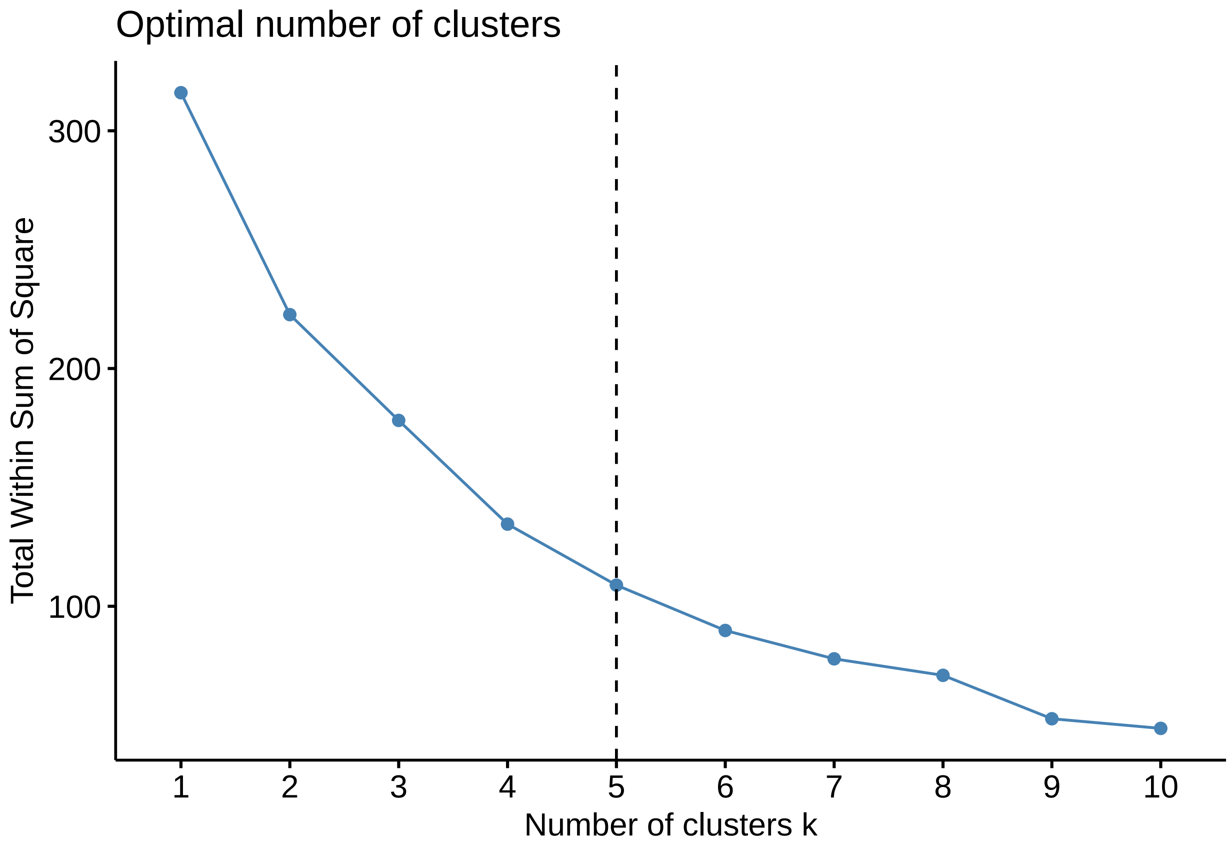
```
## K-means clustering with 5 clusters of sizes 8, 10, 14, 46, 2
##
## Cluster means:
##        onset   duration clonazepam   UPDRSIII
## 1 -0.4195348  2.2880333  0.1949255 -0.3356984
## 2 -2.0184286  0.6374467 -0.2637227 -0.3015596
## 3  0.5160944 -0.7034350 -0.2637227  1.7142565
## 4  0.3496123 -0.3415515 -0.1241341 -0.3767640
## 5  0.1165374  0.4403617  5.2400553 -0.4836333
##
## Clustering vector:
##  [1] 4 3 3 3 3 3 3 3 4 3 4 2 2 3 3 4 3 4 2 4 4 4 4 3 4 5 3 4 3 4 4 2 1 4 1 4 4 4
## [39] 4 4 4 4 4 4 4 4 4 4 4 4 4 4 2 4 4 4 2 2 4 4 4 1 4 4 4 5 1 1 1 4 1 4 4 1 2 4
## [77] 4 4 2 2
##
## Within cluster sum of squares by cluster:
## [1] 11.78540 16.09594 16.75205 43.42744 10.23792
##  (between_SS / total_SS =  68.9 %)
```

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

```
#Visualizaci?n de los clusters
#install.packages("cluster")
library(cluster)
clusplot(datos, km_clusters$cluster, lines = 0, shade = TRUE, color = TRUE, labels = 1, plotchar = FALSE
```



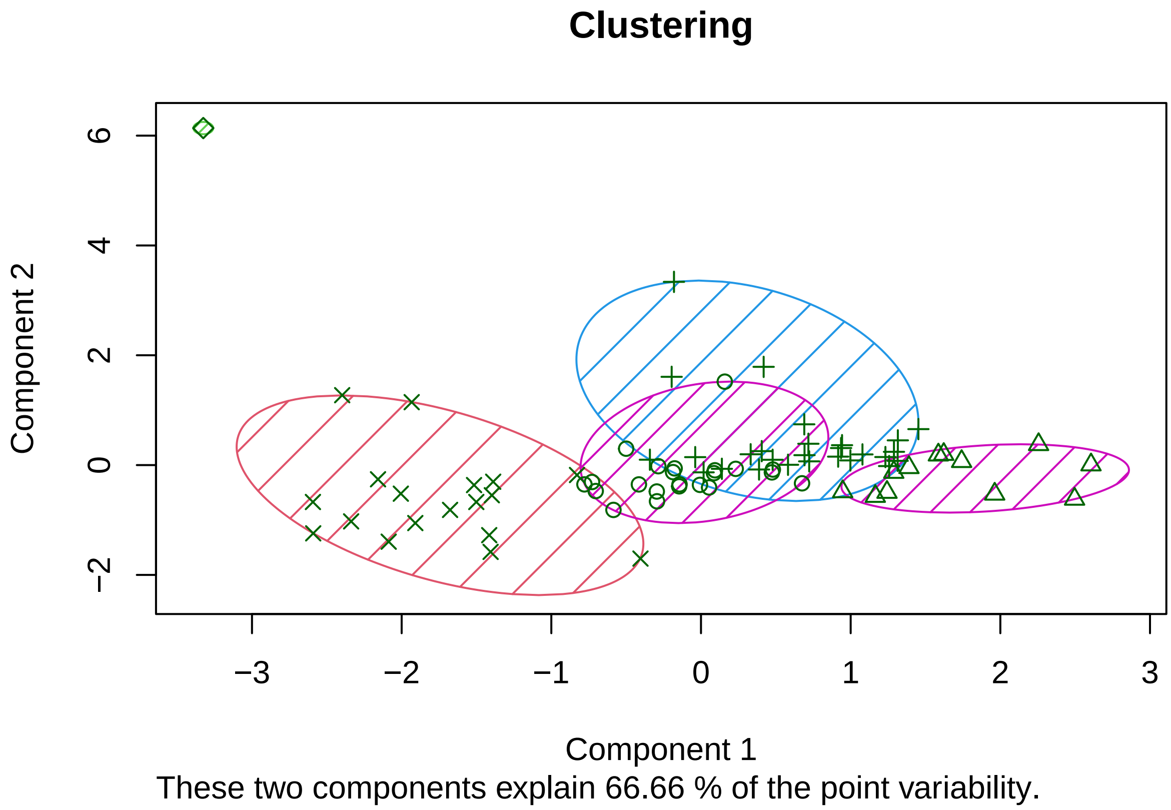
```
#K-medoides
library(ggplot2)
set.seed(123)

library(factoextra)
fviz_nbclust(x = datos, FUNcluster = pam, method = "wss",
 diss = dist(datos, method = "euclidean")) +
 geom_vline(xintercept = 5, linetype = 2)
```



```
pam_clusters <- pam(x = datos, k = 5, metric = "euclidean")
pam_clusters
```

```
## Medoids:
##      ID      onset   duration clonazepam    UPDRSIII
## [1,] 70 -0.3496123 -0.2987069 -0.2637227 -0.43811488
## [2,] 17  0.9556071 -0.5450631 -0.2637227  1.65573284
## [3,] 41  0.8623771 -0.2987069 -0.2637227 -0.07396745
## [4,] 79 -1.0022221  1.1794304 -0.2637227 -0.52915173
## [5,] 66 -0.3496123  1.6721428  7.0746480 -0.62018859
## Clustering vector:
##  [1] 1 2 3 2 2 2 2 2 3 2 3 4 4 3 2 1 2 3 4 3 3 1 3 2 3 3 2 1 2 3 3 4 4 1 4 3 1 3
## [39] 1 1 3 1 1 3 3 1 4 1 1 3 3 1 4 3 3 3 1 4 1 1 1 4 3 1 1 5 4 4 4 1 4 3 1 4 4 3
## [77] 3 3 4 4
## Objective function:
##     build      swap
## 1.0250582 0.9932326
##
## Available components:
##  [1] "medoids"    "id.med"     "clustering" "objective"  "isolation"
##  [6] "clusinfo"   "silinfo"    "diss"       "call"       "data"
```

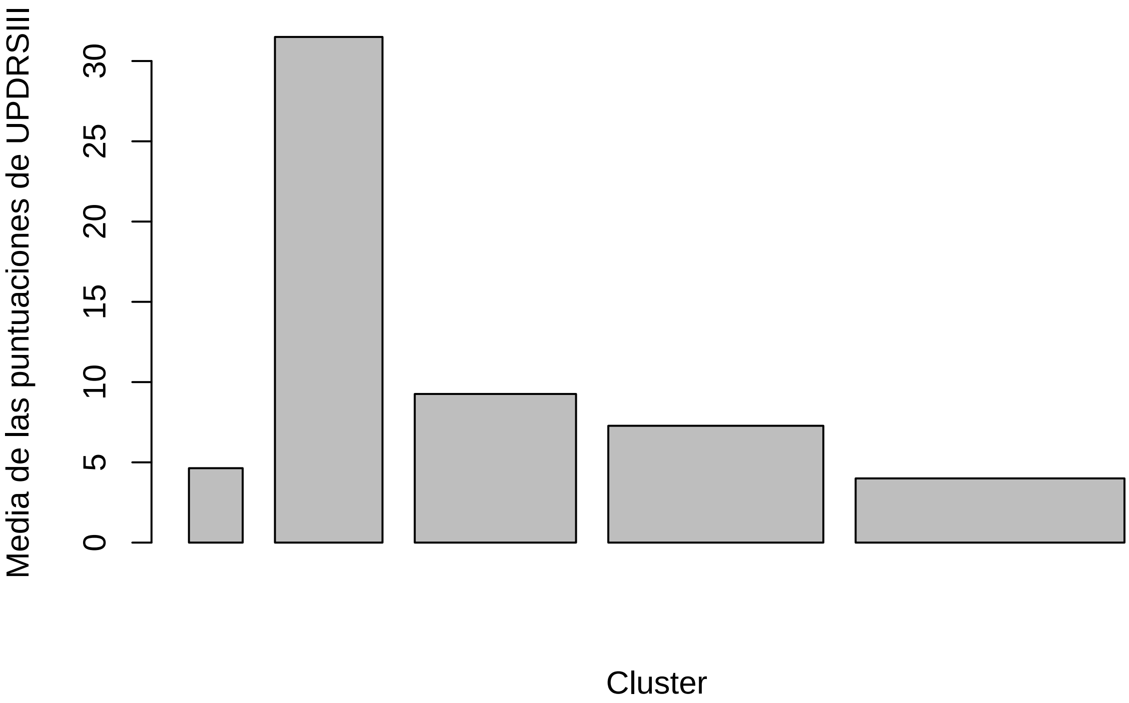
```
clusplot(datos, pam_clusters$cluster, lines = 0, shade = TRUE, color = TRUE, labels = 1, plotchar = TRU
```



```
#Observamos qué tan alta es la puntuación en la escala UPDRSIII en función del cluster al que ha sido a
datos<- cbind(cluster = pam_clusters$cluster, datos)
datos = as_tibble(datos)

clus_1 = datos1[datos$cluster == 1,]
clus_2 = datos1[datos$cluster == 2,]
clus_3 = datos1[datos$cluster == 3,]
clus_4 = datos1[datos$cluster == 4,]
clus_5 = datos1[datos$cluster == 5,]


mean1 = mean(clus_1$UPDRSIII)
mean2 = mean(clus_2$UPDRSIII)
mean3 = mean(clus_3$UPDRSIII)
mean4 = mean(clus_4$UPDRSIII)
mean5 = mean(clus_5$UPDRSIII)

means_UPDRSIII = c(mean1,mean2,mean3,mean4,mean5)
clus_updrs_mean = cbind(c(1,2,3,4,5), means_UPDRSIII)
clus_updrs_mean = as_tibble(clus_updrs_mean)
```

```
## Warning: The `x` argument of `as_tibble.matrix()` must have unique column names if `.name_repair` is
## Using compatibility `.name_repair`.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_warnings()` to see where this warning was generated.
```

```
barplot(clus_updrs_mean$means_UPDRSIII, clus_updrs_mean$V1, xlab="Cluster", ylab = "Media de las puntua
```



Vemos que la mayoría de los datos se pueden diferenciar en función de la cantidad de clonazepam que toman y por el puntuación de la variable UPDRSIII, tienen mayor variabilidad. Por lo que los clusters se forman en gran medida por estas dos variables. Es notable que en el cluster 2 los pacientes tienen un puntuaje mucho más alto.