PruebaLatex

Ejercicio 1

Lectura de los datos

```
library(readxl)
dt <- read_excel("~/ejercicio1seminario.xlsx",col_names = TRUE, col_types = "numeric")
dt <- as.data.frame(dt)
rownames(dt) <- paste0("Empresa ", 1:nrow(dt))
head(dt)
```

```
          Innovación Comunicación Eficiencia Responsabilidad
Empresa 1          4            5          4                5
Empresa 2          1            2          5                4
Empresa 3          1            2          2                3
Empresa 4          4            5          1                2
Empresa 5          5            5          5                3
Empresa 6          1            2          2                3
          Atención al Cliente Comunidad
Empresa 1                   4         3
Empresa 2                   1         4
Empresa 3                   4         2
Empresa 4                   2         1
Empresa 5                   4         3
Empresa 6                   3         3
```

Prueba de normalidad

```
library(MVN)
```

```
Warning: package 'MVN' was built under R version 4.3.3
```

```r
normalidad <- mvn(dt, mvnTest = "mardia")
normalidad
```

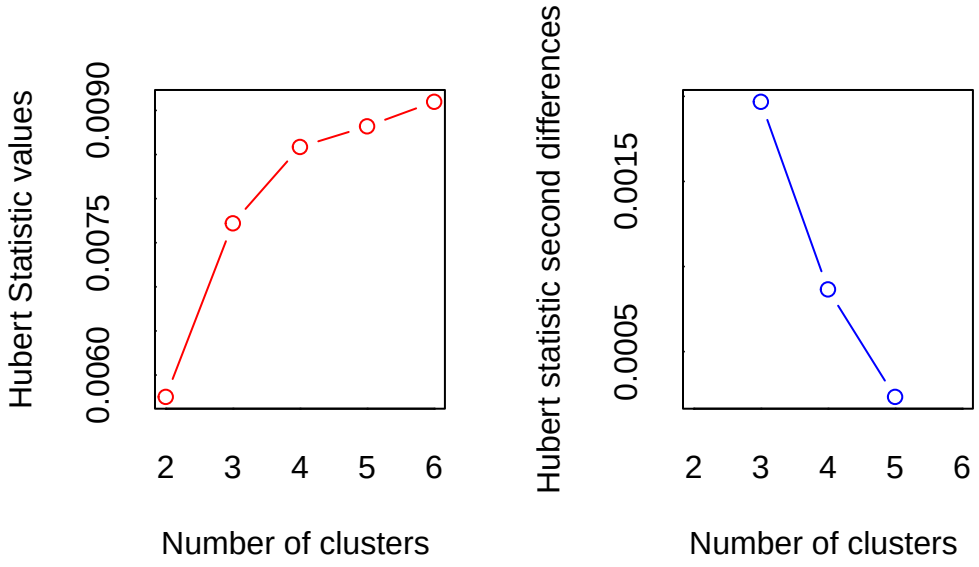
```
$multivariateNormality
             Test         Statistic            p value Result
1 Mardia Skewness  43.4819357405774  0.888829218092923    YES
2 Mardia Kurtosis -1.80347342648514 0.0713138951310004    YES
3             MVN              <NA>               <NA>    YES

$univariateNormality
              Test            Variable Statistic   p value Normality
1 Anderson-Darling     Innovación         0.8477    0.0237    NO
2 Anderson-Darling    Comunicación        1.1134    0.0049    NO
3 Anderson-Darling     Eficiencia         1.0229    0.0084    NO
4 Anderson-Darling   Responsabilidad      0.5851    0.1126    YES
5 Anderson-Darling Atención al Cliente    0.8940    0.0181    NO
6 Anderson-Darling      Comunidad         0.8734    0.0204    NO

$Descriptives
                     n Mean  Std.Dev Median Min Max 25th 75th        Skew
Innovación          20 2.65 1.308877    2.0   1   5 2.00    4  0.35755502
Comunicación        20 2.95 1.356272    2.5   1   5 2.00    4  0.32557373
Eficiencia          20 3.10 1.618967    3.0   1   5 1.75    5 -0.08200978
Responsabilidad     20 3.05 1.316894    3.0   1   5 2.00    4 -0.08637027
Atención al Cliente 20 2.95 1.431782    3.0   1   5 2.00    4 -0.01967529
Comunidad           20 2.80 1.542384    3.0   1   5 1.00    4  0.15370990
                     Kurtosis
Innovación          -1.207362
Comunicación        -1.378612
Eficiencia          -1.676884
Responsabilidad     -1.249458
Atención al Cliente -1.546286
Comunidad           -1.549809
```

Los datos presentan normalidad, así que se procede con el método de clustering.

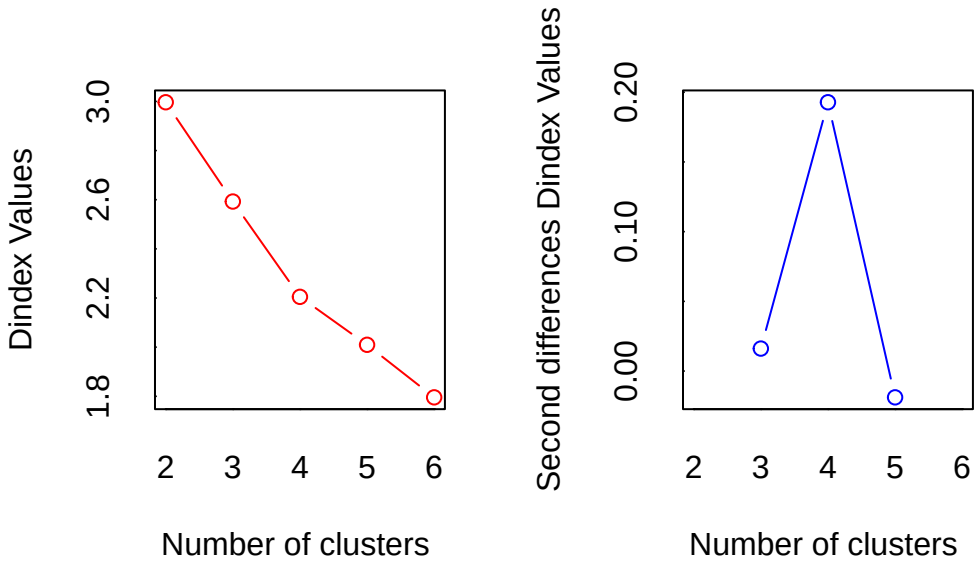
Primero se tratara de definir cuantos clusters son los ideales

```r
#install.packages("NbClust")
library(NbClust)
res.nbclust <- NbClust(dt, distance = "euclidean",
```

```
                    min.nc = 2, max.nc = 6,
                    method = "complete", index ="all")
```



*** : The Hubert index is a graphical method of determining the number of clusters.
In the plot of Hubert index, we seek a significant knee that corresponds to a
significant increase of the value of the measure i.e the significant peak in
index second differences plot.




```
*** : The D index is a graphical method of determining the number of clusters.
                In the plot of D index, we seek a significant knee (the significant peak in t
                second differences plot) that corresponds to a significant increase of the va
                the measure.

*******************************************************************
* Among all indices:
* 5 proposed 2 as the best number of clusters
* 3 proposed 3 as the best number of clusters
* 10 proposed 4 as the best number of clusters
* 5 proposed 6 as the best number of clusters

                   ***** Conclusion *****

* According to the majority rule, the best number of clusters is  4


*******************************************************************
```

Nuestro primer metodo inidca que lo adecuado serian 4 clusters.

```r
library(clValid)
```

```
Warning: package 'clValid' was built under R version 4.3.3
```

```
Loading required package: cluster
```

```r
validclus  <- clValid(dt, nClust = 2:6,
                      clMethods = c("hierarchical","kmeans","diana","fanny","pam","clara","ag
                      validation = "internal")
```

```
Warning in fanny(Dist, nc, ...): the memberships are all very close to 1/k.
Maybe decrease 'memb.exp' ?
```

```
Warning in fanny(Dist, nc, ...): the memberships are all very close to 1/k.
Maybe decrease 'memb.exp' ?
```

```
Warning in vClusters(mat, clMethods[i], nClust, validation = validation, :
fanny unable to find 3 clusters, returning NA for these validation measures
```

```
Warning in fanny(Dist, nc, ...): the memberships are all very close to 1/k.
Maybe decrease 'memb.exp' ?
```

```
Warning in vClusters(mat, clMethods[i], nClust, validation = validation, :
fanny unable to find 4 clusters, returning NA for these validation measures
```

```
Warning in fanny(Dist, nc, ...): the memberships are all very close to 1/k.
Maybe decrease 'memb.exp' ?
```

```
Warning in vClusters(mat, clMethods[i], nClust, validation = validation, :
fanny unable to find 5 clusters, returning NA for these validation measures
```

```
Warning in fanny(Dist, nc, ...): the memberships are all very close to 1/k.
Maybe decrease 'memb.exp' ?
```

```
Warning in vClusters(mat, clMethods[i], nClust, validation = validation, :
fanny unable to find 6 clusters, returning NA for these validation measures
```

```
summary(validclus)
```

```
Clustering Methods:
 hierarchical kmeans diana fanny pam clara agnes

Cluster sizes:
 2 3 4 5 6

Validation Measures:
                                 2       3       4       5       6

hierarchical Connectivity   6.0798 15.9718 20.6115 24.3917 25.9429
             Dunn           0.4913  0.4237  0.5092  0.5517  0.5517
             Silhouette     0.2082  0.1871  0.2479  0.2475  0.2357
kmeans       Connectivity  12.1103 18.8056 20.7163 23.9440 25.9429
             Dunn           0.3714  0.4410  0.5092  0.5517  0.5517
             Silhouette     0.1918  0.2228  0.2743  0.2729  0.2357
diana        Connectivity  16.1770 17.7060 20.7238 26.9929 28.8107
             Dunn           0.2981  0.2981  0.3123  0.3430  0.3651
             Silhouette     0.1776  0.0904  0.0522  0.1030  0.1287
fanny        Connectivity  13.9187      NA      NA      NA      NA
             Dunn           0.4564      NA      NA      NA      NA
             Silhouette     0.1832      NA      NA      NA      NA
pam          Connectivity  18.1060 20.4905 26.0647 28.9270 30.2659
             Dunn           0.3262  0.3536  0.4082  0.4663  0.4663
             Silhouette     0.1525  0.2198  0.2353  0.2208  0.2507
clara        Connectivity  15.2171 20.4905 25.1869 29.2813 30.6202
             Dunn           0.3444  0.3536  0.5189  0.4170  0.4170
             Silhouette     0.1562  0.2198  0.2336  0.2127  0.2310
agnes        Connectivity   6.0798 15.9718 20.6115 24.3917 25.9429
             Dunn           0.4913  0.4237  0.5092  0.5517  0.5517
             Silhouette     0.2082  0.1871  0.2479  0.2475  0.2357

Optimal Scores:

             Score  Method       Clusters
Connectivity 6.0798 hierarchical 2
Dunn         0.5517 hierarchical 5
Silhouette   0.2743 kmeans       4
```

con el segundo método, se concluye que el numero de clusters a escoger serán 4.

Se realiza el dendograma para visualizar que empresas están en cada cluster.

```
dist      <- dist(dt,method = "euclidean")
modelo    <- hclust(dist, method = "complete")

library(factoextra)
```

```
Warning: package 'factoextra' was built under R version 4.3.3
```

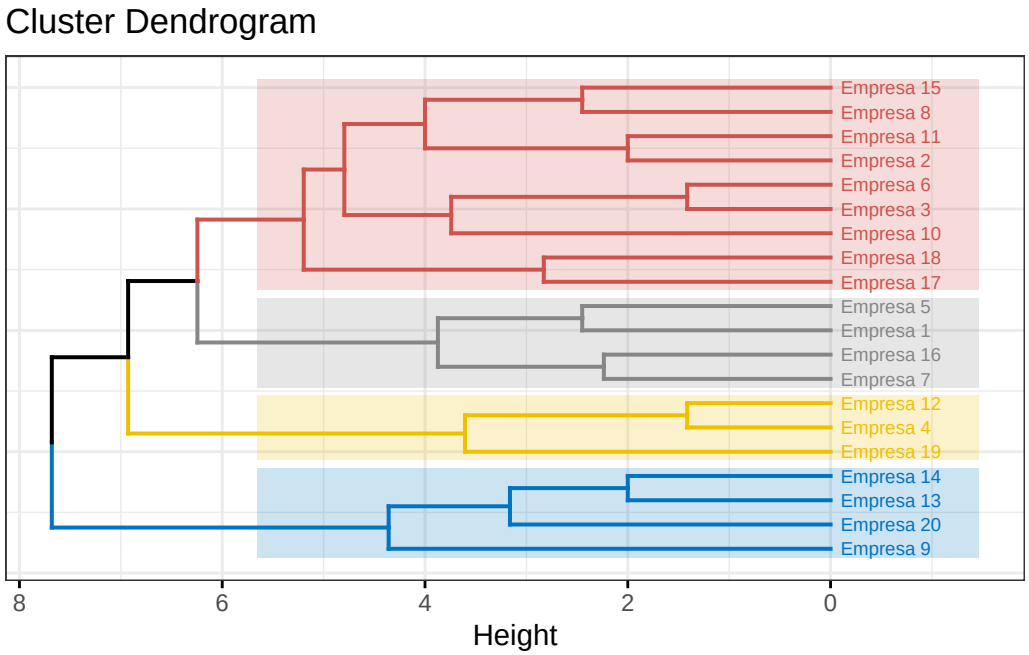
```
Loading required package: ggplot2
```

```
Warning: package 'ggplot2' was built under R version 4.3.3
```

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

```
fviz_dend(modelo, cex = 0.5, k=4,
          rect = TRUE,
          k_colors = "jco",
          rect_border = "jco",
          rect_fill = TRUE,
          horiz = TRUE,
          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>.
```



El dendograma muestra los 4 clusters que se forman y que empresas las conforman.

Análisis de las medias de los clusters

```
clust <- kmeans(dt, centers = 4)
cluster_membership <- clust$cluster
dt_clustered <- cbind(dt, cluster = cluster_membership)
summary_by_cluster <- aggregate(. ~ cluster, data = dt_clustered, FUN = mean)
print(summary_by_cluster)
```

```
  cluster Innovación Comunicación Eficiencia Responsabilidad
1       1   1.500000     1.500000   1.500000        2.000000
2       2   2.500000     3.000000   3.500000        3.000000
3       3   2.142857     2.428571   4.285714        3.428571
4       4   4.400000     4.800000   2.400000        3.400000
  Atención al Cliente Comunidad
1            3.250000  3.500000
2            4.500000  4.750000
3            1.857143  1.857143
4            3.000000  2.000000
```

Al calcular las medias de cada variable para cada cluster, podemos identificar patrones y diferencias entre los grupos. El Cluster 4 destaca por tener valores medios superiores en

general, lo que sugiere que las empresas en este grupo exhiben características más elevadas en comparación con los otros clusters.

Identificación de las mejores empresas

```
suma_medias <- rowSums(summary_by_cluster[, -1])

suma_medias_df <- data.frame(cluster = summary_by_cluster$cluster, Suma_de_Medias = suma_medi

print(suma_medias_df)
```

```
  cluster Suma_de_Medias
1       1          13.25
2       2          21.25
3       3          16.00
4       4          20.00
```

Conclusión: El Cluster 4 se distingue por sus valores medios superiores en general, sugiriendo que este grupo de observaciones representa empresas con características más elevadas en comparación con los otros clusters.

Las empresas pertenecientes al Cluster 4 son las que destacan por sus altos niveles en múltiples dimensiones, como innovación, comunicación, eficiencia, responsabilidad, atención al cliente y compromiso comunitario.

Ejercicio 2

exploramos la tabla y dimensiones del conjunto de datos.

```
#install.packages("lda")
#install.packages("rattle.data")
library(lda)
```

```
Warning: package 'lda' was built under R version 4.3.3
```

```
library(MASS)
library(car)
```

```
Loading required package: carData
```

```
library(rattle)
```

```
Warning: package 'rattle' was built under R version 4.3.3
```

```
Loading required package: tibble
```

```
Loading required package: bitops
```

```
Rattle: A free graphical interface for data science with R.
Versión 5.5.1 Copyright (c) 2006-2021 Togaware Pty Ltd.
Escriba 'rattle()' para agitar, sacudir y  rotar sus datos.
```

```
library(ggplot2)

view(wine)
dim(wine)
```

```
[1] 178  14
```

se Carga la librería MVN para hacer una prueba de normalidad

```
library(MVN)
wine_subset <- subset(wine, select = -c(Type))
result <- mvn(data = wine_subset, mvnTest = "royston")
result$multivariateNormality
```

```
     Test        H      p value MVN
1 Royston 182.5371 7.306913e-33  NO
```

No es restrictivo el metodo con que las variables sigan una distribucion normal multivariante

Matriz de datos y variable de respuesta

```
Z <- as.data.frame(wine)
head(Z)
```

```
  Type Alcohol Malic  Ash Alcalinity Magnesium Phenols Flavanoids Nonflavanoids
1    1   14.23  1.71 2.43       15.6       127    2.80       3.06          0.28
2    1   13.20  1.78 2.14       11.2       100    2.65       2.76          0.26
3    1   13.16  2.36 2.67       18.6       101    2.80       3.24          0.30
4    1   14.37  1.95 2.50       16.8       113    3.85       3.49          0.24
5    1   13.24  2.59 2.87       21.0       118    2.80       2.69          0.39
6    1   14.20  1.76 2.45       15.2       112    3.27       3.39          0.34
  Proanthocyanins Color  Hue Dilution Proline
1            2.29  5.64 1.04     3.92    1065
2            1.28  4.38 1.05     3.40    1050
3            2.81  5.68 1.03     3.17    1185
4            2.18  7.80 0.86     3.45    1480
5            1.82  4.32 1.04     2.93     735
6            1.97  6.75 1.05     2.85    1450
```

```r
Y <- Z[,1]
head(Y)
```

```
[1] 1 1 1 1 1 1
Levels: 1 2 3
```

```r
X <- Z[,2:14]
head(X)
```

```
  Alcohol Malic  Ash Alcalinity Magnesium Phenols Flavanoids Nonflavanoids
1   14.23  1.71 2.43       15.6       127    2.80       3.06          0.28
2   13.20  1.78 2.14       11.2       100    2.65       2.76          0.26
3   13.16  2.36 2.67       18.6       101    2.80       3.24          0.30
4   14.37  1.95 2.50       16.8       113    3.85       3.49          0.24
5   13.24  2.59 2.87       21.0       118    2.80       2.69          0.39
6   14.20  1.76 2.45       15.2       112    3.27       3.39          0.34
  Proanthocyanins Color  Hue Dilution Proline
1            2.29  5.64 1.04     3.92    1065
2            1.28  4.38 1.05     3.40    1050
3            2.81  5.68 1.03     3.17    1185
4            2.18  7.80 0.86     3.45    1480
5            1.82  4.32 1.04     2.93     735
6            1.97  6.75 1.05     2.85    1450
```

LDA funcion discriminante para observar las clasificaciones

```r
lda.wine <- lda(Y~., data = X, CV=TRUE)

lda.wine$class
```

```
  [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [38] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 [75] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 2 2 2 2 2 2 2 2 2 2 2 2
[112] 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
[149] 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
Levels: 1 2 3
```

ERRORES DEL METODO:

```r
table.lda <- table(Y, lda.wine$class)
table.lda
```

```
Y    1  2  3
  1 59  0  0
  2  1 69  1
  3  0  0 48
```

Se observa en la tabla que para el 1 y el 3 hay 1 variable mal agrupada por cada uno.

PROPORCION

```r
error.lda <- 178 - sum(Y==lda.wine$class)
error.lda/178
```

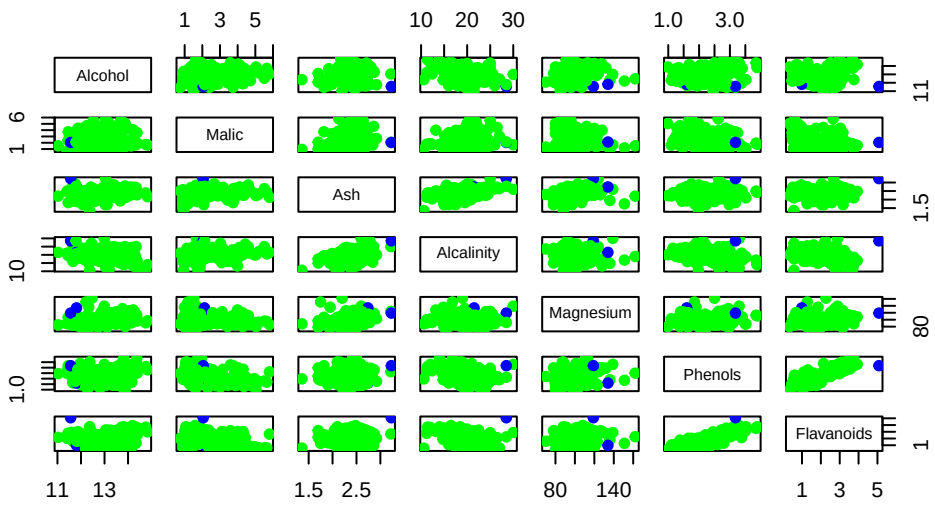
```
[1] 0.01123596
```

El metodo tiene un 1% de error lo cual es muy aceptable para trabajarlo.

Como la matriz es grande, se imprimira por partes.

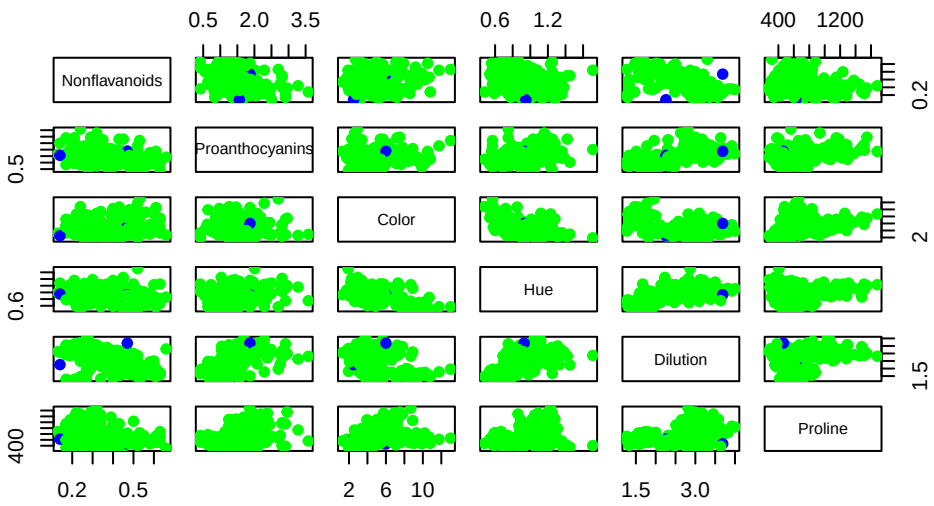
```r
#la matriz con los primeros 7
col.lda.wine <- c("blue", "green")[1*(Y==lda.wine$class)+1]
pairs(X[1:7], main = "Correctos (verde) e Incorrectos (azul) Clasificacion de vinos por Disc
      pch=19, col=col.lda.wine)
```

s (verde) e Incorrectos (azul) Clasificacion de vinos por Disc




```
#la matriz con los siguientes 6
col.lda.wine <- c("blue", "green")[1*(Y==lda.wine$class)+1]
pairs(X[8:13], main = "Correctos (rojo) e Incorrectos (negros) Clasificacion de vinos por Dis
      pch=19, col=col.lda.wine)
```

(rojo) e Incorrectos (negros) Clasificacion de vinos por Disc



PROBABILIDADES DE PERTENENCIA

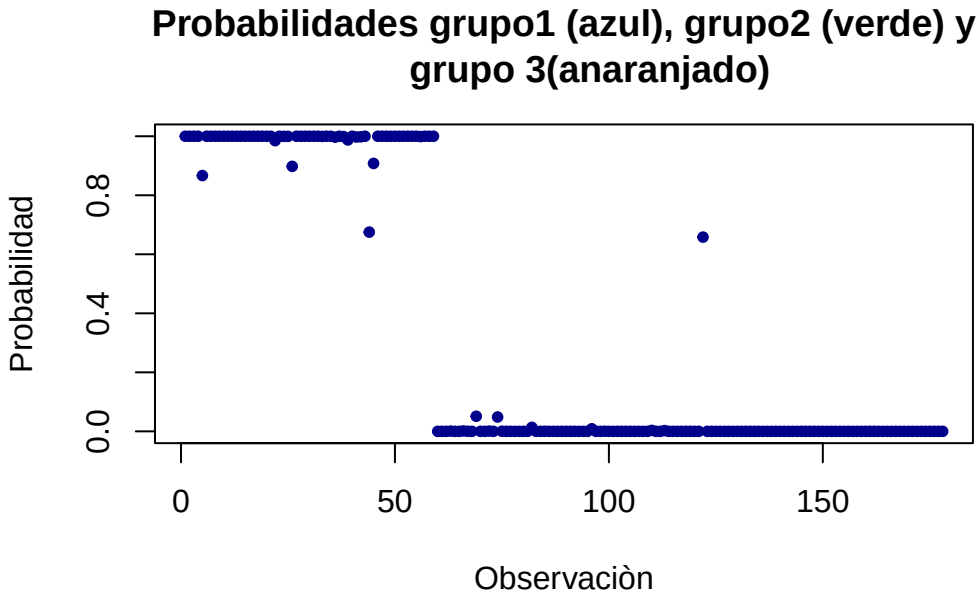
```
head(lda.wine$posterior)
```

```
          1            2            3
1 1.0000000 2.797215e-09 2.071649e-18
2 0.9999996 4.380414e-07 7.695679e-17
3 0.9999970 3.024498e-06 1.071531e-13
4 1.0000000 1.251896e-12 1.095841e-16
5 0.8667524 1.332472e-01 3.998307e-07
6 1.0000000 2.236339e-11 1.017938e-17
```

Muestra la probabilidad que pertenezcan a cada tipo.

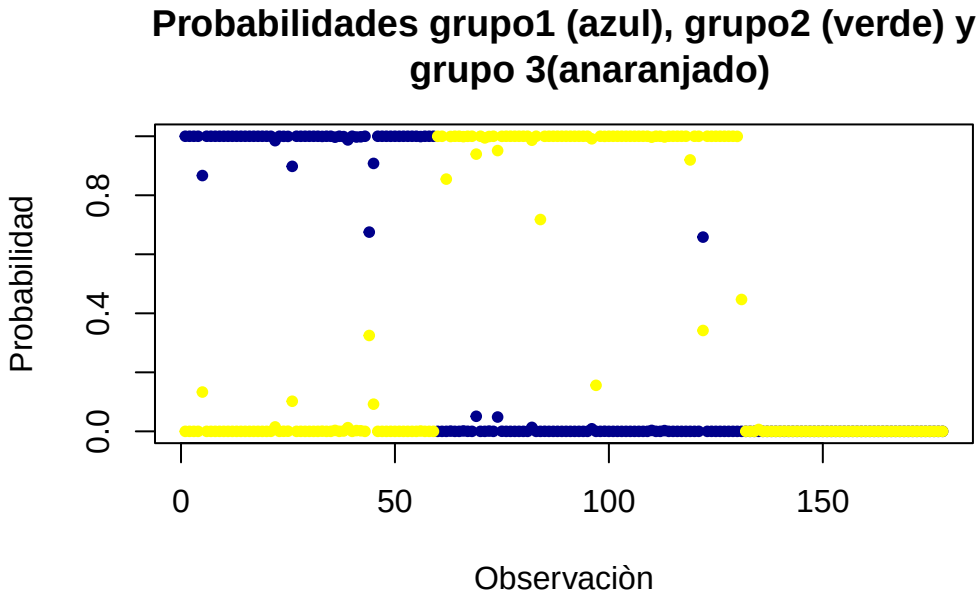
Graficamente

```
plot(1:178, lda.wine$posterior[,1], main="Probabilidades grupo1 (azul), grupo2 (verde) y
     grupo 3(anaranjado)", pch=20, col="darkblue", xlab="Observaciòn", ylab="Probabilidad")
```

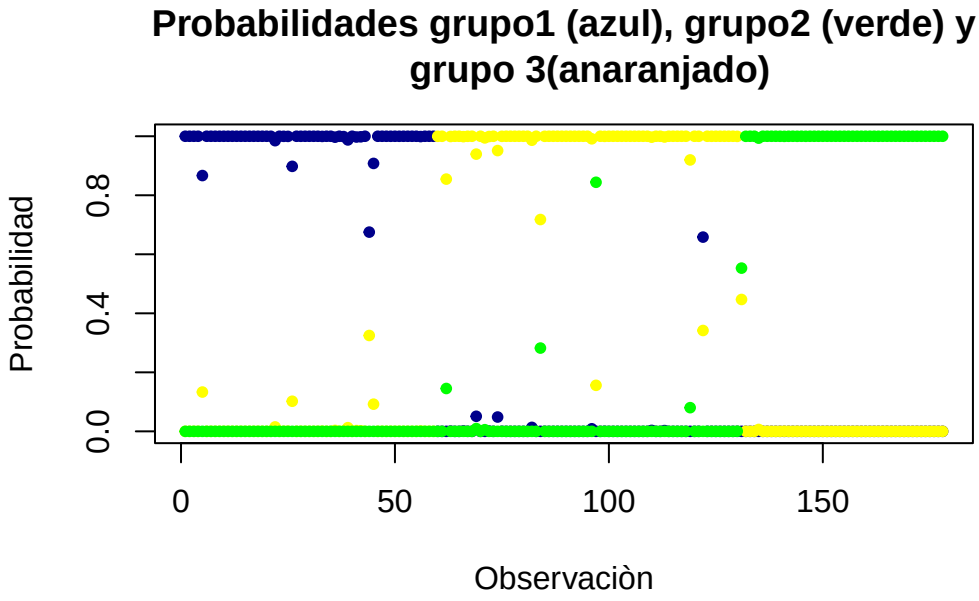


grupo 1 en azul; las que esten en 1 pertenecen al grupo 1.

```
#este amarillo
plot(1:178, lda.wine$posterior[,1], main="Probabilidades grupo1 (azul), grupo2 (verde) y
     grupo 3(anaranjado)", pch=20, col="darkblue", xlab="Observaciòn", ylab="Probabilidad")

points(1:178, lda.wine$posterior[,2], pch=20, col="yellow")
```

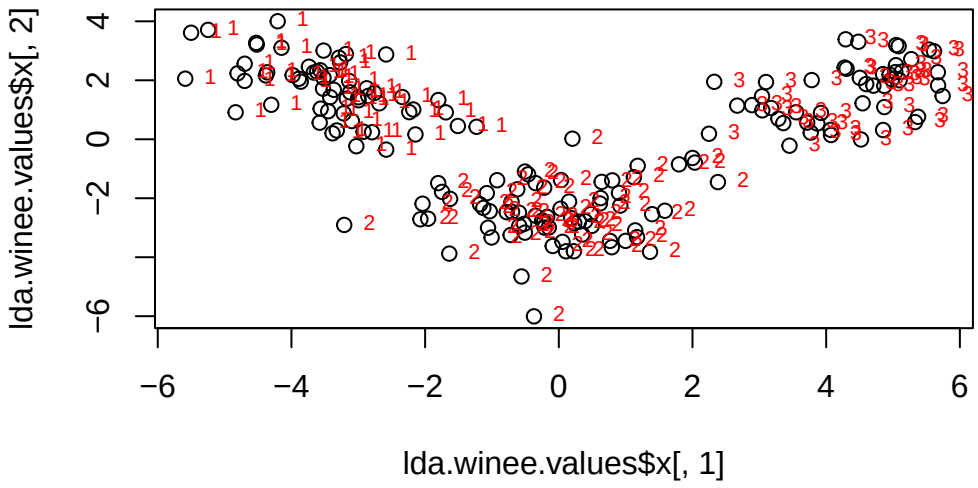



```
#verde
plot(1:178, lda.wine$posterior[,1], main="Probabilidades grupo1 (azul), grupo2 (verde) y
     grupo 3(anaranjado)", pch=20, col="darkblue", xlab="Observaciòn", ylab="Probabilidad")

points(1:178, lda.wine$posterior[,2], pch=20, col="yellow")


points(1:178, lda.wine$posterior[,3], pch=20, col="green")
```



Agrupando las funciones:

```
lda.winee <- lda(Y~., data = X)

lda.winee.values <- predict(lda.winee)

plot(lda.winee.values$x[,1],lda.winee.values$x[,2])
text(lda.winee.values$x[,1],lda.winee.values$x[,2],Y,cex=0.7,pos=4,col="red")
```




```
#el grafico de las funciones de prediccion
```

Conclusion El propósito del análisis discriminante lineal (LDA) en este ejemplo es encontrar las combinaciones lineales de las variables originales (las 13 concentraciones químicas) que proporcionan la mejor separación posible entre los grupos (variedades de vino) en nuestro conjunto de datos.