

PRACTICA 6 PROGRAMACION EN R:

**Nota: los comentarios están adjuntos al código.

The screenshot shows the RStudio interface with the title bar "RStudio" and the date "Lun 6 nov 19:24". The left pane displays a script named "Script federico P6.R" containing R code for calculating distances between vectors and loading the bioconductor package. The right pane shows the "Environment" tab with variables "a", "b", "dist.cy", "golub", and "golub.g_". The "Console" tab at the bottom shows the execution of the script, including the calculation of distances and the loading of the "golub" dataset.

```
## Script PRACTICA 6 ##
a<-c(1,1,6)
b<-c(4,5,1)
sqrt(sum((a-b)^2))
(a-b)^2
sum((a-b)^2)
#
source("http://www.bioconductor.org/biocLite.R") # cargamos paquete bioconductor
biocLite()
biocLite("multtest")
library(multtest) # libreria multtest
data(golub)
golub.gnames[1042,]
index<-grep("Cyclin",golub.gnames[,2])
index
golub.gnames[index,2]
# calculamos con funcion "dist" en este caso euclidea.
dist.cyclin<-dist(golub[,index],method="euclidean")
30:1 (Top Level) : 
```

```
Escribo 'demo()' para demostraciones, 'help()' para el sistema on-line de ayuda,
o 'help.start()' para abrir el sistema de ayuda HTML con su navegador.
Escribo 'q()' para salir de R.

[Workspace loaded from ~/.RData]

> a<-c(1,1,6)
> b<-c(4,5,1)
> sqrt(sum((a-b)^2))
[1] 7.071068
> (a-b)^2
[1] 9 16 25
> sum((a-b)^2)
[1] 50
```

The screenshot shows the RStudio interface with the title bar "RStudio" and the date "Lun 6 nov 19:24". The left pane displays the same script "Script federico P6.R" as the first session. The right pane shows the "Environment" tab with variables "a", "b", "dist.cy", "golub", and "golub.g_". The "Console" tab at the bottom shows the continuation of the script execution, specifically the loading of the "golub" dataset and its names.

```
## Script PRACTICA 6 ##
a<-c(1,1,6)
b<-c(4,5,1)
sqrt(sum((a-b)^2))
(a-b)^2
sum((a-b)^2)
#
source("http://www.bioconductor.org/biocLite.R") # cargamos paquete bioconductor
biocLite()
biocLite("multtest")
library(multtest) # libreria multtest
data(golub)
golub.gnames[1042,]
index<-grep("Cyclin",golub.gnames[,2])
index
golub.gnames[index,2]
# calculamos con funcion "dist" en este caso euclidea.
dist.cyclin<-dist(golub[,index],method="euclidean")
30:1 (Top Level) : 
```

```
Console ~/ ~
> data(golub)
> golub.gnames[1042,]
[1] "2354"           "CCND3 Cyclin D3" "M92287_at"
> index<-grep("Cyclin",golub.gnames[,2])
> index
[1] 85 962 1042 1195 1212 1354 1421 1553 1597 1937 2027 2240
> golub.gnames[index,2]
[1] "CCND2 Cyclin D2"          "CDK2 Cyclin-dependent kinase 2"
[3] "CCND3 Cyclin D3"          "CDKN1A Cyclin-dependent kinase inhibitor 1A (p21, Cip1)"
[5] "CCNH Cyclin H"            "Cyclin-dependent kinase 4 (CDK4) gene"
[7] "Cyclin G2 mRNA"           "Cyclin A1 mRNA"
[9] "Cyclin-selective ubiquitin carrier protein mRNA" "CDK6 Cyclin-dependent kinase 6"
[11] "Cyclin G1 mRNA"           "CCNF Cyclin F"
> dist.cyclin<-dist(golub[,index],method="euclidean") # calculamos con funcion "dist" en este caso euclidea.
> diam<-as.matrix(dist.cyclin) # creamos una matriz para trabajar con esta forma.
```

RStudio

Práctica6.R x Script federico P6.R x

```

14 # debemos tener "dist" en este caso euclideo.
15 index
16 golub.gnames[index,2]
17 # creamos una matriz para trabajar con esta forma.
18 dist.cyclin<-dist[golub[,index],method="euclidean"]
19 # creamos una matriz para trabajar con esta forma.
20 diam<-as.matrix(dist.cyclin)
21 # damos nombre a las columnas con tercer elemento del index.
22 colnames(diam)<-golub.gnames[index,3]
23 # damos nombre a las filas con su mismo nombre
24 rownames(diam)<-colnames(diam)
25 # damos nombre a los indices que muestre de fila 1 a 5 y columna 1 a 5.
26 diam[1:5,1:5]
27
30:1 (Top Level) R Script
```

Console ~ /

```

> index
[1] 85 962 1042 1195 1212 1354 1421 1553 1597 1937 2027 2240
> golub.gnames[index,2]
[1] "CDK2 Cyclin-dependent kinase 2"
[2] "CDK1A Cyclin-dependent kinase inhibitor 1A (p21, Cip1)"
[3] "CDK3 Cyclin D3"
[4] "CDK4 Cyclin D4"
[5] "CDK6 Cyclin E1 mRNA"
[6] "Cyclin G2 mRNA"
[7] "Cyclin-selective ubiquitin carrier protein mRNA"
[8] "CDK5 Cyclin-dependent kinase 6"
[9] "Cyclin G1 mRNA"
[10] "CCNF Cyclin F
```

> dist.cyclin<-dist[golub[,index],method="euclidean"] # calculemos con función "dist" en este caso euclideo.

> diam<-as.matrix(dist.cyclin) # creamos una matriz para trabajar con esta forma.

> colnames(diam)<-golub.gnames[index,3] # damos nombre a las columnas con tercer elemento del index.

> rownames(diam)<-colnames(diam) # damos nombre a las filas con su mismo nombre

> diam[1:5,1:5]

```

D13639_at M68520_at M92287_at U09579_at U11791_at
D13639_at 0.000000 8.821866 11.55349 10.056814 8.669112
M68520_at 8.821866 0.000000 11.70156 5.931269 2.934802
M92287_at 11.553494 11.701562 0.000000 11.991333 11.906558
U09579_at 10.056814 5.931268 11.99133 0.000000 5.598232
U11791_at 8.669112 2.934802 5.698232 0.000000
```

82% Lun 6 nov 19:27 Fedé

RStudio

Práctica6.R x Script federico P6.R x

```

29 biocLite("genefinder")
30 library("genefinder") # Cargamos la nueva librería.
31 biocLite("ALL")
32 library("ALL");
33 data(ALL)
34 closesto1389_at<- genefinder(ALL, "1389_at", 10, method = "euc")
35 closesto1389_at[[1]]$indices
36 round(closereto1389_at[[1]]$dists,1)
37 featureNames(ALL)[closereto1389_at[[1]]$indices]
38 str(closereto1389_at)
39 names<- list(c("g1","g2","g3","g4","g5"),c("p1","p2"))
40 sl.clus.dat<- matrix(c(1,1,1,1,1,3,2,2,3,5,5),ncol = 2,byrow = TRUE,dimnames = names)
41
42:1 (Top Level) R Script
```

Console ~/

```

> library("ALL");
> data(ALL)
> closesto1389_at<- genefinder(ALL, "1389_at", 10, method = "euc")
> closesto1389_at[[1]]$indices
[1] 2653 1096 6634 9255 6639 11402 9849 2274 8518 10736
> round(closereto1389_at[[1]]$dists,1)
[1] 12.6 12.8 12.8 13.0 13.1 13.2 13.3 13.4
> featureNames(ALL)[closereto1389_at[[1]]$indices]
[1] "32629_f_at" "1988_at" "36571_at" "39168_at" "36576_at" "41295_at" "39756_g_at" "32254_at" "38438_at" "40635_at"
> str(closereto1389_at)
List of 1
 $ 1389_at: list of 2
 ..$ indices: num [1:10] 2653 1096 6634 9255 6639 ...
 ..$ dists : num [1:10] 12.6 12.8 12.8 12.8 13 ...
 > names <- list(c("g1","g2","g3","g4","g5"),c("p1","p2"))
> sl.clus.dat<- matrix(c(1,1,1,1,1,3,2,2,3,5,5),ncol = 2,byrow = TRUE,dimnames = names)
> sl.clus.dat
   p1  p2
g1 1 1.0
g2 1 1.1
```

82% Lun 6 nov 19:27 Fedé

RStudio

Practica6.R

```

27 biocLite()
28 biocLite("genefilter")
29 library("genefilter") # Cargamos la nueva libreria,
30 library("ALL")
31 biocLite("ALL")
32 library("ALL");
33 data(ALL)
34 closesto1389_at<- geneFilter(ALL, "1389_at", 10, method = "euc")
35 closesto1389_at[[1]]$indices
36 round(closeto1389_at[[1]]$dists,1)
37 featureNames(ALL)[closeto1389_at[[1]]$indices]
38 str(closeto1389_at)
39
45:1 (Top Level) : R Script

```

Console ~ /

```

** building package indices
** installing vignettes
** testing if installed package can be loaded
* DONE [ALL]

The downloaded source packages are in
  '/private/var/folders/ff/crzyp23s5f5gy08f14qt2dm40000gp/T/RtmpYVwIkM/downloads_packages'
> library("ALL");
> data(ALL)
> closesto1389_at<- geneFilter(ALL, "1389_at", 10, method = "euc")
> closesto1389_at[[1]]$indices
[1] 2653 1096 6634 9255 6639 11402 9849 2274 8518 10736
> round(closeto1389_at[[1]]$dists,1)
[1] 12.6 12.8 12.8 13.0 13.1 13.2 13.3 13.4
> featureNames(ALL)[closeto1389_at[[1]]$indices]
[1] "32629_f_at" "1988_at" "36571_at" "39168_at" "36576_at" "41295_at" "39756_g_at" "32254_at" "38438_at" "40635_at"
> str(closeto1389_at)
List of 1
$.1389_at: list of 2
..$ dists: num [1:10] 2653 1096 6634 9255 6639 ...
..$ dists : num [1:10] 12.6 12.8 12.8 12.8 13 ...

```

RStudio

Practica6.R

```

39 names <- list(c("g1","g2","g3","g4","g5"),c("p1","p2"))
40 sl.clus.dat<- matrix(c(1,1,1,1.1,3,2,3,2.3,5,5),ncol = 2,byrow = TRUE,dimnames = names)
41 sl.clus.dat
42 plot(sl.clus.dat,type="n", xlim=c(0,6), ylim=c(0,6))
43 text(sl.clus.dat,labels=row.names(sl.clus.dat))
44 print(dist(sl.clus.dat,method="euclidean"),digits=3)
45 sl.out<-hclust(dist(sl.clus.dat,method="euclidean"),method="single") # funcion "hclust" hace el clustering utilizando las distancias entre
46 plot(sl.out)
47 sl.out<-hclust(dist(rnorm(20,0,1)),method="euclidean"),method="single")
48 plot(sl.out)
49 x <- c(rnorm(10,0,0.1),rnorm(10,3,0.5),rnorm(10,10,1.0))
50
51
52
53
54
55
56
57
58
59
50:1 (Top Level) : R Script

```

Console ~ /

```

..$ dists : num [1:10] 12.6 12.8 12.8 12.8 13 ...
> names <- list(c("g1","g2","g3","g4","g5"),c("p1","p2"))
> sl.clus.dat<- matrix(c(1,1,1,1.1,3,2,3,2.3,5,5),ncol = 2,byrow = TRUE,dimnames = names)
> sl.clus.dat
  p1 p2
g1 1 1.0
g2 1 1.1
g3 3 2.0
g4 3 2.3
g5 5 5.0
> plot(sl.out)
Error in plot(sl.out) : objeto 'sl.out' no encontrado
> plot(sl.clus.dat,type="n", xlim=c(0,6), ylim=c(0,6))
> text(sl.clus.dat,labels=row.names(sl.clus.dat))
> print(dist(sl.clus.dat,method="euclidean"),digits=3)
  g1  g2  g3  g4
g2 0.10
g3 2.24 2.19
g4 2.39 2.33 0.30
g5 5.66 5.59 3.61 3.36
>

```

Environment History

Data

Values

Files Plots Packages Help Viewer

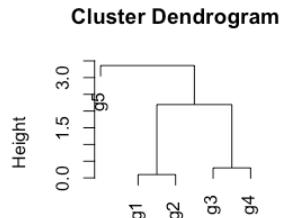
RStudio

```

Practica6.R x Script federico P6.R x
Source on Save Run Source
35 closeto1389.ot[[1]]$lists,1)
36 round(closeto1389.ot[[1]]$lists,1)
37 featureNames(ALL)closeto1389.ot[[1]]$indices
38 str(closeto1389.ot)
39 names <- list(c("g1","g2","g3","g4","g5"),c("p1","p2"))
40 sl.clus.dat<- matrix(c(1,1,1,1,3,2,3,2,3,5),ncol = 2,byrow = TRUE,dimnames = names)
41 sl.clus.dat
42 plot(sl.clus.dat,type="n", xlim=c(0,6), ylim=c(0,6))
43 text(sl.clus.dat,labels=rownames(sl.clus.dat))
44 print(dist(sl.clus.dat,method="euclidean"),digits=3)
45 sl.out<-hclust(dist(sl.clus.dat,method="euclidean"),method="single") # funcion "hclust" hace el clustering utilizando las distancias entre los datos
46 plot(sl.out)
47
47:1 (Top Level) : R Script Environment History
diam num [1:12, 1:12] 0...
golub Large matrix (1159, ...
golub.g_ Large matrix (9153, ...
sl.clus num [1:5, 1:2] 1...
Values
a num [1:3] 1 1 6
ALL Large ExpressionSet ...
b num [1:3] 4 5 1
closeto_ List of 1
dist.cy_ Class 'dist' atomic ...
Files Plots Packages Help Viewer
Zoom Export
Cluster Dendrogram
Height
0.0 1.5 3.0
g1 g2 g3 g4
dist(sl.clus.dat, method = "euclidean")
hclust (*, "single")

```

Obtenemos la siguiente grafica:



```

dist(sl.clus.dat, method = "euclidean")
hclust (*, "single")

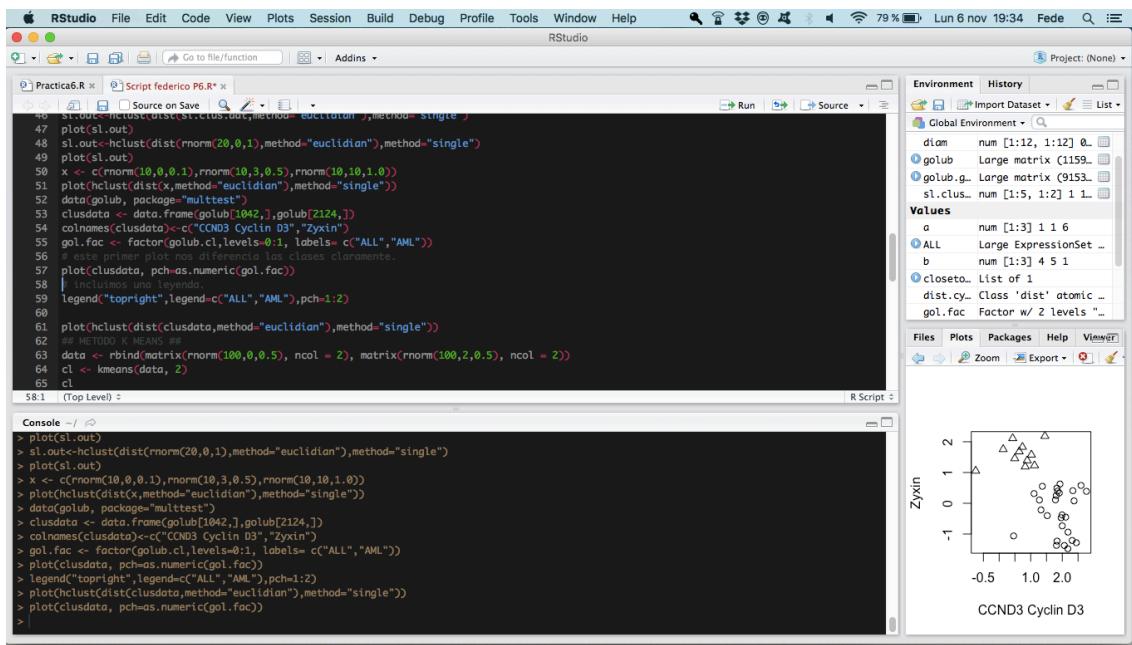
```

RStudio

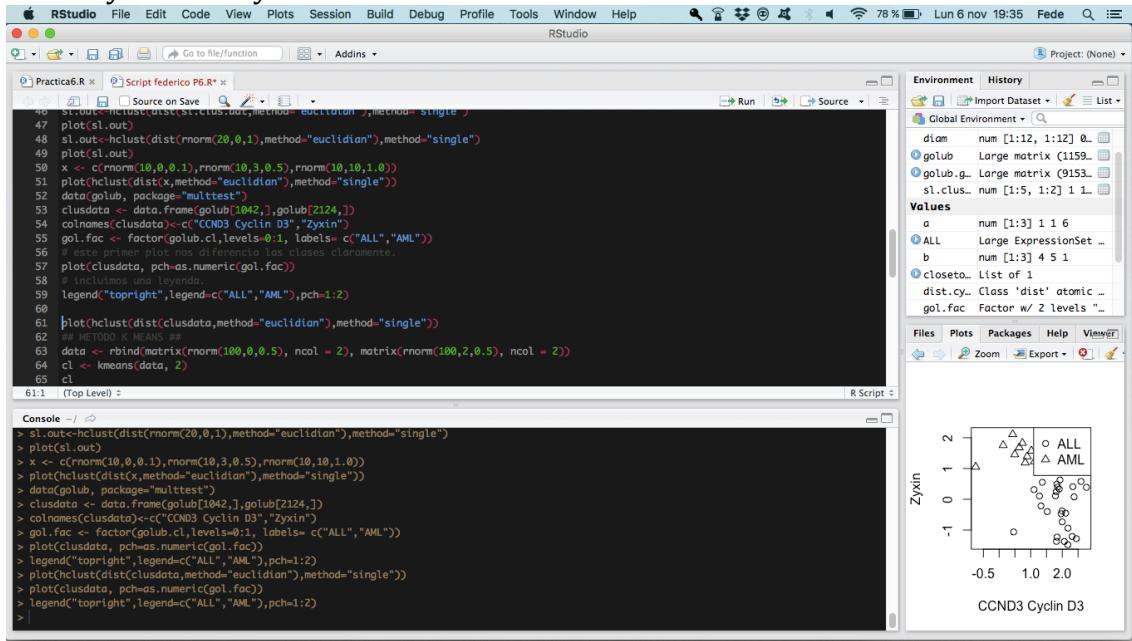
```

Practica6.R x Script federico P6.R x
Source on Save Run Source
41 sl.clus.dat
42 plot(sl.clus.dat,type="n", xlim=c(0,6), ylim=c(0,6))
43 text(sl.clus.dat,labels=rownames(sl.clus.dat))
44 print(dist(sl.clus.dat,method="euclidean"),digits=3)
45 # funcion "hclust" hace el clustering utilizando las distancias entre los datos
46 sl.out<-hclust(dist(rnorm(20,0,1)),method="euclidean"),method="single")
47 plot(sl.out)
48 sl.out<-hclust(dist(rnorm(20,0,1)),method="euclidean"),method="single")
49 plot(sl.out)
50 x <- c(rnorm(20,0,1),rnorm(10,3,0.5),rnorm(10,10,1.0))
51 plot(hclust(dist(x),method="euclidean"),method="single")
52 data(golub, package="multtest")
53 clusdata <- data.frame(golub[1042,],golub[2124,])
54 colnames(clusdata) <- data.frame(...,row.names = NULL,check.rows = FALSE, check.names = TRUE, fix.empty.names = TRUE, stringsAsFactors = default.stringsAsFactors())
55 gol.Fac <- factor(golub[,1042],levels = c("I","II","III","IV"))
56 # este print nos muestra las clases claramente.
57 plot(clusdata,pch=as.numeric(gol.Fac))
58 # incluyendo una leyenda
59 legend("topright",legend=c("ALL","AML"),pch=1:2)
60
60:25 (Top Level) : R Script Environment History
diam num [1:12, 1:12] 0...
golub Large matrix (1159, ...
golub.g_ Large matrix (9153, ...
sl.clus num [1:5, 1:2] 1...
Values
a num [1:3] 1 1 6
ALL Large ExpressionSet ...
b num [1:3] 4 5 1
closeto_ List of 1
dist.cy_ Class 'dist' atomic ...
golub.cl num [1:38] 0 0 0 0 0...
index int [1:12] 85 962 10...
names List of 2
sl.out List of 7
Files Plots Packages Help Viewer
Zoom Export
Cluster Dendrogram
Height
0 2 4
dist(x, method = "euclidean")
hclust (*, "single")

```



E incluyendo la leyenda:



Metodo K means;

Analisis de componentes principales:

The screenshot shows an RStudio interface with the following details:

- Top Bar:** Shows the RStudio logo, menu items (File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Window, Help), system status (74% battery, Lun 6 nov 19:43, Fedora), and a search bar.
- Project:** Project: (None)
- Code Editor:** The active file is "Practica6.R" (Script federico P6.R*). The code performs PCA on the Zebrafish dataset:

```
## ANALYSIS COMPONENTES PRINCIPALES
70 Z<-matrix(c(1.62,1.22,-0.40,0.79,0.93,0.97,-1.38,-1.08,-0.17,-0.96,-0.61),nrow=6,byrow=TRUE)
71 K<-eigen(Z)
72 print(K,digits=2)
73 print(Z%%K$vec, digits=2)
74 pca<-princomp(Z,center=TRUE,cor=TRUE,scores=TRUE)
75 pca<-princomp(Z,cor=TRUE,scores=TRUE)
76 pcam$comps
77 pcam$loadings
78 eigen(cor(golub))$values[1:5]
79
```

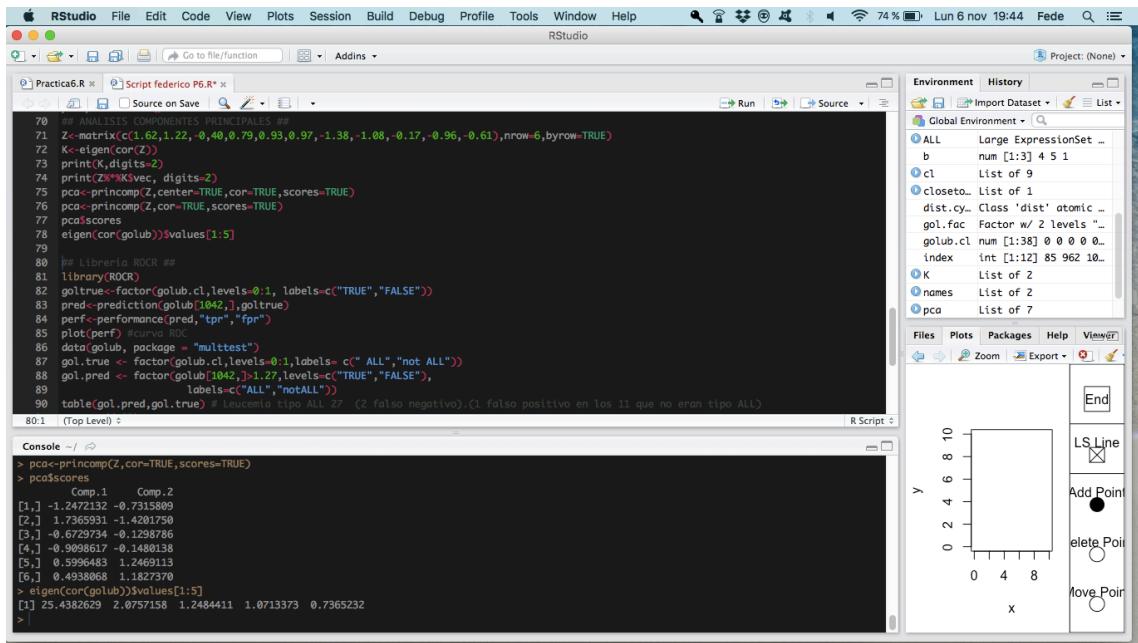
- Console:** Displays the results of the PCA command:

```
> Z<-matrix(c(1.62,1.22,-0.40,0.79,0.93,0.97,-1.38,-1.08,-0.17,-0.96,-0.61),nrow=6,byrow=TRUE)
> K<-eigen(Z)
> print(K,digits=2)
eigen() decomposition
$values
[1] 1.08 0.92

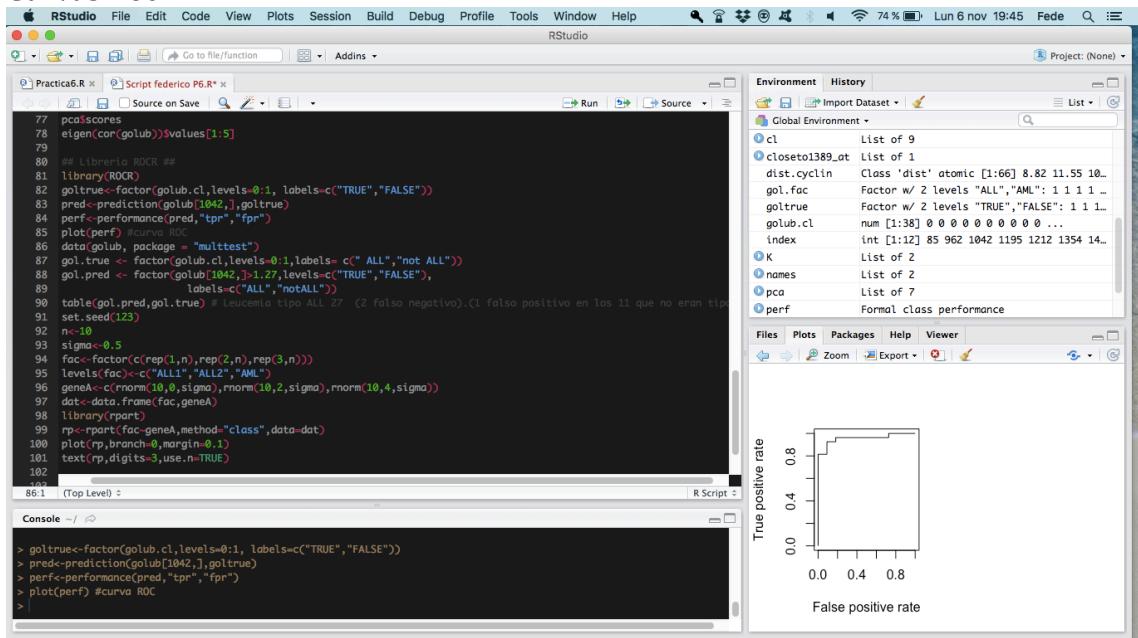
$vectors
[,1] [,2]
[1,] -0.71 -0.71
[2,] 0.71 -0.71

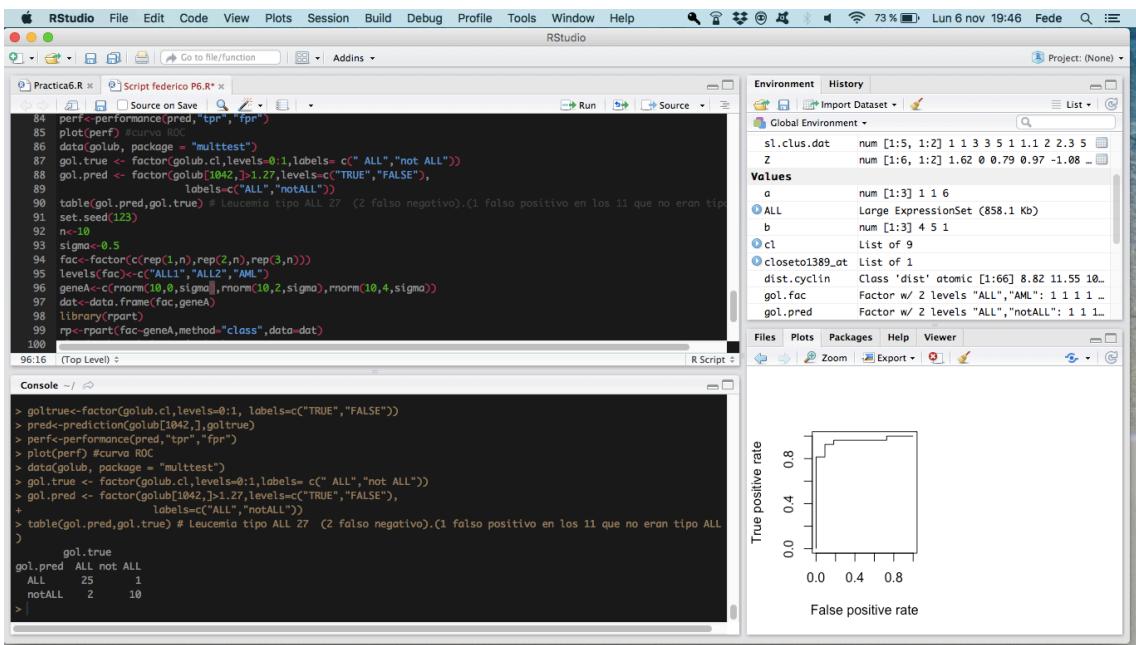
> print(Z%%K$vec, digits=2)
[,1] [,2]
[1,] -0.283 -2.01
[2,] 28.284 -28.28
[3,] 0.099 -1.22
[4,] -1.662 0.29
[5,] 0.643 0.88
[6,] 0.247 1.11
> pca<-princomp(Z,center=TRUE,cor=TRUE,scores=TRUE)
Warning message:
In princomp.default(Z, center = TRUE, cor = TRUE, scores = TRUE) :
```

- Environment:** Shows the global environment with objects like Large ExpressionSet, b, cl, closeto, dist, dist.cy, gol, golub, golub.cl, index, K, names, and pca.
- Plots:** A scatter plot of the first two principal components (PC1 vs PC2) with points labeled by row names. The plot includes controls for "End", "LS Line", "Add Point" (with a black dot at approximately (8, 4)), "Delete Point" (with a circle outline), and "Move Point" (with a circle outline).



Curvas Roc:





Leucemia tipo ALL 27 (2 falso negativo).(1 falso positivo en los 11 que no eran tipo ALL)

