Laboratorio 1

Franco Mendoza Muraira A01383399

2023-11-28

```
##
## Attaching package: 'psych'

## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

Problema 1

Descomposición espectral

```
A = matrix(c(4.4,0.8,0.8,5.6),nrow = 2,ncol = 2)
unicos = eigen(A)
valores = unicos$values
vectores= unicos$vectors

d1 = vectores[,1]%*% t(vectores[,1])* valores[1]
d2 = vectores[,2]%*% t(vectores[,2])* valores[2]
A
```

```
##      [,1] [,2]
## [1,]  4.4  0.8
## [2,]  0.8  5.6

## Descomposición 1:

##      [,1] [,2]
## [1,]  1.2  2.4
## [2,]  2.4  4.8

##
##
## Descomposición 2:

##      [,1] [,2]
## [1,]  3.2 -1.6
## [2,] -1.6  0.8
```

```
d1+d2
```

```
##      [,1] [,2]
## [1,]  4.4  0.8
## [2,]  0.8  5.6
```

Problema 2

Probabilidad

$\mu = (2.5, 4)$ $P(X \leq x), x = (2, 3)$

```
co = matrix(c(1.2,0,0,2.3),nrow=2,ncol=2)
mu = c(2.5,4)
x = c(2,3)
cat("La probabilidad de que P(X<=x) donde x' = (2,3) es:",pmnorm(x=x,mean=mu,varcov=co))
```

```
## La probabilidad de que P(X<=x) donde x' = (2,3) es: 0.08257333
```

Problema 3

Mahalanobis

```
##         x1        x2        x3
## 1 12.89603 11.716570  9.855675
## 2 13.62424 13.147912  8.534047
## 3 11.97687  6.480880  9.451111
## 4 12.51332  7.550823 10.193486
## 5 11.68629  9.047096  8.689940
## 6 12.79996  8.678760  9.982614
```

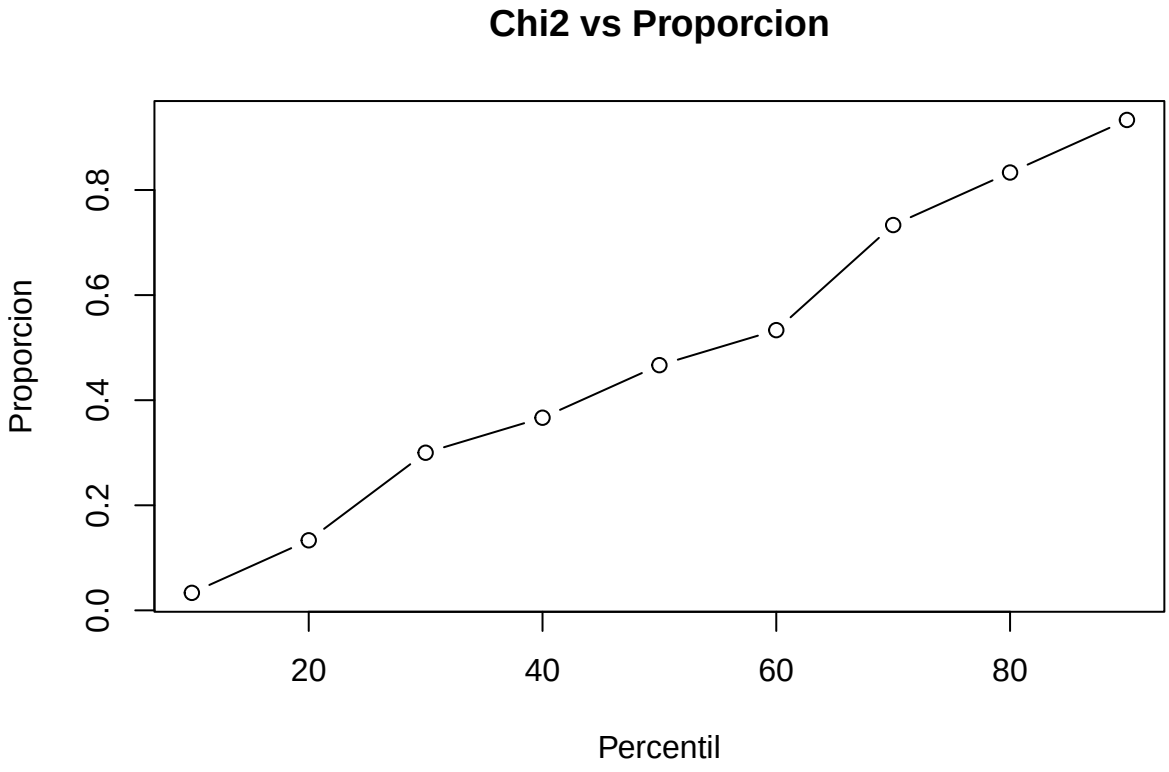
```
maha = mahalanobis(df,colMeans(df),cov(df))
maha
```

```
##  [1] 1.0608504 4.8048318 4.5991621 2.8230132 1.3010730 1.6507208 1.1891472
##  [8] 0.7734180 2.8264674 2.9562497 6.7751957 3.5069469 6.1386363 3.1158347
## [15] 1.9492944 2.0794492 0.6335702 1.0103182 3.5574749 1.8450450 3.0668657
## [22] 4.6485180 3.8619795 1.3044588 0.8957396 3.6209539 2.3639912 4.5362760
## [29] 0.4157899 7.6887282
```

```
percent = c(seq(10,90,10))
prop = c()
for (i in seq_along(percent)){
  prop[i] = sum(maha< qchisq(percent[i]/100,df = ncol(df)))
}
prop = prop/length(maha)

res = data.frame(Percentil = percent, Proporcion = prop)
plot(res,main = "Chi2 vs Proporcion", type ="both")
```

```
## Warning in plot.xy(xy, type, ...): plot type 'both' will be truncated to first
## character
```



Debido a como se ve la grafica, se puede deducir que los datos son normales, esto debido a que se ve como una linea recta, aunque no perfectamente recta.

Problema 4

```
mvn(df,mvnTest = "mardia",univariateTest = "SW")$multivariateNormality
```

```
##              Test         Statistic           p value Result
## 1 Mardia Skewness  7.68091713334412 0.659972337630838    YES
## 2 Mardia Kurtosis -1.17376271687907 0.240490081413617    YES
## 3             MVN              <NA>              <NA>    YES
```

¿Cuál es el valor p de correspondientes a los Test de sesgo y curtosis de la Prueba de normalidad multivariada de Mardia?

El valor p de la prueba de sesgo multivariada fue de 0.66, y el de curtosis fue de 0.24

```
mvn(df,mvnTest = "hz",univariateTest = "SW")
```

```
## $multivariateNormality
##            Test        HZ   p value MVN
## 1 Henze-Zirkler 0.5964568 0.5033687 YES
##
## $univariateNormality
##           Test  Variable Statistic   p value Normality
## 1 Shapiro-Wilk    x1        0.9883    0.9794    YES
## 2 Shapiro-Wilk    x2        0.9697    0.5317    YES
## 3 Shapiro-Wilk    x3        0.9756    0.7005    YES
##
## $Descriptives
##     n      Mean   Std.Dev    Median       Min      Max      25th     75th
## x1 30 12.142503 0.8932853 12.092415 10.347778 14.43922 11.681746 12.54419
## x2 30 10.431828 1.8651858 10.670746  6.480880 13.42733  9.113761 11.71012
## x3 30  9.772519 1.5218684  9.722361  7.018689 13.08560  8.573020 10.63552
##         Skew    Kurtosis
## x1  0.262059 -0.03657444
## x2 -0.254847 -0.87399859
## x3  0.361408 -0.70252667
```

¿Cual es el valor p de la prueba de normalidad multivaridad de Henze-Zirkler's?

El valor pu de la prueba de normalidad multivariada de Henze-Zirkler's fue de 0.503

A un alfa = 0.05, ¿qué se concluye?

Debido a que el valor p de la prueba de normalidad multivariada de Henze-Zirkler's fue de 0.503, se puede concluir que los datos son normales.

Problema 5

```
##   Longitud Diametro Altura PesoTotal Pesodesvainado Pesovisceras Pesocorteza
## 1    0.530    0.420  0.135    0.6770         0.2565       0.1415       0.210
## 2    0.530    0.415  0.150    0.7775         0.2370       0.1415       0.330
## 3    0.545    0.425  0.125    0.7680         0.2940       0.1495       0.260
## 4    0.550    0.440  0.150    0.8945         0.3145       0.1510       0.320
## 5    0.525    0.380  0.140    0.6065         0.1940       0.1475       0.210
## 6    0.535    0.405  0.145    0.6845         0.2725       0.1710       0.205
```

```
## [1] 30
```

A. Realice la prueba de normalidad de Mardia y la prueba de Anderson Darling con las variables X1, X2 y X3 y de la conclusión a un nivel se significación de 0.05. Interprete coeficientes de sesgo y curtosis de Mardia resultantes. Indique qué variables resultaron leptocúrticas, platicúrticas y mesocúrticas.

```
X = data[,1:3]
mvn(X,mvnTest = "mardia",univariateTest = "SW")
```

```
## $multivariateNormality
##              Test         Statistic            p value Result
## 1 Mardia Skewness  17.0312537859682 0.0736752675872285    YES
## 2 Mardia Kurtosis 0.848747898502705  0.396021587220212    YES
## 3             MVN              <NA>               <NA>    YES
##
## $univariateNormality
##           Test  Variable Statistic   p value Normality
## 1 Shapiro-Wilk Longitud     0.9310    0.0522    YES
## 2 Shapiro-Wilk Diametro     0.9352    0.0676    YES
## 3 Shapiro-Wilk  Altura      0.9741    0.6560    YES
##
## $Descriptives
##           n      Mean    Std.Dev Median   Min   Max    25th    75th      Skew
## Longitud 30 0.5416667 0.06695770 0.5375 0.440 0.705 0.50125 0.56500 0.6684604
## Diametro 30 0.4265000 0.05841513 0.4225 0.335 0.560 0.39250 0.44375 0.6664298
## Altura   30 0.1420000 0.02558286 0.1400 0.100 0.200 0.12500 0.15875 0.2730008
##             Kurtosis
## Longitud  0.07671835
## Diametro -0.04598915
## Altura   -0.58397509
```

En vez de la prueba de Anderson Darling, se utilizo la prueba de Shapiro-Wilk, debido a que hay menos de 50 datos en los datos que se nos dieron.

En las pruebas de normalidad de Mardia, podemos ver que los valores p de las pruebas son mayores a 0.05, por lo que se puede concluir que los datos son normales, tomando en cuenta que H_0 es que las variables siguen una distribución normal, y H_1 es que las variables no siguen una distribución normal multivariada.

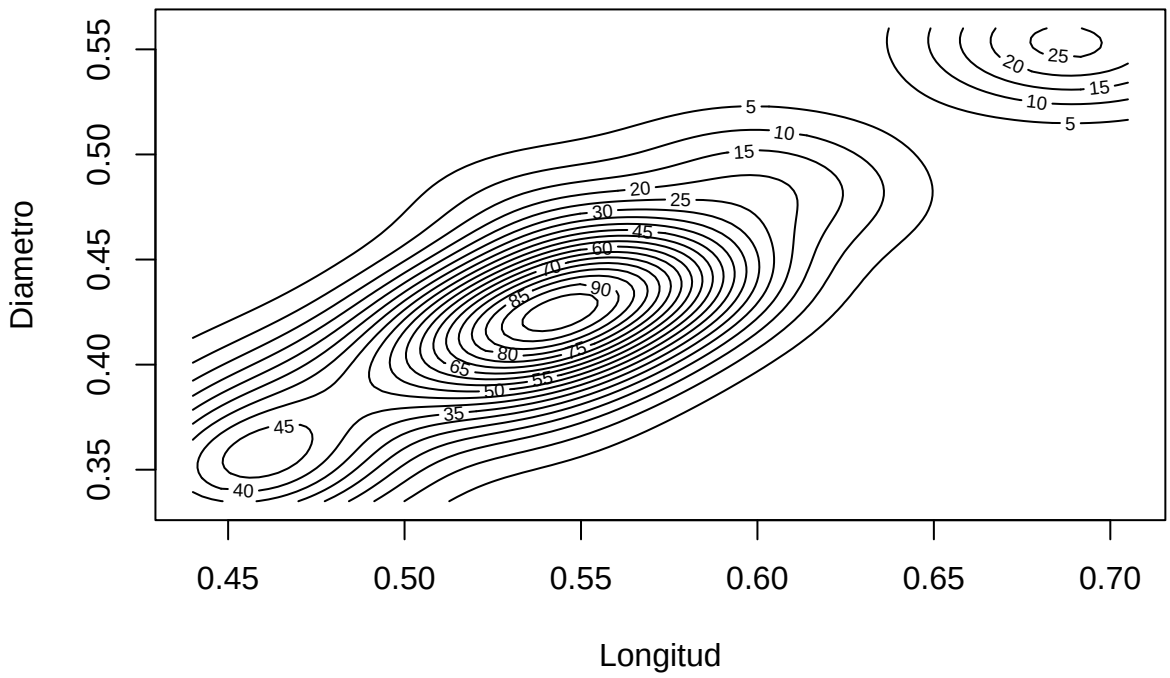
En cuanto a la curtosis de cada variable, podemos ver que la variable Longitud, con un valor de 0.0767, es leptocurtica, la variable Diametro, con un valor de -0.046, es mesocúrtica, y la variable Altura, con un valor de -0.584, es leptocurtica.

1. Mesocúrtica: Similar a una distribución normal estándar, con datos moderadamente distribuidos alrededor de la media.
2. Leptocúrtica: Más puntiaguda que una distribución normal, con datos concentrados cerca de la media y colas más pesadas.
3. Platicúrtica: Más aplanada que una distribución normal, con datos dispersos desde la media y colas más ligeras.

B. Elabore la gráfica de contorno de la normal multivariada obtenida anteriormente.

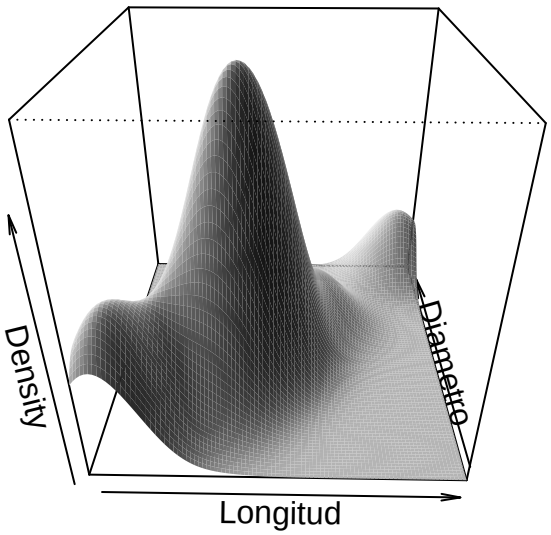
Longitud y Diametro

```
mvn(X[1:2], mvnTest ="mardia", multivariatePlot ="contour")$multivariatePlot
```




```
## NULL
```

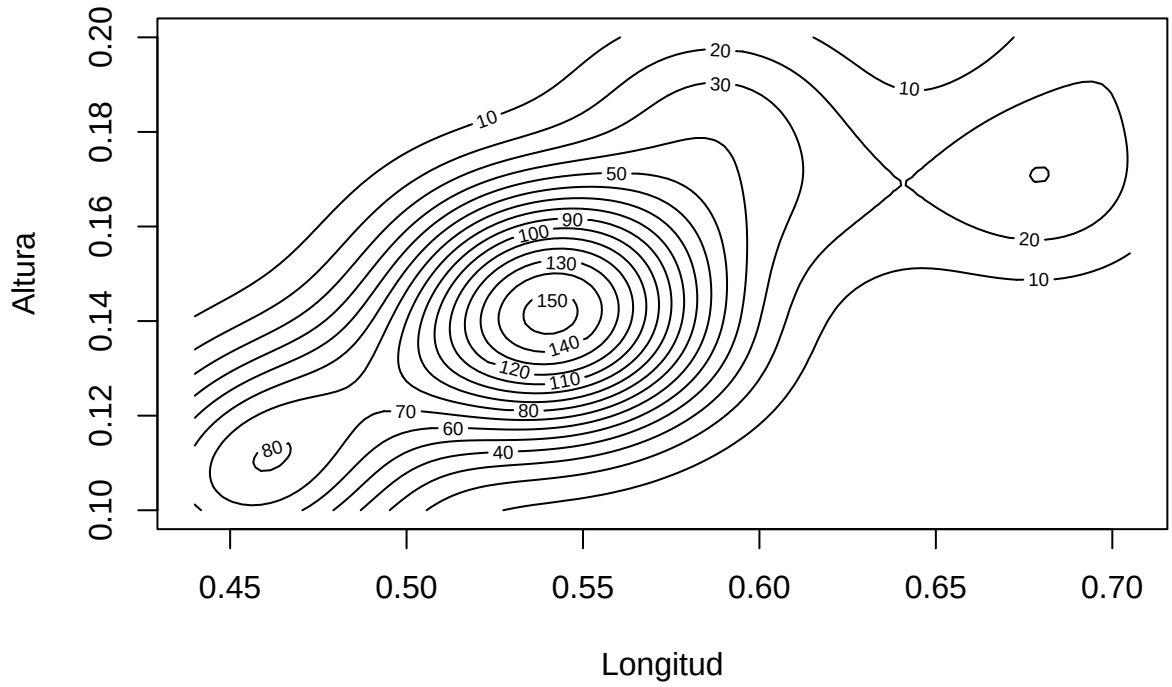
```
mvn(X[1:2], mvnTest ="mardia", multivariatePlot ="persp")$multivariatePlot
```




```
## NULL
```

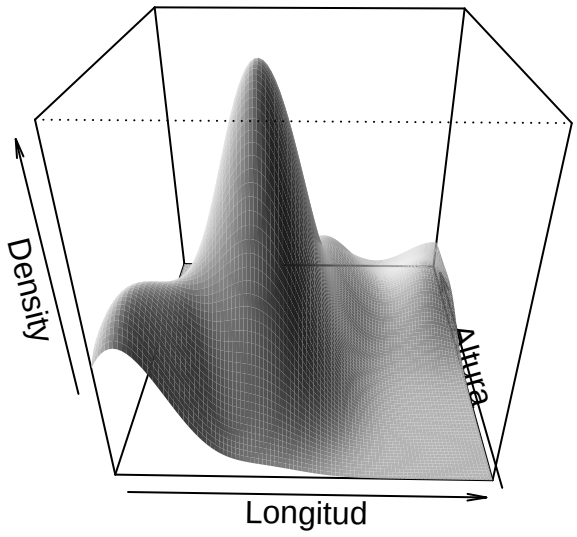
Longitud y Altura

```
mvn(X[c(1,3)], mvnTest ="mardia", multivariatePlot ="contour")$multivariatePlot
```




```
## NULL
```

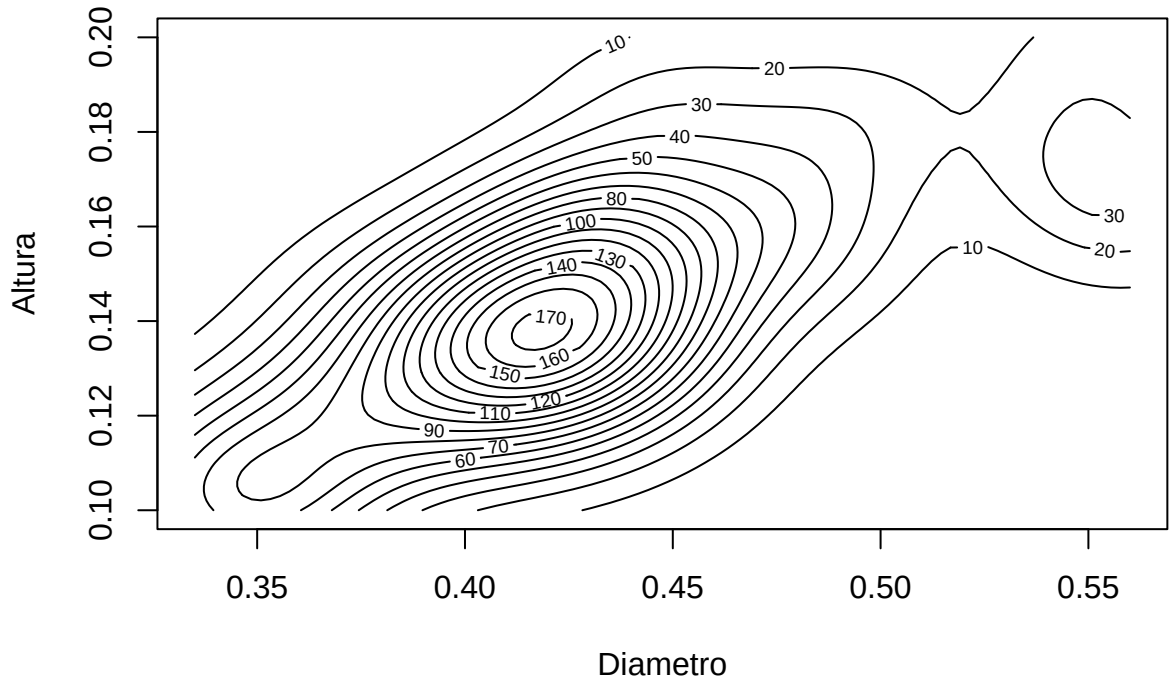
```
mvn(X[c(1,3)], mvnTest ="mardia", multivariatePlot ="persp")$multivariatePlot
```




```
## NULL
```

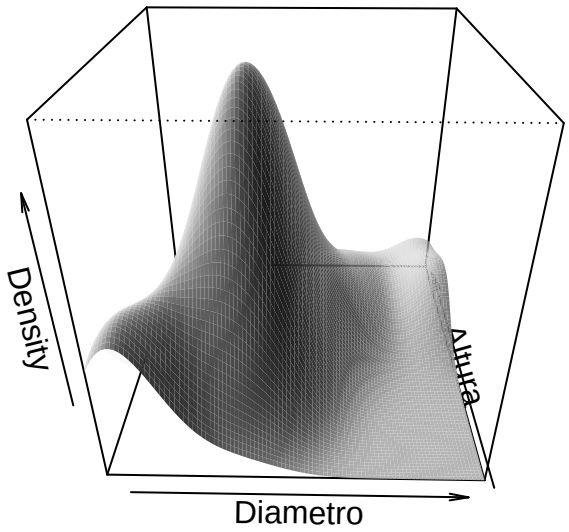
Diametro y Altura

```
mvn(X[c(2,3)], mvnTest ="mardia", multivariatePlot ="contour")$multivariatePlot
```




```
## NULL
```

```
mvn(X[c(2,3)], mvnTest ="mardia", multivariatePlot ="persp")$multivariatePlot
```




```
## NULL
```

C. Con el vector de medias y la matriz de covarianza de la normal multivariada en en el inciso A, calcule la probabilidad de que P(X <= (0.25,0.25,0.25))

```
mu = colMeans(X)
co = cov(X)
x = c(0.25,0.25,0.25)
cat("La probabilidad de que las 3 variables sean igual a el vector x =",pmnorm(x=x,mean=mu,varcov=co))
```

```
## La probabilidad de que las 3 variables sean igual a el vector x = 6.623513e-06
```

D. Con el total de datos Olmos.csv calcula la distancia de Mahalanobis de cada observación al centroide (vector de medias) con respecto a la matriz de covarianzas. ¿Qué observación está más alejada, según la distancia de Mahalanobis, del centroide? ¿Qué observación está más cercana?

```
maha = mahalanobis(data,colMeans(data),cov(data))
cat("La observación más alejada es la número",which.max(maha),"con una distancia de",max(maha))
```

```
## La observación más alejada es la número 14 con una distancia de 20.23209
```

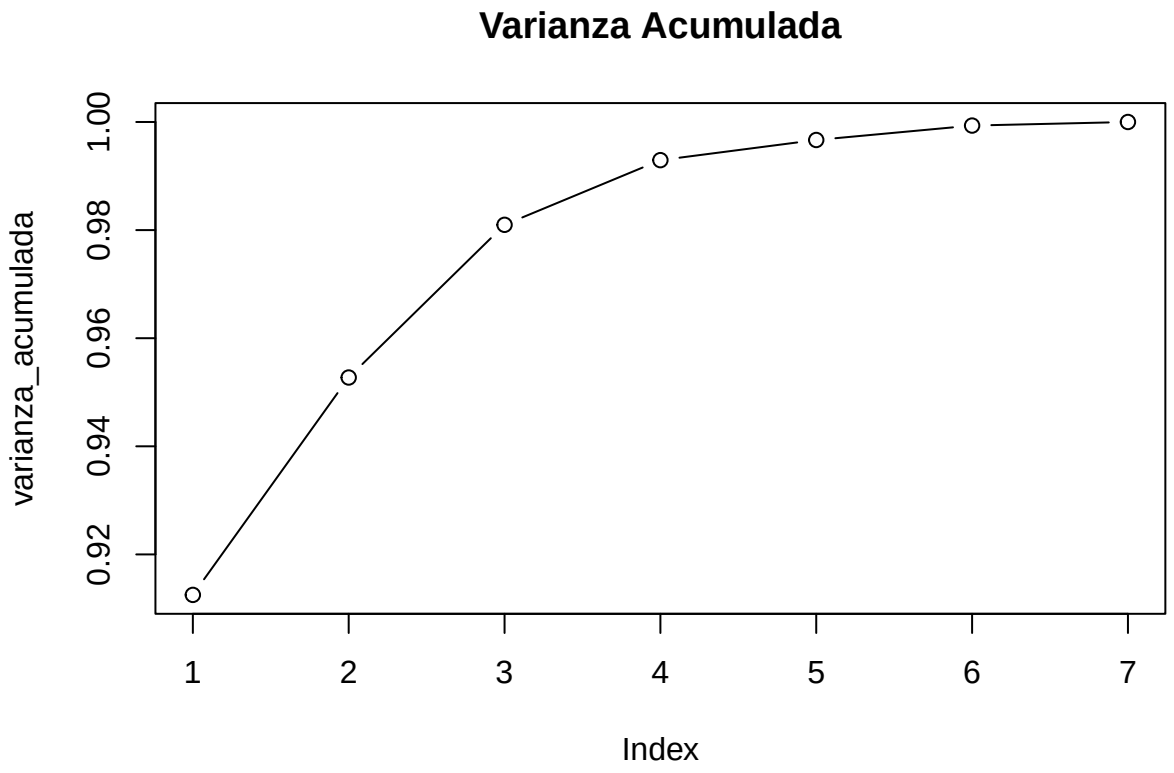
```r
cat("\nLa observación más cercana es la número",which.min(maha),"con una distancia de",min(maha))
```

```
##
## La observación más cercana es la número 6 con una distancia de 2.561069
```

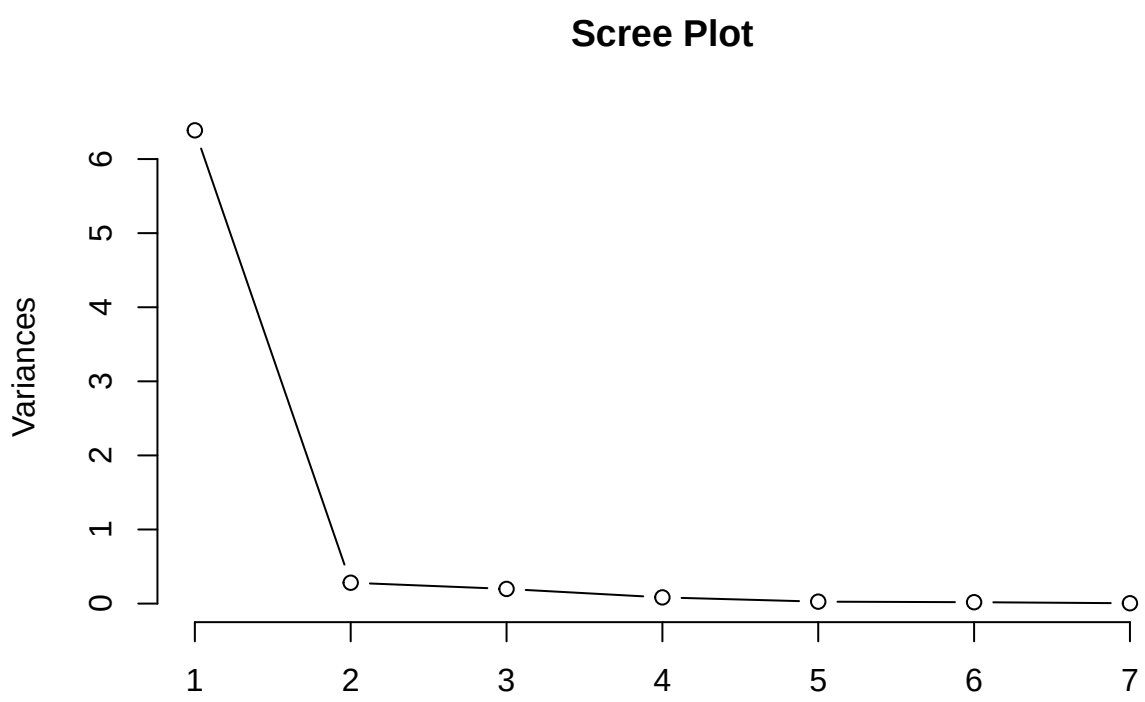
E. Aplica un análisis de componentes principales a los datos y con base en al menos tres criterios (por ejemplo, porcentaje de variación acumulada, gráfica de Scree y los valores de las cargas) determinar cuántos componentes son suficientes para explicar razonablemente la mayoría de la variación.

```r
pca = prcomp(data,center = TRUE,scale = TRUE)
varianza_acumulada = cumsum(pca$sdev^2/sum(pca$sdev^2))
plot(varianza_acumulada,type = "both",main = "Varianza Acumulada")
```

```
## Warning in plot.xy(xy, type, ...): plot type 'both' will be truncated to first
## character
```




```r
plot(pca,type = "l",main = "Scree Plot")
```




```
# Valores de las cargas
pca$rotation
```

```
##                     PC1         PC2         PC3         PC4         PC5
## Longitud       0.3829206  0.01984410 -0.11178144 -0.82798216 -0.2777313
## Diametro       0.3895735 -0.01072829 -0.23266157 -0.19008711  0.6094553
## Altura         0.3571618  0.65634059  0.55239439  0.08997344  0.2658877
## PesoTotal      0.3913471 -0.14929602 -0.20861352  0.19596466 -0.0575013
## Pesodesvainado 0.3762967 -0.53901550  0.01823927  0.25053286  0.3175768
## Pesovisceras   0.3752590 -0.32334612  0.55274938  0.10746252 -0.4720819
## Pesocorteza    0.3721137  0.38900954 -0.52806230  0.39686290 -0.3920595
##                       PC6         PC7
## Longitud       -0.27298918 -0.05719002
## Diametro        0.61556971 -0.08655058
## Altura         -0.23308811  0.05823544
## PesoTotal      -0.09146657  0.85496423
## Pesodesvainado -0.52753646 -0.35442067
## Pesovisceras    0.45211736 -0.10137310
## Pesocorteza     0.03390893 -0.34495301
```

En el primer grafico, de el porcentaje de varianza acumulada, podemos ver que con 2 componentes se explica el 95% de la varianza, y con 3 componentes se explica aproximadamente el 98% de la varianza. En el segundo grafico, podemos ver que el codo se forma en el componente23, por lo que se puede decir que a pesar de que con 3 componentes se explica la mayoría de la varianza, con 2 puede ser mejor debido a que es hasta donde hay mayor cambio en explicacion, y de ahi disminuye. Además, al analizar los valores

de carga, se observa cómo cada variable contribuye a la formación de los componentes principales. Por ejemplo, las variables de 'Longitud', 'Diámetro' y 'PesoTotal' muestran cargas similares y considerables en los primeros componentes, lo que sugiere una fuerte relación entre estas medidas físicas. Por otro lado, 'Altura', 'PesoVisceras' y 'PesoCorteza' exhiben cargas notables en diferentes componentes, indicando su influencia en aspectos distintos de la estructura de los datos. Estos valores de carga proporcionan una comprensión detallada de cómo cada atributo original contribuye a la formación de los componentes principales y, por ende, a la explicación de la variabilidad de los datos.

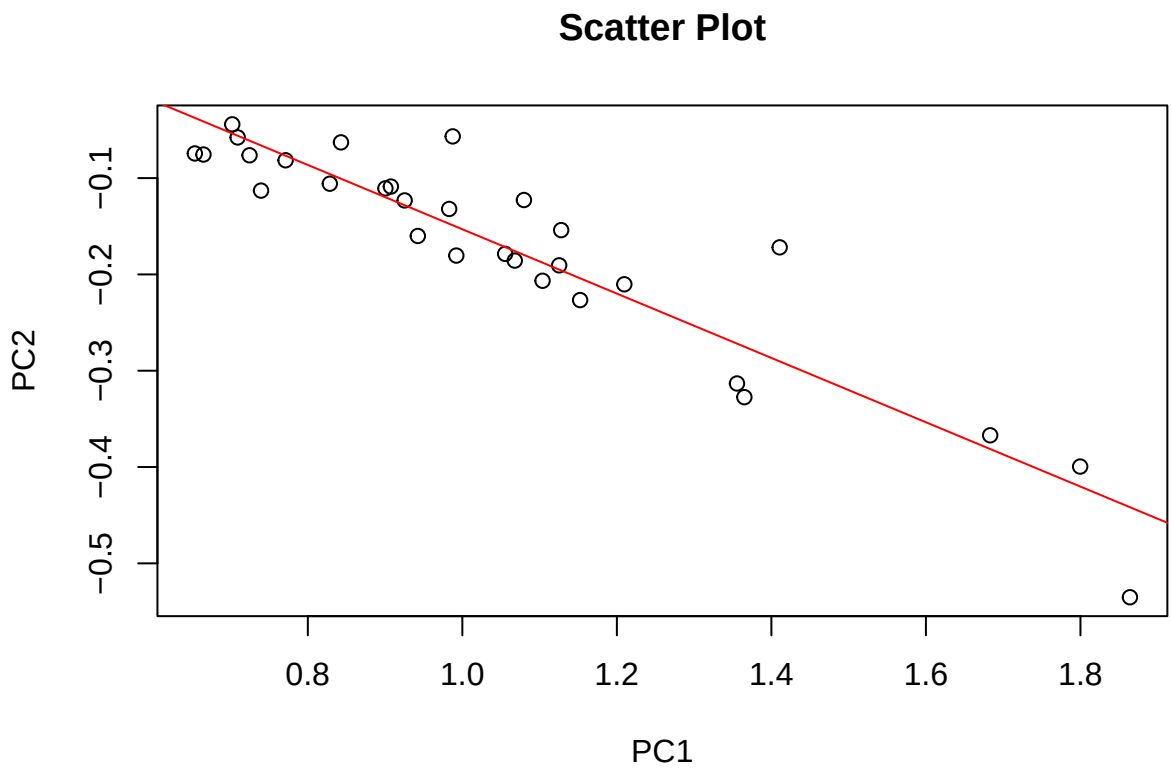
F. Escribir las combinaciones lineales de los Componentes principales en función de las variables y cargas obtenidas de los componentes principales resultantes.

```
for (i in 1:2){
  cat("PC",i,"=",round(pca$rotation[,i][1],3),"X1 +",round(pca$rotation[,i][2],3),"X2 +",round(pca$rota
}
```

```
## PC 1 = 0.383 X1 + 0.39 X2 + 0.357 X3 0.391 X4 + 0.376 X5 + 0.375 X6 + 0.372 X7
##
## PC 2 = 0.02 X1 + -0.011 X2 + 0.656 X3 -0.149 X4 + -0.539 X5 + -0.323 X6 + 0.389 X7
```

G. Utilizando los dos primeros componentes hacer una gráfica de dispersión de las puntuaciones. Comentar el gráfico en función de la variabilidad.

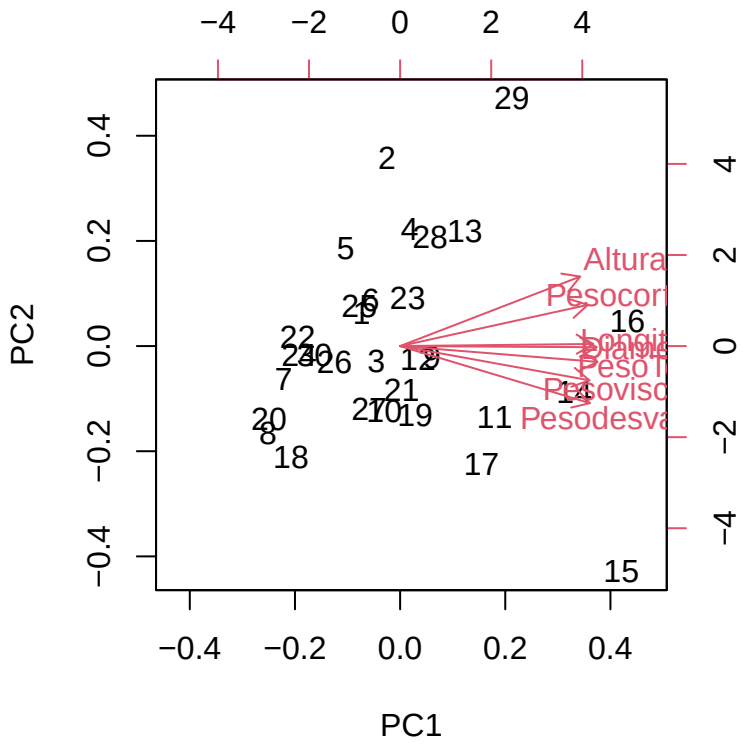
```
scores= as.matrix(data) %*% pca$rotation
model = lm(scores[,2]~scores[,1])
plot(scores[,1],scores[,2],main = "Scatter Plot",xlab = "PC1",ylab = "PC2")
abline(model,col="red")
```



Con este gráfico de dispersión, se puede ver una tendencia negativa entre las 2 componentes, lo que indica que a mayor valor de la componente 1, menor valor de la componente 2. Además, se puede ver que la mayoría de los datos se encuentran en el segundo cuadrante.

H. Hacer un gráfico vectorial de las variables e interpretar sus relaciones.

```
biplot(pca)
```



En este gráfico vectorial, se puede ver que todas las variables afectan de gran manera a el primer componente, de manera positiva, mientras que al segundo componente, las variables lo afectan poco, mas que nada Altura, y Peso desvainado son las que mas lo afectan, de manera positiva y negativa respectivamente. Tambien se puede ver que Peso corteza afeca un poco mas que las demas variables positivamente al segundo componente, pero las demas aun afectando poco estan mas del lado negativo o neutral.

Problema 6

A. Justifique por qué es adecuado el uso del Análisis factorial (hacer la prueba de esfericidad de Bartlett y KMO).

```
data = read.csv("olmos.csv")
head(data)
```

```
##   Longitud Diametro Altura PesoTotal Pesodesvainado Pesovisceras Pesocorteza
## 1    0.530    0.420  0.135    0.6770         0.2565       0.1415       0.210
## 2    0.530    0.415  0.150    0.7775         0.2370       0.1415       0.330
## 3    0.545    0.425  0.125    0.7680         0.2940       0.1495       0.260
## 4    0.550    0.440  0.150    0.8945         0.3145       0.1510       0.320
## 5    0.525    0.380  0.140    0.6065         0.1940       0.1475       0.210
## 6    0.535    0.405  0.145    0.6845         0.2725       0.1710       0.205
```

```r
cortest.bartlett(data)
```

```
## R was not square, finding R from data

## $chisq
## [1] 426.6447
##
## $p.value
## [1] 2.795635e-77
##
## $df
## [1] 21
```

```r
KMO(data)
```

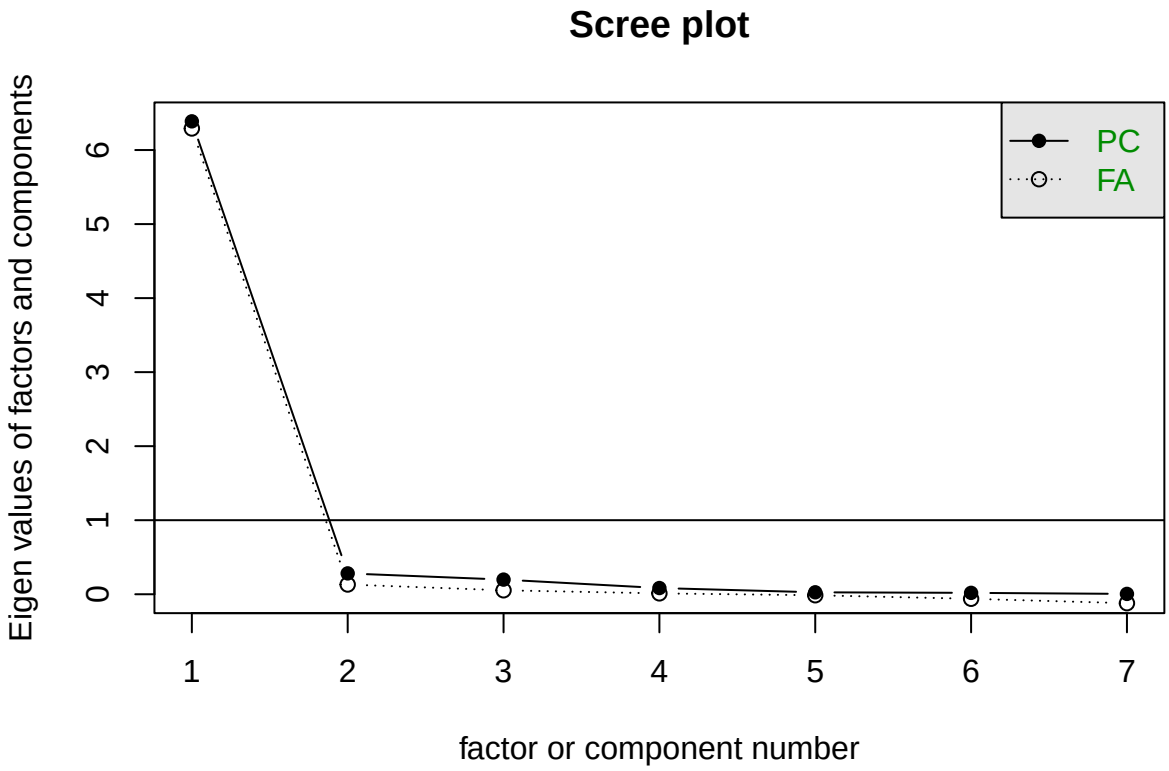
```
## Kaiser-Meyer-Olkin factor adequacy
## Call: KMO(r = data)
## Overall MSA =  0.84
## MSA for each item =
##       Longitud       Diametro         Altura      PesoTotal Pesodesvainado
##           0.92           0.93           0.82           0.78           0.83
##   Pesovisceras    Pesocorteza
##           0.83           0.76
```

Con el valor p de 2.795635e-77 en la prueba de Bartlett, que demuestra si hay o no hay independencia entre las variables, se puede ver que es menor a 0.05, por lo que se rechaza la hipotesis nula, por lo que es adecuado el uso del analisis factorial. Con el valor de KMO, que demuestra si las variables son adecuadas para el analisis factorial, se puede ver que es mayor a 0.5, por lo que se puede decir que las variables son adecuadas para el analisis factorial.

B. Justifique el número de factores principales que se utilizarán en el modelo.

```r
scree(cor(data))
```

```
## Warning in fa.stats(r = r, f = f, phi = phi, n.obs = n.obs, np.obs = np.obs, :
## The estimated weights for the factor scores are probably incorrect.  Try a
## different factor score estimation method.
```



Debido a que el codo se hace en los 2 factores, se usaran 2 factores principales.

C. Identifique las comunalidades de los factores del modelo propuesto, y los errores: interprete si se necesita un nuevo factor.

```
factor = factanal(data,factors = 2,scores = "Bartlett")
factor
```

```
##
## Call:
## factanal(x = data, factors = 2, scores = "Bartlett")
##
## Uniquenesses:
##       Longitud       Diametro         Altura      PesoTotal Pesodesvainado
##          0.092          0.034          0.229          0.005          0.005
##   Pesovisceras    Pesocorteza
##          0.095          0.027
##
## Loadings:
##                Factor1 Factor2
## Longitud       0.710   0.636
## Diametro       0.728   0.660
## Altura         0.747   0.462
## PesoTotal      0.693   0.718
```

```
## Pesodesvainado 0.497   0.865
## Pesovisceras   0.513   0.801
## Pesocorteza    0.866   0.472
##
##                Factor1 Factor2
## SS loadings      3.334   3.181
## Proportion Var   0.476   0.454
## Cumulative Var   0.476   0.931
##
## Test of the hypothesis that 2 factors are sufficient.
## The chi square statistic is 33.62 on 8 degrees of freedom.
## The p-value is 4.77e-05
```

Para 'Longitud', 'Diametro', 'Altura', 'PesoTotal', 'Pesodesvainado', 'Pesovisceras' y 'Pesocorteza', las comunalidades son relativamente altas, oscilando entre 0.005 y 0.229. Esto sugiere que una parte considerable de la varianza de estas variables está siendo explicada por los dos factores extraídos en el modelo.

Los loadings muestran la relación entre las variables originales y los factores. En este caso, los loadings son relativamente altos y sugieren una asociación significativa entre las variables y los dos factores extraídos.

La prueba de chi-cuadrado para determinar si 2 factores son suficientes indica que hay una diferencia significativa entre los datos observados y los datos esperados según el modelo de 2 factores.

En resumen, los dos factores parecen estar explicando una cantidad significativa de la varianza en los datos, ya que la mayoría de las comunalidades son altas y los loadings muestran asociaciones fuertes entre las variables y los factores. Sin embargo, la significancia estadística de la prueba chi-cuadrado sugiere que podría ser beneficioso considerar la inclusión de un tercer factor para capturar mejor la estructura subyacente de los datos. Tambien se tiene que considerar que con 2 factores ya se tiene una varianza acumulada de 0.93, por lo que no es necesario un tercer factor.

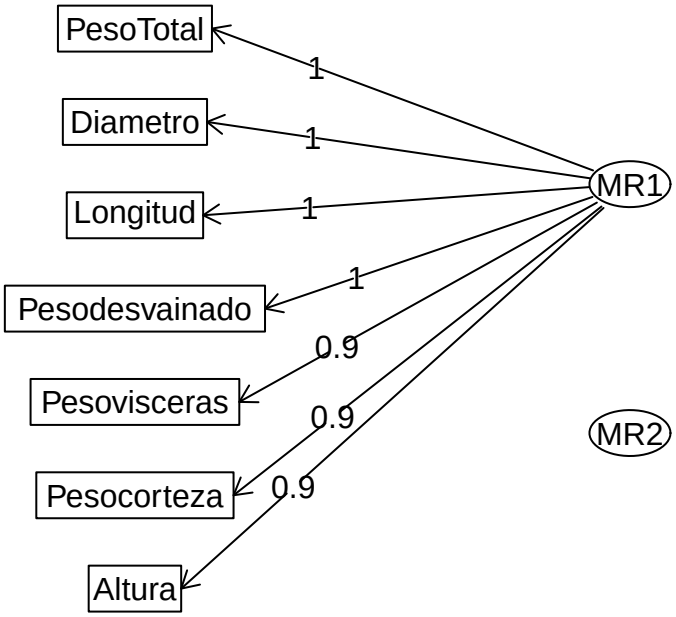
D. Encuentre con ayuda de un gráfico de variables qué conviene más sin rotación o con rotación varimax. (se puede ayudar con la función fa de la librería psych y el gráfico de la función fa.diagram)

```
factor_no_rotado = fa(data,nfactors = 2,rotate = "none")
```

```
## Warning in fa.stats(r = r, f = f, phi = phi, n.obs = n.obs, np.obs = np.obs, :
## The estimated weights for the factor scores are probably incorrect.  Try a
## different factor score estimation method.
```

```
fa.diagram(factor_no_rotado,main="Analisis Factorial sin rotar")
```

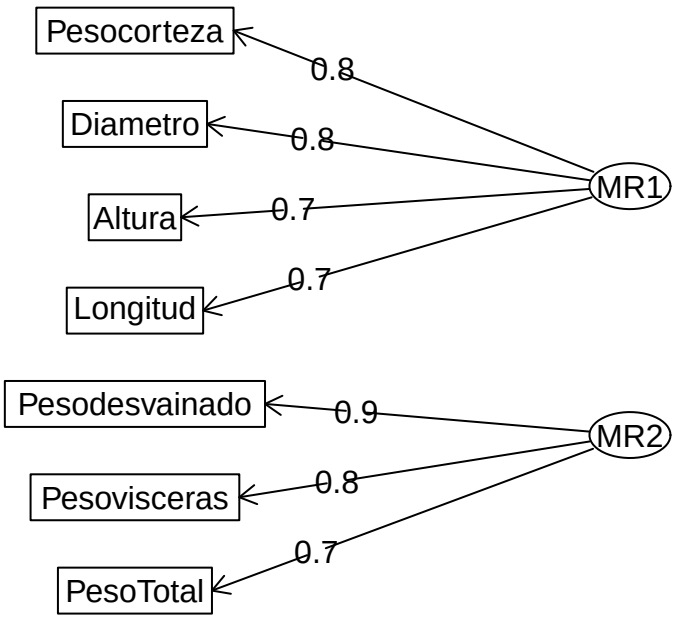
Analisis Factorial sin rotar




```
factor_rotado = fa(data,nfactors = 2,rotate = "varimax")
```

```
## Warning in fa.stats(r = r, f = f, phi = phi, n.obs = n.obs, np.obs = np.obs, :
## The estimated weights for the factor scores are probably incorrect.  Try a
## different factor score estimation method.
```

```
fa.diagram(factor_rotado,main="Analisis Factorial con varimax")
```



En las graficas se puede ver claramente que con la rotacion varimax, los factores estan mas separados, por lo que se puede decir que conviene mas con rotacion varimax, esto porque la explicacion de parte de los 2 componentes es mas clara hacia que variables afectan. Por ejemplo, en el componente 1, se pueden tomar las medidas del pez en mm o cm, y solo pesocorteza destaca, mientras que en el componente 2, se pueden tomar los pesos del pez en gramos o kg.

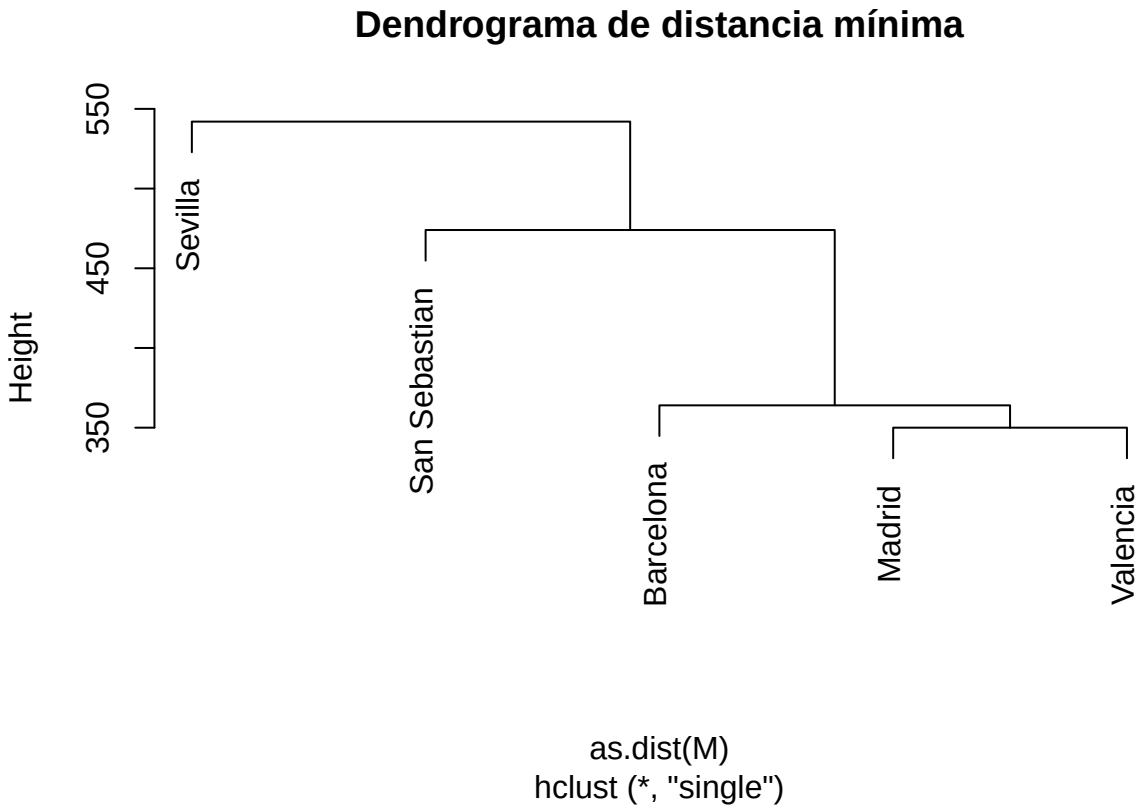
Problema 7

```
Mpre = matrix(c(0,639,606,1181,364,0,0,474,542,350,0,0,0,908,597,0,0,0,0,679,0,0,0,0,0),ncol=5, dimnames
M = Mpre + t(Mpre)
M
```

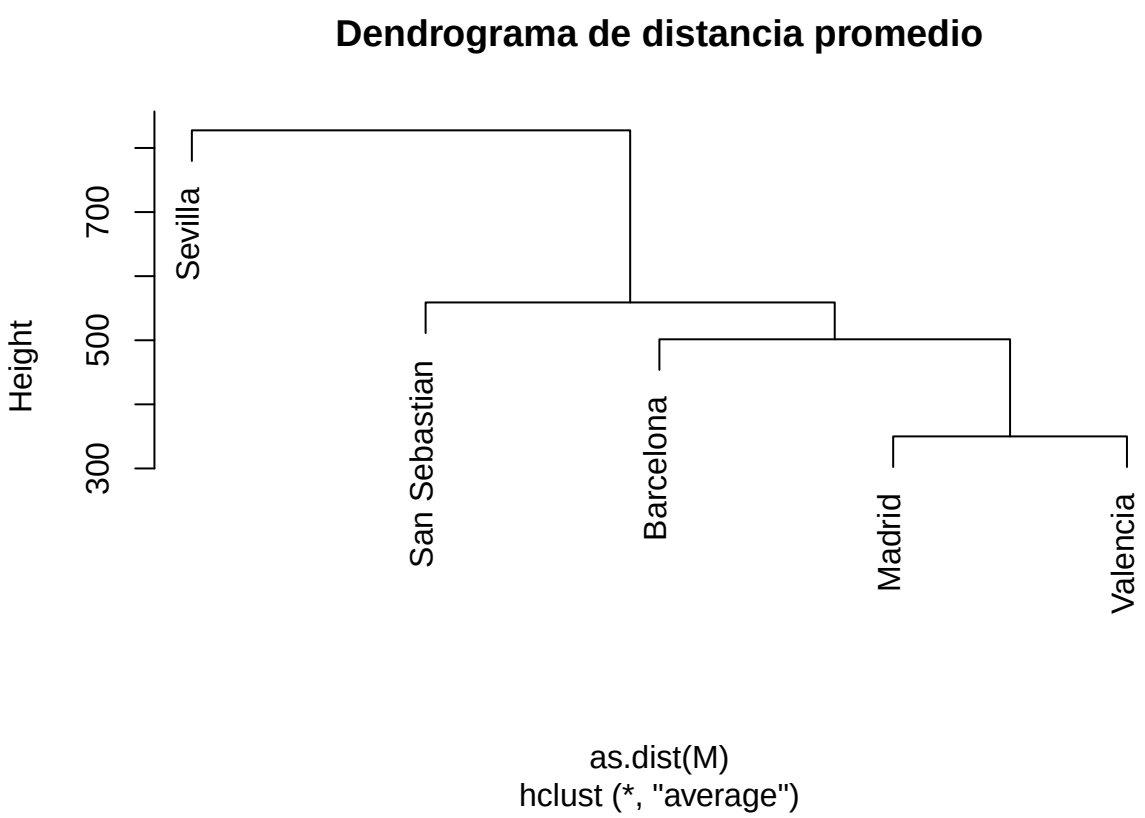
```
##               Barcelona Madrid San Sebastian Sevilla Valencia
## Barcelona             0    639           606    1181      364
## Madrid              639      0           474     542      350
## San Sebastian       606    474             0     908      597
## Sevilla            1181    542           908       0      679
## Valencia            364    350           597     679        0
```

A. Hallar las ultra-distancias (dendrogram-dist) con el método de aglomeración jerárquica: (1) distancia mínima para nuevo grupo (2) distancia promedio entre individuos. Construir el dendrograma respectivo.

```
dmin = hclust(as.dist(M),method = "single")
plot(dmin,main = "Dendrograma de distancia mínima")
```




```
dave = hclust(as.dist(M),method = "average")
plot(dave,main = "Dendrograma de distancia promedio")
```



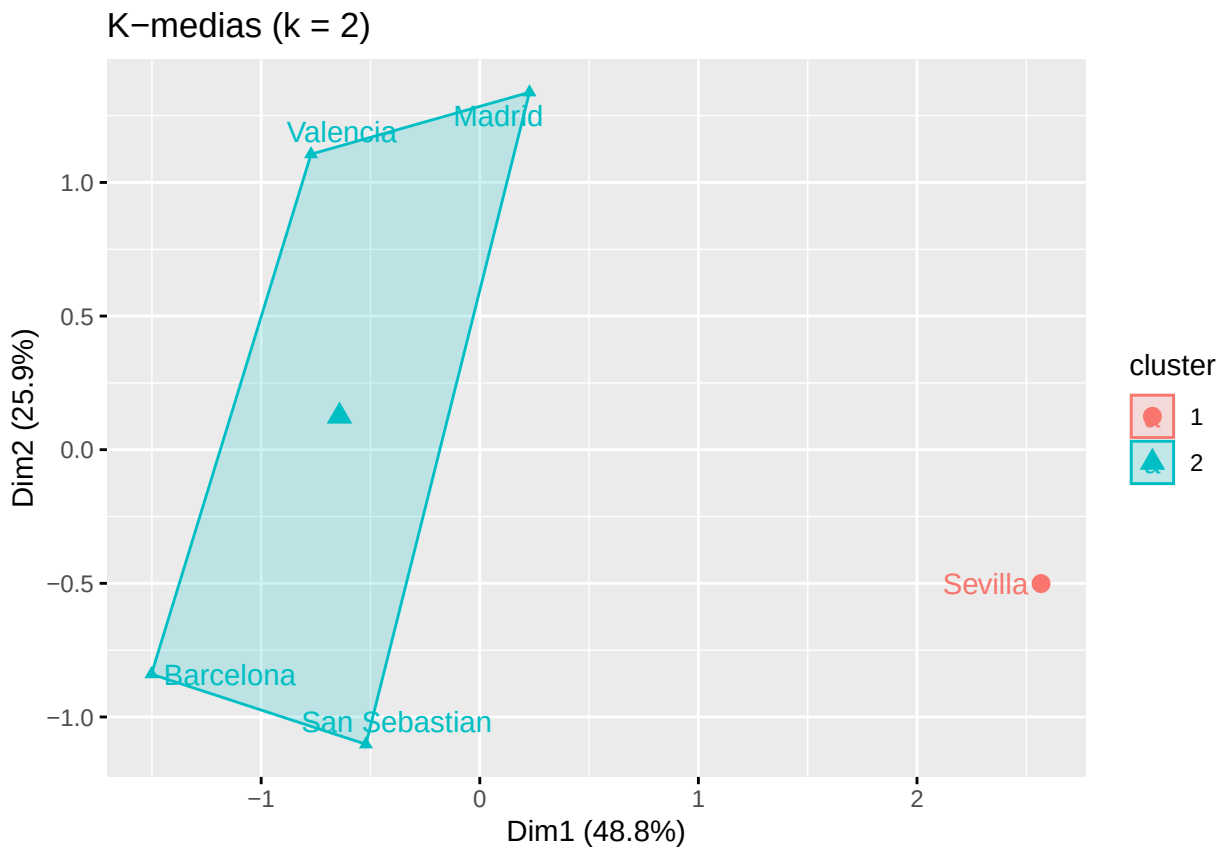
Podemos ver que en los 2 dendogramas, más que nada queda Sevilla separado de las demas ciudades.

B. Hacer el gráfico de agromeración no-jerárquica con el método de k-medias para k = 2 y para k = 3 con los datos del problema 5. Argumente porqué sería mejor usar k = 2 o k = 3, según sea su elección.

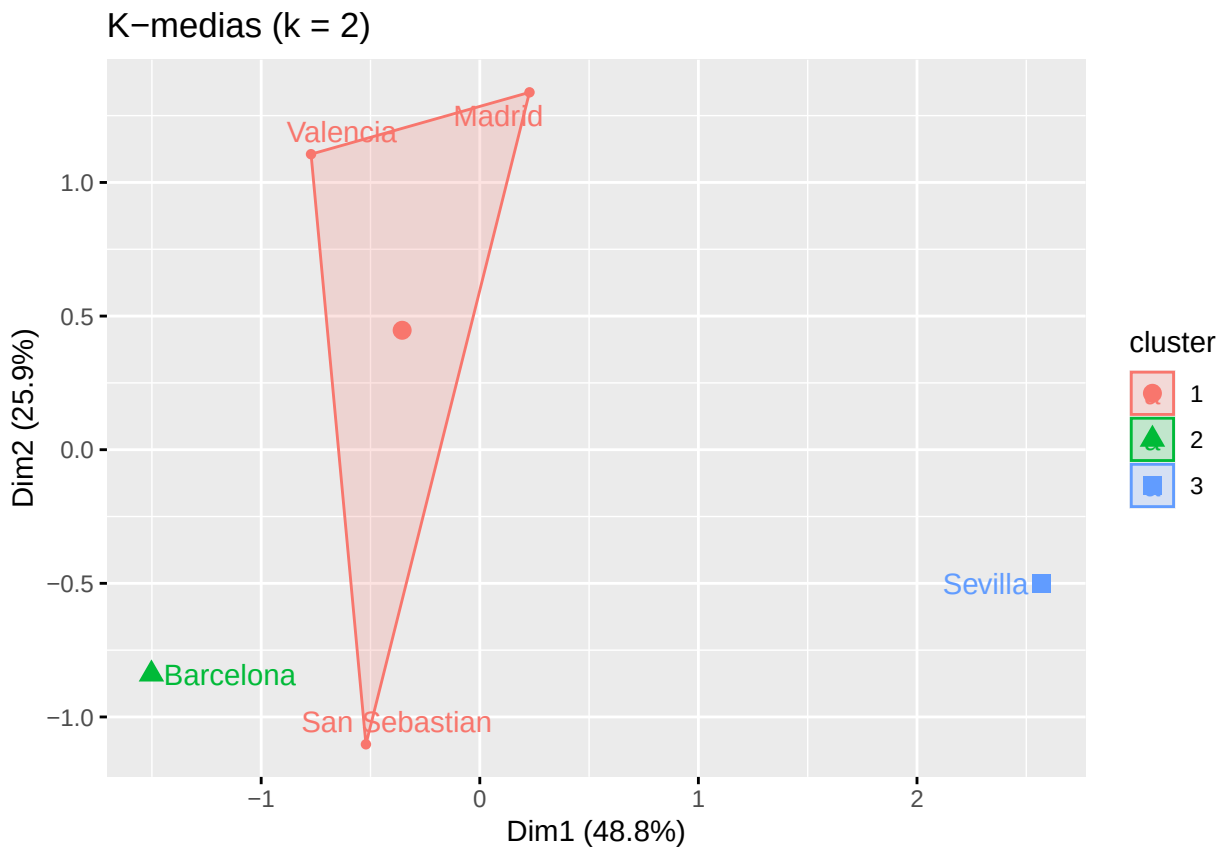
```
library(factoextra)
```

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

```
kmeans_k2 <- kmeans(M, centers = 2)
fviz_cluster(kmeans_k2, data = M, show.clust.cent = TRUE, ellipse.type="convex", star.plot=FALSE, repel=
             main = "K-medias (k = 2)")
```




```
kmeans_k3 <- kmeans(M, centers = 3)
fviz_cluster(kmeans_k3, data = M, show.clust.cent = TRUE, ellipse.type="convex", star.plot=FALSE, repel=
             main = "K-medias (k = 2)")
```



Se puede ver que con k = 2, los grupos están mejor distribuidos, ya que agarra a todos los de similares distancias y solo deja a Sevilla fuera, esto es como se veía que debría ser en los dendogramas. Con k = 3, queda separado Barcelona de todos los demás tambien, lo cual no tendría el mayor sentido debido a que está a similar distancia de todas las demás sin incluir Sevilla.