

Laboratorio 4: Analisis multivalrial.

## I. Archivos necesarios para esta sesion

- ✓ oca.txt
- ✓ nutrientes.txt

## II. Componentes principales.

Es un método multivarial para generar un nuevo sistema de variables relacionadas. Útil para establecer relaciones.

Cuando se estudia muchas variables, casi siempre las variables en estudio están correlacionadas, es decir existe alguna asociación entre ellas. Con el método de componentes se logra liberar esta dependencia a través de rotaciones, cambiando el sistema a un nuevo conjunto de variables totalmente independientes. Cada una de estas variables es función de las variables de estudio. Estas nuevas variables se les denomina componentes, y siempre se trabaja con las primeras, que son las que explican mejor la variación de la totalidad de las variables. Esto significa que si se tiene 2 variables, existen dos componentes y estas explican el 100%. Si son muchas más variables, las primeras pueden explicar por ejemplo 60%. Siempre se trabaja con 2 componentes porque así se puede visualizar en un plano cartesiano.

Para este estudio, utilizaremos el archivo: nutrientes.txt y el modulo “mva”

Si desea revisar el modulo “mva”, ingrese a “html help” del menú help

```
> rm(list=ls())
> library(mva)
> datos <- read.table("nutrientes.txt",header=TRUE)
```

# Seleccion de las variables principales (ca, mg, k) y lo almacene en el objeto “nutriente”

```
> attach(datos)
> nutriente <- data.frame(row.names=elemento,ca, mg, k)
> correlacion <- cor(nutriente)
> componentes<-eigen(correlacion)
```

En el objeto “componentes” esta todo el analisis de componentes principales.

Ahora proceda con el estudio.

1) ¿ Que % de variacion total es explica por la primera y las dos primeras componentes?

```
> total <- sum(componentes$values)
> valor1 <- componentes$values[1]
> valor2 <- componentes$values[2]
```

La primera explica:

```
> valor1*100/total
```

las dos primeras explican:

```
> (valor1+valor2)*100/total
```

2. ¿cual es el vector asociado a la primera componente y a la segunda componente?

```
> c1 <- componentes$vectores[,1]
```

```
> c2 <- componentes$vectores[,2]
```

4. Pruebe que ambos vectores son ortogonales

```
> c1%*%c2
```

El resultado debe ser cero o muy cercano a cero.

5. Halle los valores de las componentes para cada individuo y graficar las dos componentes.

Para este proceso es necesario estandarizar los datos y multiplicar por cada vector asociado (vector propio o eigen vector)

Hacer lo siguiente:

Construya una funcion para tipificar(estandarizar) las variables originales:

```
tipificar <- function(x) { (x-mean(x))/sqrt(var(x)) }
```

aplicar a sus datos:

```
> x1<-tipificar(nutriente[,1])
```

```
> x2<-tipificar(nutriente[,2])
```

```
> x3<-tipificar(nutriente[,3])
```

Juntar estos vectores en un matriz de nombre “x” y multiplicar por los vectores propios c1 y c2

```
> x<- cbind(x1,x2,x3)
```

```
> comp1<- x%*%c1
```

```
> comp2<- x%*%c2
```

las componentes “comp1” y “comp2” son las principales, las cuales pueden ser utilizadas para expresar las observaciones en un plano cartesiano.

Ahora puede dibujar los datos:

```
> plot(comp1,comp2)
```

```
> abline(h=0,v=0, col="red")
```

También puede preparar un gráfico con el nombre de los datos, en row.names del objeto “nutriente” se encuentra esta información.

```
> nombre <- row.names(nutriente)
> plot(comp1,comp2,cex=0.0,text(comp1,comp2, labels=nombre,
adj=c(0,1), cex=0.6))
> abline(h=0,v=0, col="red")
```

### III. Conglomerados (cluster).

Es otra metodología multivarial, permite formar grupos similares según las medidas que uno considere para el agrupamiento.

Hay muchas funciones para ello. Para tener un conocimiento más amplio de las metodologías, utilice las ayudas de R, en html.hlp del menú, localice los módulos “mva” y “cluster”.

Para esta aplicación se utilizará el mismo archivo del caso anterior: “nutrientes.txt”

Objetivos de esta tarea:

- determinar los conglomerados
- hacer gráficos de agrupamiento
- establecer grupos importantes en el dendrograma

#### caso #1.

```
> library(mva)
> library(cluster)

> attach(nutriente)
# Calculo de las distancias por el metodo average (distancia
euclidiana)
> distancia <- dist(nutriente)

#crear un objeto clase hclust
> hc <- hclust(distancia)

# imprimiendo informacion del metodo de conglomerado
> hc

# Sus elementos
> str(hc)

# altura de cada individuo en el dendrograma.
> hc$height
```

```
#Graficar el dendrograma:  
> par(cex=0.6)  
> plclust(hc)
```

**caso # 3**

```
# Utilice el objeto de datos creado "nutriente"  
#Calcular disimilaridades con base a la distancia euclidiana  
  
> dist <- daisy(nutriente)  
  
#Particionar en tres clusters usando la matriz de disimilaridades y  
solicitar  
#la componente clustering.  
#`pam' minimiza la suma de las disimilaridades:  
  
> clus <- pam(dist, 3, diss = TRUE)$clustering  
  
#Graficar los clusters.  
  
> par(cex=0.6)  
> clusplot(dist, clus, lines=1,diss = TRUE, color=TRUE, col.p="black",  
labels=2)
```

**caso #4**

```
> par(cex=0.8)  
  
> cl3 <- pam(nutriente,3)$clustering  
> clusplot(nutriente, cl3, color = TRUE,main = "Grupos",labels=2)  
box(col="orange",lwd=2)
```