

Distancia de Mahalanobis

Sisi Guevara García

8/6/2022

Introducción

Cargar los datos

En el presente ejercicio se usaron datos capturados en vectores de un ejercicio extraído del repertorio de Diego Calvo, sobre las ventas de una empresa.

```
ventas= c( 1054, 1057, 1058, 1060, 1061, 1060, 1061, 1062, 1062, 1064, 1062, 1062, 1064, 1056, 1066, 1070)
clientes= c(63, 66, 68, 69, 68, 71, 70, 70, 71, 72, 72, 73, 73, 75, 76, 78)
```

Los convertimos a data frame

```
datos <- data.frame(ventas ,clientes)
```

Cálculo de la distancia

El método de distancia Mahalanobis mejora el método clásico de distancia de Gauss eliminando el efecto que pueden producir la correlación entre las variables a analizar.

Determinar el número de outlier que queremos encontrar.

```
num.outliers <- 2
```

Como es un estudio con outlier determinamos cuantos serán y a partir de aquí se calculara la distancia

Ordenar los datos de mayor a menor distancia,según la métrica de Mahalanobis.

```
mah.ordenacion <- order(mahalanobis(datos, colMeans(datos), cov(datos)), decreasing=TRUE)
mah.ordenacion
```

```
## [1] 14 16 1 15 2 5 3 10 13 8 12 4 6 7 9 11
```

Ordenamos las distancias de mahalanobis de los datos, las medias de las columnas y la covarianza de los datos y ordenados de mayor a menor; lo visualizamos para observar los datos, donde observamos que los datos 14,16 y 1 las distancias de mahalanobis son mayor y en los datos 7, 9 y 11 las distancias son menores.

Generar un vector booleano los dos valores más alejados segun la distancia Mahalanobis.

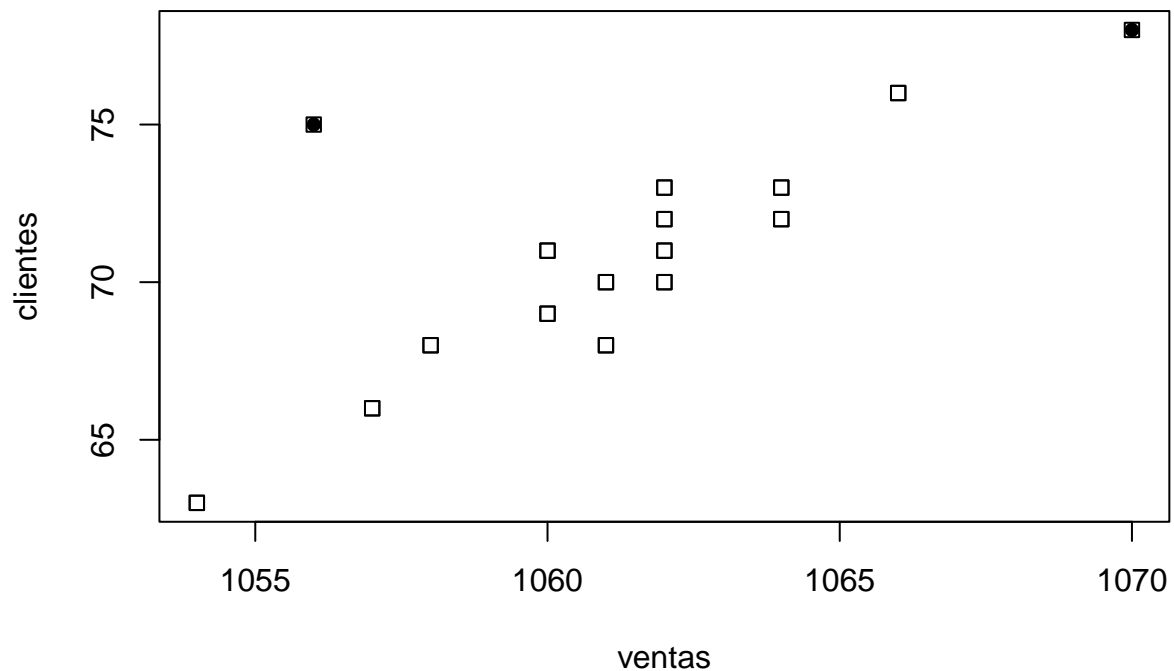
```
outlier2 <- rep(FALSE , nrow(datos))
outlier2[mah.ordenacion[1:num.outliers]] <- TRUE
```

Resaltar con un punto relleno los 2 valores outliers.

```
colorear.outlier <- outlier2 *16
```

Visualizar el gráfico con los datos destacando sus outlier.

```
plot(datos , pch=0)
points(datos , pch=colorear.outlier)
```



Despues de indicarle que punto queremos resaltar de las distancia slos gráficoamos y lo podemos ver los autliers.

Ejercicio 2

```
require(graphics)
```

```
ma <- cbind(1:6, 1:3)
(S <- var(ma))
```

```
##      [,1] [,2]
## [1,]  3.5  0.8
## [2,]  0.8  0.8
```

```
mahalanobis(c(0, 0), 1:2, S)
```

```
## [1] 5.37037
```

Se crea un vector y la varianza del mismo vector, calculando la distancia de mahalanobis a partir de la varianza del primer objeto (ma).

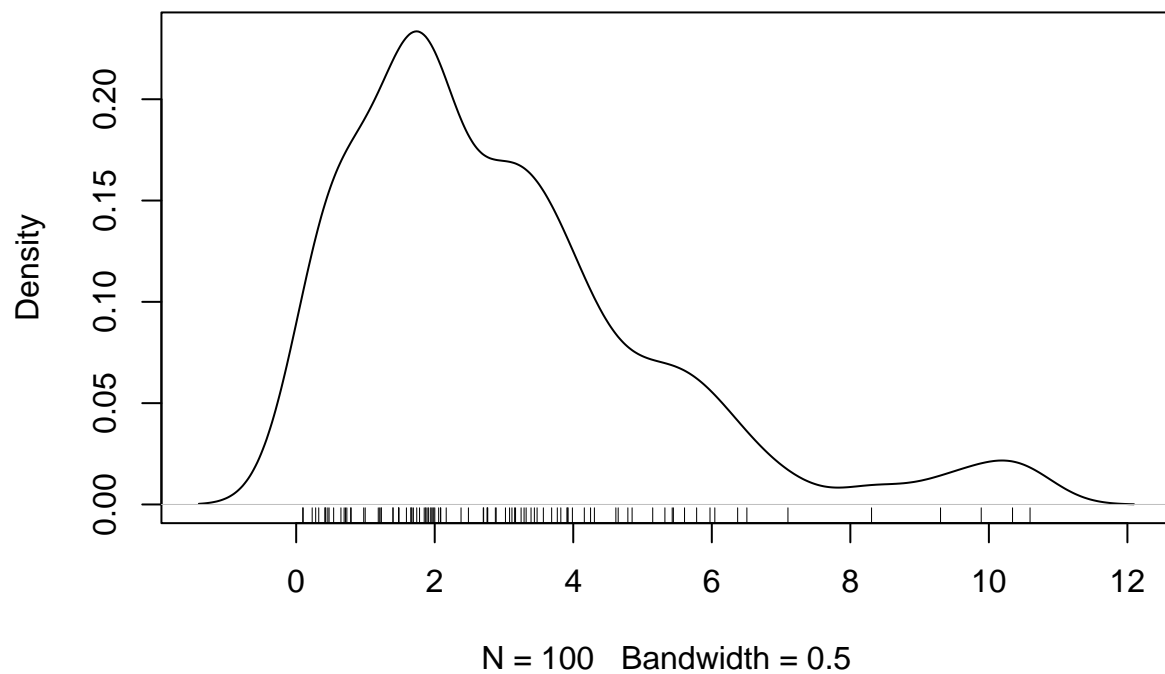
```
x <- matrix(rnorm(100*3), ncol = 3)
stopifnot(mahalanobis(x, 0,
                      diag(ncol(x))) == rowSums(x*x))
```

Creamos una matriz con **rnorm** con tres columnas después se le indica que lo resultante de “mahalanobis” lo coloque en la diagonal de la nueva matriz creada si es igual a la suma de la multiplicación de $x^T x$

Here, D^2 = usual squared Euclidean distances

```
Sx <- cov(x)
D2 <- mahalanobis(x, colMeans(x), Sx)
plot(density(D2, bw = 0.5),
     main="Squared Mahalanobis distances,
     n=100, p=3" ; rug(D2)
```

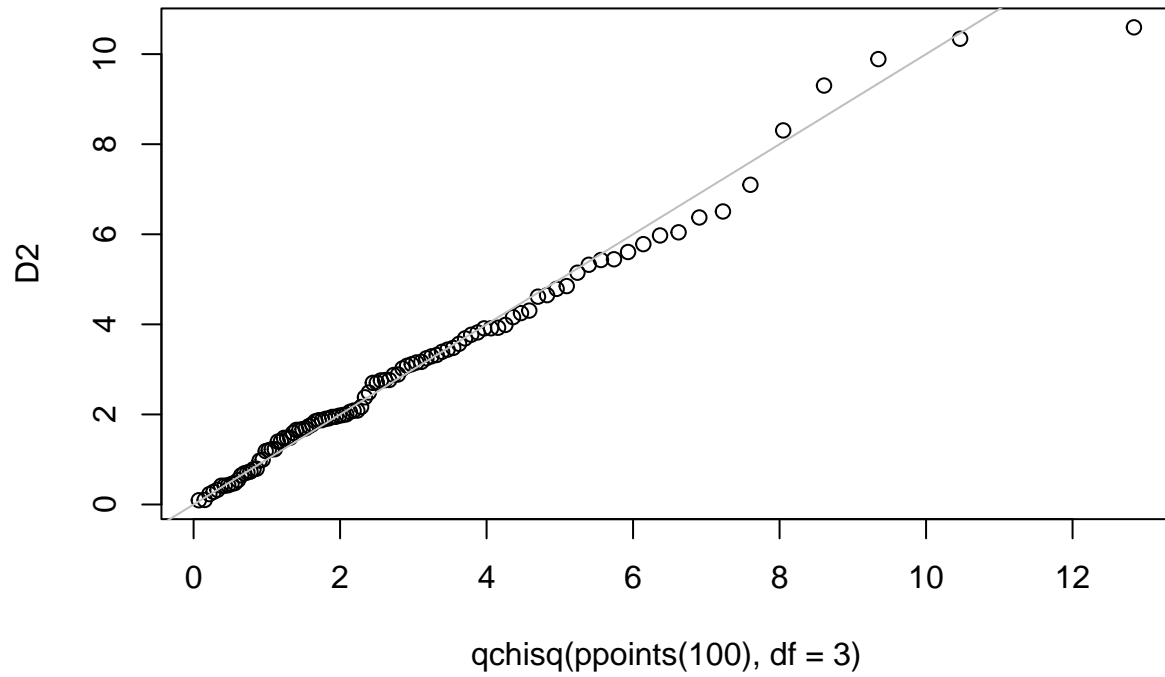
Squared Mahalanobis distances, n=100, p=3



La gráfica muestra las distancias

```
qqplot(qchisq(ppoints(100), df = 3), D2,  
       main = expression("Q-Q plot of Mahalanobis" * ~D^2 *  
                           " vs. quantiles of" * ~ chi[3]^2))  
abline(0, 1, col = 'gray')
```

Q–Q plot of Mahalanobis D^2 vs. quantiles of χ^2_3



Este gráfico muestra la distancias al cuadrado contra los cuantiles de la diatribución chi cuadrada.