Ciencia de datos, práctica 5

Juan Casado Ballesteros, Samuel García Gonzalez, Iván Anaya Martín

December 3, 2019

Abstract

Realizaremos análisis de outliers sobre las dos muestrar proporcionadas según lo indicado en la práctica. Para cada uno de los análisis realizados mostraremos los outliers obtenidos no solo de forma textual si no también con gráficos que nos ayuden a visualizar cada análisis.

Posteriormente hemos buscado un dataset que contiene medidas realizadas sobre los niveles de ozono, de temperatura, de humedad y de velocidad del viento. Realizaremos el análisis de estos datos mediante regresión para lo cual primero realizaremos un análisis de outliers.

1 Conocer los datos

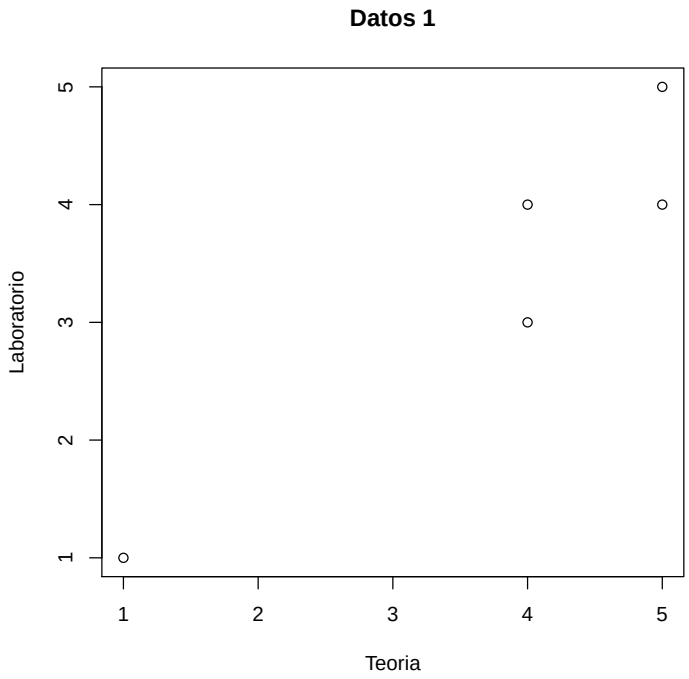
Cargamos y ostramos los datos que se nos han proporcionado para hacer la primera parte del ejercicio.

1.1 Notas

Los primeros datos representan la nota de laboratorio y de teorias evaluadas de 1 a 5.

```
> datos1 <- data.frame(read.table("datos1.txt"))
> datos1
```

```
  Teoria Laboratorio
1      4           4
2      4           3
3      5           5
4      1           1
5      5           4
```

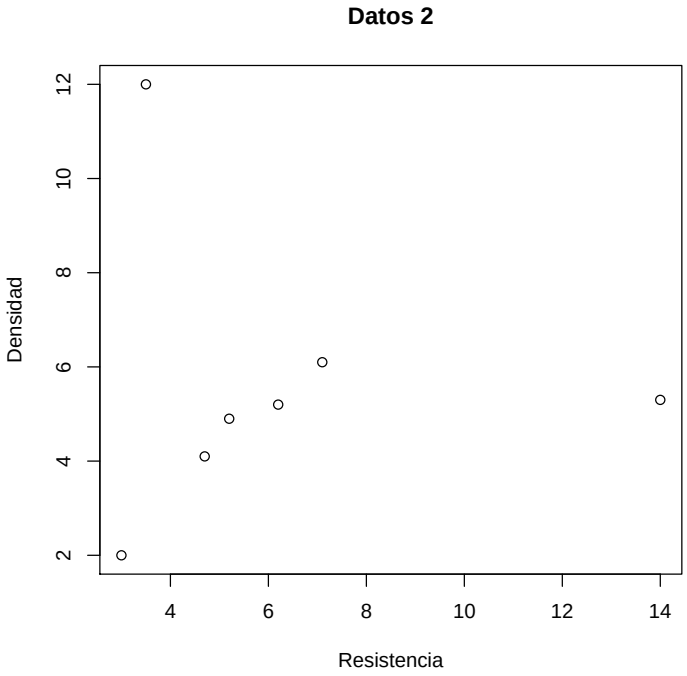


1.2 Hormigón

Los segundos representan la densidad y la resistencia del hormigón.

```
> datos2 <- data.frame(read.table("datos2.txt"))
> datos2
```

```
  Resistencia Densidad
1         3.0      2.0
2         3.5     12.0
3         4.7      4.1
4         5.2      4.9
5         7.1      6.1
6         6.2      5.2
7        14.0      5.3
```



2 K-vecinos sobre la muestra proporcionada para obtener outliers

Aplicaremos el algoritmo k-vecinos sobre la muestra que tenemos. Este algoritmo identificará de forma supervisada en la muestra datos anómalos, para poder obtener los outliers.

Calculamos las distancias euclídeas entre todos los puntos.

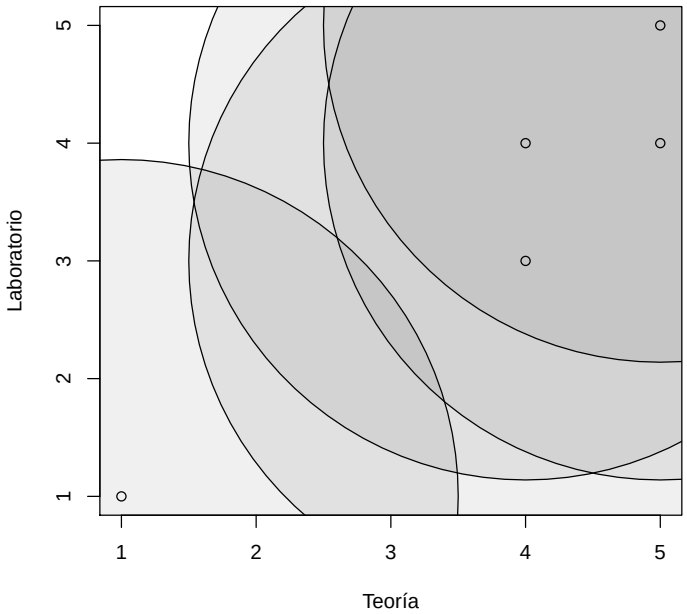
```
> distancias <- as.matrix(dist(datos1))
> distancias
```

```
          1        2        3        4        5
1 0.000000 1.000000 1.414214 4.242641 1.000000
2 1.000000 0.000000 2.236068 3.605551 1.414214
3 1.414214 2.236068 0.000000 5.656854 1.000000
4 4.242641 3.605551 5.656854 0.000000 5.000000
5 1.000000 1.414214 1.000000 5.000000 0.000000
```

Elegimos el grado a partir del cual consideraremos que un pounto es outlier. Todos los valores cuyo tercer vecino más cercano esté a una distancia superior a 2.5 los consideraremos outliers.

```
> max_radio <- 2.5
```

Mostramos en torno a cada valor un círculo con el radio indicado. Si dentro del círculo dibujado no hay al menos otros tres datos dicho valor será considerado outlier. Podemos ver que solo hay un punto para el que se da esta condición.



Calcularemos numéricamente ese valor.

Ordenamos las distancias de cada punto a todos los demás.

```
> for(i in 1:length(distancias[,1])){
+   distancias[,i] <- sort(distancias[,i])
+ }
> distanciasordenadas <- distancias
```

Reordenamos la matriz para organizarla en función de la distancia de cada punto a su vecino número 1,2,3...etc. Tras haber organizado la matriz, buscamos en el tercer vecino, que es el valor k que hemos usado en nuestro análisis para poder identificar los outliers.

```
> outliers_kvecinos = list()
> for(i in 1:length(distanciasordenadas[,1])){
+   if(distanciasordenadas[4,i]>max_radio){
+     outliers_kvecinos[[length(outliers_kvecinos)+1]] <- datos1[i,]
+   }
+ }
> outliers_kvecinos

[[1]]
  Teoria Laboratorio
4      1           1
```

3 Deteccion de datos anómalos por cuartiles

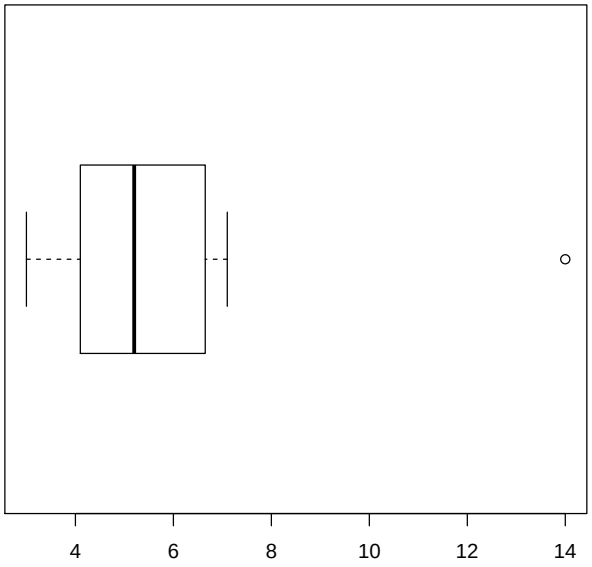
Sobre el segundo set de datos eliminaremos utilizando el método de los cuartiles.

Elegiremos el fator por el que multiplicar Q3-Q1. Todos los valores que se alegen esa distancia de Q1 hacia los negativo o de Q3 hacia los positevos serán considerados outliers.

```
> max_range = 1.5
```

3.1 Resistencia del hormigón

Mostramos el diagrama de caja y bigotes de la resistencia.



Calculamos ahora los valores que se salen del rango definido y que por tanto son outliers. Para hacer esto primero obtenemos el intercalo de valores válidos

```
> cuart1_res<-quantile(datos2$Resistencia,0.25)
> cuart3_res<-quantile(datos2$Resistencia,0.75)
> int_res=c(cuart1_res-max_range*(cuart3_res-cuart1_res),
+           cuart3_res+max_range*(cuart3_res-cuart1_res))
> int_res

   25%    75%
 0.275 10.475
```

Ahora obtendremos los valores que quedan fuera de dicho intervalo.

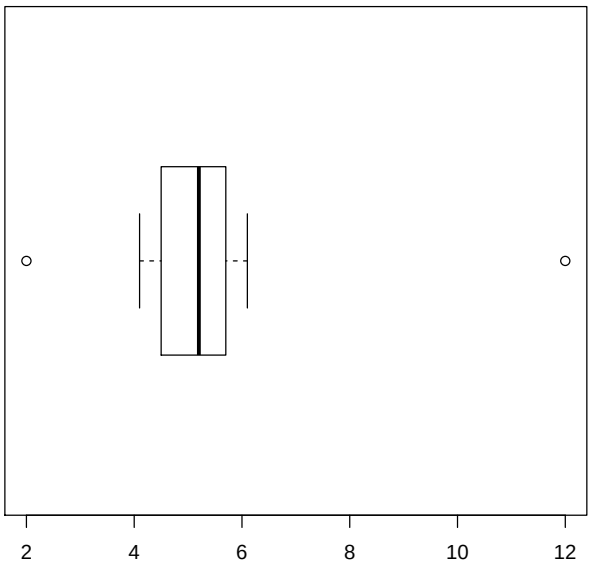
```
> outliers_cuartiles_resistencia = list()
> for(i in 1:length(datos2$Resistencia)){
+   if(datos2$Resistencia[i]<int[1]||datos2$Resistencia[i]>int[2]){
+     outliers_cuartiles_resistencia[[length(outliers_cuartiles_resistencia)+1]] <-
+         t(matrix(c(i, datos2[i,]$Resistencia), dimnames=list(c("Indice","Resistencia")))
+   }
+ }
> outliers_cuartiles_resistencia

[[1]]
     Indice Resistencia
[1,]      7          14
```

Como vemos el valor 12 es un outlier de la variable resistencia.

3.2 Densidad del hormigón

Repetimos este mismo análisis para la densidad.

Primero mostramos el diagrama de caja y bigotes para esta variable.



Calculamos el intervalo a partir del que consideramos que los datos son outliers.

```
> cuart1_den<-quantile(datos2$Densidad,0.25)
> cuart3_den<-quantile(datos2$Densidad,0.75)
> int=c(cuart1_den-max_range*(cuart3_den-cuart1_den),
+       cuart3_den+max_range*(cuart3_den-cuart1_den))
> int
```

```
25% 75%
2.7 7.5
```

Obtenemos lo datos que quedan fuera del intervalo.

```
> outliers_cuartiles_densidad <- list()
> for(i in 1:length(datos2$Densidad)){
+   if(datos2$Densidad[i]<int[1]||datos2$Densidad[i]>int[2]){
+     outliers_cuartiles_densidad[[length(outliers_cuartiles_densidad)+1]] <-
+       t(matrix(c(i, datos2[i,]$Densidad), dimnames=list(c("Indice","Densidad"))))
+   }
+ }
> outliers_cuartiles_densidad

[[1]]
     Indice Densidad
[1,]      1        2

[[2]]
     Indice Densidad
[1,]      2       12
```

Esta vez hay dos valores outlier de densidad el 2 y el 12.

4 Outliers mediante la desviación típica

Calcularemos los valores para cada que son considerados outliers por el método de la desviación tipica. Dichos valores serán aquellos que se alejen demasidado de la media de la variable analizada.

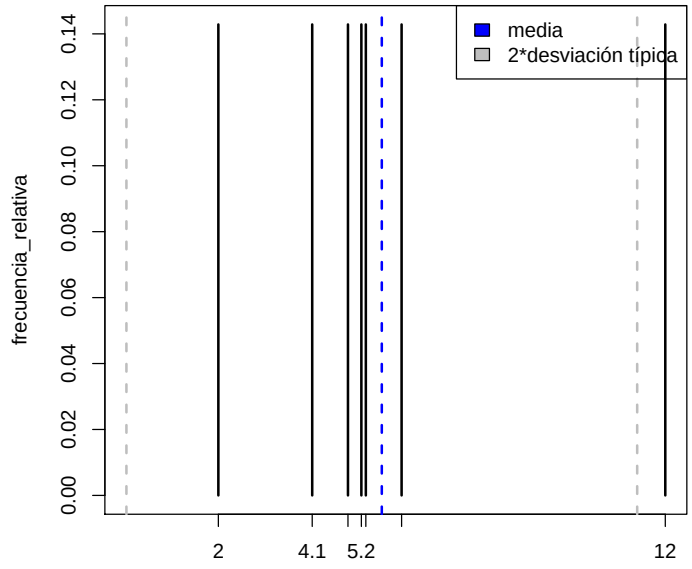
En primer lugar elegiremos el factor por el que multiplicar la desviación típica para generar el intervalo de valores no outliers entorno a la media. Metiante el teorema de tchebychev sabemos que para un valor de 2 al menos el 75% de los datos estarán dentro del intervalo generado.

```
> max_deviation = 2
```

4.1 Densidad del hormigón

Mostramos la frecuencia realativa de la densidad del hormigón con respecto a la media de esta magnitud en azul. En gris se muestra el intervalo a partir del cual los valores que queden fuera de él son considerados outliers.

```
> plotFrecuencyData(datos2$Densidad)
```



Ahora calculamos dichos valores. Primero obtenemos el intervalo.

```
> int <- c(mediaAritmetica(datos2$Densidad) - 2*desviacionTipica(datos2$Densidad),
+          mediaAritmetica(datos2$Densidad) + 2*desviacionTipica(datos2$Densidad))
> int

[1] -0.05685714 11.37114285
```

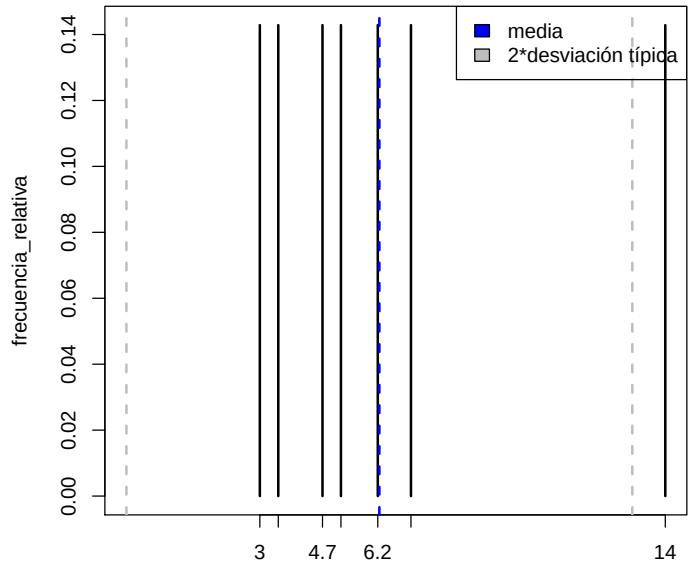
Podemos ver que el intervalo contiene valores de densidad irreales, esta no podría ser negativa. En un anális de datos realista deberíamos corregir esto consultando con alguien que sea experto en los datos que estemos analizando. Y luego los valores que quedan fuera de él.

```
> outliers_desviacion = list()
> for(i in 1:length(datos2$Densidad)) {
+   if ((datos2$Densidad[i]<int[1]) || (datos2$Densidad[i]>int[2])) {
+     outliers_desviacion[[length(outliers_desviacion)+1]] <-
+       t(matrix(c(i, datos2[i,]$Densidad), dimnames=list(c("Indice","Densidad"))))
+   }
+ }
> outliers_desviacion

[[1]]
     Indice Densidad
[1,]      2       12
```

4.2 Resistencia del hormigón

Mostramos la frecuencia realativa de la resistencia del hormigón con respecto a la media de esta magnitud en azul. En gris se muestra el intervalo a partir del cual los valores que queden fuera de él son considerados outliers.

```
> plotFrecuencyData(datos2$Resistencia)
```



Ahora calculamos dichos valores. Primero obtenemos el intervalo.

```
> int <- c(mediaAritmetica(datos2$Resistencia) - 2*desviacionTipica(datos2$Resistencia),
+          mediaAritmetica(datos2$Resistencia) + 2*desviacionTipica(datos2$Resistencia))
> int

[1] -0.6197597 13.1054740
```

Podemos ver que el intervalo contiene valores de resistencia irreales, esta no podría ser negativa. En un anális de datos realista deberíamos corregir esto consultando con alguien que sea experto en los datos que estemos analizando. Y luego los valores que quedan fuera de él.

```
> outliers_desviacion = list()
> for(i in 1:length(datos2$Resistencia)) {
+   if ((datos2$Resistencia[i]<int[1]) || (datos2$Resistencia[i]>int[2])) {
+     outliers_desviacion[[length(outliers_desviacion)+1]] <-
+       t(matrix(c(i, datos2[i,]$Resistencia), dimnames=list(c("Indice","Resistencia"))))
+   }
+ }
> outliers_desviacion

[[1]]
     Indice Resistencia
[1,]      7          14
```

5 Detección de datos anómalos sobre la regresión de la densidad en función de la resistencia

En este análisis detectamos los outliers utilizando la recta de regresión y el error estándar de los residuos. Comenzaremos por determinar el factor por el que multiplicar el error estándar para considerar que los datos son outliers.

```
> sr_factor = 2
```

Comenzamos el análisis calculando la recta de regresión de los datos.

```
> dFr=lm(datos2$Densidad~datos2$Resistencia)
```

Posteriormente, obtenemos los residuos calculados a partir de la recta de regresión y el error estándar.

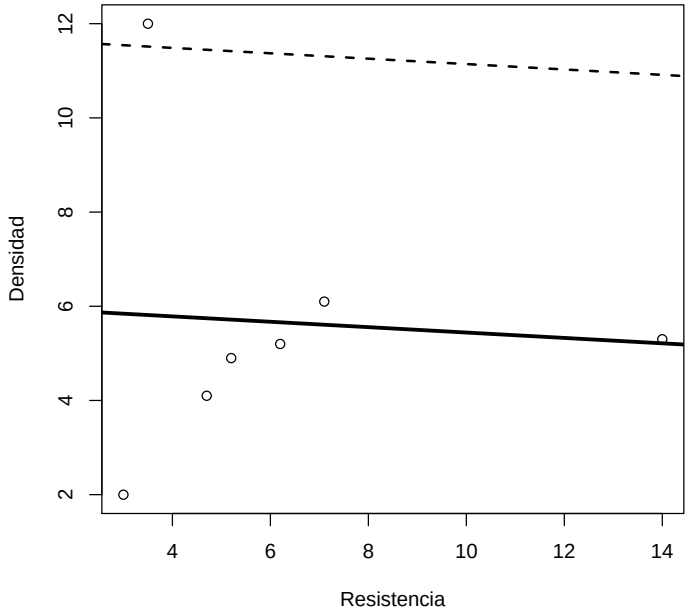
```
> res=summary(dFr)$residuals
> res
```

```
         1          2          3          4          5          6          7
-3.8427477  6.1858698 -1.6454482 -0.8168308  0.4919157 -0.4595958  0.0868370
```

```
> sr=sqrt(sum(res^2)/length(res))
> sr
```

```
[1] 2.850242
```

A partir del error estándar y de la recta de regresión obtenida podremos mostrar los datos junto a su recta de regresión. Paralela a dicha recta mostramos otras dos que marcan la frontera a partir de la que los valores se considerarán outliers. En este caso solo se ve la recta paralela superior.



Con el error estándar de los residuos, comparamos cada uno para comprobar si es mayor que el error estándar multiplicado por el factor establecido. Si se da el caso, podemos considerar ese punto como un outlier.

```
> outliers_regresion = list()
> for(i in 1:length(res)){
+   if(abs(res[i])>sr_factor*sr){
+     outliers_regresion[[length(outliers_regresion)+1]] <- datos2[i,]
+   }
+ }
> outliers_regresion

[[1]]
  Resistencia Densidad
2         3.5       12
```