

## Màster universitari en Estadística i Investigació Operativa (MESIO)

## Computación en Estadística y en Optimización

## Solución del Test 1 con R (Grupo A)

**Ejercicio 1 (0,3 + 0,6 + 0,5 + 0,5 + 0,6 + 0,7 + 0,8 + 0,5 + 1 = 5,5 puntos)**

El área de trabajo `CeoGrATestR1.RData` contiene el *data frame* `decathlon` del paquete `FactoMineR` con datos de distintos atletas de decatón del año 2004. Según la página de información sobre este conjunto de datos se trata de un

*“... data frame with 41 rows and 13 columns: the first ten columns correspond to the performance of the athletes for the 10 events of the decathlon. The columns 11 and 12 correspond respectively to the rank and the points obtained. The last column is a categorical variable corresponding to the sporting event (2004 Olympic Games or 2004 Decastar).”*

Además, se ha añadido la variable fecha de nacimiento en la columna 14.

- a) Cargad el área de trabajo `CeoGrATestR1.RData` y cambiad los nombres de las variables del *data frame* `decathlon` a minúscula.

```
> load("CeoGrATestR1.RData")
> names(decathlon) <- tolower(names(decathlon))
> summary(decathlon)
```

100m	long.jump	shot.put	high.jump	400m
Min. :10.44	Min. :6.61	Min. :12.68	Min. :1.850	Min. :46.81
1st Qu.:10.85	1st Qu.:7.03	1st Qu.:13.88	1st Qu.:1.920	1st Qu.:48.93
Median :10.98	Median :7.30	Median :14.57	Median :1.950	Median :49.40
Mean :11.00	Mean :7.26	Mean :14.48	Mean :1.977	Mean :49.62
3rd Qu.:11.14	3rd Qu.:7.48	3rd Qu.:14.97	3rd Qu.:2.040	3rd Qu.:50.30
Max. :11.64	Max. :7.96	Max. :16.36	Max. :2.150	Max. :53.20
110m.hurdle	discus	pole.vault	javeline	
Min. :13.97	Min. :37.92	Min. :4.200	Min. :50.31	
1st Qu.:14.21	1st Qu.:41.90	1st Qu.:4.500	1st Qu.:55.27	
Median :14.48	Median :44.41	Median :4.800	Median :58.36	
Mean :14.61	Mean :44.33	Mean :4.762	Mean :58.32	
3rd Qu.:14.98	3rd Qu.:46.07	3rd Qu.:4.920	3rd Qu.:60.89	
Max. :15.67	Max. :51.65	Max. :5.400	Max. :70.52	
1500m	rank	points	competition	
Min. :262.1	Min. : 1.00	Min. :7313	Decastar:13	
1st Qu.:271.0	1st Qu.: 6.00	1st Qu.:7802	Olympics:28	
Median :278.1	Median :11.00	Median :8021		
Mean :279.0	Mean :12.12	Mean :8005		
3rd Qu.:285.1	3rd Qu.:18.00	3rd Qu.:8122		
Max. :317.0	Max. :28.00	Max. :8893		
birthday				
Min. :1970-06-25				
1st Qu.:1976-08-22				
Median :1979-03-22				
Mean :1978-08-09				
3rd Qu.:1980-08-28				
Max. :1983-08-07				

- b) Cread un *data frame* con nombre `olymp` que contenga solamente los datos de los Juegos Olímpicos y borrad las variables `rank` y `competition`.

```
> olymp <- subset(decathlon, competition == "Olympics",
+               select = -c(rank, competition))
```

```
> head(olymp)
```

	100m	long.jump	shot.put	high.jump	400m	110m.hurdle	discus
Sebrle	10.85	7.84	16.36	2.12	48.36	14.05	48.72
Clay	10.44	7.96	15.23	2.06	49.19	14.13	50.11
Karpov	10.50	7.81	15.93	2.09	46.81	13.97	51.65
Macey	10.89	7.47	15.73	2.15	48.97	14.56	48.34
Warners	10.62	7.74	14.48	1.97	47.97	14.01	43.73
Zsivoczky	10.91	7.14	15.31	2.12	49.40	14.95	45.62

	pole.vault	javeline	1500m	points	birthday
Sebrle	5.0	70.52	280.01	8893	1974-11-26
Clay	4.9	69.71	282.00	8820	1980-03-01
Karpov	4.6	55.54	278.11	8725	1981-07-23
Macey	4.4	58.46	265.42	8414	1977-12-12
Warners	4.9	55.39	278.05	8343	1978-04-02
Zsivoczky	4.7	63.45	269.54	8287	1977-04-29

- c) Ordenad el *data frame* *olymp* según el apellido de los atletas.

```
> olymp <- olymp[order(rownames(olymp)), ]
> head(olymp, 4)
```

	100m	long.jump	shot.put	high.jump	400m	110m.hurdle	discus
Averyanov	10.55	7.34	14.44	1.94	49.72	14.39	39.88
Barras	11.14	6.99	14.91	1.94	49.41	14.37	44.83
Bernard	10.69	7.48	14.80	2.12	49.13	14.17	44.75
Casarsa	11.36	6.68	14.92	1.94	53.20	15.39	48.66

	pole.vault	javeline	1500m	points	birthday
Averyanov	4.8	54.51	271.02	8021	1980-02-04
Barras	4.6	64.55	267.09	8067	1980-08-01
Bernard	4.4	55.27	276.31	8225	1979-03-22
Casarsa	4.4	58.62	296.12	7404	1975-07-02

```
> tail(olymp, 3)
```

	100m	long.jump	shot.put	high.jump	400m	110m.hurdle	discus
Uldal	11.23	6.99	13.53	1.85	50.95	15.09	43.01
Warners	10.62	7.74	14.48	1.97	47.97	14.01	43.73
Zsivoczky	10.91	7.14	15.31	2.12	49.40	14.95	45.62

	pole.vault	javeline	1500m	points	birthday
Uldal	4.5	60.00	281.70	7495	1982-12-16
Warners	4.9	55.39	278.05	8343	1978-04-02
Zsivoczky	4.7	63.45	269.54	8287	1977-04-29

- d) ¿Cuántos atletas lograron saltar más de 7,5 metros en el salto de longitud (*Long jump*)?

```
> sum(olymp$long.jump > 7.5)
```

```
[1] 5
```

- e) ¿Empataron algunos atletas en puntos?

```
> any(duplicated(olymp$points))
```

```
[1] FALSE
```

- f) ¿Cuál es la marca del atleta con la mejor marca en lanzamiento de disco y qué tiempo hizo en los 1500 metros?

```
> olymp[which(olymp$discus == max(olymp$discus)), c("discus", "1500m")]
```

	discus	1500m
Karpov	51.65	278.11

- g) ¿Qué día de la semana nacieron más atletas? ¿Cuántos fueron?

```
> bdays <- table(weekdays(olymp$birthday))
> bdays[which(bdays == max(bdays))]

jueves
9
```

- h) Calculad las correlaciones entre todas las disciplinas del decatlón usando el coeficiente de Pearson.

```
> (cormatP <- round(cor(olymp[, 1:10]), 3))
```

	100m	long.jump	shot.put	high.jump	400m	110m.hurdle	discus
100m	1.000	-0.705	-0.370	-0.309	0.635	0.543	-0.233
long.jump	-0.705	1.000	0.195	0.346	-0.671	-0.538	0.250
shot.put	-0.370	0.195	1.000	0.613	-0.199	-0.245	0.666
high.jump	-0.309	0.346	0.613	1.000	-0.169	-0.326	0.517
400m	0.635	-0.671	-0.199	-0.169	1.000	0.520	-0.144
110m.hurdle	0.543	-0.538	-0.245	-0.326	0.520	1.000	-0.217
discus	-0.233	0.250	0.666	0.517	-0.144	-0.217	1.000
pole.vault	-0.260	0.285	0.024	-0.042	-0.115	-0.151	-0.184
javeline	-0.012	0.094	0.383	0.204	-0.055	-0.080	0.255
1500m	0.058	-0.147	0.130	-0.003	0.551	0.179	0.220

	pole.vault	javeline	1500m
100m	-0.260	-0.012	0.058
long.jump	0.285	0.094	-0.147
shot.put	0.024	0.383	0.130
high.jump	-0.042	0.204	-0.003
400m	-0.115	-0.055	0.551
110m.hurdle	-0.151	-0.080	0.179
discus	-0.184	0.255	0.220
pole.vault	1.000	-0.066	0.179
javeline	-0.066	1.000	-0.252
1500m	0.179	-0.252	1.000

- i) ¿Entre qué disciplinas hay la máxima correlación (en valor absoluto)? ¿Cuál es el valor de esta correlación?

```
> diag(cormatP) <- 0
> (pos <- rownames(which(abs(cormatP) == max(abs(cormatP)), arr.ind = TRUE)))

[1] "long.jump" "100m"

> cormatP[pos[1], pos[2]]

[1] -0.705
```

## Ejercicio 2 (3 puntos)

Reproducid el *boxplot* de la Figura 1, que representa la distribución de las puntuaciones de los atletas en función de la competición. Además muestra la media en ambas competiciones y el nombre del atleta con la puntuación máxima. Guardad el gráfico en formato JPG.

```
> decathlon$competition <- relevel(decathlon$competition, ref = "Olympics")
> avs <- with(decathlon, round(tapply(points, competition, mean), 1))
> maxx <- max(decathlon$points)
> ## Code for Figure 1
> library(beeswarm)
> windows(width = 7)
> par(las = 1, font = 2, font.lab = 4, font.axis = 2, pch = 16, cex.lab = 1.3)
> boxplot(points ~ competition, decathlon, col = grey(2:3 / 4),
+         xlab = "Competition", ylab = "Points",
+         pars = list(boxwex = 0.7), ylim = c(7200, maxx))
> beeswarm(points ~ competition, decathlon, add = TRUE)
> title("Points per competition", cex.main = 1.3)
```

```
> text(1:2, 7200, paste("Mean:", avs))
> text(1, maxx, rownames(decathlon)[which.max(decathlon$points)], adj = c(-0.1, 0.5))
> savePlot("BoxplotEx2", type = "jpg")
```

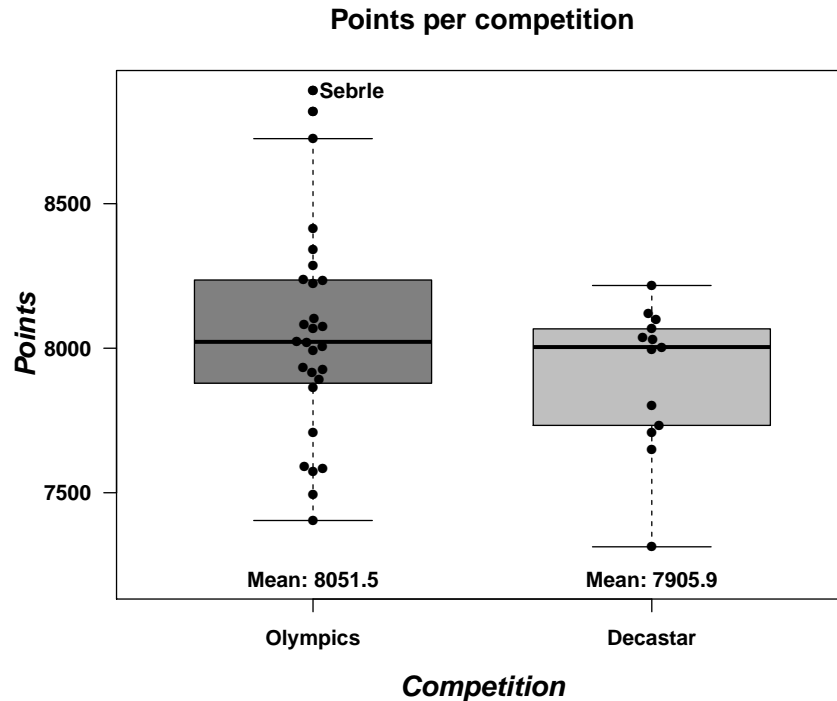


Figura 1: *Boxplot* del Ejercicio 2.

### Ejercicio 3 (1,3 + 0,2 = 1,5 puntos)

El *data frame* `dfr` del área de trabajo `CeoGrATestR1.RData` contiene los valores de dos variables numéricas que han sido importados de forma errónea desde un fichero ASCII. Como resultado en R ambas variables, `x` e `y`, son factores:

```
> dfr

  x    y
1  5 1,23
2  5  8,7
3 10    5
4  2    *
5  4    *
6  7  7,5
7  4    *
8 10    2
9  6    2
10 3  3,2

> str(dfr)

'data.frame':      10 obs. of  2 variables:
 $ x: Factor w/ 7 levels "2","3","4","5",...: 4 4 7 1 3 6 3 7 5 2
 $ y: Factor w/ 7 levels "*","1,23","2",...: 2 7 5 1 1 6 1 3 3 4
```

- a) Convertid ambas variables en variables numéricas teniendo en cuenta que el símbolo “\*” indica un valor perdido.

```
> dfr$x <- as.numeric(as.character(dfr$x))
> dfr$y <- as.numeric(sub(",", ".", dfr$y))
> dfr
```

	x	y
1	5	1.23
2	5	8.70
3	10	5.00
4	2	NA
5	4	NA
6	7	7.50
7	4	NA
8	10	2.00
9	6	2.00
10	3	3.20

```
> str(dfr)
'data.frame':      10 obs. of  2 variables:
 $ x: num  5 5 10 2 4 7 4 10 6 3
 $ y: num  1.23 8.7 5 NA NA 7.5 NA 2 2 3.2
```

- b) Guardad el *data frame* `dfr` en un fichero RDS.

```
> saveRDS(dfr, "dfr.rds")
```