

Ανάλυση κυρίων συνιστώσων

Λουκία Μελικοτσίδου

Εισαγωγή

Η ανάλυση κυρίων συνιστωσών (PCA) έχει ως στόχο την ελάττωση της διάστασης (αριθμός μεταβλητών) ενός συνόλου δεδομένων που αποτελείται από ένα μεγάλο σύνολο συσχετισμένων μεταβλητών. Αυτό επιτυγχάνεται μέσω ενός γραμμικού μετασχηματισμού των μεταβλητών προκύπτουν νέες μεταβλητές (οι κύριες συνιστώσες), οι οποίες

- είναι ασυσχέτιστες
- έχουν την ίδια συνολική διασπορά με τις αρχικές

και επιπλέον

- Ελπίζουμε ότι ένας μικρός αριθμός εξ αυτών εξηγεί το μεγαλύτερο μέρος της διασποράς των αρχικών μεταβλητών
- Εάν συμβαίνει αυτο, ίσως μπορούμε να αντικαταστήσουμε τα αρχικά δεδομένα με ένα μικρό αριθμό κυρίων συνιστωσών (χωρίς να χάσουμε πολλή πληροφορία)

Δεδομένα επτάθλου

Το σύνολο δεδομένων `heptathlon.csv` αποτελείται από δεδομένα 26 αθλητριών του επτάθλου στα 7 αγωνίσματα κατά τη διάρκεια της Ολυμπιάδας του Los Angeles το 1984. Τα αγωνίσματα είναι δρόμοι (100 μέτρα με εμπόδια, 200 μέτρα και 800 μέτρα), ρίψεις (σφαιροβολία και ακοντισμό) καθώς και άλματα (άλμα εις μήκος και άλμα εις ύψος)

Εισάγουμε τα δεδομένα στη R

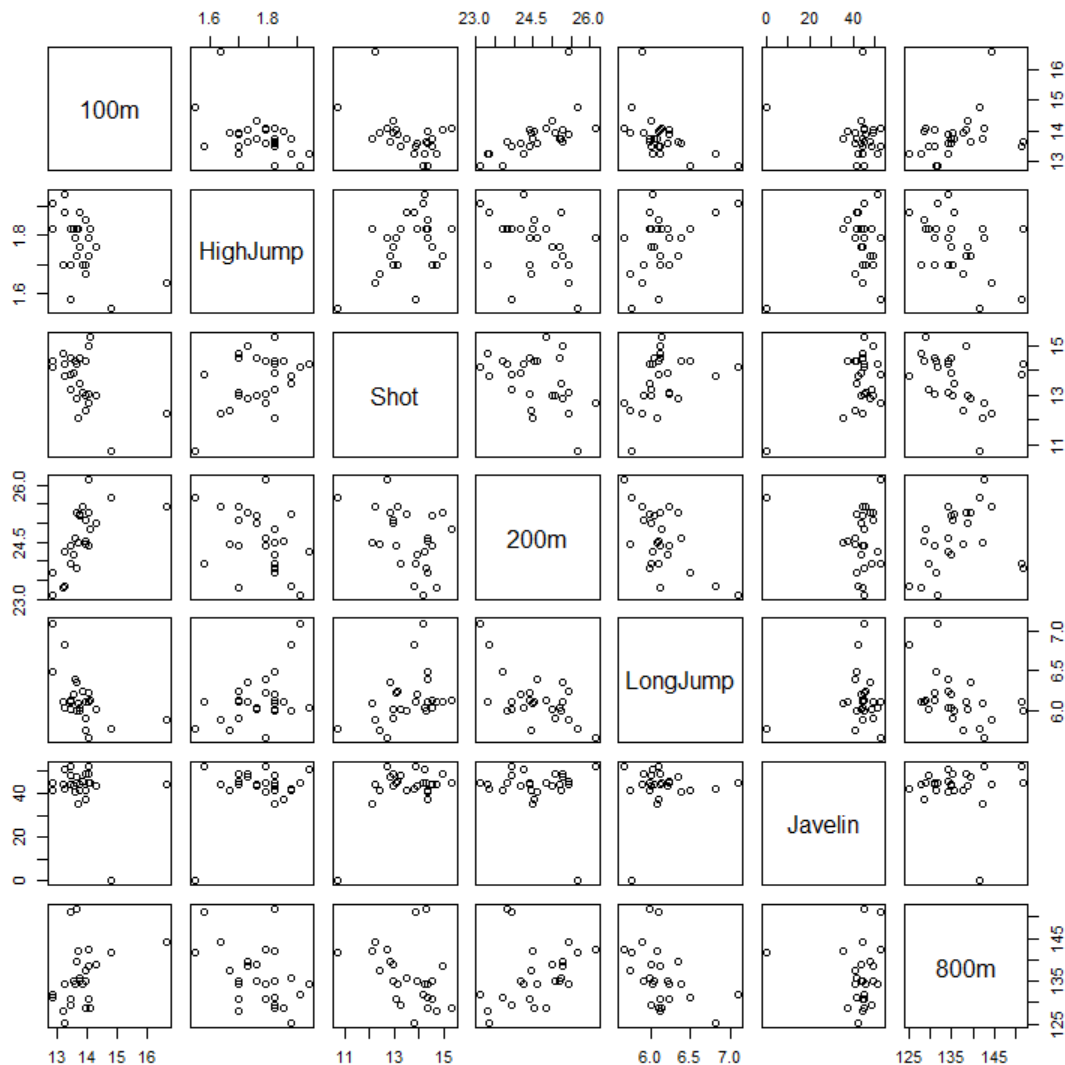
```
# Load data in R
heptathlon <- read.csv("heptathlon.csv")
head(heptathlon)
```

##	id	X100m	HighJump	Shot	X200m	LongJump	Javelin	X800m	score
## 1	1	13.87	1.70	13.11	25.44	6.23	45.42	134.31	6030
## 2	2	14.03	1.79	13.05	24.39	6.22	45.18	130.90	6251
## 3	3	14.79	1.55	10.71	25.66	5.76	0.00	141.59	4530
## 4	4	13.48	1.82	13.23	23.93	6.01	48.10	129.49	6434

```
## 5  5 13.25    1.88 13.77 23.34    6.82  41.90 125.08  6845
## 6  6 14.31    1.76 12.96 25.01    6.01  43.30 138.84  5897
```

Στην ανάλυση κυρίων συνιστωσών είναι απαραίτητο να υπάρχουν συσχετίσεις μεταξύ των μεταβλητών. Σε διαφορετική περίπτωση δεν έχει νόημα να κάνουμε PCA. Η συσχέτιση μεταξύ των μεταβλητών μπορεί να αξιολογηθεί μέσω ενός γραφήματος-πίνακα από scatterplots

```
# Examine correlations graphically
pairs(heptathlon[, -c(1, ncol(heptathlon))])
```



Οι συναρτήσεις `cor` και `cor` υπολογίζουν τους δειγματικούς πίνακες συνδιακύμανσης και συσχέτισης, αντίστοιχα. Οι πίνακες συνδιακύμανσης και συσχέτισης των δεδομένων παρουσιάζονται παρακάτω:

Covariance matrix of the data

`round(cov(heptathlon[, -c(1, ncol(heptathlon))]), 3) # Much Larger variances for Javelin and 800m`

```
##           100m HighJump   Shot   200m LongJump Javelin   800m
## 100m      0.509   -0.034 -0.374   0.349   -0.117  -1.740   1.762
## HighJump -0.034    0.009  0.044  -0.028    0.014   0.307  -0.284
## Shot      -0.374    0.044  1.098  -0.353    0.127   5.386  -2.702
## 200m       0.349   -0.028 -0.353   0.607   -0.150  -1.283   1.720
## LongJump  -0.117    0.014  0.127  -0.150    0.096   0.405  -0.928
## Javelin   -1.740    0.307  5.386  -1.283    0.405  93.659  -2.350
## 800m       1.762   -0.284 -2.702   1.720   -0.928  -2.350  45.142
```

Correlation matrix of the data

`round(cor(heptathlon[, -c(1, ncol(heptathlon))]), 3)`

```
##           100m HighJump   Shot   200m LongJump Javelin   800m
## 100m      1.000   -0.493 -0.500   0.629   -0.527  -0.252   0.368
## HighJump -0.493    1.000  0.441  -0.374    0.461   0.331  -0.442
## Shot      -0.500    0.441  1.000  -0.433    0.391   0.531  -0.384
## 200m       0.629   -0.374 -0.433   1.000   -0.621  -0.170   0.328
## LongJump  -0.527    0.461  0.391  -0.621    1.000   0.135  -0.445
## Javelin   -0.252    0.331  0.531  -0.170    0.135   1.000  -0.036
## 800m       0.368   -0.442 -0.384   0.328   -0.445  -0.036   1.000
```

`R <- cor(heptathlon[, -c(1, ncol(heptathlon))])`

phi

`p <- ncol(R)`

`phi <- sqrt((sum(R^2) - p)/(p*(p-1)))`

`phi`

```
## [1] 0.42114
```

Οι συσχετίσεις μεταξύ των μεταβλητών είναι μέτριες, υπάρχουν κάποιες ισχυρές συσχετίσεις (π.χ. τα 100 μέτρα με εμπόδια και τα 200 μέτρα) και κάποιες χαμηλές (π.χ. ο ακοντισμός έχει υψηλή συσχέτιση μόνο με τη σφαιροβολία). Το στατιστικό ϕ υπολογίστηκε στο 0.42, το οποίο δηλώνει μέτρια συσχέτιση αλλά αξίζει να διερευνηθεί). Παρατηρήστε επίσης ότι το μέγεθος των διακυμάνσεων διαφέρει πάρα πολύ (ο ακοντισμός έχει διακύμανση 93 περίπου ενώ στο άλμα σε ύψος η διακύμανση είναι μόλις 0.009). Αυτή η παρατήρηση είναι σημαντική για την ανάλυση με βάση τον πίνακα συνδιακύμανσης

Προχωράμε σε ανάλυση με βάση τον πίνακα συνδιακύμανσης αλλά και τον πίνακα συσχέτισης. Η συνάρτηση `eigen` υπολογίζει τη φασματική ανάλυση ενός πίνακα (περιλαμβάνει μια λίστα με τις ιδιοτιμές και τα αντίστοιχα ιδιοδιανύσματα)

Spectral decompositions of the covariance/correlation matrix

`S <- cov(heptathlon[, -c(1, ncol(heptathlon))])`

`R <- cor(heptathlon[, -c(1, ncol(heptathlon))])`

`eigS <- eigen(S)`

`eigR <- eigen(R)`

Στη συνέχεια, αποθηκεύουμε τα αποτελέσματα σε ένα `data.frame`

```
# In a table
eigTable <- data.frame(it = 1:(ncol(heptathlon)-2), EigValueS =
eigS$values,
                      PercS = eigS$values/sum(eigS$values))
eigTable$ceigS <- cumsum(eigTable$PercS)
eigTable$EigValueR <- eigR$values
eigTable$PercR <- eigR$values/sum(eigR$values)
eigTable$ceigR <- cumsum(eigTable$PercR)
round(eigTable[,1:4],3)

##   it EigValueS PercS ceigS
## 1  1    94.161 0.667 0.667
## 2  2    45.308 0.321 0.988
## 3  3     0.959 0.007 0.995
## 4  4     0.438 0.003 0.998
## 5  5     0.200 0.001 1.000
## 6  6     0.050 0.000 1.000
## 7  7     0.005 0.000 1.000
```

Παρατηρήστε ότι οι 2 πρώτες ιδιοτιμές του πίνακα συνδιακύμανσης είναι πολύ μεγάλες σε σχέση με τις υπόλοιπες. Επομένως, διαλέγοντας τις δύο πρώτες συνιστώσες εξηγούμε το 99% της συνολικής διακύμανσης. Αν διαλέξουμε μόνο μια συνιστώσα εξηγούμε το 66.7%

Βλέπουμε τα ιδιοδιανύσματα με βάση τον πίνακα συνδιακύμανσης

```
# Eigenvectors based on the covariance matrix
round(t(eigS$vectors),6)

##           [,1]      [,2]      [,3]      [,4]      [,5]      [,6]
## [7,]
## [1,]  0.019802 -0.003443 -0.059340  0.014941 -0.004937 -0.996535
##      0.052393
## [2,]  0.037744 -0.005976 -0.054564  0.037455 -0.020285  0.057007
##      0.995238
## [3,] -0.491883  0.024709  0.616033 -0.588679  0.148971 -0.051861
##      0.080738
## [4,]  0.319610 -0.011340  0.780255  0.524139 -0.114575 -0.031103
##      0.010309
## [5,]  0.805739 -0.026062  0.071453 -0.583656  0.065764  0.002578 -
##      0.003639
## [6,] -0.061431 -0.066489  0.008981 -0.190797 -0.977351 -0.000086 -
##      0.010312
## [7,] -0.033084 -0.997052  0.004458  0.007154  0.068589  0.002106 -
##      0.003479
```

Οι δύο πρώτες συνιστώσες σχεδόν ταυτίζονται με τον ακοντισμό και τα 800 μέτρα αντίστοιχα (σημειώστε ότι τα πρόσημα δεν παίζουν κάποιο ρόλο). Αυτές ήταν οι δύο μεταβλητές με τη μεγαλύτερη διακύμανση. Επομένως, τα αποτελέσματα δεν

έχουν κάποιο ενδιαφέρον και αυτό οφείλεται στις μεγάλες διαφορές των μεταβλητών ως προς τις διακυμάνσεις τους

Όταν η ανάλυση βασίζεται στον πίνακα συσχέτισης

```
round(eigTable[,5:7],3)

##   EigValueR PercR ceigR
## 1      3.439 0.491 0.491
## 2      1.141 0.163 0.654
## 3      0.781 0.112 0.766
## 4      0.546 0.078 0.844
## 5      0.458 0.065 0.909
## 6      0.325 0.046 0.956
## 7      0.310 0.044 1.000
```

Οι 2 πρώτες συνιστώσες εξηγούν μόλις το 65% της συνολικής διακύμανσης, ενώ οι 3 ανεβαίνουν στο 76%.

Η ανάλυση με βάση τα ιδιοδιανύσματα του πίνακα συσχέτισης

```
# Eigenvectors based on the correlation matrix
round(t(eigR$vectors),6)

##           [,1]      [,2]      [,3]      [,4]      [,5]      [,6]
## [1,]  0.429654 -0.391738 -0.401327  0.407340 -0.408511 -0.240791
## [2,]  0.082156  0.066686  0.377340  0.230331 -0.316381  0.770490
## [3,] -0.292121 -0.381523 -0.073663 -0.511351  0.159720  0.036265
## [4,]  0.100602 -0.759488  0.491387 -0.113056 -0.068450 -0.019332
## [5,]  0.655432 -0.063855 -0.105760 -0.030354  0.693881  0.267612
## [6,]  0.193857  0.226798  0.661771  0.162148  0.128028 -0.522687
## [7,]  0.496004  0.253373  0.023199 -0.692493 -0.452730 -0.043105
```

Τώρα κάθε συνιστώσα εξαρτάται από πολλές μεταβλητές (το θέλουμε αυτό). Η πρώτη συνιστώσα έχει θετικό πρόσημο στους δρόμους και αρνητικό στα υπόλοιπα αγωνίσματα. Άρα (ερμηνεία)?

Παρατηρήστε επίσης ότι το άθροισμα τετραγώνων των συντελεστών κάθε συνιστώσας είναι 1

```
# Eigenvectors are normalized
colSums(eigS$vectors^2)

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

```
colSums(eigR$vector^2)
```

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

Στους δρόμους οι καλές επιδόσεις είναι οι μικρές επιδόσεις (μικρότερος χρόνος τερματισμού) ενώ στις ρίψεις και τα άλματα οι μεγάλες επιδόσεις. Συνεπώς αυτή η διαφορά που βλέπουμε στα πρόσημα δεν σημαίνει σύγκριση αλλά είναι ένας σταθμισμένος μέσος όρος των επιδόσεων των αθλητριών. Για να γίνει αυτό πιο φανερό, αλλάζουμε πρόσημο στα αγωνίσματα των δρόμων

```
# Change signs so that that 'Large' values are 'good'
```

```
heptathlon[,c("X100m", "X200m", "X800m")] <- -  
heptathlon[,c("X100m", "X200m", "X800m")]
```

```
S <- cov(heptathlon[, -c(1, ncol(heptathlon))])
```

```
R <- cor(heptathlon[, -c(1, ncol(heptathlon))])
```

```
eigS <- eigen(S)
```

```
eigR <- eigen(R)
```

Όλες οι μεταβλητές είναι θετικά συσχετισμένες τώρα

```
# ALL variables are positively correlated now
```

```
round(R, 3)
```

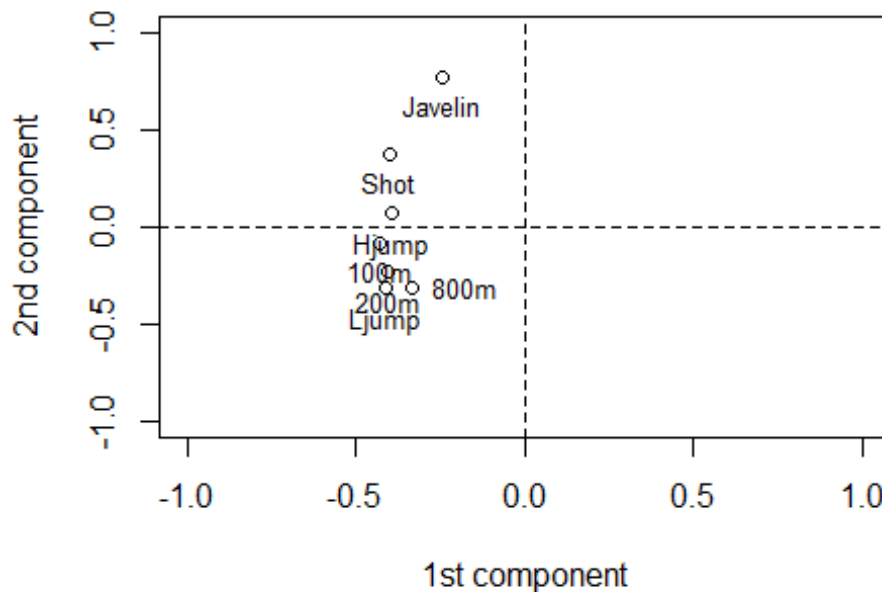
```
##           100m HighJump Shot  200m LongJump Javelin  800m  
## 100m      1.000    0.493 0.500 0.629    0.527   0.252 0.368  
## HighJump 0.493    1.000 0.441 0.374    0.461   0.331 0.442  
## Shot     0.500    0.441 1.000 0.433    0.391   0.531 0.384  
## 200m     0.629    0.374 0.433 1.000    0.621   0.170 0.328  
## LongJump 0.527    0.461 0.391 0.621    1.000   0.135 0.445  
## Javelin  0.252    0.331 0.531 0.170    0.135   1.000 0.036  
## 800m     0.368    0.442 0.384 0.328    0.445   0.036 1.000
```

```
round(t(eigR$vector), 6)
```

```
##           [,1]      [,2]      [,3]      [,4]      [,5]      [,6]  
## [,7]  
## [1,] -0.429654 -0.391738 -0.401327 -0.407340 -0.408511 -0.240791 -  
## 0.331796  
## [2,]  0.082156 -0.066686 -0.377340  0.230331  0.316381 -0.770490  
## 0.315615  
## [3,]  0.292121 -0.381523 -0.073663  0.511351  0.159720  0.036265 -  
## 0.689474  
## [4,] -0.100602 -0.759488  0.491387  0.113056 -0.068450 -0.019332  
## 0.392117  
## [5,]  0.655432  0.063855  0.105760 -0.030354 -0.693881 -0.267612  
## 0.033736  
## [6,]  0.193857 -0.226798 -0.661771  0.162148 -0.128028  0.522687  
## 0.396429  
## [7,] -0.496004  0.253373  0.023199  0.692493 -0.452730 -0.043105  
## 0.053611
```

Δημιουργούμε τώρα ένα γράφημα με τις δύο πρώτες κύριες συνιστώσες με βάση την εντολή `plot`. Στο άξονα x βάζουμε την πρώτη κύρια συνιστώσα (`eigR$vectors[,1]`) ενώ στον άξονα y τη δεύτερη (`-eigR$vectors[,2]`). Η συνάρτηση `text` προσθέτει κείμενο στο γράφημα ως `character vector` στην επιλογή `labels` = ενώ με το `x` = και `y` = δηλώνουμε τις συντεταγμένες. `pos=1` σημαίνει ότι το κείμενο θα εμφανιστεί κάτω από τις συντεταγμένες και `pos=4` δεξιά από τις συντεταγμένες.

```
# Plots with the coefficient of the first two principal components
plot(eigR$vectors[,1],-eigR$vectors[,2],xlim = c(-1,1),ylim = c(-1,1),
     xlab = "1st component",ylab = "2nd component")
abline(h = 0 ,v = 0,lty = 2)
text(x = eigR$vectors[-7,1],y = -eigR$vectors[-7,2],
     labels = c("100m","Hjump","Shot","200m","Ljump","Javelin"),
     pos = 1,cex = 0.8)
text(x = eigR$vectors[7,1],y = -eigR$vectors[7,2],
     labels = c("800m"),
     pos = 4,cex = 0.8)
```



Υπολογίζουμε τα σκορ των δύο πρώτων κυρίων συνιστωσών. Επειδή δουλεύουμε με τον πίνακα συσχέτισης, πρέπει πρώτα τα τυποποιήσουμε τα δεδομένα ώστε να έχουν μέση τιμή 0 και τυπική απόκλιση 1 (συνάρτηση `scale`). Για κάθε άτομο, το

σκορ των κύριων συνιστωσών είναι ένας γραμμικός συνδυασμός των τυποποιημένων δεδομένων με βάρη βάσει των ιδιοδιανυσμάτων. Επομένως, πολλαπλασιάζουμε τον πίνακα των τυποποιημένων μεταβλητών με τον πίνακα που περιλαμβάνει τις δύο πρώτες κύριες συνιστώσες.

```
# 1st and 2nd component scores
heptathlon$Comp1 <- NULL
heptathlon$Comp2 <- NULL
heptathlon[,c("Comp1", "Comp2")] <- scale(heptathlon[, -
c(1, ncol(heptathlon))]) %*% eigR$eigenvectors[,1:2]
```

Θέλουμε να συγκρίνουμε την πραγματική βαθμολογία με την βαθμολογία των αθλητριών που θα προέκυπτε εάν χρησιμοποιούσαμε την 1η κύρια συνιστώσα για να βαθμολογούσαμε τις αθλήτριες. Η εντολή `heptathlon[order(-heptathlon$score),]` διατάσσει το `data.frame` `heptathlon` σε φθίνουσα σειρά όσον αφορά στο τελικό σκορ. Επομένως η `1:nrow(dd)` μας δείχνει την κατάταξη κάθε αθλήτριας. Στη συνέχεια, διατάσσουμε το `data.frame` σε αύξουσα σειρά σύμφωνα με τον κωδικό της αθλήτριας και η μεταλητή `rankScore` μας δείχνει την κατάταξη της αθλήτριας με βάση το τελικό σκορ.

```
# Ranks according to the total score
dd <- heptathlon[order(-heptathlon$score),]
head(dd)
```

```
##      id  100m HighJump  Shot   200m LongJump Javelin    800m score
Comp1
## 14 14 -12.85      1.91 14.13 -23.12      7.10   44.98 -131.75  7044 -
3.6680002
## 5   5 -13.25      1.88 13.77 -23.34      6.82   41.90 -125.08  6845 -
2.9358388
## 7   7 -13.25      1.94 14.23 -24.27      6.02   51.12 -134.35  6649 -
1.5901747
## 19 19 -12.86      1.82 14.34 -23.70      6.49   41.30 -131.22  6619 -
2.2033443
## 12 12 -13.23      1.70 14.68 -23.31      6.11   44.48 -127.90  6464 -
1.5670497
## 4   4 -13.48      1.82 13.23 -23.93      6.01   48.10 -129.49  6434 -
0.9075069
##              Comp2
## 14  1.2711381
## 5   1.5837531
## 7   -0.8832393
## 19  0.7818384
## 12  0.3311000
## 4   0.0929018
```

```
dd$rankScore <- 1:nrow(dd)
dd <- dd[order(dd$id),]
heptathlon$rankScore <- dd$rankScore
head(dd)
```



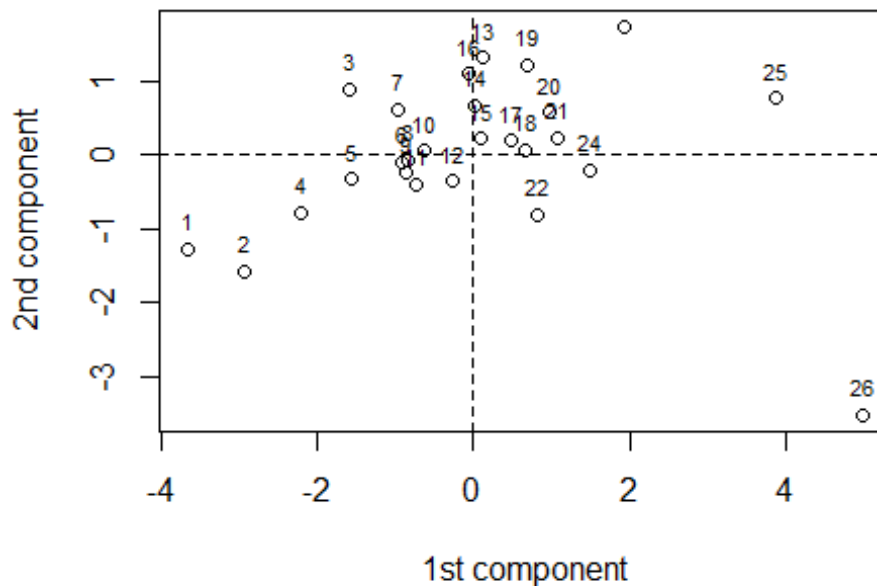
```
##   id  100m HighJump  Shot   200m LongJump Javelin    800m score
Comp1
## 1  1 -13.87      1.70 13.11 -25.44      6.23   45.42 -134.31  6030
0.6684539
## 2  2 -14.03      1.79 13.05 -24.39      6.22   45.18 -130.90  6251 -
0.2781540
## 3  3 -14.79      1.55 10.71 -25.66      5.76    0.00 -141.59  4530
4.9780847
## 4  4 -13.48      1.82 13.23 -23.93      6.01   48.10 -129.49  6434 -
0.9075069
## 5  5 -13.25      1.88 13.77 -23.34      6.82   41.90 -125.08  6845 -
2.9358388
## 6  6 -14.31      1.76 12.96 -25.01      6.01   43.30 -138.84  5897
1.0869050
##           Comp2 rankScore
## 1 -0.06047369      18
## 2  0.35951070      12
## 3  3.53230307      26
## 4  0.09290180       6
## 5  1.58375312       2
## 6 -0.24001042      21
```

Επαναλαμβάνουμε την ίδια διαδικασία για την κατάταξη των αθλητριών με βάση την πρώτη κύρια συνιστώσα

```
# Ranks according to the 1st principal component
dd <- heptathlon[order(heptathlon$Comp1),]
dd$rankComp1 <- 1:nrow(dd)
dd <- dd[order(dd$id),]
heptathlon$rankComp1 <- dd$rankComp1
```

Δημιουργούμε ένα γραφήμα των 2 κύριων συνιστωσών με την πραγματική κατάταξη να εμφανίζεται πάνω από κάθε σημεία

```
plot(heptathlon$Comp1, -heptathlon$Comp2,
     xlab = "1st component", ylab = "2nd component")
abline(h = 0, v = 0, lty = 2)
text(heptathlon$Comp1, -heptathlon$Comp2, labels =
heptathlon$rankScore, pos = 3, cex = 0.7)
```



Επιλογή αριθμού συνιστωσών

```
round(eigTable[5:7],4)
```

```
##   EigValueR  PercR  ceigR
## 1    3.4392 0.4913 0.4913
## 2    1.1408 0.1630 0.6543
## 3    0.7811 0.1116 0.7659
## 4    0.5461 0.0780 0.8439
## 5    0.4580 0.0654 0.9093
## 6    0.3248 0.0464 0.9557
## 7    0.3100 0.0443 1.0000
```

Kaiser Εάν χρησιμοποιήσουμε το κριτήριο του Kaiser, πρέπει να διαλέξουμε τόσες κύριες συνιστώσες όσες οι ιδιοτιμές που είναι μεγαλύτερες του 1. Έχουμε 2 ιδιοτιμές πάνω από 1, επομένως διαλέγουμε 2 συνιστώσες.

Συνολική διακύμανση Με βάση το κριτήριο της συνολικής διακύμανσης > 80%: διαλέγουμε 4 συνιστώσες ενώ αν ρίξουμε το ποσοστό αυτό στο 70% χρειαζόμαστε 3 συνιστώσες

Scree plot Το scree plot είναι ένα απλό γραφημα των ιδιοτιμών σε φθίνουσα σειρά. Διαλέγουμε όλες τις ιδιοτιμές μέχρι να διαπιστώσουμε ότι αρχίζει να αλλάζει η κλίση

```
# Scree plot  
plot(eigTable$it,eigTable$EigValueR,type = "b",col = "lightblue",lwd =  
2,  
      xlab = "Component number",ylab = "Eigenvalue",pch = 19)
```

