

Analisi delle Componenti Principali

1. Caricare i dati marks

```
marks <- read.table("http://www.maths.leeds.ac.uk/~charles/mva-data/openclosedbook.dat",header = TRUE)
X = as.matrix(marks)
# assegno i nomi alle variabili
colnames(X) <- c("Mechanics", "Vectors", "Algebra", "Analysis", "Statistics")
# guardo le prime righe
head(X)
```

```
##      Mechanics Vectors Algebra Analysis Statistics
## [1,]         77      82      67         67         81
## [2,]         63      78      80         70         81
## [3,]         75      73      71         66         81
## [4,]         55      72      63         70         68
## [5,]         63      63      65         70         63
## [6,]         53      61      72         64         73
```

```
n = nrow(X)
p = ncol(X)
```

2. Calcolare la matrice dei dati centrati \tilde{X} come trasformazione lineare $\tilde{X} = \frac{1}{n} X A' + \frac{1}{n} b'$ con $q = p$, $A = \frac{I}{p \times p}$
e $b = -\frac{1}{n} \bar{x}$

```
A = diag(rep(1,p))
one.n = matrix(rep(1,n))
b = (1/n) * t(X) %*% one.n

Xtilde = X %*% t(A) + one.n %*% (-t(b))
```

2. Calcolare il voto medio di ciascun studente come combinazione lineare $y_j = \sum_{n=1}^n \frac{1}{n} a_j$ con $a_j = 1/p$,
 $j = 1, \dots, p$

```
a = matrix(rep(1/p, p), ncol=1)
y = X %*% a
```

2. Calcolare la prima componente principale di \tilde{X} come $y_1 = \sum_{n=1}^n \tilde{X} v_1$ dove v_1 è il primo autovettore di $S = \frac{1}{n} \tilde{X}' \tilde{X}$ associato all'autovalore più grande λ_1 . Verificare che la varianza di y_1 è pari a λ_1 e che è maggiore della varianza della combinazione lineare normalizzata $y_j = \sum_{n=1}^n \tilde{X} a_j$ con $a_j = 1/\sqrt{p}$,
 $j = 1, \dots, p$.

```
# decomposizione spettrale di S
S = (1/n) * t(Xtilde) %*% Xtilde
eigen = eigen(S)
Lambda = diag(eigen$values)
V = eigen$vectors

# pesi (loadings) della 1ma componente principale
v1 = V[,1, drop=FALSE]
v1

##      [,1]
## [1,] -0.5054457
## [2,] -0.3683486
```

```
## [3,] -0.3456612
## [4,] -0.4511226
## [5,] -0.5346501

# punteggi (scores) della 1ma componente principale
y1 = Xtilde %*% v1

# varianza di y1
var(y1) * (n-1)/n # coincide con Lambda[1,1]

##          [,1]
## [1,] 679.1831

# confronto con altra combinazione lineare normalizzata
a = matrix(rep(1/sqrt(p), p), ncol=1)
y = Xtilde %*% a
var(y) * (n-1)/n
```

```
##          [,1]
## [1,] 662.6463
```

3. Calcolare le p componenti principali $Y = \tilde{X} V$. Verificare che il vettore medio di Y è nullo, la matrice di varianze/covarianze S^Y di Y è pari a Λ , che la varianza totale e generalizzata di S^Y è pari a quella di S

```
# p componenti principali Y
Y = Xtilde %*% V

# vettore medio di Y
round(
(1/n) * t(Y) %*% one.n
, 8)
```

```
##          [,1]
## [1,]      0
## [2,]      0
## [3,]      0
## [4,]      0
## [5,]      0
```

```
# matrice di varianze covarianze di Y
S_Y = (1/n) * t(Y) %*% Y # coincide con Lambda
round(
S_Y
, 8)
```

```
##          [,1]      [,2]      [,3]      [,4]      [,5]
## [1,] 679.1831  0.0000  0.0000  0.00000  0.00000
## [2,]  0.0000 199.8144  0.0000  0.00000  0.00000
## [3,]  0.0000  0.0000 102.5684  0.00000  0.00000
## [4,]  0.0000  0.0000  0.0000 83.66873  0.00000
## [5,]  0.0000  0.0000  0.0000  0.00000 31.78791
```

```
# varianza totale di S_Y
sum(diag(S_Y)) # coincide con sum(diag(S_Y))
```

```
## [1] 1097.022
```

```
# varianza generalizzata di S_Y
det(S_Y) # coincide con det(S)
```

```
## [1] 37021339491
```

4. Calcolare le componenti principali con il comando `princomp()` e `prcomp()`

```
pca = princomp(X)
summary(pca) # Standard deviation coincide con sqrt(diag(Lambda))
```

```
## Importance of components:
```

```
##              Comp.1      Comp.2      Comp.3      Comp.4
## Standard deviation 26.061142 14.1355705 10.12760414 9.14706148
## Proportion of Variance 0.619115 0.1821424 0.09349705 0.07626893
## Cumulative Proportion 0.619115 0.8012575 0.89475453 0.97102347
##              Comp.5
## Standard deviation 5.63807655
## Proportion of Variance 0.02897653
## Cumulative Proportion 1.00000000
```

```
# pesi
```

```
pca$loadings[,] # coincide con V
```

```
##              Comp.1      Comp.2      Comp.3      Comp.4      Comp.5
## Mechanics -0.5054457 0.74874751 0.2997888 0.296184264 -0.07939388
## Vectors -0.3683486 0.20740314 -0.4155900 -0.782888173 -0.18887639
## Algebra -0.3456612 -0.07590813 -0.1453182 -0.003236339 0.92392015
## Analysis -0.4511226 -0.30088849 -0.5966265 0.518139724 -0.28552169
## Statistics -0.5346501 -0.54778205 0.6002758 -0.175732020 -0.15123239
```

```
# punteggi
```

```
head(pca$scores) # coincide con head(Y)
```

```
##              Comp.1      Comp.2      Comp.3      Comp.4      Comp.5
## [1,] -66.32077 6.447125 7.0736275 -9.6463833 -5.4557651
## [2,] -63.61810 -6.754424 0.8599283 -9.1490636 7.5656517
## [3,] -62.92626 3.080258 10.2297139 -3.7238434 0.3841125
## [4,] -44.53775 -5.577218 -4.3780192 -4.4816746 -4.4065605
## [5,] -43.28425 1.133228 -1.5314139 5.8059805 -0.7378218
## [6,] -42.55249 -10.972900 4.8671678 -0.4788987 7.1021171
```

```
pca = prcomp(X, center = TRUE)
```

```
summary(pca) # Standard deviation coincide con sqrt(diag(Lambda)*(n/n-1))
```

```
## Importance of components:
```

```
##              PC1      PC2      PC3      PC4      PC5
## Standard deviation 26.2105 14.2166 10.1856 9.19948 5.67039
## Proportion of Variance 0.6191 0.1821 0.0935 0.07627 0.02898
## Cumulative Proportion 0.6191 0.8013 0.8948 0.97102 1.00000
```

```
# pesi
```

```
pca$rotation[,] # coincide con V
```

```
##              PC1      PC2      PC3      PC4      PC5
## Mechanics -0.5054457 -0.74874751 0.2997888 -0.296184264 -0.07939388
## Vectors -0.3683486 -0.20740314 -0.4155900 0.782888173 -0.18887639
## Algebra -0.3456612 0.07590813 -0.1453182 0.003236339 0.92392015
## Analysis -0.4511226 0.30088849 -0.5966265 -0.518139724 -0.28552169
```

```
## Statistics -0.5346501  0.54778205  0.6002758  0.175732020 -0.15123239
```

```
# punteggi
```

```
head( pca$x ) # coincide con head(Y)
```

```
##          PC1          PC2          PC3          PC4          PC5
## [1,] -66.32077 -6.447125  7.0736275  9.6463833 -5.4557651
## [2,] -63.61810  6.754424  0.8599283  9.1490636  7.5656517
## [3,] -62.92626 -3.080258 10.2297139  3.7238434  0.3841125
## [4,] -44.53775  5.577218 -4.3780192  4.4816746 -4.4065605
## [5,] -43.28425 -1.133228 -1.5314139 -5.8059805 -0.7378218
## [6,] -42.55249 10.972900  4.8671678  0.4788987  7.1021171
```