7 Febbraio 2020 - Analisi Esplorativa (Analisi Statistica Multivariata)

Cognome: .....

Nome: .....

Matricola: .....

Prova scritta - fila A

Si svolgano gli esercizi riportando il risultato dove indicato. Durata: 90 minuti

---

Esercizio 1 (Punti 2)

Si dimostri che $\underset{n\times n}{J} = I - H$ è una matrice idempotente, dove $\underset{n\times n}{I}$ è la matrice identità e $\underset{n\times n}{H}$ è la matrice di centramento, giustificando tutti i passaggi.

Esercizio 2 (Punti 4)

Si consideri la seguente matrice $\underset{n\times p}{X} = \begin{bmatrix} 3 & 1 & 0 \\ 6 & 4 & 6 \\ 4 & 2 & 2 \\ 7 & 0 & 3 \\ 5 & 3 & 4 \end{bmatrix}$

a. (Punti 1) Calcolare la matrice dei dati centrati $\tilde{X}$

b. (Punti 2) Verificare che le colonne di $\tilde{X}$ sono linearmente dipendenti, determinando un vettore non-nullo $\underset{p\times 1}{c}$ tale che $\tilde{X}c = \underset{n\times 1}{0}$

c. (Punti 1) Calcolare la matrice di varianze e covarianze S e la varianza generalizzata

$\tilde{X} = \begin{bmatrix} & \\ & \\ & \\ & \\ & \end{bmatrix}$ $c = \begin{bmatrix} \\ \\ \end{bmatrix}$ $S = \begin{bmatrix} & \\ & \end{bmatrix}$ $det(S) =$

```r
X = matrix(c(3,1,0,6,4,6,4,2,2,7,0,3,5,3,4),
           byrow=T,ncol=3)
n = nrow(X)
# a.
Xtilde = scale(X, center=T, scale=F)[,]
Xtilde
```

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

```r
# b.
c = matrix(c(1,1,-1),ncol=1)
c
```

```
     [,1]
[1,]    1
[2,]    1
[3,]   -1
```

```r
Xtilde %*% c
```

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

```r
# c.
S = (1/n)*t(Xtilde)%*%X
S
```

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

```r
# d.
det(S)
```

```
[1] 0
```

Esercizio 3 (Punti 4)

Alla matrice di varianze/covarianze $\underset{p\times p}{S}$ sono associati i seguenti autovalori $\lambda_1 = 6, \lambda_2 = 4$ e autovettori normalizzati $v_1 = \begin{bmatrix} 1/\sqrt{5} \\ 2/\sqrt{5} \end{bmatrix}$, $v_2 = \begin{bmatrix} 2/\sqrt{5} \\ -1/\sqrt{5} \end{bmatrix}$.

a. (Punti 2) Riportare la matrice di correlazione $\underset{p\times p}{R}$ e la matrice $\underset{p\times p}{S^{2/3}}$ (arrotondando i risultati al secondo decimale):

$$R = \begin{bmatrix} & \\ & \end{bmatrix} \qquad S^{2/3} = \begin{bmatrix} & \\ & \end{bmatrix}$$

b. (Punti 2) Calcolare la correlazione tra la prima colonna $\underset{n\times 1}{\tilde{x}_1}$ di $\underset{n\times p}{\tilde{X}}$ e i punteggi $\underset{n\times 1}{y_1}$ della prima componente principale, arrotondando il risultato al secondo decimale:

```
rm(list=ls())
Lambda = diag(c(6,4))
V = matrix(c(1/sqrt(5),2/sqrt(5),2/sqrt(5),-1/sqrt(5)), byrow=F, ncol=2)
S = V %*% Lambda %*% t(V)
# a.
R = diag(diag(S)^(-1/2)) %*% S %*% diag(diag(S)^(-1/2))
round(R,2)
```

```
     [,1] [,2]
[1,] 1.00 0.16
[2,] 0.16 1.00
```

```
round(V %*% Lambda^(2/3) %*% t(V),2)
```

```
     [,1] [,2]
[1,] 2.68 0.31
[2,] 0.31 3.15
```

```
# b.
round( V[1,1]*sqrt(Lambda[1,1])/sqrt(S[1,1]) , 2)
```

```
[1] 0.52
```

Esercizio 4 (Punti 3)

Si consideri la seguente matrice di distanza:

	1	2	3	4
1	0			
2	3	0		
3	7	9	0	
4	8	6	5	0

a. (Punti 2) Si utilizzi il metodo gerarchico agglomerativo con il legame completo, riportando ad ogni passo dell'algoritmo la matrice delle distanze tra gruppi

passo 1)

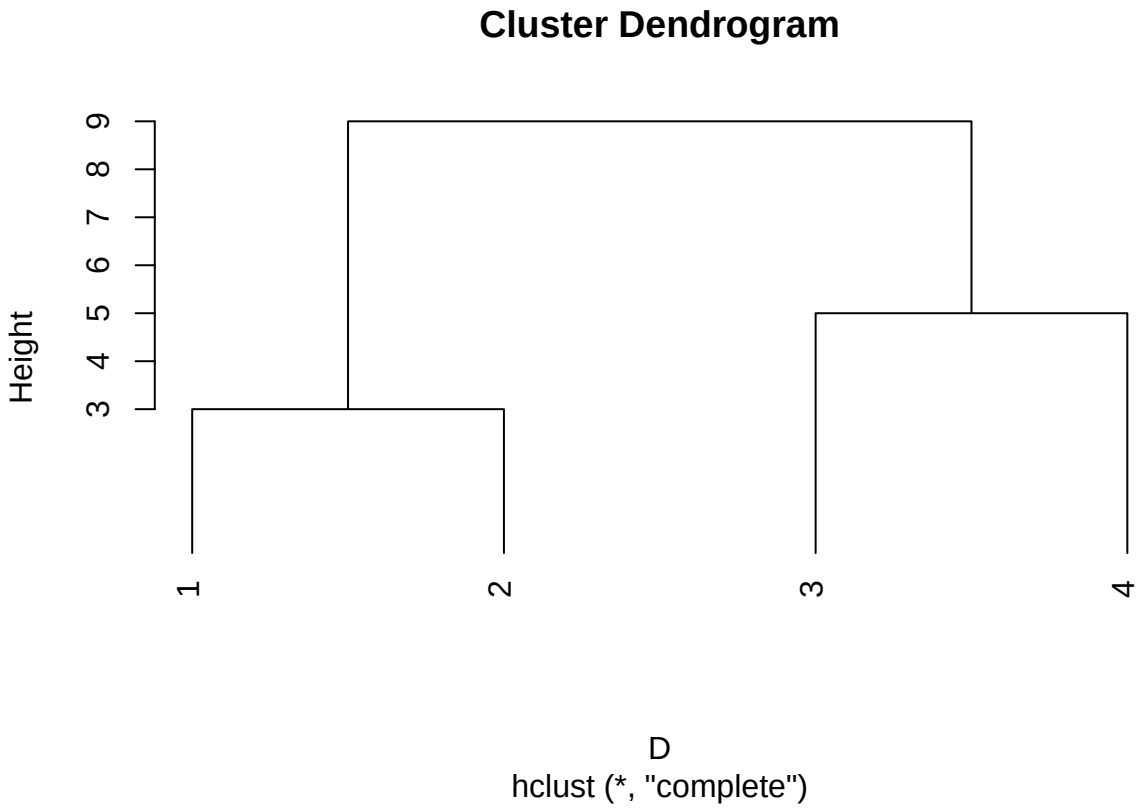
	.....	.....	.....
.....	0		
.....	.....	0	
.....	.....	.....	0

passo 2)

	.....	.....
.....	0	
.....	.....	0

b. (Punti 1) Riportare il dendogramma corrispondente al punto a.

```
D = as.dist(matrix(c(0,3,7,8,3,0,9,6,7,9,0,5,8,6,5,0), ncol=4))
# b.
hc = hclust(D, "complete")
plot(hc, hang=-1)
```



Esercizio 5 (Punti 9)

Si considereri la seguente matrice di correlazione calcolata sulla base di $n = 50$ osservazioni:

```
rm(list=ls())
R = round(cor(USArrests), 1)
R
```

```
         Murder Assault UrbanPop Rape
Murder      1.0     0.8      0.1  0.6
Assault     0.8     1.0      0.3  0.7
UrbanPop    0.1     0.3      1.0  0.4
```

```
Rape        0.6     0.7      0.4  1.0
```

a. (Punti 1) Sulla base dalla matrice di correlazione, si stimi il modello fattoriale con $k = 1$ fattore utilizzando il metodo della massima verosimiglianza senza effettuare alcuna rotazione. Arrontondando al terzo decimale, si riportino le stime dei pesi fattoriali:

$\hat{\lambda}_1 = \ldots\ldots\ldots\ldots$ $\hat{\lambda}_2 = \ldots\ldots\ldots\ldots$ $\hat{\lambda}_3 = \ldots\ldots\ldots\ldots$ $\hat{\lambda}_4 = \ldots\ldots\ldots\ldots$

```
hatLambda = matrix(factanal(covmat=R, factors=1, n.obs=50, rotation ="none")$ loadings[,], ncol=1)
round(hatLambda,3)
```

```
      [,1]
[1,] 0.819
[2,] 0.975
[3,] 0.303
[4,] 0.723
```

b. (Punti 2) Si determini il punteggio fattoriale con il metodo di Thompson (arrotondando alla quarta cifra decimale) per l'unità statistica "Arizona" sapendo che i suoi valori nelle quattro variabili standardizzate sono

	Murder	Assault	UrbanPop	Rape
Arizona	1.2426	0.7828	-0.5209	-0.0034


```
z = c(1.2426, 0.7828, -0.5209, -0.0034)
Rinv = solve(R)
round( t(hatLambda) %*% Rinv %*% z, 4)
```

```
       [,1]
[1,] 0.7822
```

c. (Punti 2) Si riporti la stima delle comunalità per le quattro variabili, arrotondando al secondo decimale. Indicare quale variabile è spiegata meglio dal modello, motivando la risposta.

	Murder	Assault	UrbanPop	Rape
Comunalità				

Variabile spiegata meglio:

```
# c
round(t(hatLambda)^2,2)
```

```
     [,1] [,2] [,3] [,4]
[1,] 0.67 0.95 0.09 0.52
```

```
# variabile spiegata meglio
c("Murder","Assault","UrbanPop","Rape")[which.max(hatLambda^2)]
```

```
[1] "Assault"
```

d. (Punti 1) Si valuti l'opportunità di stimare un modello a 2 fattori, motivando la risposta.

e. (Punti 3) Si calcolino le correlazioni tra la j-sima variabile standardizzata $\underset{n\times 1}{z_j}$ e la k-sima componente principale $\underset{n\times 1}{y_k}$ per $j = 1, \ldots, 4$ e $k = 1, \ldots, 4$, riportandole nella seguente tabella (arrotondando al secondo decimale).

	y_1	y_2	y_3	y_4
z_1				
z_2				
z_3				
z_4				

Potrebbe tornare utile sapere che le covarianze tra variabili standardizzate e componenti principali sono date dalla matrice

$$\frac{1}{n} Z'Y$$

dove $\underset{n\times p}{Y} = ZV$ è la matrice dei punteggi delle componenti principali e $R = V\Lambda V'$ è la matrice di correlazione.

```
V= eigen(R)$vectors
Lambda = diag(eigen(R)$values)
round( V %*% Lambda^(0.5),2)
```

```
      [,1]  [,2]  [,3]  [,4]
[1,] -0.85  0.40 -0.23  0.26
[2,] -0.92  0.16 -0.15 -0.31
[3,] -0.46 -0.87 -0.19  0.06
[4,] -0.87 -0.10  0.48  0.05
```

Esercizio 6 (Punti 4)

Si consideri il dataset `swiss` presente nella libreria `datasets`, che contiene $n = 47$ unità statistiche (provincie) relative alle seguenti 6 variabili:

- *Fertility* : common standardized fertility measure
- *Agriculture* : % of males involved in agriculture as occupation
- *Examination* : % draftees receiving highest mark on army examination
- *Education* : % education beyond primary school for draftees
- *Catholic* : % catholic (as opposed to protestant)
- *Infant.Mortality* : live births who live less than 1 year

a. (Punti 2) Per ciascuna unità statistica, si calcoli la distanza di Mahalanobis dal baricentro, e si riportino i valori delle distanze superiori a 3.35 (arrotondando al terzo decimale) con i rispettivi nomi delle unità (provincie)

```
# a
rm(list=ls())
```

```
X = as.matrix(swiss)
n = nrow(X)
xbar = matrix(colMeans(X), ncol=1)
S = var(X)*((n-1)/n)
InvS = solve(S)
dM2 = apply(X,1, function(u) t(u-xbar) %*% InvS %*% (u - xbar) )
dM = sqrt(dM2)
round(dM[which(dM > 3.35)],3)
```

```
  Porrentruy    La Vallee    Neuchatel V. De Geneve
       3.599        3.974        3.361        4.519
```

b. (Punti 2) Standardizzare i dati `swiss`, e utilizzare l'algoritmo delle K-medie (`algorithm = Hartigan-Wong`) per $K = 2, 4, 6$, inizializzando i centroidi con le osservazioni di riga $1, 2, \ldots, K$. Riportare per ciascun valore di K il rispettivo valore dell'indice Calinski and Harabasz (arrotondando al terzo decimale).

```
# b
Z = scale(X, center=T, scale= diag(S)^(1/2))
K = c(2,4,6)
CH <- vector()
for (i in 1:length(K)){
  k = K[i]
  km = kmeans(Z, centers = Z[1:k,])
  W = km$tot.withinss
  B = km$betweenss
  CH[i] = (B/(k-1)) / (W/(n-k))
}
rbind(K, round(CH,3))
```

```
    [,1]   [,2]   [,3]
K  2.000  4.000  6.000
  24.719 24.305 19.853
```

K	2	4	6
Indice CH	.....	.....	.....