# Analisi Esplorativa (Analisi Statistica Multivariata)

Prova d'esame

27 Aprile 2022

*Tempo a disposizione: 120 minuti*

*Modalità di consegna: svolgere gli esercizi di teoria (parte A) riportando le soluzioni sul foglio protocollo, e consegnare il foglio protocollo assieme al testo della prova d'esame. Successivamente, accedere alla piattaforma esamionline tramite computer e svolgere gli esercizi di analisi dei dati (parte B). In questo caso la consegna si svolge tramite piattaforma esamionline. Il tempo da dedicare alla parte A e alla parte B è a discrezione dello studente.*

*Compilare con nome, cognome e numero di matricola. E' obbligatorio consegnare il testo della prova d'esame all'interno del foglio protocollo contente le soluzioni degli esercizi di teoria.*

---

NOME:

COGNOME:

MATRICOLA:

---

## PARTE A: esercizi di teoria

*Esercizi da svolgere sul foglio protocollo senza l'ausilio di R/Rstudio.*

### Problema 1

1. Si consideri il modello fattoriale

$$z_1 = 0.9f + u_1$$
$$z_2 = 0.7f + u_2$$
$$z_3 = 0.5f + u_3$$

relativo alle variabili standardizzate $z = (z_1, z_2, z_3)^t$, dove $\mathbb{V}\text{ar}(f) = 1, \mathbb{C}\text{ov}(u, f) = 0$ e

$$\Psi = \mathbb{C}\text{ov}(u) = \begin{bmatrix} 0.19 & 0 & 0 \\ 0 & 0.51 & 0 \\ 0 & 0 & 0.75 \end{bmatrix}$$

a) Calcolare la matrice di varianze/covarianze $\Sigma = \mathbb{C}\text{ov}(\underset{3\times 1}{z})$

b) Calcolare le comunalità $h_i^2$, $i = 1, 2, 3$

c) Calcolare $\mathbb{C}\text{orr}(z_i, f)$, $i = 1, 2, 3$

2. Sia $R = \begin{bmatrix} 1 & r \\ r & 1 \end{bmatrix}$ la matrice di correlazione dei dati standardizzati $Z$, con $r \in (-1, 0)$.

a) Determinare gli autovalori $\lambda_1$ e $\lambda_2$ di $R$ (rispettando l'ordinamento $\lambda_1 > \lambda_2$).

b) Determinare gli autovettori normalizzati $v_1$ e $v_2$ di $R$ corrispondenti a $\lambda_1$ e $\lambda_2$.

c) Determinare l'equazione del vettore dei punteggi $\underset{n\times 1}{y_1}$ della prima componente principale di $Z$, i.e.

$$y_{i,1} = \ldots\ldots\ldots \quad i = 1, \ldots, n$$

dove $y_{i,1}$ è l'$i$-esimo elemento del vettore $y_1$.

3. Si dimostri che una matrice quadrata $\underset{n\times n}{A}$ idempotente ha autovalori $\lambda_i \in \{0, 1\}$ per $i = 1, \ldots, n$, giustificando tutti i passaggi.

---

# PARTE B: esercizi di analisi dei dati

*Esercizi da svolgere con il computer sulla piattaforma esamionline con l'ausilio di R/Rstudio.*

## Problema 2

Si consideri il dataset `state.x77` presente nella libreria `datasets`. Questo dataset descrive 50 stati degli Stati Uniti d'America misurati su 8 variabili. Sia $X$ la matrice $49 \times 8$ corrispondente al dataset `state.x77`, escludendo la riga 1 (nel vostro esercizio il numero di riga potrebbe essere diverso)

1. Calcolare il coseno dell'angolo (espresso in radianti) compreso tra $\tilde{x}_1$ e $\tilde{x}_2$ (dove $\tilde{x}_1$ e $\tilde{x}_2$ sono i vettori scarto dalla media corrispondenti alle variabili `Population` e `Income`). Riportare il risultato arrotondando al **terzo decimale** (si ricordi l'uso della **virgola** per i decimali).

```
rm(list=ls())
X = state.x77[-1,]
R = cor(X)
a = R[1,2]
round(a,3)
```

```
## [1] 0.208
```

2. Calcolare l'elemento di posizione [2, 2] (riga 2, colonna 2) della matrice $R^{-1/2}$, dove $R$ indica la matrice di correlazione associata a $X$. Riportare il risultato arrotondando al **terzo decimale** (si ricordi l'uso della **virgola** per i decimali).

```
V = eigen(R)$vectors
lambdas = eigen(R)$values
b = V %*% diag(1/sqrt(lambdas)) %*% t(V)
round(b[2,2],3)
```

```
## [1] 1.307
```

3. Svolgere l'analisi delle componenti principali, decidendo opportunamente se basarla sui dati originali o sui dati standardizzati. Calcolare la correlazione in valore assoluto tra il vettore dei punteggi $y_1$ della prima componente principale e la variabile `Area` (standardizzata oppure no a seconda della scelta effettuata). Riportare il risultato arrotondando al **terzo decimale** (si ricordi l'uso della **virgola** per i decimali).

```
# 3
c = abs( sqrt(lambdas[1]) * V[8,1] )
round(c, 3)
```

```
## [1] 0.045
```

## Problema 3

Si consideri il dataset `state.center` (anch'esso presente nella libreria `datasets`), che riporta la longitudine (con segno negativo, variabile `x`) e la latitudine (variabile `y`) del centro geografico di ogni stato considerato nel Problema 2. Sia $X$ la matrice $49 \times 2$ corrispondente al dataset `state.center`, escludendo la riga 1 (nel vostro esercizio il numero di riga potrebbe essere diverso).

1. Sulla base di $X$, calcolare la matrice $D$ delle distanze Euclidee tra gli Stati, e riportare il valore della distanza (non nulla) più piccola, arrotondando al **terzo decimale** (si ricordi l'uso della **virgola** per i decimali).

```
rm(list=ls())
Y = data.frame(longitudine = state.center$x,
               latitudine = state.center$y)
X = Y[-1,]
n = nrow(X)
p = ncol(X)
D = dist(X)
a = min(D)
round( a, 3)
```

```
## [1] 0.896
```

2. Calcolare il numero di osservazioni anomale verificando se la distanza di Mahalanobis al quadrato di ciascuna osservazione dal baricento è superiore alla soglia $s$, dove $s$ corrisponde al quantile 0.95 di una variabile casuale $\chi^2_p$ (dove $p$ è il numero di colonne di $X$). Riportare il numero di osservazioni anomale.

```
xbar = matrix(colMeans(X), nrow=p, ncol=1)
S = var(X) * ((n-1)/n)
InvS = solve(S)
dM2 = apply(X,MARGIN=1, function(u) t(u-xbar) %*% InvS %*% (u - xbar) )
s = qchisq(0.95, df=p)
b = sum(dM2 > s)
b
```

```
## [1] 2
```

3. Rimuovere da $X$ le osservazioni anomale individuate al punto precedente e svolgere l'analisi dei gruppi utilizzando l'algoritmo agglomerativo gerarchico con il legame completo. Decidere il numero di gruppi $K$ ottimale secondo il criterio della *silhouette*, per $K = 2, \ldots, 20$. Riportare il valore medio della *silhouette* corrispondente alla scelta effettuata, arrotondando al **terzo decimale** (si ricordi l'uso della **virgola** per i decimali).

```
out = which(dM2 > s)
X_no_out = X[-out, ]
D_no_out = dist(X_no_out)
require(cluster)
```

```
## Loading required package: cluster
```

```
sil_media <- vector()
Ks = 2:20
for (k in Ks){
hc = hclust(D_no_out, method="complete")
gruppi = cutree(hc, k=k)
sil_media[k-1] <- summary(silhouette(gruppi, dist=D_no_out))$avg.width
}
K_opt = which.max(sil_media) + 1
```

```r
c = max(sil_media)
round( c, 3)
```

```
## [1] 0.518
```