

Esempio esame

Stefano Bussolon

15 maggio 2019

Domanda 1

Variabili

```
variabile1 <- c(5:9,TRUE,1,7,8)
mode_v1 <- mode(variabile1)
length_v1 <- length(variabile1)
isna_v1 <- is.na(variabile1)
names_v1 <- names(variabile1)
```

Output

Associare alla variabile il corrispondente output.

- A) FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
- B) numeric
- C) 9
- D) 5 6 7 8 9 1 1 7 8
- E)

Risposte

(A,B,C,D,E) variabile1: []; mode_v1: []; length_v1: []; isna_v1: []; names_v1: [];

Domanda 2

Creiamo una matrice, ed applichiamo ad essa alcune funzioni.

Variabili

```
variabile2 <- matrix( c(11,12,13,14,21,22,23,24,31,32,33,34),nrow=4,ncol=3)
dim_v2 <- dim(variabile2)
length_v2 <- length(variabile2)
asvector_v2 <- as.vector(variabile2)
t_v2 <- t(variabile2)
```

Output

Associare alla variabile il corrispondente output.

- A) 12
- B) 11 12 13 14 21 22 23 24 31 32 33 34
- C) 4 3
- D)

```
      [,1] [,2] [,3] [,4]
[1,]   11   12   13   14
[2,]   21   22   23   24
[3,]   31   32   33   34
```

Risposte

(A,B,C,D,E) str_v2: []; dim_v2: []; length_v2: []; asvector_v2: []; t_v2: [];

Domanda 3

Creiamo il vettore `variabile3`, ed applichiamo alcuni filtri.

Variabili

```
variabile3 <- c(2,4,6,8,10,12,14,16,18,20)
filtro3_1 <- variabile3[-c(2,4,6,8)]
filtro3_2 <- variabile3[4:9]
filtro3_3 <- variabile3[c(FALSE,TRUE)]
filtro3_4 <- variabile3[variabile3>9]
filtro3_5 <- which(variabile3>9)
```

Output

Associare alla variabile il corrispondente output.

- A) 8 10 12 14 16 18
- B) 10 12 14 16 18 20
- C) 4 8 12 16 20
- D) 2 6 10 14 18 20
- E) 5 6 7 8 9 10

Risposte

(A,B,C,D,E) filtro3_1: []; filtro3_2: []; filtro3_3: []; filtro3_4: []; filtro3_5: [];

Domanda 4

Creiamo la matrice `matrice4` ed applichiamo alcuni filtri.

Variabili

```
matrice4 <- matrix(c(11,12,13,14,21,22,23,24,31,32,33,34),
                  nrow=4,ncol=3)
colnames(matrice4) <- c("C_A", "C_B", "C_C")
rownames(matrice4) <- c("R1", "R2", "R3", "R4")
matrice4
```

```
   C_A C_B C_C
R1  11  21  31
R2  12  22  32
R3  13  23  33
R4  14  24  34
```

```
filtro4_1 <- matrice4[4,2]
filtro4_2 <- matrice4[c(1,3),3]
filtro4_3 <- matrice4[,2]
filtro4_4 <- matrice4["R3",]
filtro4_5 <- matrice4["R4",c(1,2)]
```

Output

Associare alla variabile il corrispondente output.

- A) 13 23 33
- B) 24
- C) 14 24
- D) 31 33
- E) 21 22 23 24

Risposte

(A,B,C,D,E) filtro4_1: []; filtro4_2: []; filtro4_3: []; filtro4_4: []; filtro4_5: [];

Domanda 5

Creiamo il dataframe `dataframe_5` ed applichiamo alcuni filtri.

Variabili

```
var5_1 <- c("A", "B", "C", "D", "E")
var5_2 <- c(54, 75, 12, 66, 95)
var5_3 <- c("X", "Y", "Y", "Y", "X")
var5_4 <- c(TRUE, TRUE, FALSE, TRUE, FALSE)
```

```
dataframe_5 <- data.frame (var5_1,var5_2,var5_3,var5_4)
dataframe_5
```

```
  var5_1 var5_2 var5_3 var5_4
1      A     54      X   TRUE
2      B     75      Y   TRUE
3      C     12      Y FALSE
4      D     66      Y   TRUE
5      E     95      X FALSE
```

```
filtro5_1 <- dataframe_5 [2]
filtro5_2 <- dataframe_5 [1:3,2:3]
filtro5_3 <- dataframe_5$var5_2
filtro5_4 <- dataframe_5[dataframe_5$var5_3=="X",c(1,2)]
filtro5_5 <- dataframe_5$var5_4[dataframe_5$var5_2>60]
```

Output

Associare alla variabile il corrispondente output.

A)

```
  var5_1 var5_2
1      A     54
5      E     95
```

B) TRUE TRUE FALSE

C) 54 75 12 66 95

D)

```
  var5_2
1      54
2      75
3      12
4      66
5      95
```

E)

```
  var5_2 var5_3
1      54      X
2      75      Y
3      12      Y
```

Risposte

(A,B,C,D,E) filtro5_1: []; filtro5_2: []; filtro5_3: []; filtro5_4: []; filtro5_5: [];

Domanda 6

Variabili

```
variabile6 <- sample(1:9,10, replace=TRUE)
v6_min <- min(variabile6)
v6_max <- max(variabile6)
v6_sum <- sum(variabile6)
v6_mean <- mean(variabile6)
v6_sd <- sd(variabile6)
```

Output

Associare alla variabile il corrispondente output.

- A) 58
- B) 3
- C) 9
- D) 5.8
- E) 2.529822

Risposte

(A,B,C,D,E) v6_min: []; v6_max: []; v6_sum: []; v6_mean: []; v6_sd: [];

Domanda 7

Variabili

```
v7_rep <- rep(c(2,4,2,2),times=5)
v7_seq <- seq(2,40,length=20)
v7_sample <- sample(1:20)
v7_sort <- sort(sample(1:20,20,replace = TRUE))
v7_rev <- rev(seq(1,40)[sample(c(TRUE,FALSE),40,replace = TRUE)])
```

Output

Associare alla variabile il corrispondente output.

- A) 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40
- B) 40 39 38 37 31 29 27 22 21 15 6 5 2
- C) 16 15 14 10 20 8 11 19 2 12 6 17 7 9 3 5 1 13 4 18
- D) 1 1 2 3 5 6 6 6 8 8 11 11 13 13 13 13 16 18 19 20
- E) 2 4 2 2 2 4 2 2 2 4 2 2 2 4 2 2 2 4 2 2

Risposte

(A,B,C,D,E) v7_rep: []; v7_seq: []; v7_sample: []; v7_sort: []; v7_rev: [];

Domanda 8

Variabili

```
var8_a <- sample(1:15,8)
var8_b <- sample(1:15,8)
v8_intersect <- intersect(var8_a,var8_b)
v8_union <- union(var8_a,var8_b)
v8_setdiff <- setdiff(var8_a,var8_b)
v8_setequal <- setequal(var8_a,var8_b)
```

Output

Associare alla variabile il corrispondente output.

- A) 10 5 15 13 7 6 11 9
- B) 4 1 14 2
- C) 4 1 15 14 2 10 9 13 5 7 6 11
- D) 15 10 9 13
- E) FALSE

Risposte

(A,B,C,D,E) var8_b: []; v8_intersect: []; v8_union: []; v8_setdiff: []; v8_setequal: [];

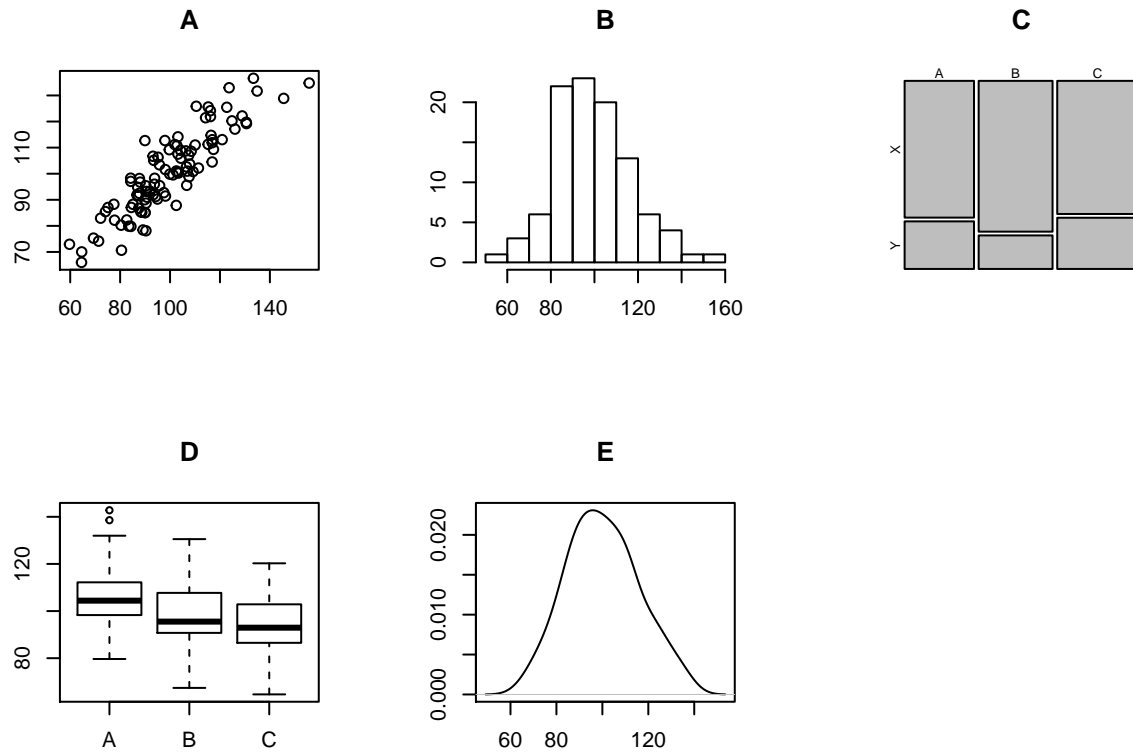
Domanda 9

Variabili

```
fig9_1 <- plot(table(var9_1,var9_2))
fig9_2 <- plot(var9_4~var9_5)
fig9_3 <- boxplot(var9_6~var9_1)
fig9_4 <- plot(density(var9_4))
fig9_5 <- hist(var9_5)
```

Output

Associare alla funzione il corrispondente grafico.



Risposte

(A,B,C,D,E) fig9_1: []; fig9_2: []; fig9_3: []; fig9_4: []; fig9_5: [];

Domanda 10

Variabili

Attaching package: 'cowplot'

The following object is masked from 'package:ggplot2':

```
ggsave
plot10_1 <- ggplot(dataframe_9,aes(x="", fill=var9_2)) +
  geom_bar(width = 0.5) +
  coord_polar("y", start=0)

plot10_2 <- ggplot(data=dataframe_9, aes(var9_4,var9_5)) +
  geom_point(aes(col=var9_3))

plot10_3 <- ggplot(dataframe_9,aes(var9_1)) +
  geom_bar(aes(fill=var9_2), width = 0.5)
```

```

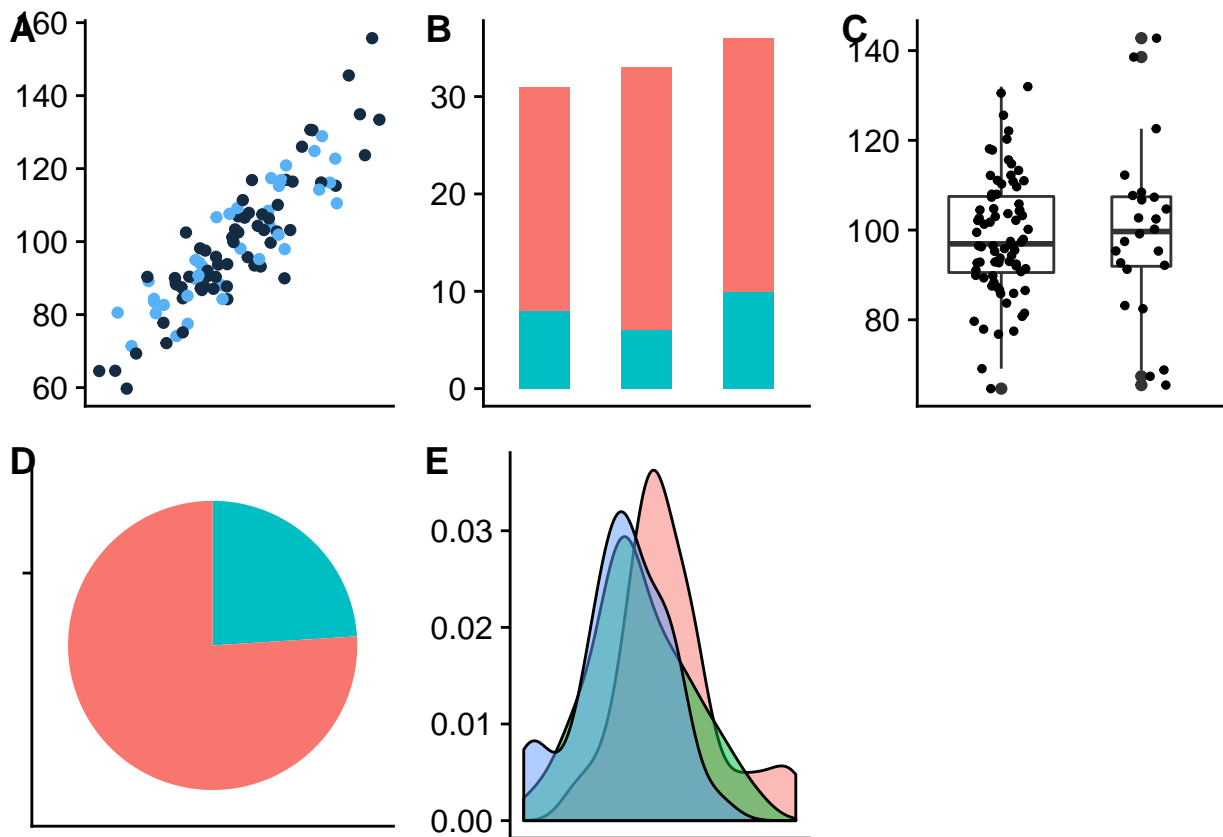
plot10_4 <- ggplot(data=dataframe_9, aes(var9_2,var9_6)) +
  geom_boxplot(varwidth=T) +
  geom_jitter(width = .2, size=1)

plot10_5 <- ggplot(data=dataframe_9, aes(var9_6)) +
  geom_density(aes(fill=factor(var9_1)), alpha=0.5)

```

Output

Associare alla funzione il corrispondente grafico.



Risposte

(A,B,C,D,E) ggplot10_1: []; ggplot10_2: []; ggplot10_3: []; ggplot10_4: []; ggplot10_5: [];

Domanda 11

Funzioni e variabili

```

funzione11_1 <- function (vettore) {
  ritorno <- rep(0,length(vettore))
  precedente <- vettore[length(vettore)]
}

```



```

for (i in 1:length(vettore)) {
  ritorno[i] <- vettore[i] + precedente
  precedente <- vettore[i]
}
return(ritorno)
}

funzione11_2 <- function(vettore) {
  ritorno <- rep(0, length(vettore))
  for (i in 1:length(vettore)) {
    if (vettore[i] < mean(vettore)) {
      ritorno[i] <- vettore[i] + 10
    } else {
      ritorno[i] <- vettore[i] + 1
    }
  }
  return(ritorno)
}

var11_x <- c(3,10,6,7,8)
var11_y <- c(6,5,1,9,10)

out11_1x <- funzione11_1(var11_x)
out11_1y <- funzione11_1(var11_y)
out11_2x <- funzione11_2(var11_x)
out11_2y <- funzione11_2(var11_y)

```

Output

Associare alla variabile il corrispondente output.

- A) 13 11 16 8 9
- B) 11 13 16 13 15
- C) 16 15 11 10 11
- D) 16 11 6 10 19

Risposte

(A,B,C,D) out11_1x: []; out11_1y: []; out11_2x: []; out11_2y: [];

Domanda 12

Scaricare in locale ed aprire il dataframe xxxx.

Per ognuna delle seguenti coppie di variabili, generare l'appropriata rappresentazione grafica bivariata, valutare quando opportuno gli assunti (normalità, omoschedasticità) e le appropriate statistiche inferenziali.

- nominale1 e nominale3
- intervalli1 e intervalli2
- nominale1, nominale2 e intervalli4

Riportare i risultati in un file rMarkdown.