Analiza podataka i obrada informacija - službeni šalabahter

Osnove

```
print("Hello, World!")  # ispis niza znakova
paste("Hello", "World") # spajanje više objekata u jedan tekstualni niz

a <- 42
class(a) # provjera vrste podataka
rm(a)    # brisanje varijable


if (1 == 0) { # if / else
  print(1)
} else {
  print(0)
}

for (i in 1:10) { # for petlja
  print(paste("i = ", i))
}

naziv_funkcije <- function(argument_1, argument_2, ...) { # funkcija
  # tijelo funkcije/operacije
  return(varijabla/vrijednost)
}
```

Učitavanje podataka

```
data("mtcars") # učitavanje ugrađenih R podataka
cars <- mtcars

getwd() # vraća aktivni radni direktorij
podaci <- read.csv("data/primjer.csv") # učitavanje CSV datoteke

podaci <- read.table("data/primjer.txt", header = TRUE, sep = " ") # učitavanje TXT
datoteke

install.packages("readxl")
library(readxl)
podaci <- read_excel("data/primjer.xlsx", sheet = 1) # učitavanje XLSX datoteke
```

Provjera i čišćenje podataka

```
is.na(podaci) # provjera nedostajućih vrijednosti
colSums(is.na(podaci)) # broj nedostajućih vrijednosti po stupcima

podaci_clean <- na.omit(podaci) # uklanjanje redaka s nedostajućim vrijednostima
podaci_clean2 <- podaci[complete.cases(podaci),] # uklanjanje redaka s nedostajućim
vrijednostima
```

Kreiranje i korištenje slučajnog uzorka

```
SEED <- 1234567890
set.seed(SEED)

podaci <- c("Marko", "Ana", "Ivan")
imena <- sample(podaci, 2) # biranje 2 nasumična podatka

redci <- sample(nrow(podaci), 20) # biranje 20 nasumičnih redaka
stupci <- sample(ncol(podaci), 5)  # biranje 5 nasumičnih stupaca
```

Rad s vektorima

```
podaci <- c(8, 5, 7, 1, 2) # izrada vektora

podaci[2] # dohvaća drugi element - 5
podaci[podaci > 5] # dohvaća elemente - c(8, 7)

podaci_1 <- c(8, 5, 7)
podaci_2 <- c(1, 5)

podaci_3 <- c(8, 5, 7, podaci_2) # spajanje vektora - c(8, 5, 7, 1, 5)
podaci_3 <- c(podaci_1, podaci_2) # spajanje vektora - c(8, 5, 7, 1, 5)
podaci_3 <- setdiff(podaci_1, podaci_2) # razlika vektora - c(8, 7)

ifelse(podaci < 4 | podaci > 7, podaci, 0) # inline uvjet - c(8, 0, 0, 1, 2)
```

Funkcije za rad s vektorima

```
podaci <- c(1, 2, 4, 2, 7, 2)

length(podaci)         # vraća broj elemenata u vektoru                    - 6
sum(podaci)            # zbraja sve elemente u vektoru                     - 18

min(podaci)            # vraća minimalnu vrijednost u vektoru              - 1
max(podaci)            # vraća maksimalnu vrijednost u vektoru             - 7
range(podaci)          # vraća minimalnu i maksimalnu vrijednost           - c(1, 7)

mean(podaci)           # vraća prosječnu vrijednost vektora                - 3
median(podaci)         # vraća srednju vrijednost sortiranog vektora       - 2

quantile(podaci, 0.75) # vraća 75. percentil (gornji kvartil)              - 3.5
quantile(podaci, 0.25) # vraća 25. percentil (donji kvartil)               - 2
IQR(podaci)            # vraća interkvartilni raspon                       - 1.5

sd(podaci)             # vraća standardnu devijaciju vektora               - 2.136976
var(podaci)            # vraća varijancu vektora                           - 4.566667

sort(podaci)           # sortira elemente u vektoru                        - c(1, 2, 2,
2, 4, 7)
rev(podaci)            # obrće redoslijed elemenata u vektoru              - c(2, 7, 2,
4, 2, 1)

which(podaci > 3)      # vraća indekse elemenata koji zadovoljavaju uvjet - c(3, 5)
any(podaci > 5)        # vraća TRUE ako jedan element zadovoljava uvjet   - TRUE
all(podaci > 6)        # vraća TRUE ako svi elementi zadovoljavaju uvjet  - FALSE

diff(podaci)           # vraća razlike između susjednih elemenata          - c(1, 2,
-2, 5, -5)
unique(podaci)         # vraća jedinstvene vrijednosti                     - c(1, 2, 4,
7)
table(podaci)          # vraća frekvenciju svake jedinstvene vrijednosti   - 1   2   4
7
                       #                                                   - 1   3   1
1
```

Rad s okvirima

```
podaci <- data.frame(
  Ime = c("Ana", "Marko", "Iva", "Pero"),
  Dob = c(25, 30, 22, 25),
  Visina = c(168, 175, 160, 190)
) # izrada okvira

podaci$Ime # vraća vektor stupca: c('Marko', 'Ana', 'Ivan', 'Pero')
podaci['Ime'] # vraća podokvir stupca
podaci[['Ime']] # vraća vektor stupca
podaci[c('Ime', 'Dob')] # vraća podokvir s danim stupcima

podaci[2, ]   # vraća drugi redak kao podokvir
podaci[, 3]   # vraća treći stupac kao podokvir
podaci[2, 3]  # Vraća vrijednost u drugom retku i trećem stupcu kao podokvir

podaci$Težina <- c(55, 80, 60, 87) # dodavanje novog stupca ili ažuriranje vrijednosti
stupca
podaci$Visina <- NULL # brisanje stupca

podaci[podaci$Dob > 23, ] # filtriranje redaka
podaci[podaci$Dob > 23 & podaci$Visina < 180, ] # filtriranje redaka
podaci[podaci$Dob > 23, c('Ime', 'Dob')] # filtriranje redaka i stupaca

podaci[which.min(podaci$Dob),] # filtriranje retka s min vrijednosti
podaci[which.max(podaci$Visina),] # filtriranje retka s max vrijednosti
podaci[which(podaci$Dob == max(podaci$Dob)),] # isto radi što i linija gore
podaci[podaci$Dob == max(podaci$Dob),] # isto radi što i linija gore

podaci[order(podaci$Dob, decreasing = TRUE), ] # sortiranje redaka
```

Funkcije za rad s okvirima

```
     Ime  Dob  Visina
1    Ana   25     168
2  Marko   30     175
3    Iva   22     160
4   Pero   25     191

colnames(podaci) # vraća nazive stupaca
colnames(podaci) <- c("Name", "Age", "Height") # promjena naziva svih stupaca
colnames(podaci)[2] <- "Year" # promjena naziva određenog stupca

nrow(podaci)     # vraća broj redaka okvira            - 4
ncol(podaci)     # vraća broj redaka okvira            - 3
dim(podaci)      # vraća broj redaka i stupaca okvira  - c(4, 3)

head(podaci, n = 2L) # vraća prva 2 retka
tail(podaci, n = 4L) # vraća zadnja 4 retka

str(podaci)     # prikazuje strukturu objekta
summary(podaci) # vraća sažetak podataka
# (min, max, medijan, prosjek za numeričke podatke, frekvencije za kategorijske)

# tablica kontingencije
table(podaci2$Ime, podaci2$Dob) # prikazuje raspodjelu dviju kategorijskih varijabli
proportions(tablica)            # izračunavanje proporcionalnih vrijednosti
addmargins(tablica)             # dodaje sumarne stupce i retke

aggregate(Visina ~ Dob, data = podaci, FUN = mean)
# grupira podatke i primjenjuje funkciju sažimanja (mean, sum, max, ...)

scale(podaci[-1])    # normalizacija podataka (Z-score)
colSums(podaci[-1])  # računa zbroj svih vrijednosti u svakom stupcu
colMeans(podaci[-1]) # računa prosječnu vrijednost svih vrijednosti u svakom stupcu

sapply(podaci[-1], function(x) { return(mean(x)) })
# primjenjuje zadanu funkciju na svaki element vektora

colMean <- function (x) {
  return(mean(x))
}
sapply(podaci[-1], colMean ) # koristeći vlastitu funkciju
sapply(podaci[-1], mean )    # koristeći postojeću funkciju
```

X/Y plot graf

```
x <- 1:10
y <- c(1,3,5,2,4,6,4,8,6,2)

plot(
    x,                  # vrijednosti na osi x
    y,                  # vrijednosti na osi y

    xlim = c(1, 9),     # raspon x vrijednost
    ylim = c(1, 8),     # raspon y vrijednost

    main = "Naslov",    # naslov grafa
    xlab = "X os",      # naziv osi x
    ylab = "y os",      # naziv osi y

    cex.main = 2,       # veličina naslova
    cex.lab = 1.5,      # veličina naziva osi
    cex.axis = 1.25,    # veličina labela osi

    type = "b",         # vrsta linija/točaka
    lth  = 2,           # vrsta linije

    pch  = 21,          # vrsta točke
    lwd  = 3,           # debljina linije i obruba točki
    cex  = 2,           # veličina točki

    col  = "brown",     # boja linije i točki
    bg   = "orange",    # boja ispune točke
    fg   = "tomato",    # boja obruba grafa
)

# dodavanje linija na graf
abline(lm(mpg ~ wt), col="red",   lwd=2)          # linija linearne regresije
abline(h=mean(mpg),  col="blue",  lwd=2, lty=2)   # linija aritmetičke sredine mpg
abline(v=median(wt), col="green", lwd=2, lty=2)   # linija medijana wt

# dodavanje legende na graf
legend(
  "topright",                                     # pozicija
  legend=c("lm(y ~ x)", "mean(y)", "median(x)"), # nazivi
  col=c("red", "blue", "green"),                  # boje
  lty=c(1,2,4),                                   # linije
)
```

Histogram

```
vrijednosti <- c(2,3,4,3,4,3,3,2,3,2)
hist(
    vrijednosti,        # vrijednosti

    breaks = 4,         # razredi

    xlim = c(2, 4),     # raspon x vrijednost
    ylim = c(1, 5),     # raspon y vrijednost

    main = "Naslov",    # naslov
    xlab = "X os",      # naziv osi x
    ylab = "Y os",      # naziv osi y

    cex.main = 2,       # veličina naslova
    cex.lab = 1.5,      # veličina naziva osi
    cex.axis = 1.25,    # veličina labela osi

    col = "lightblue",  # boja stupaca
)
```

Boxplot

```
vrijednosti <- c(106, 102, 141, 121, 116, 168, 119, 71, 164, 67, 54, 98, 82, 240)
boxplot(
    vrijednosti,        # vrijednosti

    ylim = c(50, 250),  # raspon y vrijednost

    main = "Naslov",    # naslov
    xlab = "X os",      # naziv osi x
    ylab = "Y os",      # naziv osi y

    cex.main = 2,       # veličina naslova
    cex.lab = 1.5,      # veličina naziva osi
    cex.axis = 1.25,    # veličina labela osi

    col = "lightblue",  # boja boxplot-a
)
```

Rad s grafovima

```
par(mfrow = c(retci, stupci)) # omogućuje prikaz više grafova u istom

# dodavanje legende na graf
legend(
  "topright",                                    # pozicija
  legend=c("lm(y ~ x)", "mean(y)", "median(x)"), # nazivi
  col=c("red", "blue", "green"),                 # boje
  lty=c(1,2,4),                                  # linije
)
```

Vrste linija: `p` za točke, `l` za linije, `o` za preklopljene točke i linije, `b` za točke povezane linijama, `c` za prazne točke povezane linijama, `s` i `S` za korake, `h` za okomite linije te `n` za isključivanje točaka i linija.