1. Uvod u R

R je nastao početkom devedesetih kao implementacija programskog jezika S. Danas je S implementiran kao programski paket S-PLUS. Iako postoje razlike u odnosu na R, većina programa se jednako može izvršiti u oba sustava.

Instalacijski paket može se pronaći na adresi http://www.r-project.org/. Iako ova instalacija omogućava sve funkcionalnosti R-a, dodatno se mogu koristiti praktičnija i vizualno ugodnija okruženja kao što je primjerice RStudio. Instalacija se može pronaći na https://www.rstudio.com/ (odabrati RStudio Desktop Open Source Edition).

Korisničko sučelje RStudia standardno je podijeljeno na četiri dijela: otvorene skripte za upisivanje koda, povijest naredbi i memorija, konzola i output prozor u kojem se prikazuju datoteke, slike, paketi i pomoć. Za osnovne funkcionalnosti grafičkog sučelja pogledati https://www.rstudio.com/wp-content/uploads/2016/01/rstudio-IDE-cheatsheet.pdf (dostupno i u izborniku na Help → Cheatsheets → RStudio IDE Cheat Sheet). Iz istog dokumenta vrijedi zapamtiti nekoliko kratica:

- Ctrl+Enter – slanje naredbe iz skripte na izvršavanje u konzoli,
- Tab – izbor nadopunjavanja započete naredbe,
- Ctrl+Shift+C – komentiranje retka.

Popis svih kratica može se dobiti s Alt+Shift+K.

Od osnovih svojstava R sintakse treba napomenuti da je R osjetljiv na mala i velika slova, a komentari započinju znakom #. Dozvoljeno je koristiti točku u imenovanju varijabli i funkcija pa su riječi u nazivima često razdvojene točkom, primjerice funkcija read.table().

1 Osnovne naredbe

R je kalkulator, upisivanje jednostavnih računskih naredbi odmah daje rezultat:

```
2+2
exp(1)
```

Help se poziva na nekoliko različitih načina. Help je standardno u obliku html datoteke što omogućuje navigiranje između različitih dijelova. U RStudiu help će se prikazati u donjem desnom dijelu sučelja, a za pozivanje helpa dovoljno je označiti ključnu riječ i stisnuti F1.

```
#pozivanje helpa
help("if")
?"if"
??histogram
?hist
```

```
example(hist)
```

Za učitavanje datoteka važno je definirati radni direktorij (working directory – wd) u kojem se nalaze datoteke. Osim direktne naredbe, u RStudiu to je moguće napraviti putem izbornika: u donjem desnom dijelu prozora na kartici Files odabrati željeni direktorij te zatim na More → Set As Working Directory. Drugi način je u glavnom izborniku otići na Session → Set Working Directory gdje se može odabrati direktorij ili jednostavno postaviti na mjesto trenutno otvorene skripte.

```
#postavljanje direktorija  (uobičajeno Windows adresiranje direktorija s \
    ovdje je s \\ ili s /)
getwd()
setwd("C:\\Users\\Danijel\\Desktop\\")
getwd()
setwd("C:/Users/Danijel/Desktop/")
getwd()
```

Sve varijable s kojima radimo, učitane su u radnu memoriju. Sljedećim naredbama se može vidjeti njihov popis i obrisati neke ili sve. Operator <- je operator pridruživanja i odgovara = ili := u drugim programskim jezicima. Može se koristiti i obratno, ->. U RStudio <- se brzo dobije s Alt+-.

```
#brisanje svih varijabli iz memorije
x <- 0
0 -> x
ls()              #popis svih varijabli u memoriji
rm(x)
rm(list=ls())
```

2 Objekti u R-u i manipulacija

Objekt	Različiti tipovi podataka u jednom objektu?	Objašnjenje
vector	Ne	Niz podataka istog tipa
factor	Ne	Kategorijalna varijabla
array	Ne	k-dimenzionalno polje podataka istog tipa
matrix	Ne	Specijalan slučaj array za $k = 2$
data frame	Da	Skup nekoliko vektora ili factora istog ili različitog tipa
ts	Ne	Podaci vremenskog niza
list	Da	Lista objekata bilo kojeg tipa

2.1 Vektori

Vektor je niz podataka istog tipa (numerički, znakovni...). Najčešće se zadaje funkcijom c() – konkatenacija. Prilikom bilo kakvog pridruživanja u konzoli se ne prikazuje nikakav ispis. Ukoliko želimo ispis, stavljamo naredbu u zagrade. Uobičajene operacije i funkcije djeluju na vektore po komponentama.

```
#Vektor
x <- c(1, 4, 5, 2, 3)
(x <- c(1, 4, 5, 2, 3))
y <- c(2, 3, 1, 9, 4)
y
c(x,y)

exp(x)
x*y
x+y
x^2
```

Postoji nekoliko zgodnih načina kako možemo jednostavno zadati vektore, primjerice, korištenjem operatora :, te funkcija seq i rep.
Funkcija seq može imati više argumenata (pogledati help) od kojih svaki ima svoje ime. Općenito u R-u, kada pozivamo funkciju možemo argumente naznačiti navodeći ime argumenta = vrijednost argumenta, i u tom slučaju redoslijed nije bitan (npr. seq(from=1, to=12, by=2) je isto kao i seq(to=12, from=1, by=2)). Ukoliko ne navodimo ime argumenta, tada je bitno da se vrijednosti argumenta predaju onim redoslijedom kako je to definirano za funkciju (piše u helpu). Primjerice, prva tri argumenta funkcije seq su from, to i by. Stoga je poziv seq(from=1,to=12,by=2) ekvivalentan pozivu seq(1,12,2). Ako eventualno neki argument preskačemo (kad nije obavezan), onda za sve argumente poslije tog moramo napisati njegovo ime.
Više naredbi može se zapisati u jednom retku ako se odvoje s ;.

```
x <- 3:10
(x <- seq(from = 1, to = 12, by = 2))
(x <- seq(to = 12, from = 1, by = 2))
(x <- seq(1, 12, 2))
seq(1, 10, length = 7)
rep(1:4, 2); rep(1:4, 1:4); rep(1:4, each=3)
```

Osim numeričkih važni su i tzv. logički vektori i znakovni vektori.

```
#Logički vektori
(x <- 4:9)
mode(x)
vek1 <- x>5
vek1
mode(vek1)
y <- c(0, 3, 6, 7, 10, 11)
(vek2 <- x==y)
```

```
vek1 & vek2
vek1 | vek2

#vektori znakova
(znakovi <- c("d", "3", "R", "v"))
mode(znakovi)
```

Elementi vektora dohvaćaju se indeksima unutar uglatih zagrada. Negativan indeks daje vektor bez tog elementa. Indeks može biti i vektor koji sadrži redne brojeve elemenata koji će biti dohvaćeni. Posebno korisna je mogućnost indeksiranja logičkim vektorom iste duljine. Na taj način moguće je dohvaćati elemente na osnovu logičkih uvjeta.

```
#Indeksiranje vektora
length(x)
x <- 1:20
x[3:15]
x[-5]
x[1:10][-5]
x[seq(1, length(x), 2)]
x[x<=5]
```

2.2 Faktori

Faktori predstavljaju poseban tip namijenjen kategorijalnim varijablama koje mogu poprimiti konačno mnogo različitih vrijednosti. Korisnost ovog tipa dolazi do izražaja pri statističkoj analizi podataka.

```
#Neka je x vektor podataka o spolu u kojima 0 označava muški i 1 ženski
    spol
x <- c(1, 0, 0, 1, 1, 1, 0, 1)
factor(x)
(x.f <- factor(x, labels = c("M", "Ž"))) #redoslijed kategorija odgovara
    numeričkom redoslijedu
```

2.3 Matrice

Matrica se može zadati spajanjem vektora kao retke – rbind() (row bind) ili spajanjem vektora kao stupce – cbind() (column bind). Funkcija t() vrši transponiranje matrice. Drugi način zadavanja matrice je funkcijom matrix().

```
#Matrice
rbind(1:4, 1:4)
cbind(1:4, 1:4)
t(cbind(1:4, 1:4))

matrix(data = 1:15, nrow = 5, ncol = 3)
```

```
matrix(data = 1:15, nrow = 5, ncol = 3, byrow = TRUE)
matrix(8, 2, 3)
matrix(1:6, nrow = 2)

diag(1, 3)
diag(c(4, 5))
```

Standardne matrične operacije:

```
#Matrične operacije
(A <- matrix(1:16, 4, 4, byrow = TRUE))
(B <- matrix(16:1, 4, 4, byrow = TRUE))
A*B
A %*% B
diag(A)

A <- matrix(rnorm(16, 0, 1), 4, 4)
x <- 1:4
b <- A %*% x
solve(A, b)
inverz <- solve(A)
inverz %*% A
```

Dohvaćanje elemenata matrice obavlja se zadavanjem dva indeksa odvojena zarezom unutar uglatih zagrada. Prvi indeks označava redak, a drugi stupac. Ako jedan od indeksa nije naveden, ispisuje se cijeli redak ili cijeli stupac. Redci i stupci matrice mogu se proizvoljno imenovati. Osim toga matrice se mogu mijenjati unutar ugrađenog tabličnog editora.

```
#Indeksiranje elemenata matrice
(A <- matrix(1:12, nrow = 3, byrow = TRUE))
A[2,3]
A[2,]
A[2:3,-1]

dim(A)

rownames(A) <- c("a", "b", "c")
A
data.entry(A)
A
```

2.4 Ostali tipovi objekata

Data frame je objekt najsličniji tablici podataka s kakvima se susrećemo u statističkim analizama. Za razliku od matrice, stupci mogu biti različitih tipova (osim numerički, stupci mogu biti znakovni ili faktori). Pri tome, svaki stupac ima svoj naziv. Podaci koji se učitavaju obično se spremaju u data frame.

Lista je niz objekata bilo kojeg tipa. Primjerice, lista može sadržavati i listu. Većina procedura kao rezultat daje listu iz koje se onda dohvaćaju podaci. Elementi liste dohvaćaju se operatorom $ (djelomično odgovara . u nekim objektno orijentiranim jezicima).

```
#Ostali tipovi objekata
lista <- list(x = c(1, 2, 3), y = c(5, 6), z = list(a = 2, b = 9))
lista$x
lista$x[2]
lista$z
lista$z$b
```

Osim numeričkih, znakovnih i faktor podataka postoje i neki specijalni kao što su: Inf – beskonačno, NaN – Not A Number, NA – Not Available (missing value – nedostajuća vrijednost).

```
#Posebne vrijednosti
is.finite(pi)
is.finite(Inf)
?NaN
is.nan(NaN)
?NA
is.na(NA)
is.na(1:10)
is.na(c(1, 2, 3, NA, NA, 2))
```

2.5 Kontrola toka i petlje

Sintaksa osnovnih naredbi za kontrolu toka i petlji:

```
#If, for i funkcije
x <- 5
if (x<=6) print("manji") else print("veći")
if (x<=6) {print("manji"); print("stvarno manji")} else print("veći")

x <- 0
for (i in 1:10) {
x <- x + i
}
x

zbroj <- function(a, b) {
return(a + b)
}

zbroj(2,3)
```

3 R paketi

Bogatstvo R-a leži u preko 13000 paketa koji se jednostavno instaliraju i sadrže gotove procedure za brojne statističke probleme. Pokretanjem se učitava oko 20 paketa, a svi ostali se instaliraju i učitavaju po potrebi. Instalacija i učitavanje može se izvršiti naredbama:

```
#Instaliranje i učitavanje paketa
install.packages("optimx")
library("optimx")
```

Putem izbornika u RStudiu, pakete je moguće instalirati na Tools → Install Packages.

4 Učitavanje podataka

Podaci koji se učitavaju mogu biti u različitim formatima, primjerice .txt, .csv, .dat, .xls...

Tekstualne datoteke formata txt mogu se učitavati funkcijom read.table. Prije toga treba postaviti radni direktorij na mjesto gdje se datoteka nalazi. Važne opcije funkcije read.table su: header koja označava ima li datoteka prvi redak u kojem su zapisani nazivi stupaca. Ako su podaci iz Excela, tada je obično decimalna oznaka zarez, a ne točka kao u R-u, pa to treba naglasiti stavljanjem dec=",". Objekt koji dobijemo učitavanjem je data frame. Kad su podaci učitani, važno je provjeriti da je sve kako treba funkcijom str. Funkcijom str može se vidjeti ako neki podaci eventualno nisu učitani kao numerički a trebali bi biti. To bi se dogodilo primjerice ako propustimo postaviti dec=",".

```
#Učitavanje podataka
getwd()
Lignje <- read.table(file = "lignje.txt", header = TRUE, dec = ",")
Lignje
str(Lignje) #structure
names(Lignje)
dim(Lignje)
```

Funkcija read.table može učitavati podatke i iz drugih tipova datoteka, primjerice csv (comma separated values). Kraći oblik u ovom slučaju omogućen je funkcijom read.csv2 koja će po defaultu koristiti zarez kao decimalnu oznaku i ; kao sep argument (default vrijednost za sep je prazan prostor). Sljedeći pozivi su ekvivalentni:

```
hormon <- read.csv2('hormon.csv')
str(hormon)
hormon1 <- read.table('hormon.csv', header = TRUE, dec = ",", sep = ";")
str(hormon1)
hormon2 <- read.table('hormon.txt', header = TRUE, dec = ",", sep = "\t")
str(hormon2)
```

Podaci koje želimo analizirati najčešće su spremljeni u obliku jedne Excel tablice. Takve je podatke moguće učitati direktno iz Excela (vidjeti primjerice paket "xlsx" ili

"xlsReadWrite"): Uobičajeno se podaci učitavaju indirektno tako da se najprije spreme u csv formatu. Prije toga, prvo treba pripremiti Excel datoteku. Ako su nazivi u prvom retku (header), ti nazivi ne smiju sadržavati hrvatske dijakritičke znakove i ne smiju biti neka ključna riječ (npr. data je ključna riječ). Zatim u Excelu, idemo na Save As i kao tip odaberemo CSV (MS-DOS). Tako dobivenu datoteku možemo učitati u R kao csv.

U RStudiu učitavanje se može obaviti i putem glavnog izbornika na File → Import Dataset, odabrati datoteku i eventualno korigirati parametre učitavanja. Isto je moguće i u gornjem desnom dijelu klikom na Import Dataset. Novije verzije RStudia omogućavaju i direktno učitavanje iz Excela.

Nakon učitavanja, podaci se spremaju u tip data.frame. Stupcu dobivenog data framea pristupa se operatorom $, a svaki stupac je vektor čijim elementima pristupamo indeksiranjem s uglatim zagradama. Svi stupci se mogu učitati u memoriju tako da ime stupca postane ime varijable. Pri tome treba biti pažljiv jer to može prebrisati neku staru varijablu istog imena.

```
#pristupanje ucitanim podacima
Lignje$GSI
GSI
attach(Lignje)
GSI
detach(Lignje)
GSI
attach(Lignje)

#sortiranje podataka
Lignje[order(Mjesec),]
Lignje

#snimanje podataka
LignjeM <- Lignje[Spol==1, ]
write.table(LignjeM, file = "LignjeM.txt", sep = "\t", quote = FALSE,
    append = FALSE, na = "NA")
```

5 Grafika u R-u

Osnovni grafički paket koji se učitava pokretanjem R-a je graphics. Osim njega, postoje i neki drugi kao što su lattice ili ggplot. Grafičke funkcije dijele se u osnovne skupine:

- High-level funkcije koje daju kompletan graf
- Low-level funkcije koje služe dodavanju elemenata na postojeći graf dobiven pozivom high-level funkcije
- Funkcije za interaktivan rad s grafom

Osnovna grafička funkcija je plot(). To je generička funkcija, što znači da kao ulazne varijable može primati razne vrste objekata.

Dobivene slike lako se spremaju u druge oblike. Mogu se spremiti korištenjem odgovarajućih funkcija pdf(), jpeg(), png(), odnosno direktno u RStudiu korištenjem Export u donjem desnom dijelu ili izbornika Plots → Save as Image.

```
#Grafika u R-u
pressure
str(pressure)
plot(pressure)
plot(pressure$temperature, pressure$pressure)
plot(pressure ~ temperature, data = pressure)

pdf("sl1.pdf", width = 7, height = 5)
plot(pressure ~ temperature, data = pressure)
dev.off()
```

Opća sintaksa funkcije plot je oblika plot(x,y) gdje je x vektor x-koordinata točaka, a y vektor y-koordinata točaka. Točke će biti spojene ako to odredimo odgovarajućim atributom. Osnovni atributi funkcije plot:

- type – osnovni izgled grafa, za točke p, za linije l itd. (pogledati help: ?plot)
- lwd – debljina linije ako je tip l
- lty - tip linije, npr. dashed za isprekidanu
- col – boja grafa (boja se može zadati na razne načine, osnovne boje su dostupne kao engleske riječi, za ostale može se koristiti funkcija rgb)
- main – glavni naziv grafa
- sub – podnaslov
- xlab, ylab – naziv x, y osi
- xlim, ylim – granice koje određuju koliki dio x i y osi će biti prikazan

Svaki od ovih atributa je opcionalan i služi uređivanju grafa ako nismo zadovoljni standardnim postavkama.

```
plot(pressure$temperature, pressure$pressure, type = "l", lwd = 1.2, lty =
    "dashed", col = "red", main = "Glavni naslov", sub = "Podnaslov", xlab =
     "temperatura", ylab = "tlak", asp = 0.2)

x <- 1:20
y <- rnorm(20, 5, 1)
plot(x, y, main = "Naziv grafa", xlab = "Naziv x-osi", ylab = "Naziv y-osi"
    , xlim = c(0, 22), ylim = c(0, 10), type = "p")
plot(x, y, main = "Naziv grafa", xlab = "Naziv x-osi", ylab = "Naziv y-osi"
    , xlim = c(0, 22), ylim = c(0, 10), type = "b")
```

```
plot(x, y, main = "Naziv grafa", xlab = "Naziv x-osi", ylab = "Naziv y-osi"
    , xlim = c(0, 22), ylim = c(0, 10), type = "o", pch = 22, col = "red",
    bg = "yellow", cex = 2)
```

Funkcija plot je high-level funkcija i njome dobivamo jedan graf i uređujemo okvir slike, nazive i sl. Sve druge grafove dodajemo low-level funkcijama. Osnovne takve funkcije su points() za točke, abline() za pravac, lines() za bilo kakvu liniju, legend() za legendu itd.

```
#Dodavanje drugih elemenata na graf
plot(x, y, main = "Naziv grafa", xlab = "Naziv x-osi", ylab = "Naziv y-osi"
    , xlim = c(0, 22), ylim = c(0, 10), type = "o")
points(3, 3)
abline(2, 1)
lines(c(1, 2, 3), c(5, 5, 4), col = "red")

plot(x, y, main = "Naziv grafa", xlab = "Naziv x-osi", ylab = "Naziv y-osi"
    , xlim = c(0, 22), ylim = c(0, 10), type = "o", pch = 22, col = "red",
    bg = "yellow", cex = 2)
abline(2, 1, col="blue")
legend("topleft", legend = c("Podaci", "pravac"), col = c("red", "blue"),
    lty = c(1, 1), lwd = c(2, 2))  #lty=1 - solid line, lwd-debljina linije

x <- seq(-3, 3, length = 100)
plot(x, x^2, type = "l", xaxt = "n", yaxt = "n", bty = "n", lty = "dashed",
     col = "purple", lwd = 2, ylim = c(0, 3), ylab = "")
lines(x, abs(x), type = "l", xaxt = "n", yaxt = "n", bty = "n", lty = "
    dotted", col = "red", lwd = 2)
axis(1, pos = c(0, 0))
axis(2, pos = c(0, 0), las = 2)
arrows(-2.5, 1, -2, 2, lwd = 2, length = 0.1)
text(-2.5, 0.8, labels = expression(f[1] (x) == group("|",x,"|")))
arrows(2, 1, 1.5, 1.5^2, lwd = 2, length = 0.1)
text(2, 0.8, labels = expression(f[2] (x) == x^2))
```

Funkcije zadane izrazima možemo jednostavno crtati funkcijom curve(). Funkcija curve() za razliku od plot može primiti izraz kao argument, pa tako i crtati definirane funkcije (default oznaka za argument je x).

```
#Crtanje funkcija
curve(x^2, from = -2, to = 2)
curve(x^2, -2, 2)
curve(t^2, -2, 2, xname = "t")

plot(-5:5, (-5:5)^2)
curve(x^2, -5, 5, add = TRUE)
```

6 Dodatne napomene vezane uz RStudio

U RStudiu moguće je vrlo jednostavno generirati izvještaje iz postojeće R skripte. Takav izvještaj u sebi sadrži pripadni R kod i output koji se pri tome generira. Izvještaje je moguće generirati kao Word, pdf ili html dokument. Nakon što u R skripti napravimo analizu, iz izbornika odaberemo File → Compile Report zatim odaberemo tip dokumenta i kliknemo Compile. U R skriptama iz kojih generiramo izvještaj treba izbaciti naredbe install.packages() koje mogu proizvesti probleme.

Zatvaranjem skripte, RStudio će automatski na lokaciji skripte snimiti i Rhistory datoteku koja sadrži povijest korištenih naredbi. Ovu mogućnost moguće je isključiti na Tools → Global Options → Always save history.

Zadatak 1. *Sa web stranice kolegija preuzmite datoteku Baze.zip te pronađite bazu podataka djelatnici.xls.*

(a) Učitajte podatke u R.

(b) Odaberite podatke iz varijable placa_prije posebno za muškarce i žene.

(c) Odaberite podatke iz varijable placa_prije za muškarce sa srednjom stručnom spremom.

(d) Odaberite podatke iz varijable placa_prije za muškarce sa srednjom ili visokom stručnom spremom.

(e) Odaberite podatke iz varijabli placa_prije i placa_poslije za odjel IS.

(f) Nacrtajte graf parova podataka iz varijabli placa_prije i placa_poslije. Uredite graf.

(g) Nacrtajte prethodni graf tako da podatke za muškarce prikažete plavim točkama, a za žene crvenim točkama. Dodajte legendu na graf.

```
#(a) učitati poadtke putem izbornika
str(djelatnici)
attach(djelatnici)
#(b)
placa_prije[spol == "Z"]
placa_prije[spol == "M"]
#(c)
placa_prije[spol == "M" & obrazovanje == "SSS"]
#(d)
placa_prije[spol == "M" & (obrazovanje == "SSS" | obrazovanje == "VSS")]
#(e)
djelatnici[odjel == "IS", 7:8]
djelatnici[odjel == "IS", c("placa_prije", "placa_poslije")]
#(f)
plot(placa_prije, placa_poslije)
```

```
plot(placa_prije, placa_poslije, xlab = "Plaća prije", ylab = "Plaća
    poslije", pch = 16)
#(g)
plot(placa_prije[spol == "M"], placa_poslije[spol == "M"], pch = 16, col="
    blue")
points(placa_prije[spol == "Z"], placa_poslije[spol == "Z"], pch = 16, col=
    "red")
legend("topleft", legend = c("muškarci", "žene"), col = c("blue", "red"),
    pch = c(16, 16))
```