

## Osnovne naredbe u R-u

**Napomena:** R radi sa skupovima podataka. Osnovni skup podataka je vektor, tj. niz elemenata. Varijablu čija je vrijednost broj je vektor dužine 1. Neka je x vektor sa više elemenata, i izračunajmo funkciju sin(x). U matematici je funkcija sin definirana na skupu realnih brojeva i kao rezultat daje realni broj. Rezultat u R-u će biti vektor koji ima isti broj elemenata kao i x, ali u kojem će se na odgovarajućim pozicijama nalaziti vrijednosti funkcije sin. U ovom pregledu navedene su funkcije sa uobičajenim parametrima, kako se najčešće upotrebljavaju. Zato, kod prve upotrebe neke funkcije korisno je pogledati pomoć (help). Tamo se nalaze opisi svih funkcija i objekata.

### Razlikuje velika i mala slova.

Nazivi koje korisnik definira **nesmiju** počinjati sa brojkom.

; (**to ka - zarez**) - odvajanje naredbi u istom redu

# - označava komentar (do kraja reda)

{ } - grupiranje izraza u blokove

+ - ako naredba ne stane u jedan red, R prikazuje ovaj znak. Može se promijeniti !

**MS Windows:** kod decimalnog broja koristi se točka za odvajanje decimalnih mjesta, bez obzira na "regional settings". Postoji varijabla LC\_NUMERIC koja definira decimalnu točku, ali se ne preporučuje upotreba, osim privremeno kada se rade grafički prikazi.

<- - operator pridruživanja, kod novijih verzija R-a može se koristiti znak jednakosti (=).

**q()** - završava rad s R-om.

**source("ime\_datoteke")** - izvršava naredbe iz datoteke.

**sink("ime\_datoteke")** - rezultati se spremaju u datoteku.

**sink()** - preusmjerava izlazne rezultate na ekran.

**objects()** , **ls()** - ispisuju sve objekte u zadanom okruženju. Ako ih izvršimo bez argumenata u radnom prostoru, ispisuju sve objekte u radnom prostoru. Ako ih pozovemo bez argumenata unutar funkcije, ispisuju sve lokalne objekte te funkcije.

**rm(O1,O2,...)** - uklanja objekte O1,O2,... iz memorije.

**rm(list=ls(all=TRUE))** - uklanja sve objekte iz memorije.

**dir()** - prikazuje sve datoteku u radnom direktoriju.

**getwd()** - ispisuje putanju radnog direktorija

**setwd("path\_dir\_name")** - mijenja putanju radnog direktorija u "path\_dir\_name"

**search()** - daje listu "priključenih" paketa, R - objekata ili spremnika podataka.

**searchpaths()** - daje listu svih putanja (paths) koje se momentalno vide u radnoj okolini.

**attach(what,pos,name= deparse(substitute(what)))**

- priključuje **what** na pretražnu putanju. **what** može biti spremnik podataka, lista ili R datoteka kreirana sa "save".

**detach(name, pos, version)** - odstranjuje objekt iz pretražne putanje. **pos** je pozicija u pretražnoj putanji, dok je **version** - verzija učitano paketa. ako nije navedena uzima se predefinicirana.

**library(package, help, pos = 2, lib.loc = NULL, character.only = FALSE, logical.return = FALSE, warn.conflicts = TRUE, keep.source = getOption("keep.source.pkgs"), verbose = getOption("verbose"), version)**

- učitava dodatni paket.

**require(package, quietly = FALSE, warn.conflicts = TRUE, keep.source = getOption("keep.source.pkgs"), character.only = FALSE, version, save = TRUE)**

- učitava dodatne pakete.

**getOption(x)** - (ili **options(...)**, **.Options**) - mijenja ili daje vrijednost od **x**-a. Služi za definiranje radnih parametara u radnom okruženju. Vidi help(getOption) za listu sistemskih parametara koje se mogu mijenjati ovom funkcijom. **Posebno važna funkcija !** Treba posebno proučiti help. Mogu se promijeniti mnogi parametri koji utječu na rad R-a. Od oblika "prompt"-a do ponašanja R-a prilikom izvršavanja programa.

Lista nekih predefiniciranih vrijednosti:

'prompt'	''> ''
'continue'	''+ ''
'width'	'80'
'digits'	'7'
'expressions'	'5000'
'keep.source'	'interactive()'
'show.signif.stars'	'TRUE'
'show.coef.Pvalues'	'TRUE'
'na.action'	'na.omit'
'timeout'	'60'
'ts.eps'	'1e-5'
'error'	'NULL'
'show.error.messages'	'TRUE'
'warn'	'0'
'warning.length'	'1000'
'echo'	'TRUE'
'verbose'	'FALSE'
'scipen'	'0'
'locatorBell'	'TRUE'

Primjer: options(prompt = ":"), options(digit = 10),...

R: brza kuharica, ver. 1.1

Postoji tekst datoteka RHOME/etc/**rprofile** koja se izvršava prilikom pokretanja R-a. Ako želimo zadržati promijenjene neke vrijednosti ovih parametara, onda ih zapišemo u obliku: **options(name = value)**.

**Sys.getlocale()**, **Sys.setlocale()** - prikazuje i mijenja neke parametre lokalnog okruženja.

**print(x,...)** - prikazuje objekt, generička funkcija, ima različito značenje za različite objekte.

**format(x,...)** -formatira R objekt za bolji ispis.

**open()**, **close()** - obično se upotrebljavaju u paru, prilikom uspostavanja veza (connections). Npr, otvaranje datoteke za pisanje ili čitanje, veza između procesa (pipe) ili internet veza. Za specifične veze vidjeti naredbe: **isOpen()**, **fifo()**, **pipe()**, **file()**, **socketConnection**, ....

## Pretraživanje pomo i

**help(ime\_naredbe)**

**help(ime\_naredbe,package="ime\_paketa")**

?**ime\_naredbe** - pomoć za **ime\_naredbe**

**help.start()**

- pokreće pomoć u HTML formatu.

**help.search("traženi\_pojam")**

- pretražuje pomoć i izlistava sve pozicije gdje se nalazi traženi pojam.

**example(ime\_naredbe)** - daje primjer upotrebe za traženu naredbu.

**apropos("topic")** - daje listu svih objekata na pretražnoj putanji koji u nazivu sadrže zadani niz znakova.

U prozoru gdje je prikazana pomoć možete se kretati osim sa mišem i tipkom **<razmaknica>** - (space) a za izlaz možete također koristiti tipku **<Q>**.

## Upotreba varijabli i objekata

**c(a,b,c,...)**

- kreira vektor sa komponentama a,b,c,... .

- **x <- c(a,b,c,...)**, pridružuje niz brojeva a,b,c,... vektoru.

**assign("var\_name",c(a,b,c,...))**

- assign("x",c(a,b,c,...)) je isto kao: **x <- c(a,b,c,...)**

**mode(x)** - mijenja i očitava tip objekta.

**typeof(x)** - čita internu R-ovu definiciju objekta.

Pr.: **x <- 3**, funkcija **mode(3)** daje tip "numeric" dok funkcija **typeof(3)** daje "integer".

**dim(x)** - mijenja ili ispisuje dimenziju objekta.

**dimnames(x)** - mijenja ili ispisuje nazive "dimenzija", tj. komponenta objekata.

**length(object)** - daje broj elemenata vektora ili broj objekata sadržanih u gledanom objektu.

**attributes(object)** - daje listu svojstava objekta.

**attr(object,name)** - mijenja i daje svojstvo **name** objekta **object**.

**class(object)** - daje ime klase kojoj objekt pripada. Ovo svojstvo je važno, jer ga koriste mnoge R-ove funkcije i njihovo ponašanje zavisi o klasi kojoj objekt pripada. Ako želimo, da se ne vidi kojoj klasi objekt pripada koristimo **unclass(object)**.

**methods(object)** - ispisuje sve metode objekta, tj. funkcije koje objekt ima definirane. Posebno korisna funkcija, jer možete odmah saznati sve mogućnosti nekog objekta.

**factor(x)** - kreira objekt tipa factor, tj. grupira komponente vektora iste duljine.

**levels(x)** - daje nivoe faktora.

**apply(vector1,factor1,func)** - primjenjuje funkciju **func** na svaku grupu komponenata vektora **vector1**, a nivou komponenata su definirani faktorom **factor1** nekog drugog vektora iste dužine. Postoje slične funkcije **apply()**, **lapply()**, **by()**.

**rep(x,times=n)** - vraća objekt koji sadrži n-kopija **x**-a. Objekti su vektori i liste.

**rep(x, times, length.out, each, ...)** - općenita sintaksa (vidi: help(rep)).

**str(x)** - prikazuje internu strukturu R objekta.

**ls.str()** - kombinacija funkcije **ls()** i **str()**, tj. prikazuje internu strukturu objekata na pretražnom putu. Za moguće parametre pogledajte help.

**save(file = "f-name",...)** - sprema specificirani objekt (...) u XDR OS nezavisnom binarnom formatu.

**load()** - učitava objekt zapisan sa **save()**.

**save.image(filename = "f-name")** - sprema sve objekte u radnom prostoru.

**paste(..., sep = " ", collapse = NULL)** - kreira znakovni niz (string) od navedenih komponenata, odvojenih znakom **sep** i **collapse**. Numeričke vrijednosti pretvara u znakove. Posebno, treba obratiti pažnju kako radi sa vektorima.

**substr(x,start,stop)** - daje znakovni podniz iz niza **x** od pozicije **start** do pozicije **stop**.

**strsplit(x,split)** - dijeli znakovni vektor **x** prema podnizu **split**.

**grep(pattern,x)** - traži izraz definiran s **pattern** u **x**, pogledaj ?regex.

**gsub(pattern,replacement,x)** - pretražuje znakovni vektor i/ili tekst te vrši zamjenu svih nađenih uzoraka, za razliku od **sub()** koji to radi samo na prvo nađenom uzorku.

**tolower(x)** - prebacuje sve znakove u mala slova.

**toupper(x)** - prebacuje sve znakove u velika slova.

**match(x,table)** - daje vektor pozicija prvih uparivanja vektora **x** s **table**.

**x %in% table** - slično kao **match** samo daje vektor logičkih vrijednosti.

Funkcije koje rade s stringovima:

**pmatch(x,table)** - parcijalno pretraživanje znakovnog niza, gdje je **x** vrijednost koja se traži, dok je **table** skup vrijednosti koji se pretražuju.

**nchar(x)** - broj znakova u znakovnom nizu.

**sprintf** - omotač za C funkciju **sprintf**. Funkcija vrši formatiranje raznih izraza.

**split(x,f)** - daje listu vektora iz **x**, koji nastaju cijepanjem vektora **x** prema faktoru **f**.

**unsplit()** - suprotna funkcija od **split**.

**cut(x,...)** - konvertira numerički vektor u faktor.

**which.max(x)** - daje indeks najvećeg elementa od **x**.

**which.min(x)** - daje indeks najmanjeg elementa od **x**.

**rev(x)** - daje vektor s obrnutim poretkom elemenata.

**which(x == a, arr.ind = FALSE)** - daje vektor indeksa za koji je operacija uspoređivanja TRUE. Prvi argument mora biti logička varijabla ili vrijednost. Ako je **arr.ind = TRUE** onda nam daje polje indeksa, što je naročito korisno ako pretražujemo matricu.

**na.omit(x,...)** - prekida kontrolu objekta kada naiđe na NA vrijednost (kod matrice i spremnika podataka vraća indeks prvog reda gdje je naišao na NA vrijednost).

**na.fail(x)** - ispisuje poruku greške ako objekt ima najmanje jednu NA vrijednost. Pogledati help za ostale funkcije koje upravljaju sa NA vrijednostima.

**unique(x)** - ako je **x** vektor ili spremnik podataka, daje nam sličan objekt kao **x** ali bez ponavljajućih elemenata.

### Konverzije varijabli

**as.array(x)**, **as.data.frame(x)**, **as.numeric(x)**,

**as.logical(x)**, **as.complex(x)**, **as.character(x)** - tipovi konverzije. Za ispis svih tipova koristite: **methods(as)**.

Za kreiranje određenog tipa varijabli možemo koristiti **numeric()**, **double()**, **character()**, ...

### Ispitivanje tipova varijabli

**is.na(x)**, **is.null(x)**, **is.array(x)**, **is.data.frame(x)**,

**is.numeric(x)**, **is.complex(x)**, **is.character(x)** - testovi tipova. Za ispis svih funkcija koristite: **methods(is)**.

### Generiranje nizova brojeva

: - dvotočka, označava niz brojeva. Pr.: 10:20 označava vektor čiji su elementi cijeli brojevi od 10 do 20. Prilikom izvršavanja naredbi ovaj operator ima **najve** i prioritet. **seq()** - generira niz brojeva. Sintaksa ima više oblika: **seq(-5, 5, by=.2)** - uobičajeni oblik: **from,to,by**. Ako je raspored promijenjen, onda treba navesti parametre s imenima, npr.: **seq(length=51, from=-5, by=.2)**.

### Logi ki operatori i vrijednosti

Vrijednosti logičkih izraza: **TRUE**, **FALSE**, **NA** (not available - nije definirano). Paziti, **velika slova** !  
Logički operatori:

<, <=, >, >=, ==

! - negacija

& - presjek ( logičko i - AND)

&& - presjek (logičko i - AND). Uspoređuje samo prvi element vektora, odnosno ide po vektoru do elementa gdje s može izvršiti operacija. (Možemo imati NA vrijednost).

| - unija (logičko ili - OR)

|| - unija (logičko ili - OR). Ponašanje je isto kao i kod &&. **is.na(x)** - vraća vektor iste dužine kao i **x** sa vrijednostima TRUE ako element nije definiran, inače vraća FALSE.

**is.nan(x)** - radi isto kao **is.na(x)**, samo gleda markere za NaN (Not a Number).

### Matemati ke funkcije

Operacije na vektorima izvode se element po element. Ako vektori nisu iste dužine primjenjuje se pravilo cikličkog nadopunjavanja (recycling).

**\***, **/**, **+**, **-**, **^** - uobičajeni operatori množenja, dijeljenja, zbrajanja, oduzimanja i potenciranja.

**eval()** - računa zadani izraz. Za detalje pogledati help.

**round(x, n)** - zokružuje **x** na **n** decimala.

**log** - logaritam baze e (prirodni log.).

**log10** - logaritam baze 10.

**log2** - logaritam baze 2.

**logb(x, base = broj)** - općenita funkcija logaritma

**exp** - eksponencijalna funkcija baze e.

**sin,cos,tan** - trigonometrijske funkcije

**sqrt** - kvadratni korijen

**max(x)**, **min(x)** - maksimalna i minimalna vrijednost vektora.

**pmax(x)** , **pmin(x)** - "paralelni" maksimum i minimum. Služi za uspoređivanje više vektora istih dužina. Vraća vektor iste dužine, čiji elementi su maksimumi odnosno minimumi odgovarajućih elemenata zadanih vektora.

**range(x)** - = **c(min(x), max(x))**.

**length(x)** - broj elemenata vektora.

**sum(x)** - suma elemenata vektora.

**diff(x,lag = 1, differences = 1,...)** - daje niz računajući razlike između svih elemenata međusobno udaljenih za **lag**.

**prod(x)** - produkt elemenata vektora.

**mean(x)** - srednja vrijednost elemenata vektora.

**median(x)** - medijan elemenata od **x**.

**quantile(x,probs=)** - uzorak kvantila koji odgovara zadanim vjerojatnostima (predefinicirana vrijednost je 0,0,25,0,5,0,75,1). **probs** je numerički vektor vjerojatnosti čije vrijednosti moraju biti u intervalu [0,1].

**weighted.mean(x, w)** - računa srednju vrijednost elemenata od **x**-a, gdje je **w** težinski vektor iste dužine kao i **x**.

**rank(x)** - računa rang elemenata. **x** je numerički, kompleksni ili znakovni vektor.

**var(x)** - varijanca elemenata vektora.

**sd(x)** - standardna devijacija od **x**.

**var(x, y)** - kovarijanca između **x** i **y**. Ako su **x** i **y** matrice ili spremnici podataka onda nam daje kovarijancu između odgovarajućih stupaca.

**cov(x, y)** - isto kao i predhodno.

**cor(x)** - korelaciona matrica od **x**, ako je **x** matrica ili spremnik podataka (1 iako je **x** vektor).

**cor(x, y)** - linearna korelacija između **x** i **y**, ili korelaciona matrica ako su **x** i **y** matrice ili spremnici podataka.

**sort(x)** - vraća vektor sortiran po uzlaznom poretku.

**order(x)** - sortira sam vektor po zadanim parametrima (vidi: help(order)).

**sort.list** - sortira vektor (vidi: help(sort.list)).

**choose(n,k)** - broj kombinacija **k**-tog reda od **n** elemenata (=  $n!/(n-k)!k!$ ).

**Re(x)** - realni dio kompleksnog broja.

**Im(x)** - imaginarni dio kompleksnog broja

**Mod(x)** - modul kompleksnog broja

**Arg(x)** - kut u radijanima kompleksnog broja.

**Conj(x)** - konjugirani broj kompleksnog broja.

**convolve(x,y)** - računa više vrsta konvolucija dva niza.

**fft(x)** - Fourier-ova transformacije (Fast Fourier Transform - FFT) polja.

**mvfft(x)** - FFT za svaki stupac matrice.

**filter(x,filter)** - primjenjuje linearni filter na jednu vremensku seriju ili na svaku seriju posebno ako imamo višestruke serije.

Mnoge matematičke funkcije imaju logički parametar **na.rm = FALSE** za definiranje ponašanja u slučaju da se pojavi nedostajajuća vrijednost (NA). Ako je **na.rm = TRUE** onda se nedostajajuće vrijednosti uklanjaju tj. ne učestvuju u računskim operacijama.

## Polja i matrice

**array(data\_vector,dim\_vector)** - kreira polje.

**is.array(X)** - provjerava da li je **X** polje. Vraća TRUE ili FALSE.

Aritmetičke operacije na poljima vrše se element po element. Pr.:  $D \leftarrow 2*A*B + C + 1$ , znači  $D[i,j] = 2*A[i,j]*B[i,j]+C[i,j]+1$ , za svako  $i,j$ , uz uvjet da su polja iste dimenzije i da se na njima mogu vršiti aritmetičke operacije.

**X%o%Y** - vanjski produkt polja **X** i **Y**. Množi se svaki element od **X** sa svakim elementom od **Y**. Rezultat je polje dimenzije **c(dim(X), dim(Y))**.

**outer(X,Y,f,par)** - općenita definicija vanjskog produkta, gdje je **f** funkcija koja se primjenjuje na sve parove elemenata iz **X** i **Y**, dok je **par** dodatni parametar za funkciju **f**.

**aperm(X,perm)** - vrši permutaciju elemenata polja, gdje je **perm** neka permutacija indeksa polja.

**matrix()** - kreira matricu.

**X%\*%Y** - množenje matrica.

**t(X)** - vrši operaciju transponiranja matrice **X**.

**nrow(X)** - daje broj redova matrice **X**.

**ncol(X)** - daje broj stupaca matrice **X**.

**crossprod(X,Y)** - isto što i **t(X)%\*%Y**, gdje su **X** i **Y** matrice.

**diag(x)** - gdje je **x** vektor, kreira dijagonalnu matricu gdje su elementi od **x**-a elementi glavne dijagonale.

**diag(X)** - gdje je **X** matrica daje vektor čiji su elementi elementi glavne dijagonale matrice.

**diag(x = a, nrow = b, ncol = c)** - kreira matricu s **b** redaka i **c** stupaca čiji elementi na glavnoj dijagonali imaju vrijednost **a**. Umjesto **a** možemo staviti **c(a,b,c,...)**, gdje će **a,b,c,...** biti vrijednosti elemenata na glavnoj dijagonali matrice.

**solve(A,b)** - rješava sistem linearnih jednadžbi, gdje je **A** matrica sistema, dok je **b** vektor slobodnih članova.

**solve(A)** - inverzna matrica od **A**.

**eigen(A)** - računa svojstvene vrijednosti i svojstvene vektore matrice **A**.

**svd(A)** - računa singularnu dekompoziciju matrice **A**.

**qr()** - računa **QR** dekompoziciju matrice.

**det(X)** - računa determinantu kvadratne matrice.

**cbind(arg1,arg2,arg3,...)** - kreira matricu iz veličina **arg1, arg2, arg3**, koji mogu biti ili vektori ili matrice. Ako su vektori nejednakih dužina, tada se primjenjuje pravilo recikliranja.

**rbind()** - isto kao i **cbind()** samo argumente tretira kao retke.

**as.vector(X)** - pretvara matricu u vektor uzimajući kolonu po kolonu ("**column major order**").

**as.matrix(x)** - postoji ali nema smisla. Dovoljno je vektoru promijeniti dimenziju:  $\dim(x) \leftarrow c(n\_col, n\_row)$ .

**is.matrix()** - da li je matrica, vraća TRUE ili FALSE.

**table()** - računa tablice frekvencija iz faktora jednake dužine.

**subset(x,...)** - daje potskup podataka prema zadanim kriterijima. **x** može biti vektor, matrica ili spremnik podataka.

**sample(x, size, replace = FALSE, prob = NULL)** - daje uzorak veličine **size** iz skupa **x**. Uzorci mogu biti sa ili bez ponavljanja. **prob** je težinski vektor dobiven ako je uzorak iz vektora.

## Liste i spremnici podataka

**list(...)** - kreira listu objekata različitih tipova ili vrsta. Vidi help(list) za ostale mogućnosti sintakse (**pairlist(...)**, **as.list(x,...)**, **as.pairlist(...)**, **as.list.environment(x,...)**, **is.list(x)**, **is.pairlist(x)**)

**data.frame(..., row.names = NULL, check.rows = FALSE, check.names = TRUE)**

- kreira spremnik podataka.

**as.data.frame()** - pretvara listu u spremnik podataka.

**aggregate(x,by,FUN)** - dijeli spremnik podataka **x** u podgrupe, te računa statistiku za svaku grupu i vraća rezultat u pogodnoj formi. **by** je lista elemenata po kojoj se vrši grupiranje ,dok je **FUN** skalarna funkcija koja računa sumarnu statistiku i može se primjeniti na sve potskupove.

**stack(x, ...)** - transformira podatke grupirane po stupcima u spremnik podataka ili listu u jedan stupac.

**unstack(x, ...)** - obratno od **stack()**.

**reshape(x, ...)** - vrši formatiranje podataka u spremniku podataka.

**Učitavanje - spremanje podataka**

```
read.table(file, header = FALSE, sep = "",
  quote = "\"\"'", dec = ".", row.names,
  col.names, as.is = FALSE, na.strings = "NA",
  colClasses = NA, nrow = -1, skip = 0,
  check.names = TRUE, fill = !blank.lines.skip,
  strip.white = FALSE, blank.lines.skip = TRUE,
  comment.char = "#")
```

- učitava podatke iz tekst datoteke. Rezultat je spremnik podataka. Opis pogledati u help-u.

```
read.csv("filename", header=TRUE) - isto kao i
read.table, podaci separirani zarezom (comma-delimited).
read.delim("filename", header=TRUE) - isto kao
read.table, podaci su separirani sa tabulatorom - ASCII 9
(tab-delimited)
```

```
read.fwf(file, widths, header=FALSE, sep="",
  as.is=FALSE)
```

- podaci su zadane dužine (fixed width formatted) i učitavaju se u spremnik podataka, **widths** je vektor cijelih brojeva koji definiraju duljinu kolona. Detaljni opis je u help-u.

```
scan(file = "", what = double(0), nmax = -1, n = -1,
  sep = "",
  quote = if(identical(sep, "\n")) "" else "\"\"'",
  dec = ".", skip = 0, nlines = 0,
  na.strings = "NA", flush = FALSE, fill = FALSE,
  strip.white=FALSE, quiet=FALSE,
  blank.lines.skip = TRUE, multi.line = TRUE,
  comment.char = "", allowEscapes = TRUE)
```

- učitava podatke iz tekst datoteke. Rezultat je vektor ili lista podataka. Opis pogledati u help-u.

**file()** - služi za rad s vanjskim datotekama. Sve mogućnosti mogu se vidjeti sa **methods(file)**. Pr. 1.:

**file.edit(file.choose())** - prikazuje sve datoteke u tekućem direktoriju, dozvoljava odabir, te sadržaj prikaže u editoru gdje se može mijenjati.

Pr. 2.:

**f <- file("test.txt", "w")** - kreira datoteku i otvori ju za pisanje, pridruži datoteci varijabu **f** (file handle) preko koje komuniciramo sa datotekom.

**close(f)** - kada smo gotovi sa pisanjem moramo datoteku zatvoriti.

**cat("tekst 1", "tekst 2", file=f, sep="\n")** - zapisuje u datoteku definiranu sa **file = f**.

**readLines()** - čita podatke iz datoteke.

**writelines()** - piše podatke u datoteku.

```
write.table(x, file="filename", row.names=TRUE,
  col.names=TRUE, sep=" ")
```

- zapisuje **x** nakon konverzije u spremnik podataka. Ako je **quote = TRUE** znakovi ili stupci se stavljaju u navodnike, **sep** je separator podataka, **eol** je separator linija, **na** je znakovni niz koji označava nedostajajuće vrijednosti. Upotrebite **col.names=NA** da bi se dodao prazan naziv stupca radi lijepšeg ispisa.

**data()** - prikazuje listu svih skupova podataka koji dolaze s R-om. Podaci služe za učenje i testiranje R-ovih funkcija.

**data(ime\_skupa)** - Učitava podatke koji su u skupu **ime\_skupa**. Ovo je stari način (prije ver. 2.0). Novi način je da se direktno napiše **ime\_skupa**. Sada se neki učitavaju na stari a neki na novi način.

**data(ime\_skupa, package = "ime\_paketa")** - učitavanje podataka **ime\_skupa** iz paketa **ime\_paketa**. Obratite pažnju da je **ime\_paketa** u navodnicima.

**data.entry(data)** - otvara prozor sa tabelarnim prikazom, gdje možemo vršiti izmjene.

**edit(data)** - pregled i izmjena podataka. Prema obliku zapisa podataka, podaci će se prikazati u obliku tabele (slično Excelu) ili kao tekst editor. S time da se u datoteci RHOME/etc/**rprofile** može definirati vanjski editor.

**Napomena:** Mnoge input/output (I/O) funkcije imaju **file** kao argument. Obično je to znakovni niz koji zadovoljava konvenciju OS-a ili konekcije. **file= ""** znači standardni I/O, obično prikaz na ekranu. Konekcija može uključivati datoteke, pipe, kompresirane datoteke ili R varijable. Na MS Windows OS-u konekcija može biti upotrebljena sa parametrom **description = "clipboard"**. Za čitanje tabele iz **Excel-a** možete upotrebiti:

```
x <- read.delim("clipboard")
```

Za prebacivanje tabele u **Excel** možete koristiti

```
write.table(x, "clipboard", sep="\t", col.names = NA)
```

(Nemojte zaboraviti da Excel komunicira sa clipboard-om putem naredbi copy (CTRL + C) i paste (CTRL + V))

Za interakciju sa raznim bazama podataka možete koristiti pakete: RODBC, DBI, RMySQL, RPostgreSQL i ROracle. Za čitanje raznih drugih formata datoteka možete pogledati XML, hdf5, netCDF pakete.

**Rad s indeksima****Indeksiranje vektora:**

```
x[n] - n-ti element
x[-n] - svi samo ne n-ti element
x[1:n] - prvih n elemenata
x[-(1:n)] - elementi od n+1 do kraja
x[c(1,7,5)] - određeni element
x["name"] - element naziva "name"
x[x > 3] - svi elementi veći od 3
```

```
x[x > 3 & x < 5] - svi elementi između 3 i 5
x[x %in% c("a", "bla", "ccc")] - element u zadanom skupu.
```

**Indeksiranje lista:**

```
x[n] - lista sa n elemenata
x[[n]] - n-ti element liste
x[["name"]] - element liste s imenom "name"
x$name - isto kao i predhodno.
```

**Indeksiranje matrica:**

```
x[i,j] - element u redu i, stupcu j
x[i,] - i-ti red
x[,j] - j-ti stupac
x[,c(2,5)] - stupci 2 i 5
x[["name",]] - redak s imenom "name"
```

**Indeksiranje spremnika podataka:**

Isto kao indeksiranje matrice plus slijedeće:

```
x[["name"]] - redak s imenom "name"
x$name - isto kao predhodno.
```

**Rad sa grafi kim okruženjem**

Treba voditi računa da pojam "grafičko okruženje" nije nužno samo dio ekrana (obično ga zovemo prozor - window) gdje se prikazuje crtež. To može biti neki drugi uređaj - program (device driver).

U R-u postoje dva skupa grafičkih naredbi: Naredbe za grafiku visoke razine i za grafiku niske razine. Naredbe za grafiku visoke razine uvijek kreiraju novi crtež - graf i te naredbe mogu biti vrlo sofisticirane u smislu da analiziraju podatke i grafički prikaz prilagode zadanim podacima. Naredbe niske razine se koriste da se postojeći grafički prikaz prilagodi potrebama i željama korisnika.

Postoji niz funkcija za kopiranje, spremanje, tiskanje grafičkih prikaza: **dev.copy()**, **dev.print()**, **dev.control()**, **dev.off()**, **dev.open()**, **png()**, **jpg()**, **postscript()**, ...

**split.screen(figsize, screen, erase = TRUE)** - dijeli grafičko područje u više dijelova.

**figs** - vektor sa dvije komponente, prva definira broj redova, a druga broj stupaca. Pr.: **c(2,1)** definira dva reda i jedan stupac, tj. daje nam dva područja za crtanje, jedno ispod drugoga. Ako stavimo **c(2,2)** imamo dva stupca i dva retka, tj. četiri područja za crtanje. Ako parametar **screen** nije definiran, onda se naredba odnosi na cijelo grafičko područje, a ako želimo podijeliti već podijeljeno područje, onda stavimo broj područja koje želimo dijeliti. Pr.: **screen(c(2,2),4)** - područje 4 želimo podijeliti na četiri područja.

**screen(n=xx)** - Grafičko područje **xx** postaje aktivno, tj. sve naredbe ise izvršavaju u tom području. Bez parametra vraća broj aktivnog grafičkog područja.

**erase.screen(n=xx)** - briše sadržaj grafičkog područja **xx**.

**close.screen(n=xx)** - zatvara grafičko područje **xx**. Naredba **close.screen(all = TRUE)** poništava naredbu **split.screen**, R izlazi iz moda za više grafičkih područja.

**layout()** - funkcija koja također dijeli grafičko područje u više dijelova. Nije sukladna sa funkcijom **split.screen** i ne možemo ih istovremeno upotrebljavati.

### Grafika visoke razine

**plot(x)** - crta točkasti graf (**i,x[i]**), gdje **i** ima vrijednosti od 1 do **length(x)**.

**plot(x, y)** - crta točkasti graf (**x[i],y[i]**) gdje **i** ima vrijednosti od 1 do **length(x)** (**x** i **y** moraju biti iste dužine)..

**hist(x)** - prikazuje histogram frekvencija od **x**-a.

**barplot(x)** - histogram vrijednosti od **x**.

**dotchart(x)** - ako je **x** spremnik podataka, crta "Cleveland" točkasti graf.

**pie(x)** - prikaz u krugu (pie-chart).

**boxplot(x)** - "box-and-whiskers" graf.

**sunflowerplot(x, y)** - prikazuje isto kao i **plot()**. Višestruke točke su prikazane drugačije radi boljeg prikaza.

**coplot(x~y | z)** - prikazuje **x** u zavisnosti od **y** za dane vrijednosti od **z**-a.

**interaction.plot (f1, f2, y)** - ako su **f1** i **f2** faktori, prikazuje srednju vrijednost od **y** (na y osi) s obzirom na vrijednosti **f1** (na x-osi) i **f2** (različite krivulje). Parametar **fun** dozvoljava da dobijemo sumarnu statistiku (predefinirana vrijednost je **fun=mean**).

**matplot(x,y)** - prikazuje graf zavisnosti između stupaca prve matrice sa drugom. Tj. prvog stupca od **x** i prvog stupca od **y**, zatim drugog stupca od **x** sa drugim stupcem od **y**, itd.

**assocplot(x)** - daje Cohen-Friendly graf koji pokazuje devijaciju iz nezavisnosti redaka i stupaca u dvodimenzionalnoj kontigencijskoj tabeli.

**mosaicplot(x)** daje "mozaik" graf reziduala iz log-linearne regresije kontingencijske tabele.

**pairs(x)** - ako je **x** matrica ili spremnik podataka prikazuje sve moguće grafove između stupaca od **x**.

**plot.ts(x)** - ako je **x** objekt klase "**ts**" (ts - time series - vremenski niz), prikazuje **x** u zavisnosti od vremena, **x**

može sadržavati više nizova podataka ali moraju imati istu frekvenciju i vrijeme.

**ts.plot(x)** - isto kao i predhodno ali ako **x** sadrži više vremenskih nizova oni mogu imati različita vremena ali moraju imati istu frekvenciju.

**qqnorm(x)** - daje kvantile od **x** s obzirom na očekivanu vrijednost normalne distribucije.

**qqplot(x, y)** - prikazuje zavisnost kvantila od **y** s obzirom na kvantile od **x**.

**contour(x, y, z)** - crta krivulju (podaci su interpolirani da se prikazuje neprekidna krivulja), **x** i **y** moraju biti vektori i **z** mora biti matrica tako da je **dim(z)=c(length(x),length(y))** (**x** i **y** mogu biti izostavljeni).

**filled.contour(x, y, z)** - isto kao predhodno samo su površine između krivulja obojane i nactana je legenda boja.

**image(x, y, z)** - isto kao i predhodno, sa bojama i podaci nisu interpolirani.

**persp(x, y, z)** - podaci su prikazani u perspektivi.

**stars(x)** - ako je **x** matrica ili spremnik podataka, prikazuje se graf sa segmentima ili zvjezdicama gdje je svaki redak prikazan zvjezdicom a stupci daju veličinu segmenta.

**symbols(x, y, ...)** - crta u točki (**x,y**) simbole (krug, kvadrat, pravokutnik, zvjezdicu,..) čija se veličina, boja,... mogu definirati s dodatnim parametrima.

**termplot(mod.obj)** - crta (parcijalne) efekte regresijskog modela (mod.obj).

Slijedeći parametri su zajednički većini grafičkih funkcija:

**add=FALSE** ako je TRUE dodaje crtež na već postojeći (ako postoji).

**axes=TRUE** ako je FALSE ne crta koordinatne osi i okvir crteža.

**type="p"** definira tip grafa, "p": točkasti, "l": linijski, "b": točke povezane linijama, "o": linije iznad točaka, "h": verikalne linije, "s": razmak, podaci su prikazani na vrhu vertikalnih linija, "S": podaci su prikazani na dnu vertikalnih linija.

**xlim=, ylim=** - definira donju i gornju granicu koord. osi.

**xlab=, ylab=** - nazivi na koord. osima.

**main=** - glavni naziv crteža.

**sub=** - podnaslov crteža prikazan sa manjim slovima.

### Grafika niske razine

**points(x, y)** - dodaje točke na postojeći graf (mogu se koristiti dodatni parametri).

**lines(x, y)** - dodaje linije na postojeći graf.

**text(x, y, labels, ...)** - dodaje tekst u točki (x,y). Tipična upotreba: **points(x, y, type="n"); text(x, y, "tekst")**

**mtext(text, side=3, line=0, ...)** - dodaje tekst na postojeći crtež. Tekst se dodaje na vanjski rub u zavisnosti od parametra side (1 - dolje, 2 - lijevo, 3 - gore, 4 - desno). Za detaljniji opis koristite **help(mtext)**.

**segments(x0, y0, x1, y1)** - crta liniju od točke (x0,y0) do točke (x1,y1).

**arrows(x0, y0, x1, y1, angle= 30, code=2)** - isto kao predhodno samo dodaje vrh strijelice u zavisnosti od parametra code. Ako je code = 2 vrh se nalazi u točki (x0,y0), ako je code = 1 vrh se nalazi u točki (x1,y1), odnosno ako je code=3 na oba kraja linije. Parametar angle je kut između linije i ruba strijelice.

**abline(a,b)** - crta liniju (pravac), gdje je a odrezak na osi y (intercept), dok je b koeficijent smjera pravca.

**abline(h=y)** - crta horizontalnu liniju na udaljenosti y od osi x.

**abline(v=x)** - crta vertikalnu liniju na udaljenosti x od osi y.

**abline(lm.obj)** - crta regresionu liniju zadanu sa **lm.obj**.

**rect(x1, y1, x2, y2)** - crta pravokutnik, gdje su (x1,y1) i (x2,y2) nasuprotni vrhovi.

**polygon(x, y)** - crta poligon povezujući toče definirane sa x i y.

**legend(x, y, legend)** dodaje tekst kod točke (x,y).

**title()** - dodaje naslov i podnaslov.

**axis(side, vect)** - dodaje koordinatne osi (vidi: **help(axis)**)

**rug(x)** - crta vrijednosti od x kao male vertikalne linije na osi x.

**locator(n, type="n", ...)** - čita poziciju pokazivača u grafičkom prozoru kada se pritisne lijeva tipka miša, gdje je n je broj mogućih točaka. Kada dostignemo broj mogućih točaka ili prekinemo operaciju ranije (pritiskom desne tipke miša ili pritisnemo tipku stop u izborniku) prikažu se koordinate (x,y) u vektorima x i y koje je korisnik odabrao. Parametrom type možemo nacrtati simbole prilikom odabira točaka: "p" crta točke, dok "l" spaja točke linijama. Predefinirana vrijednost je "n", tj. ne crta.

## Grafi ki parametri

Slijedeći parametri mogu se definirati globalno koristeći funkciju **par(...)**, a mnogi mogu biti zadani putem parametra u grafičkim funkcijama.

**adj** - definira poziciju teksta (0 - lijevo poravnan, 0.5 - centrirani, 1 - desno poravnan).

**bg** - definira boju pozadine (tj. : bg="red", bg="blue", . . . Lista svih 657 boja može se dobiti funkcijom colors()).

**bty** - definira tip okvira oko crteža, dozvoljene vrijednosti su: "o", "l", "7", "c", "u", "]" (okvir izgleda kao odgovarajući znak). Ako je **bty="n"** okvir neće biti prikazan.

**cex** - parametar koji definira veličinu teksta i simbola s obzirom na predefiniranu vrijednost. **cex.axis** - definira brojeve na koordinatnim osima, **cex.lab** - definira tekstove na crtežu, **cex.main** - glavni naslov, **cex.sub** - podnaslov.

**col** - definira boju simbola i linija. Upotrebite eng. nazive boja: "red", "blue". Funkcija **colors()** daje nam sve definirane boje na sistemu. Možete koristiti i RGB oblik "#RRGGBB". Također postoje funkcije **rgb()**, **hsv()**, **gray()**, i **rainbow()** za definiranje boja. Također, postoje parametri (slično kao i cex): **col.axis**, **col.lab**, **col.main**, **col.sub**.

**font** - broj koji definira stil teksta (1 - normalni, 2 - zakošeni (italic), 3 - pojačani (bold), 4 - zakošeni i pojačani (bold italics)). Analogno parametru cex, postoje parametri: **font.axis**, **font.lab**, **font.main**, **font.sub**.

**las** - broj koji definira smjer teksta koji opisuje koordinatne osi (0 - paralelno sa osima, 1 - horizontalno, 2 - okomito na osi, 3 - vertikalno).

**lty** - definira tip linije, može biti cijeli broj (1 - puna linija, 2 - crtkana linija, 3 - točkasta linija, 4 - crta-točka, 5 - dugačka crtkana linija, 6 - dvostruka crtkana linija), ili znakovni niz do osam znakova (između "0" i "9") koji definiraju dužinu, u točkama ili pikselima elemenata i razmaka između njih. Npr. lty="44" je isto kao i lty=2.

**lwd** - broj koji definira širinu linija, predefinirana vrijednost je 1.

**mar** - vektor od 4 broja koji definiraju razmak između koordinatnih osi i ruba crteža oblika **c(bottom, left, top, right)**, predefinirane vrijednosti su c(5.1, 4.1, 4.1, 2.1).

**mfc** - vektor oblika **c(nr,nc)** koji dijeli grafički prozor kao matricu od **nr** linija i **nc** stupaca. Crteži se prikazuju u stupcima.

**mfrow** - isto kao prethodno, samo se crteži prikazuju u redovima.

**pch** - definira tip simbola (ugrađenih u R) za točke na grafu. Detaljni opis može se dobiti sa help(points) i example(points). Može biti znak ili broj koji označava jedan od skupova simbola. Cijeli skup simbola može se dobiti s brojevima od 1 do 18. Postoje još i dodatni simoli: 19 - puni kružić, 20 - mali kružić, 21 - kružić, 22 - kvadratić, 23 - dijamant, 24,25 - trokutići. Brojevi od 26 do 32 se ne upotrebljavaju. Od 33 do 255 su jedno-bajtni lokalni znakovi. Ako je **pch** broj ili znak tipa "NA" ili je znak prazan tada se točke ne prikazuju. Veličinu simbola kontroliramo sa parametrom **cex** (koji ima predefiniranu vrijednost 1). **cex = 0.5** će smanjiti veličinu simbola za 50 % dok će **cex = 2** udvostručiti veličinu.

**ps** - broj koji definira veličinu točaka i teksta. Koristi se samo kod funkcija grafike niske razine.

**pty** - znak koji definira tip grafičkog područja: "s"-kvadratno, "m" - maksimalno.

**tck** - vrijednost koja definira dužinu oznaka razdiobe na koordinatnim osima. Ako je **tck=1** dobijemo mrežu. Upotrebljiva veličina se nalazi oko broja -0.01. Ako je broj negativan onda se ove oznake nalaze s druge strane koordinatnih osi.

**tcl** - isto kao i prethodno, samo je veličina definirana u omjeru prema tekstu (predefinirana vrijednost je **tcl=0.5**).

**xaxt** - ako je **xaxt="n"** tada je x os definirana ali nije nacrtana (obično se koristi zajedno sa axis(side=1, ...)).

**yaxt** - ako je **yaxt="n"** tada je y os definirana ali nije nacrtana (obično se koristi zajedno sa axis(side=2, ...)).

## Programiranje

**if (expr\_1) {expr\_2} else {expr\_3}** - uvjetno izvršavanje. Ako je **expr\_1** TRUE tada se izvršava **expr\_2** inače se izvršava **expr\_3**. Zgrade nisu potrebne ako imamo samo jednu naredbu, a ako ih imamo više treba ih odvojiti sa ; unutar zagrada.

Ako uvjet **expr\_1** nije definiran (NA) R javlja grešku.

**stopifnot(...)** - ako jedan od izraza u ... nije 'all' ili 'TRUE' izraz daje poruku greške i indikator koji izraz nije dobar.

**ifelse(cond, a, b)** - vektorski oblik. Ako je **cond** TRUE vraća vektor a, inače vraća vektor b.

**function( arglist ) {expr}** - definiranje funkcije

**return(value)** - prilikom izlaza iz funkcije vraća vrijednost **value**.

**for(var in seq) {expr}** - for petlja. **var** - kontrolna varijabla, **seq** - niz iz kojeg **var** uzima vrijednosti, **expr** - izraz koji se izvršava.

**while(cond) {expr}** - dok je **cond** TRUE izvršava se **expr**.

**repeat {expr}** - petlja u kojoj se izvršava **expr**, može se prekinuti samo sa **break**.

**break** - prekida izvođenje nekog bloka programskog koda.

**next** - u **for**, **while** i **repeat** prekida izvršavanje koda i vraća se na početak bloka.

**do.call(funname, args)** - izvršava funkciju **funname** i daje joj **args** kao parametre.

**browser()** - prekida izvođenje programa i pokreće kopiju R interpretera, pa nam omogućuje pregled radne okoline. Ovu funkciju koristimo da bismo našli greške u programu. S naredbom **ls()** možemo dobiti ispis svih lokalnih varijabli, a također možemo ispisati njihove vrijednosti. Prekid ove funkcije i nastavak izvršavanja programa vršimo pritiskom na tipku <c>, dok pritiskom na tipku <n> nastavljamo izvršavanje programa korak po korak (step-through-debugger).

**menu(choices, graphics = FALSE, title = "")** - interaktivni izbornik, gdje je **choices =**

**c("opt1", "opt2", "opt3")**. Vraća redni broj selekcije.

**switch(EXPR, ...)** - evaluira **EXPR** i izvršava kôd koji se nalazi u "...". Najjednostavniji primjer switch(ch,{kôd 1},{kôd 2},{kôd 3}). Prema vrijednosti varijable ch (1,2 ili 3) izvršava odgovarajući kôd.

## Statistika, optimalizacija, prilago avanje modela

### Raspodjele:

Raspodjela	R naziv	dodatni argumenti
beta	beta	shape1, shape2, ncp
binomna	binom	size, prob
Cauchyjeva	cauchy	location, scale
Hi-kvadratna	chisq	df, ncp
eksponencijalna	exp	rate
F	f	df1, df1, ncp
gama	gamma	shape, scale
geometrijska	geom	prob
hipergeometrijska	hyper	m, n, k
log-normalna	lnorm	meanlog, sdlog
logistička	logis	location, scale
negativna binomna	nbinom	size, prob
normalna	norm	mean, sd
Poissonova	pois	lambda
Studentova	t	df, ncp
uniformna	unif	min, max
Weibullova	weibull	shape, scale

**Pozivanje raspodje:****dxxx** - gustoća raspodjele**pxxx** - CDF - Common Data Form**qxxx** - funkcija kvantila**rxxx** - simulacija slučajnih odstupanja.gdje **xxx** znači jedno od gornjih R imena.

Pr.: dbeta, pbeta, qbeta, rbeta.

Za detalje pogledati help.

Za kreiranje nizova slučajnih brojeva možemo koristiti i funkciju **.Random.seed**.**summary()** - sumarni rezultati za različite statističke modele usklađivanja funkcija. (minimum, prvi kvantil, medijan, srednja vrij., treći kvantil, maksimum). Ova funkcija ima "generičko" značenje, tj. na različitim objektima daje drugačije prikaze.**ptukey()** - funkcija raspodjele studentiziranog raspona uzoraka iz normalne raspodjele.**qtukey()** - inverz od **ptukey** - kvantilna funkcija.**fivenum()** - pet Tukey-evih brojeva.**stem()** - daje nam tzv. "stem and leaf" grafikon.**lsfit(x, y, wt = NULL, intercept = TRUE, tolerance = 1e-07, yname = NULL)**- metoda najmanjih kvadrata, procjenjuje **b** u modelu **y = Xb + e**.**ls.diag()** - računa osnovnu statistiku, uključujući standardnu pogrešku, t- i p-vrijednosti za regresijske koeficijente.**aov(formula)** - analiza modela varijance.**anova(fit,...)** - analiza tablica varijance (ili devijance) za jedan ili više prilagođenih objekata..**density(x)** - procjenjuje gustoću prema zadanoj gustoći i širini pojasa.**binom.test()**, **pairwise.t.test()**, **power.t.test()**, **prop.test()**, **t.test()**, ... - upotrebite help.search("test") za opis ovih funkcija.**optim(par, fn, method = c("Nelder-Mead", "BFGS", "CG", "L-BFGS-B", "SANN"))**- općeniti model optimizacije, baziran na Nelder-Mead algoritmu. **par** - početne vrijednosti, **fn** je funkcija koja se optimizira.**nlm(f,p)** - minimizira funkciju **f** koristeći Newtonovu vrstu algoritma sa početnom vrijednošću **p**.**lm(formula)** - osnovna funkcija za usklađivanje običnih višestrukih linearnih modela.**glm(formula, family=family.generator, data=data.frame)** - generalizirana funkcija za usklađivanje linearnih modela. Za definiranje **family.generator** pogledajte pod help(family).**nls(formula)** - Daje procjenu nelinearne metode najmanjih kvadrata za model nelinearnih parametara.**approx(x,y)** - daje listu točaka linearno interpoliranih u dani skup podataka. **x** i **y** su vektori koji predstavljaju koordinate skupa podataka.**spline(x,y=)** - kubična spline interpolacija.**loess(formula)** - prilagođava neku neparametarsku regresiju s pomoću lokalno ponderirane regresije.

Mnoge funkcije za modeliranje zadane formulom imaju više zajedničkih parametara:

**data** - spremnik podataka, **subset** - podskup varijabli koje se koriste kod modela prilagođavanja, **na.action** - obrada nepostojećih vrijednosti: "na.fail", "na.omit" ili korisnička funkcija.

Sličjeće funkcije su generičke i često se primjenjuju na modele prilagođavanja:

**predict(fit,...)** - daje predviđene vrijednosti iz modela prilagođavanja ulaznih podataka.**df.residual(fit)** - daje preostali broj stupnjeva slobode prilagođenog modela.**coef(fit)** - čita procijenjene koeficijente modela (ponekad sa greškama).**residuals(fit)** - ekstrahira (matricu) ostataka, po potrebi ponderiranih.**deviance(fit)** - daje rezidualni zbroj kvadrata, po potrebi ponderiran.**fitted(fit)** - daje prilagođene vrijednosti.**AIC(fit)** - računa Akaike informacijski kriterij (AIC).