

Vježbe 1: Uvod u kvantitativnu ekolo- giju i programski jezik R

1.1 Uvod

Ove vježbe osmišljene su za studente koji prvi put koriste programski jezik R. Ne očekujemo nikakvo prethodno programersko iskustvo. Vježbe su organizirane kronološki – **svaka vježba gradi na prethodnoj**, pa je važno da ih radite redom.

Kako koristiti ove vježbe

1. **Pročitajte pažljivo objašnjenja** – ne preskačite tekst
2. **Pišite kod vlastoručno** – nemojte samo kopirati-zalijepiti
3. **Eksperimentirajte** – promijenite brojeve, isprobajte nove stvari
4. **Ako nešto ne radi** – pročitajte poruku o grešci, provjerite tipfeler
5. **Koristite znak # za komentare** – objašnjavajte sami sebi što radite

Što trebate prije početka

- Instaliran R (<https://www.r-project.org/>)
- Instaliran RStudio (<https://posit.co/downloads/>)
- Kreiran projektni direktorij (prema uputama u poglavlju)

1.2 Vježba 1: Prvi koraci u R-u

1.2.1 Cilj vježbe

Upoznati se s osnovnim konceptima: varijable, tipovi podataka, i kako R evaluira izraze.

1.2.2 Teorijska podloga

R možete koristiti kao kalkulator, ali njegova prava snaga je u **spremanju vrijednosti u varijable**. Varijabla je kao kutija u koju stavljamo vrijednost i dajemo joj ime kako bismo tu vrijednost mogli kasnije koristiti.

Operatori dodjele

U R-u postoje dva glavna načina dodjele vrijednosti varijabli:

- `<-` (preporučeno u R zajednici)
- `=` (također funkcionira)

Primjer:

```
1   x <- 5           # klasičan R stil
2   y = 10          # također funkcionira
```

Osnovni tipovi podataka

Tip	Primjer	Opis
numeric	42, 3.14	Brojevi
character	"hrast", "Risnjak"	Tekst (u navodnicima)
logical	TRUE, FALSE	Logičke vrijednosti
integer	42L	Cijeli broj (rijetko potreban)

Tablica 1.1: Osnovni tipovi podataka u R-u

1.2.3 Zadaci

Zadatak 1.1: Jednostavne matematičke operacije

Otvorite RStudio i u **Console**-u (donji lijevi panel) upišite sljedeće izraze, pritiskom Enter nakon svakog. R će odmah izračunati i prikazati rezultat.

```
1  # Zbrajanje
2  5 + 3
3
4  # Oduzimanje
5  10 - 7
6
7  # Množenje
8  4 * 6
9
10 # Dijeljenje
11 20 / 4
12
13 # Potenciranje
14 2^3
15
16 # Ostatak pri dijeljenju (modulo)
17 17 %% 5
18
19 # Cjelobrojno dijeljenje
20 17 %/% 5
```

Pitanje za razmišljanje: Što vraća `17 %% 5`? Zašto bi ova operacija mogla biti korisna?

Zadatak 1.2: Stvaranje varijabli

Sada ćemo spremiti vrijednosti u varijable.

```
1  # Broj vrsta na lokalitetu 1
2  broj_vrsta_lokalitet1 <- 45
3
4  # Broj vrsta na lokalitetu 2
5  broj_vrsta_lokalitet2 <- 38
6
7  # Izračunajte ukupan broj vrsta
8  ukupno_vrsta <- broj_vrsta_lokalitet1 + broj_vrsta_lokalitet2
9
10 # Prikažite rezultat
11 ukupno_vrsta
12
13 # Provjerite što je u radnom okruženju
```

```
14 ls()
```

Napomena: Funkcija `ls()` prikazuje sve varijable koje ste stvorili. Također ih možete vidjeti u **Environment** panelu (gornji desni panel u RStudio).

Zadatak 1.3: Rad s različitim tipovima podataka

```
1  # Numerički podaci
2  temperatura <- 22.5
3  vlaznost <- 68
4
5  # Tekstualni podaci (character)
6  vrsta <- "Canis lupus"
7  lokalitet <- "Nacionalni park Risnjak"
8
9  # Logički podaci (logical)
10 zatvoreno_podrucje <- TRUE
11 invazivna_vrsta <- FALSE
12
13 # Proujerite tipove podataka
14 class(temperatura)
15 class(vrsta)
16 class(zatvoreno_podrucje)
17
18 # Prikažite sve varijable
19 ls()
20
21 # Prikažite vrijednosti
22 temperatura
23 vrsta
24 zatvoreno_podrucje
```

Važno: Imena varijabli u R-u su *case-sensitive*, što znači da `Temperatura` i `temperatura` su različite varijable!

Zadatak 1.4: Pravila imenovanja varijabli

```
1  # DOBRI primjeri imena:
2  broj_jedinki <- 150
3  brojJedinki <- 150      # camelCase stil
4  broj.jedinki <- 150     # točka je dozvoljena
5  gustoca_m2 <- 12.5
6
7  # LOŠI primjeri (neće raditi ili nisu preporučeni):
8  # 1broj <- 100          # ne može početi brojem
9  # broj-jedinki <- 50    # crtice su zabranjene
10 # broj jedinki <- 30    # razmaci su zabranjeni
11 # TRUE <- 10           # ne možete prepisati rezervirane riječi
12
13 # Pokušajte sami:
14 # Stvorite varijablu za prosječnu duljinu tijela (u cm)
15 prosjecna_duljina_cm <- 85.3
16
17 # Stvorite varijablu za ime istraživača
18 istrazivac <- "Marko Horvat"
19
```

```

20  # Stvorite varijablu za broj opažanja
21  broj_opazanja <- 247

```

Zadatak 1.5: Jednostavni ekološki izračuni

```

1  # Podatci sa terena
2  površina_hektara <- 25
3  broj_stabala_hrasta <- 156
4  broj_stabala_bukve <- 203
5
6  # Izračunajte gustoću stabala po hektaru
7  ukupno_stabala <- broj_stabala_hrasta + broj_stabala_bukve
8  gustoca_stabala_ha <- ukupno_stabala / površina_hektara
9
10 # Prikažite rezultat
11 gustoca_stabala_ha
12
13 # Izračunajte postotak hrasta u sastojini
14 postotak_hrasta <- (broj_stabala_hrasta / ukupno_stabala) *
15   100
16 postotak_hrasta
17
18 # Zaokružite na 2 decimale
19 round(postotak_hrasta, 2)

```

Za razmišljanje: Koliko stabala ima na svakom hektaru? Koji je omjer hrasta i bukve?

1.2.4 Provjera razumijevanja

Bez gledanja u rješenja, pokušajte:

1. Stvoriti varijable s brojem opaženih jelena (12), divljih svinja (8) i srna (15)
2. Izračunati ukupan broj opaženih životinja
3. Izračunati postotak jelena u ukupnom uzorku
4. Stvoriti varijablu tekstualnog tipa s imenom parka gdje je provedeno opažanje
5. Stvoriti logičku varijablu koja označava je li bilo kiše tijekom opažanja

1.2.5 Česte greške i njihova rješenja

Greška	Uzrok	Rješenje
Error: object 'x' not found	Varijabla nije definirana ili ima tipfelr	Provjerite pravopis
Error: unexpected symbol	Zaboravljen operator ili navodnick	Provjerite sintaksu
+ u novom redu konzole	R čeka nastavak naredbe	Pritisnite ESC i počnite iznova

Tablica 1.2: Najčešće greške početnika

1.3 Vježba 2: Vektori – temelj R-a

1.3.1 Cilj vježbe

Naučiti raditi s vektorima – osnovnom strukturom podataka u R-u koja omogućava rad s više vrijednosti istovremeno.

1.3.2 Teorijska podloga

Vektor je slijed vrijednosti istog tipa. Možete ga zamisliti kao niz kutija postavljenih jednu za drugom, gdje svaka kutija sadrži vrijednost.

Zašto su vektori važni?

U ekologiji rijetko radimo s jednom vrijednošću. Obično imamo:

- Mjerenja s više lokacija
- Opažanja kroz više vremenskih točaka
- Brojeve jedinki različitih vrsta
- Okolišne varijable s različitih mjernih postaja

Vektori omogućavaju rad sa svim tim podacima odjednom!

Stvaranje vektora

Koristimo funkciju `c()` (od "combine" ili "concatenate"):

```
1 temperature <- c(18, 22, 19, 24, 21)
```

1.3.3 Zadaci

Zadatak 2.1: Stvaranje i pregled vektora

```
1 # Brojnosti ptica na 5 lokacija
2 brojnost_sjenice <- c(12, 8, 15, 10, 14)
3
4 # Pogledajte vektor
5 brojnost_sjenice
6
7 # Provjerite duljinu vektora
8 length(brojnost_sjenice)
9
10 # Provjerite tip podataka
11 class(brojnost_sjenice)
12
13 # Struktura objekta
14 str(brojnost_sjenice)
15
16 # Osnovne statistike
17 summary(brojnost_sjenice)
```

Zadatak 2.2: Operacije na vektorima

Jedna od najmoćnijih osobina R-a je **vektorizacija** – operacije se primjenjuju na sve elemente odjednom.

```
1 # Temperature u Celzijusima
2 temp_C <- c(15, 18, 22, 19, 16)
3
4 # Pretvorite u Fahrenheite
5 temp_F <- (temp_C * 9/5) + 32
6 temp_F
```

```
7
8  # Dodajte konstantu svim vrijednostima
9  temp_C_plus5 <- temp_C + 5
10 temp_C_plus5
11
12 # Brojnosti vrapčeva na istim lokacijama
13 brojnost_vrapac <- c(25, 30, 28, 22, 26)
14
15 # Izračunajte ukupne brojnosti
16 ukupno_ptica <- brojnost_sjenice + brojnost_vrapac
17 ukupno_ptica
18
19 # Izračunajte omjer sjenica i vrapčeva
20 omjer <- brojnost_sjenice / brojnost_vrapac
21 omjer
22
23 # Zaokružite na 2 decimale
24 round(omjer, 2)
```

Zadatak 2.3: Indeksiranje vektora

Indeksiranje omogućava pristup određenim elementima vektora.

```
1  # Kreirajte vektor s pH vrijednostima tla
2  pH_tla <- c(6.5, 7.2, 5.8, 6.9, 7.5, 6.1, 7.0)
3
4  # Prvi element
5  pH_tla[1]
6
7  # Treći element
8  pH_tla[3]
9
10 # Zadnji element (dva načina)
11 pH_tla[7]
12 pH_tla[length(pH_tla)]
13
14 # Više elemenata odjednom
15 pH_tla[c(1, 3, 5)]
16
17 # Raspon elemenata
18 pH_tla[2:5]
19
20 # Svi elementi osim prvog
21 pH_tla[-1]
22
23 # Svi elementi osim prvog i zadnjeg
24 pH_tla[-c(1, 7)]
25
26 # Elementi koji zadovoljavaju uvjet
27 pH_tla[pH_tla > 7]  # svi elementi veći od 7
28 pH_tla[pH_tla < 6]  # svi elementi manji od 6
```

Važno: R koristi 1-bazirano indeksiranje (za razliku od Python-a koji koristi 0-bazirano).

Zadatak 2.4: Statističke funkcije

```

1  # Težine ženki jelena (kg)
2  tezina_jelena <- c(85, 92, 88, 78, 95, 87, 91, 84)
3
4  # Osnovne statističke mjere
5  mean(tezina_jelena)      # aritmetička sredina
6  median(tezina_jelena)    # medijan
7  sd(tezina_jelena)        # standardna devijacija
8  var(tezina_jelena)       # varijanca
9  min(tezina_jelena)       # minimum
10 max(tezina_jelena)       # maksimum
11 range(tezina_jelena)     # raspon (min i max)
12 quantile(tezina_jelena)  # kvartili
13
14 # IQR - interkvartilni raspon
15 IQR(tezina_jelena)
16
17 # Suma
18 sum(tezina_jelena)
19
20 # Broj elemenata
21 length(tezina_jelena)
22
23 # Proujerite ima li nedostajućih vrijednosti
24 any(is.na(tezina_jelena))

```

Zadatak 2.5: Rad s nedostajućim vrijednostima

U ekologiji često imamo nepotpune podatke. R označava nedostajuće vrijednosti s NA.

```

1  # Brojnosti organizama, s nekim nedostajućim podacima
2  brojnost <- c(12, 8, NA, 15, 10, NA, 14, 9)
3
4  # Pokušajte izračunati sredinu
5  mean(brojnost)           # Rezultat: NA
6
7  # Ispravno: ignorirajte NA vrijednosti
8  mean(brojnost, na.rm = TRUE)
9  sum(brojnost, na.rm = TRUE)
10 sd(brojnost, na.rm = TRUE)
11
12 # Proujerite koji su elementi NA
13 is.na(brojnost)
14
15 # Broj NA vrijednosti
16 sum(is.na(brojnost))
17
18 # Elementi koji NISU NA
19 brojnost[!is.na(brojnost)]
20
21 # Uklonite NA iz vektora
22 brojnost_cist <- na.omit(brojnost)
23 brojnost_cist

```

Zadatak 2.6: Sekvence i ponavljanja

```

1  # Kreirajte sekvence - korisno za godine, transekte, parcele

```

```

2  godine <- 2015:2024
3  godine
4
5  # Sekvenca s korakom
6  c(0, 5, 10, 15, 20)          # ručno
7  seq(from = 0, to = 20, by = 5) # funkcija seq()
8
9  # 10 brojeva između 0 i 100
10 seq(from = 0, to = 100, length.out = 10)
11
12 # Ponavljanje vrijednosti
13 rep(5, times = 10)           # broj 5, 10 puta
14 rep(c(1, 2, 3), times = 3)   # vektor [1,2,3] tri puta
15 rep(c("kontrola", "tretman"), each = 5) # svaki element 5
    puta
16
17 # Kombinacija - korisno za eksperimentalni dizajn
18 tretman <- rep(c("A", "B", "C"), each = 4)
19 tretman

```

Zadatak 2.7: Praktični ekološki primjer

```

1  # Simulirajmo podatke s terenskog istraživanja
2  # 10 transekt, brojanje biljnih vrsta
3
4  transekt <- 1:10
5  brojnost_vrsta <- c(23, 28, 19, 31, 25, 22, 27, 24, 29, 26)
6
7  # Osnovna analiza
8  mean(brojnost_vrsta)
9  sd(brojnost_vrsta)
10
11 # Koji transekt ima najviše vrsta?
12 max(brojnost_vrsta)
13 which.max(brojnost_vrsta) # vraća POZICIJU
14 transekt[which.max(brojnost_vrsta)] # vraća BROJ transekt
15
16 # Koji transekt ima najmanje vrsta?
17 transekt[which.min(brojnost_vrsta)]
18
19 # Transekti s više od 25 vrsta
20 transekt[brojnost_vrsta > 25]
21 brojnost_vrsta[brojnost_vrsta > 25]
22
23 # Postotak transekt s više od 25 vrsta
24 sum(brojnost_vrsta > 25) / length(brojnost_vrsta) * 100

```

1.3.4 Provjera razumijevanja

1. Kreirajte vektor s temperaturama izmjerenima tijekom tjedna: 18, 22, 25, 23, 19, 21, 24 °C
2. Izračunajte prosječnu temperaturu
3. Koje su bile temperature iznad prosjeka?
4. Koliki je bio raspon temperatura (razlika između max i min)?

5. Dodajte jednu nedostajuću vrijednost (NA) na kraj vektora i ponovno izračunajte prosjek

1.3.5 Napredni izazov

```
1  # Za one koji žele dodatni izazov:
2  # Shannonomov indeks raznolikosti za zajednicu s 5 vrsta
3
4  brojnost <- c(45, 32, 18, 8, 3)
5
6  # Izračunajte proporcije
7  ukupno <- sum(brojnost)
8  proporcije <- brojnost / ukupno
9
10 # Shannonomov indeks:  $H = -\sum(p * \log(p))$ 
11 H <- -sum(proporcije * log(proporcije))
12 H
13
14 # Usporedite s funkcijom iz paketa vegan (ako je instaliran):
15 # library(vegan)
16 # diversity(brojnost, index = "shannon")
```

1.4 Vježba 3: Data frame – rad s tablicama

1.4.1 Cilj vježbe

Naučiti raditi s `data.frame` objektima – primarnom strukturom za ekološke podatke u R-u.

1.4.2 Teorijska podloga

Data frame je tablica podataka – ima redove (obično opažanja) i stupce (obično varijable). Možete ga zamisliti kao Excel tablicu unutar R-a, ali s mnogo većim mogućnostima.

Karakteristike data frame-a:

- Svaki stupac je vektor
- Svi vektori moraju biti iste duljine
- Različiti stupci mogu imati različite tipove podataka
- Svaki stupac i redak može imati ime

1.4.3 Zadaci

Zadatak 3.1: Kreiranje data frame-a

```
1  # Kreirajmo jednostavan dataset s monitoring podacima
2
3  # Prvo kreiramo vektore
4  lokalitet <- c("Risnjak", "Plitvice", "Velebit",
5  "Kopački rit", "Medvednica")
6  broj_vrsta <- c(45, 52, 38, 41, 35)
7  površina_ha <- c(6400, 29600, 20000, 17700, 3900)
8  god_zastite <- c(1953, 1949, 1981, 1967, 1981)
9
10 # Kombiniramo u data frame
11 nacionalni_parkovi <- data.frame(
12   lokalitet = lokalitet,
13   broj_vrsta = broj_vrsta,
14   površina_ha = površina_ha,
15   god_zastite = god_zastite
16 )
17
18 # Pogledajte što smo stvorili
19 nacionalni_parkovi
20
21 # Alternativno, možemo pisati direktno:
22 nacionalni_parkovi <- data.frame(
23   lokalitet = c("Risnjak", "Plitvice", "Velebit",
24   "Kopački rit", "Medvednica"),
25   broj_vrsta = c(45, 52, 38, 41, 35),
26   površina_ha = c(6400, 29600, 20000, 17700, 3900),
27   god_zastite = c(1953, 1949, 1981, 1967, 1981)
28 )
```

Zadatak 3.2: Pregled data frame-a

```
1  # Osnovno o strukturi
2  dim(nacionalni_parkovi)      # dimenzije (redovi, stupci)
3  nrow(nacionalni_parkovi)     # broj redova
4  ncol(nacionalni_parkovi)     # broj stupaca
5  names(nacionalni_parkovi)    # nazivi stupaca
6  colnames(nacionalni_parkovi) # isto što i names()
```

```

7   rownames(nacionalni_parkovi) # nazivi redaka
8
9   # Prikaz prvih i zadnjih redaka
10  head(nacionalni_parkovi)      # prvih 6 redaka (default)
11  head(nacionalni_parkovi, 3)   # prvih 3 retka
12  tail(nacionalni_parkovi, 2)  # zadnja 2 retka
13
14  # Struktura podataka
15  str(nacionalni_parkovi)
16
17  # Statistički sažetak
18  summary(nacionalni_parkovi)
19
20  # Provjera tipova stupaca
21  class(nacionalni_parkovi$lokalitet)
22  class(nacionalni_parkovi$broj_vrsta)

```

Zadatak 3.3: Pristup podacima

Postoji nekoliko načina pristupanja podacima u data frame-u:

```

1   # 1. Znak $ - pristup cijelom stupcu
2   nacionalni_parkovi$broj_vrsta
3   nacionalni_parkovi$lokalitet
4
5   # 2. Uglate zagrade [redak, stupac]
6   nacionalni_parkovi[1, ]      # prvi redak, svi stupci
7   nacionalni_parkovi[, 2]      # svi redci, drugi stupac
8   nacionalni_parkovi[1, 2]     # prvi redak, drugi stupac
9   nacionalni_parkovi[1:3, ]    # prva tri retka
10
11  # 3. Pristup po imenu
12  nacionalni_parkovi[, "broj_vrsta"]
13  nacionalni_parkovi[, c("lokalitet", "broj_vrsta")]
14
15  # 4. Pristup s uvjetima (logičko indeksiranje)
16  nacionalni_parkovi[nacionalni_parkovi$broj_vrsta > 40, ]
17
18  # 5. Kombinacija
19  nacionalni_parkovi[nacionalni_parkovi$povrsina_ha > 10000,
20  c("lokalitet", "povrsina_ha")]

```

Zadatak 3.4: Dodavanje novih stupaca

```

1   # Dodajte novi stupac - gustoća vrsta po hektaru
2   nacionalni_parkovi$gustoca_vrsta <-
3   nacionalni_parkovi$broj_vrsta / nacionalni_parkovi$povrsina_ha
4
5   # Dodajte kategorijalni stupac - veličina parka
6   nacionalni_parkovi$velicina <- ifelse(
7   nacionalni_parkovi$povrsina_ha > 15000,
8   "Velik",
9   "Mali"
10  )
11
12  # Pogledajte rezultat
13  nacionalni_parkovi

```

```

14
15   # Dodajte stupac - broj godina zaštite
16   trenutna_godina <- 2024
17   nacionalni_parkovi$godine_zastite <-
18   trenutna_godina - nacionalni_parkovi$god_zastite
19
20   # Provjera
21   nacionalni_parkovi

```

Zadatak 3.5: Sortiranje podataka

```

1   # Sortirajte po broju vrsta (uzlazno)
2   nacionalni_parkovi[order(nacionalni_parkovi$broj_vrsta), ]
3
4   # Sortirajte po broju vrsta (silazno)
5   nacionalni_parkovi[order(nacionalni_parkovi$broj_vrsta,
6   decreasing = TRUE), ]
7
8   # Sortirajte po površini
9   nacionalni_parkovi[order(nacionalni_parkovi$povrsina_ha,
10  decreasing = TRUE), ]
11
12  # Spremite sortiranu verziju
13  parkovi_sortirani <- nacionalni_parkovi[
14  order(nacionalni_parkovi$broj_vrsta, decreasing = TRUE),
15  ]
16
17  parkovi_sortirani

```

Zadatak 3.6: Filtriranje podataka

```

1   # Parkovi s više od 40 vrsta
2   veliki_diverzitet <- nacionalni_parkovi[
3   nacionalni_parkovi$broj_vrsta > 40,
4   ]
5   veliki_diverzitet
6
7   # Parkovi zaštićeni prije 1970.
8   stari_parkovi <- nacionalni_parkovi[
9   nacionalni_parkovi$god_zastite < 1970,
10  ]
11  stari_parkovi
12
13  # Kombinacija uvjeta - veliki parkovi s puno vrsta
14  veliki_i_raznolik <- nacionalni_parkovi[
15  nacionalni_parkovi$povrsina_ha > 10000 &
16  nacionalni_parkovi$broj_vrsta > 40,
17  ]
18  veliki_i_raznolik
19
20  # Parkovi koji su zaštićeni 1981. ILI imaju više od 50 vrsta
21  subset(nacionalni_parkovi,
22  god_zastite == 1981 | broj_vrsta > 50)

```

Logički operatori:

- == jednako
- != različito od
- >, <, >=, <= usporedbe
- & logičko I (AND)
- | logičko ILI (OR)
- ! logička negacija (NOT)

Zadatak 3.7: Agregatne funkcije

```

1  # Prosječan broj vrsta
2  mean(nacionalni_parkovi$broj_vrsta)
3
4  # Ukupna površina svih parkova
5  sum(nacionalni_parkovi$povrsina_ha)
6
7  # Park s najvećom površinom
8  nacionalni_parkovi[which.max(nacionalni_parkovi$povrsina_ha),
9  "lokalitet"]
10
11 # Najstariji park
12 nacionalni_parkovi[which.min(nacionalni_parkovi$god_zastite),
13 "lokalitet"]
14
15 # Agregacija po kategorijama
16 aggregate(broj_vrsta ~ velicina,
17 data = nacionalni_parkovi,
18 FUN = mean)

```

Zadatak 3.8: Rad s realističnijim podacima

```

1  # Kreirajmo dataset koji simulira monitoring ptica kroz godine
2
3  monitoring <- data.frame(
4  godina = rep(2020:2024, each = 3),
5  lokalitet = rep(c("Risnjak", "Plitvice", "Velebit"), times =
6    5),
7  brojnost_sjenica = c(12, 15, 8, 14, 17, 10, 15, 18, 11,
8    16, 19, 13, 17, 20, 14),
9  brojnost_zeba = c(25, 28, 22, 27, 30, 24, 29, 32, 26,
10    31, 34, 28, 33, 36, 30),
11  temperatura_C = c(8.5, 9.2, 7.8, 8.8, 9.5, 8.1, 9.1, 9.8, 8.4,
12    9.3, 10.0, 8.7, 9.6, 10.3, 9.0)
13 )
14
15 # Pogledajte podatke
16 head(monitoring, 10)
17 str(monitoring)
18
19 # Analize:
20
21 # 1. Prosječna brojnost sjenica po lokalitetu
22 aggregate(brojnost_sjenica ~ lokalitet,
23 data = monitoring,
24 FUN = mean)
25
26 # 2. Prosječna temperatura po godini

```

```

26 aggregate(temperatura_C ~ godina,
27 data = monitoring,
28 FUN = mean)
29
30 # 3. Filtrirajte podatke za Plitvice
31 plitvice_podaci <- monitoring[monitoring$lokalitet == "
    Plitvice", ]
32 plitvice_podaci
33
34 # 4. Podaci iz 2024. godine s temperaturom iznad 9°C
35 monitoring_2024 <- monitoring[
36 monitoring$godina == 2024 & monitoring$temperatura_C > 9,
37 ]
38 monitoring_2024
39
40 # 5. Dodajte novi stupac - ukupan broj ptica
41 monitoring$ukupno_ptica <- monitoring$brojnost_sjenica +
42 monitoring$brojnost_zeba
43
44 # 6. Dodajte stupac - omjer sjenica i zeba
45 monitoring$omjer_sjenice_zebe <-
46 monitoring$brojnost_sjenica / monitoring$brojnost_zeba
47
48 # Prikaz
49 head(monitoring)

```

1.4.4 Provjera razumijevanja

Koristeći gornji monitoring dataset, odgovorite:

1. Koliko ukupno opažanja (redaka) ima dataset?
2. Koji lokalitet ima najviši prosječan ukupan broj ptica?
3. Koliki je raspon temperatura kroz sve godine i lokalitete?
4. Koliko opažanja ima brojnost sjenica veću od 15?
5. Koji je prosječan omjer sjenica i zeba?

1.4.5 Napredni zadatak

```

1 # Za dodatni izazov:
2 # 1. Izračunajte godišnje promjene brojnosti za svaki
    lokalitet
3 # 2. Identifi cijte lokalitet i godinu s najvećom ukupnom
    brojnošću
4 # 3. Istražite postoji li korelacija između temperature i
    brojnosti
5
6 # Primjer za korelaciju:
7 cor(monitoring$temperatura_C, monitoring$brojnost_sjenica)
8 cor(monitoring$temperatura_C, monitoring$brojnost_zeba)
9
10 # Vizualizacija
11 plot(monitoring$temperatura_C, monitoring$brojnost_sjenica,
12 xlab = "Temperatura (°C)",
13 ylab = "Brojnost sjenica",
14 main = "Odnos temperature i brojnosti")

```

1.5 Vježba 4: Vizualizacija podataka

1.5.1 Cilj vježbe

Naučiti osnovne metode grafičkog prikaza ekoloških podataka koristeći `base` R grafiku i uvod u `ggplot2`.

1.5.2 Teorijska podloga

"A picture is worth a thousand words" – i u ekologiji, dobra vizualizacija često otkriva obrasce koje brojke ne mogu.

Vizualizacija podataka služi za:

- Eksploratornu analizu (razumijevanje podataka)
- Detekciju outliera i grešaka
- Komunikaciju rezultata
- Testiranje pretpostavki statističkih modela

1.5.3 Zadaci

Zadatak 4.1: Osnovna grafika - histogram

```

1  # Koristimo monitoring dataset iz prethodne vježbe
2  # Ako ga nemate, ponovno ga kreirajte
3
4  monitoring <- data.frame(
5    godina = rep(2020:2024, each = 3),
6    lokalitet = rep(c("Risnjak", "Plitvice", "Velebit"), times =
7      5),
8    brojnost_sjenica = c(12, 15, 8, 14, 17, 10, 15, 18, 11,
9      16, 19, 13, 17, 20, 14),
10   brojnost_zeba = c(25, 28, 22, 27, 30, 24, 29, 32, 26,
11     31, 34, 28, 33, 36, 30),
12   temperatura_C = c(8.5, 9.2, 7.8, 8.8, 9.5, 8.1, 9.1, 9.8, 8.4,
13     9.3, 10.0, 8.7, 9.6, 10.3, 9.0)
14 )
15
16 # Jednostavan histogram
17 hist(monitoring$brojnost_sjenica)
18
19 # Poboljšana verzija
20 hist(monitoring$brojnost_sjenica,
21   main = "Distribucija brojnosti sjenica",
22   xlab = "Brojnost sjenica",
23   ylab = "Frekvencija",
24   col = "lightblue",
25   border = "darkblue",
26   breaks = 10)
27
28 # Dodajte vertikalnu liniju za prosjek
29 abline(v = mean(monitoring$brojnost_sjenica),
30   col = "red",
31   lwd = 2,
32   lty = 2)
33
34 # Dodajte legendu
35 legend("topright",

```

```

35 legend = "Prosjek",
36 col = "red",
37 lty = 2,
38 lwd = 2)

```

Zadatak 4.2: Scatter plot (dijagram raspršenja)

```

1  # Odnos temperature i brojnosti
2  plot(monitring$temperatura_C,
3  monitring$brojnost_sjenica)
4
5  # Poboljšana verzija
6  plot(monitring$temperatura_C,
7  monitring$brojnost_sjenica,
8  main = "Odnos temperature i brojnosti sjenica",
9  xlab = "Temperatura (°C)",
10 ylab = "Brojnost sjenica",
11 pch = 19,           # tip točke
12 col = "darkgreen",  # boja
13 cex = 1.5)          # veličina točke
14
15 # Dodajte liniju trenda
16 abline(lm(brojnost_sjenica ~ temperatura_C, data = monitring)
17 ,
18 col = "red",
19 lwd = 2)
20
21 # Alternativno: različite boje po lokalitetu
22 plot(monitring$temperatura_C,
23 monitring$brojnost_sjenica,
24 main = "Brojnost sjenica po temperaturi i lokalitetu",
25 xlab = "Temperatura (°C)",
26 ylab = "Brojnost sjenica",
27 pch = 19,
28 col = as.factor(monitring$lokalitet),
29 cex = 1.5)
30
31 # Legenda
32 legend("topleft",
33 legend = levels(as.factor(monitring$lokalitet)),
34 col = 1:3,
35 pch = 19)

```

Zadatak 4.3: Box plot (kutijasti dijagram)

Box plot pokazuje distribuciju podataka kroz kvartile.

```

1  # Jednostavan boxplot
2  boxplot(brojnost_sjenica ~ lokalitet,
3  data = monitring)
4
5  # Poboljšana verzija
6  boxplot(brojnost_sjenica ~ lokalitet,
7  data = monitring,
8  main = "Brojnost sjenica po lokalitetima",
9  xlab = "Lokalitet",
10 ylab = "Brojnost sjenica",

```



```

11 col = c("lightblue", "lightgreen", "lightyellow"),
12 border = "darkblue",
13 notch = TRUE) # notch pokazuje interval pouzdanosti medijana
14
15 # Dodajte točke za individualna opažanja
16 points(jitter(as.numeric(as.factor(monitored$lokalitet))),
17 monitored$brojnost_sjenica,
18 pch = 19,
19 col = "red",
20 cex = 0.8)

```

Tumačenje box plot:

- Crta u kutiji = medijan
- Kutija = interkvartilni raspon (IQR, 25-75 percentil)
- Brkovi = $1.5 \times \text{IQR}$
- Točke izvan brkova = potencijalni outlieri

Zadatak 4.4: Linijski grafikon

Korisno za vremenske serije.

```

1  # Izračunajte prosječnu brojnost po godini
2  prosjek_godina <- aggregate(brojnost_sjenica ~ godina,
3  data = monitoring,
4  FUN = mean)
5
6  # Linijski grafikon
7  plot(prosjek_godina$godina,
8  prosjek_godina$brojnost_sjenica,
9  type = "l", # "l" za liniju
10 main = "Trend brojnosti sjenica kroz godine",
11 xlab = "Godina",
12 ylab = "Prosječna brojnost",
13 col = "blue",
14 lwd = 2)
15
16 # Dodajte točke
17 points(prosjek_godina$godina,
18 prosjek_godina$brojnost_sjenica,
19 pch = 19,
20 col = "blue",
21 cex = 1.5)
22
23 # Dodajte grid
24 grid()
25
26 # Alternativno: sve lokalitete na jednom grafikonu
27 risnjak <- monitoring[monitoring$lokalitet == "Risnjak", ]
28 plitvice <- monitoring[monitoring$lokalitet == "Plitvice", ]
29 velebit <- monitoring[monitoring$lokalitet == "Velebit", ]
30
31 plot(risnjak$godina, risnjak$brojnost_sjenica,
32 type = "b", # "b" za both (i linija i točke)
33 main = "Trend brojnosti po lokalitetima",
34 xlab = "Godina",
35 ylab = "Brojnost sjenica",

```

```

36 col = "red",
37 pch = 19,
38 ylim = c(5, 25)) # fiksni y-raspon za sve linije
39
40 lines(plitvice$godina, plitvice$brojnost_sjenica,
41 type = "b", col = "blue", pch = 19)
42
43 lines(velebit$godina, velebit$brojnost_sjenica,
44 type = "b", col = "green", pch = 19)
45
46 legend("topleft",
47 legend = c("Risnjak", "Plitvice", "Velebit"),
48 col = c("red", "blue", "green"),
49 lty = 1,
50 pch = 19)

```

Zadatak 4.5: Barplot (stupčasti dijagram)

```

1 # Prosječna brojnost po lokalitetu
2 prosjek_lokalitet <- aggregate(brojnost_sjenica ~ lokalitet,
3 data = monitoring,
4 FUN = mean)
5
6 # Jednostavan barplot
7 barplot(prosjek_lokalitet$brojnost_sjenica,
8 names.arg = prosjek_lokalitet$lokalitet)
9
10 # Poboljšana verzija
11 barplot(prosjek_lokalitet$brojnost_sjenica,
12 names.arg = prosjek_lokalitet$lokalitet,
13 main = "Prosječna brojnost sjenica po lokalitetima",
14 ylab = "Brojnost",
15 col = c("lightblue", "lightgreen", "lightyellow"),
16 border = "darkblue",
17 ylim = c(0, 20))
18
19 # Dodajte numeričke vrijednosti iznad stupaca
20 text(x = 1:3 * 1.2 - 0.5, # pozicija x
21 y = prosjek_lokalitet$brojnost_sjenica + 1, # pozicija y
22 labels = round(prosjek_lokalitet$brojnost_sjenica, 1),
23 cex = 1.2)
24
25 # Dodajte standardne devijacije kao error bars
26 sd_lokalitet <- aggregate(brojnost_sjenica ~ lokalitet,
27 data = monitoring,
28 FUN = sd)
29
30 bp <- barplot(prosjek_lokalitet$brojnost_sjenica,
31 names.arg = prosjek_lokalitet$lokalitet,
32 main = "Brojnost sjenica (prosjeak ± SD)",
33 ylab = "Brojnost",
34 col = "lightblue",
35 ylim = c(0, 22))
36
37 # Dodajte error bars
38 arrows(x0 = bp,

```

```

39 y0 = prosjek_lokalitet$brojnost_sjenica -
    sd_lokalitet$brojnost_sjenica,
40 y1 = prosjek_lokalitet$brojnost_sjenica +
    sd_lokalitet$brojnost_sjenica,
41 angle = 90,
42 code = 3,
43 length = 0.1)

```

Zadatak 4.6: Spremanje grafikona

```

1  # Spremanje kao PNG
2  png("output/figures/brojnost_sjenica_trend.png",
3      width = 800, height = 600)
4
5  plot(prosjek_godina$godina,
6       prosjek_godina$brojnost_sjenica,
7       type = "b",
8       main = "Trend brojnosti sjenica kroz godine",
9       xlab = "Godina",
10      ylab = "Prosječna brojnost",
11      col = "blue",
12      lwd = 2,
13      pch = 19,
14      cex = 1.5)
15  grid()
16
17  dev.off() # Važno! Zatvara grafički uređaj
18
19  # Spremanje kao PDF (vektorski format, bolji za publikacije)
20  pdf("output/figures/brojnost_lokaliteti.pdf",
21      width = 8, height = 6)
22
23  boxplot(brojnost_sjenica ~ lokalitet,
24         data = monitoring,
25         main = "Distribucija brojnosti po lokalitetima",
26         xlab = "Lokalitet",
27         ylab = "Brojnost sjenica",
28         col = "lightblue")
29
30  dev.off()
31
32  # Spremanje kao SVG (također vektorski)
33  svg("output/figures/scatter_temp_brojnost.svg",
34      width = 8, height = 6)
35
36  plot(monitoring$temperatura_C,
37       monitoring$brojnost_sjenica,
38       main = "Odnos temperature i brojnosti",
39       xlab = "Temperatura (°C)",
40       ylab = "Brojnost sjenica",
41       pch = 19,
42       col = "darkgreen")
43
44  dev.off()

```

Zadatak 4.7: Uvod u ggplot2

ggplot2 je moćniji i fleksibilniji sistem za grafiku, ali ima strmiju krivulju učenja.

```

1  # Instalirajte i učitajte paket (samo prvi put)
2  # install.packages("ggplot2")
3  library(ggplot2)
4
5  # Osnovni scatter plot
6  ggplot(data = monitoring,
7  aes(x = temperatura_C, y = brojnost_sjenica)) +
8  geom_point()
9
10 # Poboljšana verzija
11 ggplot(data = monitoring,
12 aes(x = temperatura_C, y = brojnost_sjenica,
13 color = lokalitet)) +
14 geom_point(size = 3) +
15 geom_smooth(method = "lm", se = TRUE) +
16 labs(title = "Odnos temperature i brojnosti sjenica",
17 x = "Temperatura (°C)",
18 y = "Brojnost sjenica",
19 color = "Lokalitet") +
20 theme_minimal()
21
22 # Box plot s ggplot2
23 ggplot(monitoring, aes(x = lokalitet, y = brojnost_sjenica,
24 fill = lokalitet)) +
25 geom_boxplot() +
26 geom_jitter(width = 0.1, alpha = 0.5) +
27 labs(title = "Distribucija brojnosti po lokalitetima",
28 x = "Lokalitet",
29 y = "Brojnost sjenica") +
30 theme_classic() +
31 theme(legend.position = "none")
32
33 # Linijski grafikon s ggplot2
34 # Prvo izračunajte prosječne vrijednosti
35 library(dplyr) # korisno za manipulaciju podataka
36
37 prosjeci <- monitoring %>%
38 group_by(godina, lokalitet) %>%
39 summarise(prosjek = mean(brojnost_sjenica),
40 sd = sd(brojnost_sjenica))
41
42 ggplot(prosjeci, aes(x = godina, y = prosjek,
43 color = lokalitet)) +
44 geom_line(linewidth = 1) +
45 geom_point(size = 3) +
46 geom_errorbar(aes(ymin = prosjek - sd, ymax = prosjek + sd),
47 width = 0.2) +
48 labs(title = "Trend brojnosti sjenica po lokalitetima",
49 x = "Godina",
50 y = "Brojnost (prosjek ± SD)",
51 color = "Lokalitet") +
52 theme_minimal() +
53 scale_x_continuous(breaks = 2020:2024)

```

```

54
55   # Spremanje ggplot grafikona
56   ggsave("output/figures/ggplot_trend.png",
57   width = 10, height = 6, dpi = 300)

```

1.5.4 Provjera razumijevanja

Koristeći monitoring dataset:

1. Napravite histogram distribucije temperatura
2. Napravite scatter plot odnosa temperature i brojnosti zeba
3. Napravite box plot brojnosti zeba po godinama
4. Napravite barplot ukupnog broja vrsta (sjenice + zebi) po lokalitetima
5. Spremite najmanje jedan grafikon kao PNG datoteku

1.5.5 Napredni zadatak

```

1   # Za dodatni izazov - kombinirani grafikon:
2   # Četiri panela u jednoj slici (2x2)
3
4   par(mfrow = c(2, 2)) # 2 retka, 2 stupca
5
6   # Panel 1: Histogram
7   hist(monitring$brojnost_sjenica,
8   main = "Distribucija brojnosti",
9   xlab = "Brojnost sjenica",
10  col = "lightblue")
11
12  # Panel 2: Scatter plot
13  plot(monitring$temperatura_C, monitring$brojnost_sjenica,
14  main = "Temperatura vs. brojnost",
15  xlab = "Temperatura (°C)",
16  ylab = "Brojnost",
17  pch = 19,
18  col = "darkgreen")
19
20  # Panel 3: Box plot
21  boxplot(brojnost_sjenica ~ lokalitet,
22  data = monitring,
23  main = "Brojnost po lokalitetima",
24  col = "lightyellow")
25
26  # Panel 4: Linijski grafikon
27  prosjek_god <- aggregate(brojnost_sjenica ~ godina,
28  data = monitring,
29  FUN = mean)
30  plot(prosjek_god$godina, prosjek_god$brojnost_sjenica,
31  type = "b",
32  main = "Trend kroz godine",
33  xlab = "Godina",
34  ylab = "Prosjek",
35  pch = 19,
36  col = "red")
37
38  # Vratite postavke natrag
39  par(mfrow = c(1, 1))

```

```
40
41   # Spremite kompletan grafikon
42   png("output/figures/kombinir ani_grafikon.png",
43       width = 1000, height = 1000)
44   # ... ponovite gornji kod ...
45   dev.off()
```

1.6 Vježba 5: Učitavanje i čišćenje pravih podataka

1.6.1 Cilj vježbe

Naučiti raditi s pravim ekološkim podacima – učitavanje iz različitih formata, identifikacija problema, čišćenje podataka.

1.6.2 Teorijska podloga

U pravom istraživačkom radu rijetko dobivate savršeno pripremljene podatke. Obično morate:

- Učitati podatke iz različitih formata (CSV, Excel, TXT)
- Provjeriti kvalitetu podataka
- Riješiti probleme s nedostajućim vrijednostima
- Ispraviti greške u unosu
- Standardizirati formate
- Transformirati podatke u potreban oblik

1.6.3 Zadaci

Zadatak 5.1: Učitavanje CSV datoteka

CSV (Comma-Separated Values) je najčešći format za ekološke podatke.

```

1  # Osnovno učitavanje
2  podaci <- read.csv("data/raw/monitoring_ptice.csv")
3
4  # S dodatnim parametrima
5  podaci <- read.csv("data/raw/monitoring_ptice.csv",
6  header = TRUE,          # prvi red su nazivi stupaca
7  sep = ",",              # separator je zarez
8  dec = ".",              # decimalna točka
9  stringsAsFactors = FALSE) # ne pretvori tekst u faktore
10
11 # Učitavanje s hrvatskim postavkama (točka-zarez separator)
12 # Ako su podaci iz Excel-a sa hrvatskim postavkama
13 podaci <- read.csv2("data/raw/monitoring_ptice.csv") # sep =
    ";"
14
15 # Alternativa s readr paketom (brže i bolje)
16 # install.packages("readr")
17 library(readr)
18
19 podaci <- read_csv("data/raw/monitoring_ptice.csv")
20
21 # Pregled učitanih podataka
22 head(podaci)
23 str(podaci)
24 summary(podaci)

```

Zadatak 5.2: Učitavanje Excel datoteka

```

1  # Instalirajte paket za Excel (samo prvi put)
2  # install.packages("readxl")
3  library(readxl)
4
5  # Učitajte prvi sheet
6  podaci_excel <- read_excel("data/raw/monitoring.xlsx")

```

```

7
8 # Učitajte specifičan sheet po imenu
9 podaci_excel <- read_excel("data/raw/monitoring.xlsx",
10 sheet = "Plitvice")
11
12 # Učitajte specifičan sheet po broju
13 podaci_excel <- read_excel("data/raw/monitoring.xlsx",
14 sheet = 2)
15
16 # Preskočite prvih n redaka (ako ima zaglavlja)
17 podaci_excel <- read_excel("data/raw/monitoring.xlsx",
18 skip = 3)
19
20 # Učitajte samo određeni raspon ćelija
21 podaci_excel <- read_excel("data/raw/monitoring.xlsx",
22 range = "A1:E100")

```

Zadatak 5.3: Identifikacija problema u podacima

```

1 # Kreirajmo dataset s problemima
2 messy_data <- data.frame(
3   lokacija = c("Risnjak", "Plitvice", "Velebit", NA, "Risnjak",
4   "plitvice", "VELEBIT"),
5   brojnost = c(12, 15, NA, 10, "N/A", 18, 20),
6   datum = c("2024-01-15", "2024-02-20", "2024-01-", "2024-03-10"
7   ,
8   "01.04.2024", "2024-05-15", "2024-06-20"),
9   temperatura = c(8.5, 12.3, 7.8, NA, 25.0, 9.1, 10.5),
10  observer = c("A.M.", "A.M.", "B.K.", "B.K.", "am", "A.M.", "b.
11  k.")
12 )
13
14 # Problema:
15 # 1. NA vrijednosti
16 sum(is.na(messy_data))
17 colSums(is.na(messy_data))
18
19 # 2. Nekonzistentna velika/mala slova
20 table(messy_data$lokacija)
21
22 # 3. brojnost je character, ne numeric!
23 str(messy_data)
24
25 # 4. Različiti formati datuma
26
27 # 5. Temperatura od 25°C u siječnju - sumnjivo
28
29 # 6. Nekonzistentni kodovi opažača

```

Zadatak 5.4: Čišćenje podataka

```

1 # 1. Riješite problem s velika/mala slova
2 messy_data$lokacija <- tolower(messy_data$lokacija) # sve
3   malo
4 messy_data$lokacija <- tools::toTitleCase(messy_data$lokacija)
5   # prvo veliko

```



```

4
5 # 2. Standardizirajte lokacije
6 messy_data$lokacija <- gsub("plitvice", "Plitvice",
7 messy_data$lokacija,
8 ignore.case = TRUE)
9 messy_data$lokacija <- gsub("velebit", "Velebit",
10 messy_data$lokacija,
11 ignore.case = TRUE)
12
13 # 3. Riješite problem s brojnošću
14 # "N/A" treba biti NA
15 messy_data$brojnost[messy_data$brojnost == "N/A"] <- NA
16
17 # Pretvorite u numeric
18 messy_data$brojnost <- as.numeric(messy_data$brojnost)
19
20 # 4. Standardizirajte datum
21 # install.packages("lubridate")
22 library(lubridate)
23
24 # Pokušajte parsirati različite formate
25 messy_data$datum_clean <- ymd(messy_data$datum)
26
27 # 5. Provjerite outliere u temperaturi
28 # Temperatura od 25°C u siječnju je sumnjiva
29 boxplot(messy_data$temperatura)
30
31 # Identifi cijte sumnjivu vrijednost
32 sumnjivo <- messy_data[messy_data$temperatura > 20 &
33 !is.na(messy_data$temperatura), ]
34 sumnjivo
35
36 # Možda je trebalo biti 2.5 umjesto 25.0?
37 # Ili postavite na NA i kontaktirajte terenske istraživače
38
39 # 6. Standardizirajte kodove opažača
40 messy_data$observer <- toupper(messy_data$observer)
41 messy_data$observer <- gsub("\\.", "", messy_data$observer) #
42     ukloni točke
43
44 # Konačan pregled
45 str(messy_data)
46 summary(messy_data)

```

Zadatak 5.5: Rad s nedostajućim vrijednostima

```

1 # Identificirajte stupce s NA
2 colSums(is.na(messy_data))
3
4 # Postotak nedostajućih po stupcu
5 colMeans(is.na(messy_data)) * 100
6
7 # Vizualizacija nedostajućih vrijednosti
8 # install.packages("naniar")
9 library(naniar)
10

```

```

11  # Grafički prikaz nedostajućih
12  gg_miss_var(messy_data)
13
14  # Opcije za rješavanje:
15
16  # 1. Uklonite redove s bilo kojim NA (rigorozno)
17  clean_complete <- na.omit(messy_data)
18  nrow(messy_data)
19  nrow(clean_complete)
20
21  # 2. Uklonite samo redove gdje je brojnost NA
22  clean_brojnost <- messy_data[!is.na(messy_data$brojnost), ]
23
24  # 3. Imputacija - zamijenite NA s prosječnom vrijednošću
25  # (pazite - ovo može biti problematično!)
26  messy_data$temperatura_imputed <- messy_data$temperatura
27  messy_data$temperatura_imputed[is.na(
28    messy_data$temperatura_imputed)] <-
29  mean(messy_data$temperatura, na.rm = TRUE)
30
31  # 4. Zamijenite NA s medijanom (robustnije)
32  median_temp <- median(messy_data$temperatura, na.rm = TRUE)
33  messy_data$temperatura_imputed2 <- messy_data$temperatura
34  messy_data$temperatura_imputed2[is.na(
35    messy_data$temperatura_imputed2)] <-
36  median_temp
37
38  # VAŽNO: Uvijek dokumentirajte što ste učinili s NA
39  # vrijednostima!

```

Zadatak 5.6: Provjera i validacija podataka

```

1  # Osnovne provjere
2
3  # 1. Provjerite raspone vrijednosti
4  range(messy_data$brojnost, na.rm = TRUE)
5  range(messy_data$temperatura, na.rm = TRUE)
6
7  # 2. Provjerite kategoričke varijable
8  table(messy_data$lokacija)
9  table(messy_data$observer)
10
11  # 3. Provjera logičkih uvjeta
12  # Npr. brojnost ne može biti negativna
13  any(messy_data$brojnost < 0, na.rm = TRUE)
14
15  # 4. Provjera duplikata
16  # Postoje li dupli unosi?
17  duplicated(messy_data)
18  sum(duplicated(messy_data))
19
20  # Uklonite duplikate
21  clean_data <- messy_data[!duplicated(messy_data), ]
22
23  # 5. Detaljnija provjera duplikata po ključnim varijablama
24  duplicated(messy_data[, c("lokacija", "datum")])

```

```

25
26 # 6. Vizualna provjera outliera
27 par(mfrow = c(1, 2))
28 boxplot(messy_data$brojnost, main = "Brojnost")
29 boxplot(messy_data$temperatura, main = "Temperatura")
30 par(mfrow = c(1, 1))
31
32 # 7. Kreirajte funkciju za validaciju
33 validate_data <- function(data) {
34   cat("=== VALIDACIJA PODATAKA ===\n\n")
35
36   cat("Dimenzije:", nrow(data), "redaka,", ncol(data), "
       stupaca\n\n")
37
38   cat("Nedostajuće vrijednosti:\n")
39   print(colSums(is.na(data)))
40   cat("\n")
41
42   cat("Duplikati:", sum(duplicated(data)), "\n\n")
43
44   cat("Numeričke varijable - rasponi:\n")
45   num_cols <- sapply(data, is.numeric)
46   print(sapply(data[, num_cols], range, na.rm = TRUE))
47
48   invisible(data)
49 }
50
51 # Koristite funkciju
52 validate_data(messy_data)

```

Zadatak 5.7: Spremanje čistih podataka

```

1 # Spremite očišćene podatke
2
3 # 1. Selektirajte samo potrebne stupce
4 final_data <- messy_data[, c("lokacija", "datum_clean",
5 "brojnost", "temperatura",
6 "observer")]
7
8 # Preimenujte stupce
9 names(final_data) <- c("lokalitet", "datum", "brojnost",
10 "temperatura_C", "opazac")
11
12 # 2. Spremite kao CSV
13 write.csv(final_data,
14 "data/processed/monitoring_clean.csv",
15 row.names = FALSE)
16
17 # 3. Spremite kao RDS (R-ov format, očuva tipove)
18 saveRDS(final_data,
19 "data/processed/monitoring_clean.rds")
20
21 # 4. Spremite metapodatke
22 sink("data/processed/README.txt")
23 cat("Očišćeni monitoring podaci\n")
24 cat("Datum obrade:", as.character(Sys.Date()), "\n")

```

```

25 cat("Izvorni podaci: data/raw/monitoring_ptice.csv\n")
26 cat("\nObrada:\n")
27 cat("- Standardizirane lokacije\n")
28 cat("- Pretvoren format datuma\n")
29 cat("- Identificirana i uklonjena sumnjiva mjerenja\n")
30 cat("- Standarize irani kodovi opažača\n")
31 sink()
32
33 # 5. Učitavanje RDS datoteke
34 podaci_ucitani <- readRDS("data/processed/monitoring_clean.rds
  ")

```

1.6.4 Provjera razumijevanja

1. Koja je razlika između `read.csv` i `read.csv2`?
2. Zašto je važno provjeriti `str()` nakon učitavanja podataka?
3. Kada biste koristili imputaciju za nedostajuće vrijednosti, a kada biste uklonili redove?
4. Zašto je RDS format ponekad bolji od CSV-a?
5. Kako provjeriti postoje li dupli unosi u podacima?

1.6.5 Napredni zadatak

```

1  # Za dodatni izazov:
2  # Kreirajte potpuni pipeline za obradu podataka
3
4  clean_ecological_data <- function(input_file, output_file) {
5    # Učitajte podatke
6    data <- read.csv(input_file, stringsAsFactors = FALSE)
7
8    # Broj redaka na početku
9    n_start <- nrow(data)
10
11    # Uklonite duplikate
12    data <- data[!duplicated(data), ]
13    n_after_duplicates <- nrow(data)
14
15    # Identificirajte outliere (3 SD od prosjeka)
16    for(col in names(data)) {
17      if(is.numeric(data[[col]])) {
18        m <- mean(data[[col]], na.rm = TRUE)
19        s <- sd(data[[col]], na.rm = TRUE)
20        outliers <- abs(data[[col]] - m) > 3 * s
21        data[[col]][outliers & !is.na(outliers)] <- NA
22      }
23    }
24
25    # Uklonite redove s previše NA (>50%)
26    na_prop <- rowMeans(is.na(data))
27    data <- data[na_prop < 0.5, ]
28    n_after_na <- nrow(data)
29
30    # Spremite rezultat
31    write.csv(data, output_file, row.names = FALSE)
32
33    # Izvještaj

```

```
34     cat("=== PIPELINE ZA ČIŠĆENJE PODATAKA ===\n")
35     cat("Ulazna datoteka:", input_file, "\n")
36     cat("Početni broj redaka:", n_start, "\n")
37     cat("Nakon uklanjanja duplikata:", n_after_duplicates, "\n")
38     cat("Nakon uklanjanja redaka s previše NA:", n_after_na, "\n")
39     cat("Uklonjeno redaka:", n_start - n_after_na, "\n")
40     cat("Procenat zadržanih redaka:",
41         round(n_after_na/n_start * 100, 2), "%\n")
42
43     return(data)
44 }
45
46 # Koristite funkciju
47 # clean_data <- clean_ecological_data(
48 #     "data/raw/monitoring.csv",
49 #     "data/processed/monitoring_clean.csv"
50 # )
```

1.7 Vježba 6: Integrativna vježba - Analiza biorazno-likosti

1.7.1 Cilj vježbe

Kombinirati sve naučene vještine u jednu sveobuhvatnu analizu - od učitavanja podataka do vizualizacije i interpretacije.

1.7.2 Scenarij

Provodite istraživanje biorazno-likosti u tri hrvatska nacionalna parka kroz pet godina. Trebate analizirati trendove, usporediti lokalitete, i vizualizirati rezultate za izvještaj upravi parka.

1.7.3 Zadatak

Korak 1: Priprema podataka

```

1  # Kreirajte kompleksniji dataset
2  # (U stvarnosti biste učitali iz datoteke)
3
4  set.seed(123) # Za reproducibilnost
5
6  # Osnovni podaci
7  n_loc <- 3
8  n_god <- 5
9  n_vrsta <- 8
10
11  lokaliteti <- c("Risnjak", "Plitvice", "Velebit")
12  godine <- 2020:2024
13  vrste <- c("Canis lupus", "Lynx lynx", "Ursus arctos",
14            "Tetrao urogallus", "Aquila chrysaetos",
15            "Bombina variegata", "Salamandra salamandra",
16            "Rosalia alpina")
17
18  # Generirajte realističnije podatke
19  monitoring <- expand.grid(
20    godina = godine,
21    lokalitet = lokaliteti,
22    vrsta = vrste
23  )
24
25  # Dodajte brojnost s realističnim varijacijama
26  monitoring$brojnost <- rpois(nrow(monitoring), lambda = 15)
27
28  # Dodajte trend (povećanje brojnosti kroz godine)
29  monitoring$brojnost <- monitoring$brojnost +
30    (monitoring$godina - 2020) * sample(c(-1, 0, 1, 2),
31    nrow(monitoring),
32    replace = TRUE)
33
34  # Dodajte okolišne varijable
35  monitoring$temperatura_C <- rnorm(nrow(monitoring), mean = 10,
36    sd = 3)
37  monitoring$oborina_mm <- rnorm(nrow(monitoring), mean = 1200,
38    sd = 200)
39  monitoring$pokrov_suma_pct <- runif(nrow(monitoring), min =
40    40, max = 95)

```

```

38
39 # Dodajte neke NA vrijednosti za realizam
40 monitoring$brojnost[sample(1:nrow(monitoring), 5)] <- NA
41 monitoring$temperatura_C[sample(1:nrow(monitoring), 3)] <- NA
42
43 # Dodajte metapodatke
44 monitoring$metoda <- sample(c("transekt", "kamera", "akustika"
45 ),
46 nrow(monitoring),
47 replace = TRUE)
48 monitoring$istrazivac <- sample(c("I.H.", "M.K.", "A.P."),
49 nrow(monitoring),
50 replace = TRUE)
51
52 # Pregledajte dataset
53 head(monitoring, 10)
54 str(monitoring)
55 summary(monitoring)

```

Korak 2: Eksplorativna analiza

```

1 # 1. Osnovne statističke mjere
2 cat("=== DESKRIPTIVNA STATISTIKA ===\n\n")
3
4 # Po lokalitetu
5 cat("Prosječna brojnost po lokalitetu:\n")
6 aggregate(brojnost ~ lokalitet, data = monitoring,
7 FUN = function(x) c(mean = mean(x, na.rm = TRUE),
8 sd = sd(x, na.rm = TRUE),
9 n = sum(!is.na(x))))
10
11 # Po vrsti
12 cat("\nProsječna brojnost po vrsti:\n")
13 vrsta_stat <- aggregate(brojnost ~ vrsta, data = monitoring,
14 FUN = function(x) {
15   c(mean = mean(x, na.rm = TRUE),
16     sd = sd(x, na.rm = TRUE),
17     min = min(x, na.rm = TRUE),
18     max = max(x, na.rm = TRUE))
19 })
20 print(vrsta_stat)
21
22 # 2. Vremenski trendovi
23 cat("\n=== VREMENSKI TRENDVI ===\n\n")
24
25 trend_godina <- aggregate(brojnost ~ godina, data = monitoring
26 ,
27 FUN = mean, na.rm = TRUE)
28 print(trend_godina)
29
30 # Test linearnog trenda
31 model_trend <- lm(brojnost ~ godina,
32 data = monitoring[!is.na(monitoring$brojnost), ])
33 summary(model_trend)
34
35 # 3. Korelacija s okolišnim varijablama

```

```

35   cat("\n=== KORELACIJE ===\n\n")
36
37   cor_mat <- cor(monitring[, c("brojnost", "temperatura_C",
38   "oborina_mm", "pokrov_suma_pct")],
39   use = "complete.obs")
40   print(round(cor_mat, 3))

```

Korak 3: Vizualizacija

```

1   # Kompleksna višepanelna vizualizacija
2   png("output/figures/bioraznolikost_analiza.png",
3   width = 1400, height = 1000, res = 120)
4
5   par(mfrow = c(2, 3), mar = c(4, 4, 3, 2))
6
7   # Panel 1: Distribucija brojnosti
8   hist(monitring$brojnost,
9   main = "Distribucija brojnosti",
10  xlab = "Brojnost",
11  col = "lightblue",
12  breaks = 20)
13
14  # Panel 2: Brojnost po lokalitetima
15  boxplot(brojnost ~ lokalitet, data = monitoring,
16  main = "Brojnost po lokalitetima",
17  ylab = "Brojnost",
18  col = c("lightgreen", "lightblue", "lightyellow"))
19
20  # Panel 3: Trend kroz godine
21  trend_god <- aggregate(brojnost ~ godina, data = monitoring,
22  FUN = mean, na.rm = TRUE)
23  plot(trend_god$godina, trend_god$brojnost,
24  type = "b", pch = 19, col = "darkblue", lwd = 2,
25  main = "Trend brojnosti kroz godine",
26  xlab = "Godina", ylab = "Prosječna brojnost")
27  abline(lm(brojnost ~ godina, data = trend_god),
28  col = "red", lty = 2)
29
30  # Panel 4: Temperatura vs brojnost
31  plot(monitring$temperatura_C, monitoring$brojnost,
32  pch = 19, col = rgb(0, 0, 1, 0.3),
33  main = "Brojnost vs temperatura",
34  xlab = "Temperatura (°C)", ylab = "Brojnost")
35  abline(lm(brojnost ~ temperatura_C, data = monitoring),
36  col = "red", lwd = 2)
37
38  # Panel 5: Brojnost po vrstama (top 5)
39  vrsta_suma <- aggregate(brojnost ~ vrsta, data = monitoring,
40  FUN = sum, na.rm = TRUE)
41  vrsta_suma <- vrsta_suma[order(-vrsta_suma$brojnost), ][1:5, ]
42
43  barplot(vrsta_suma$brojnost,
44  names.arg = substr(vrsta_suma$vrsta, 1, 10),
45  las = 2, col = "steelblue",
46  main = "Top 5 vrsta (ukupna brojnost)",
47  ylab = "Ukupna brojnost")

```



```

48
49 # Panel 6: Pokrov šume vs brojnost
50 plot(monitring$pokrov_suma_pct, monitoring$brojnost,
51 pch = 19, col = as.factor(monitring$lokalitet),
52 main = "Brojnost vs pokrov šume",
53 xlab = "Pokrov šume (%)", ylab = "Brojnost")
54 legend("topright", legend = unique(monitring$lokalitet),
55 col = 1:3, pch = 19, cex = 0.8)
56
57 par(mfrow = c(1, 1))
58 dev.off()
59
60 cat("\nGrafikon spremljen: output/figures/
    bioraznolikost_analiza.png\n")

```

Korak 4: Napredna analiza s ggplot2

```

1  library(ggplot2)
2  library(dplyr)
3
4  # 1. Trend po lokalitetima s error bars
5  trend_lok <- monitoring %>%
6  group_by(godina, lokalitet) %>%
7  summarise(
8    prosjek = mean(brojnost, na.rm = TRUE),
9    sd = sd(brojnost, na.rm = TRUE),
10    n = sum(!is.na(brojnost)),
11    se = sd / sqrt(n)
12  )
13
14  p1 <- ggplot(trend_lok, aes(x = godina, y = prosjek,
15    color = lokalitet)) +
16  geom_line(linewidth = 1) +
17  geom_point(size = 3) +
18  geom_errorbar(aes(ymin = prosjek - se, ymax = prosjek + se),
19    width = 0.2) +
20  labs(title = "Trend brojnosti po lokalitetima",
21    subtitle = "Prikazano: prosjek ± SE",
22    x = "Godina",
23    y = "Brojnost",
24    color = "Lokalitet") +
25  theme_minimal() +
26  theme(legend.position = "bottom")
27
28  ggsave("output/figures/trend_lokaliteti.png", p1,
29    width = 10, height = 6, dpi = 300)
30
31  # 2. Faceted plot po vrstama
32  # Odaberite 6 najčešćih vrsta
33  top_vrste <- monitoring %>%
34  group_by(vrsta) %>%
35  summarise(ukupno = sum(brojnost, na.rm = TRUE)) %>%
36  arrange(desc(ukupno)) %>%
37  head(6) %>%
38  pull(vrsta)
39

```

```

40 monitoring_top <- monitoring %>%
41 filter(vrsta %in% top_vrste)
42
43 p2 <- ggplot(monitring_top,
44 aes(x = godina, y = brojnost, color = lokalitet)) +
45 geom_point(alpha = 0.6) +
46 geom_smooth(method = "lm", se = FALSE) +
47 facet_wrap(~ vrsta, scales = "free_y", ncol = 3) +
48 labs(title = "Trendovi po vrstama i lokalitetima",
49 x = "Godina",
50 y = "Brojnost",
51 color = "Lokalitet") +
52 theme_bw() +
53 theme(legend.position = "bottom",
54 strip.background = element_rect(fill = "lightblue"))
55
56 ggsave("output/figures/trend_vrste_facet.png", p2,
57 width = 12, height = 8, dpi = 300)
58
59 # 3. Heatmap brojnosti
60 vrsta_lok <- monitoring %>%
61 group_by(vrsta, lokalitet) %>%
62 summarise(prosjek_brojnost = mean(brojnost, na.rm = TRUE))
63
64 p3 <- ggplot(vrsta_lok, aes(x = lokalitet, y = vrsta,
65 fill = prosjek_brojnost)) +
66 geom_tile(color = "white") +
67 scale_fill_gradient(low = "lightyellow", high = "darkred") +
68 labs(title = "Prosječna brojnost po vrsti i lokalitetu",
69 fill = "Prosječna\nbrojnost") +
70 theme_minimal() +
71 theme(axis.text.x = element_angle(45, hjust = 1))
72
73 ggsave("output/figures/heatmap_brojnosti.png", p3,
74 width = 10, height = 8, dpi = 300)

```

Korak 5: Generiranje izvještaja

```

1 # Generirajte tekstualni izvještaj
2
3 sink("output/izvjestaj_bioraznolikost.txt")
4
5 cat("=rep(=", 60), "\n")
6 cat("          IZVJEŠTAJ O MONITORINGU BIORAZNOLIKOSTI\n")
7 cat("=rep(=", 60), "\n\n")
8
9 cat("Datum izvještaja:", as.character(Sys.Date()), "\n")
10 cat("Autor: Ekološki tim\n\n")
11
12 cat("1. OSNOVNI PODACI\n")
13 cat(strrep("-", 60), "\n")
14 cat("Lokaliteti:", paste(unique(monitring$lokalitet),
15 collapse = ", "), "\n")
16 cat("Period:", min(monitring$godina), "-",
17 max(monitring$godina), "\n")
18 cat("Broj vrsta:", length(unique(monitring$vrsta)), "\n")

```

```

19 cat("Ukupno opažanja:", nrow(monitored), "\n")
20 cat("Opažanja s podacima:", sum(!is.na(monitored$brojnost)),
    "\n\n")
21
22 cat("2. OPĆA BROJNOST\n")
23 cat(strrep("-", 60), "\n")
24 cat("Prosječna brojnost:",
25 round(mean(monitored$brojnost, na.rm = TRUE), 2), "\n")
26 cat("Medijan brojnosti:",
27 median(monitored$brojnost, na.rm = TRUE), "\n")
28 cat("Standardna devijacija:",
29 round(sd(monitored$brojnost, na.rm = TRUE), 2), "\n")
30 cat("Raspon:", min(monitored$brojnost, na.rm = TRUE), "-",
31 max(monitored$brojnost, na.rm = TRUE), "\n\n")
32
33 cat("3. BROJNOST PO LOKALITETIMA\n")
34 cat(strrep("-", 60), "\n")
35 lok_sum <- aggregate(brojnost ~ lokalitet, data = monitored,
36 FUN = function(x) {
37   c(n = sum(!is.na(x)),
38     mean = mean(x, na.rm = TRUE),
39     sd = sd(x, na.rm = TRUE))
40 })
41 print(lok_sum)
42 cat("\n")
43
44 cat("4. VREMENSKI TREND\n")
45 cat(strrep("-", 60), "\n")
46 model <- lm(brojnost ~ godina,
47 data = monitored[!is.na(monitored$brojnost), ])
48 cat("Linearni model: brojnost ~ godina\n")
49 cat("Nagib:", round(coef(model)[2], 3), "\n")
50 cat("P-vrijednost:", format.pval(summary(model)$coefficients
51 [2, 4]), "\n")
52 if(summary(model)$coefficients[2, 4] < 0.05) {
53   if(coef(model)[2] > 0) {
54     cat("Interpretacija: Statistički značajan PORAST brojnosti
55       kroz godine.\n")
56   } else {
57     cat("Interpretacija: Statistički značajan PAD brojnosti
58       kroz godine.\n")
59   }
60 } else {
61   cat("Interpretacija: Nema statistički značajnog trenda.\n")
62 }
63 cat("\n")
64
65 cat("5. TOP 5 NAJČEŠĆIH VRSTA\n")
66 cat(strrep("-", 60), "\n")
67 top5 <- aggregate(brojnost ~ vrsta, data = monitored,
68 FUN = sum, na.rm = TRUE)
69 top5 <- top5[order(-top5$brojnost), ][1:5, ]
70 for(i in 1:nrow(top5)) {
71   cat(i, ". ", top5$vrsta[i], ": ", top5$brojnost[i], "\n",
72     sep = "")
73 }

```

```

70   cat("\n")
71
72   cat("6. KORELACIJE S OKOLIŠNIM VARIJABLAMA\n")
73   cat(strrep("-", 60), "\n")
74   cor_temp <- cor.test(monitring$brojnost,
75   monitoring$temperatura_C,
76   use = "complete.obs")
77   cat("Brojnost vs Temperatura:\n")
78   cat("  r =", round(cor_temp$estimate, 3), "\n")
79   cat("  p =", format.pval(cor_temp$p.value), "\n\n")
80
81   cor_oborine <- cor.test(monitring$brojnost,
82   monitoring$oborina_mm,
83   use = "complete.obs")
84   cat("Brojnost vs Oborine:\n")
85   cat("  r =", round(cor_oborine$estimate, 3), "\n")
86   cat("  p =", format.pval(cor_oborine$p.value), "\n\n")
87
88   cat("7. ZAKLJUČCI I PREPORUKE\n")
89   cat(strrep("-", 60), "\n")
90   cat("- Analiza obuhvaća", length(unique(monitring$lokalitet))
91   ,
92   "lokaliteta kroz", length(unique(monitring$godina)), "godina\
93   n")
94   cat("- Praćeno je", length(unique(monitring$vrsta)),
95   "različitih vrsta\n")
96
97   if(summary(model)$coefficients[2, 4] < 0.05) {
98     cat("- Utvrđen je statistički značajan trend brojnosti\n")
99   } else {
100     cat("- Brojnost je relativno stabilna kroz promatrano
101     razdoblje\n")
102   }
103
104   cat("- Preporučuje se nastavak monitoringa\n")
105   cat("- Potrebna je dodatna analiza utjecaja klimatskih
106   varijabli\n")
107
108   cat("\n", strrep("=", 60), "\n")
109   cat("KRAJ IZVJEŠTAJA\n")
110   cat(strrep("=", 60), "\n")
111
112   sink()
113
114   cat("\nIzvjestaj spremljen: output/izvjestaj_bioraznolikost.
115   txt\n")

```

1.7.4 Završni zadatak

Sada samostalno analizirajte podatke i odgovorite:

1. Koji lokalitet pokazuje najveći porast brojnosti kroz godine?
2. Koja vrsta ima najveću varijabilnost u brojnosti?
3. Postoji li sezonski obrazac (ako dodamo informaciju o mjesecu)?
4. Kako biste poboljšali ovaj monitoring program?

5. Koje dodatne varijable biste mjerili?

1.7.5 Proširenje vježbe

Za dodatni izazov:

```

1  # 1. Dodajte analizu raznolikosti
2  # install.packages("vegan")
3  library(vegan)
4
5  # Transformirajte podatke u matricu za vegan
6  zajednica <- monitoring %>%
7  group_by(lokalitet, vrsta) %>%
8  summarise(brojnost_ukupno = sum(brojnost, na.rm = TRUE)) %>%
9  tidyr::pivot_wider(names_from = vrsta,
10 values_from = brojnost_ukupno,
11 values_fill = 0)
12
13 zajednica_mat <- as.matrix(zajednica[, -1])
14 rownames(zajednica_mat) <- zajednica$lokalitet
15
16 # Shannonomov indeks raznolikosti
17 shannon <- diversity(zajednica_mat, index = "shannon")
18 simpson <- diversity(zajednica_mat, index = "simpson")
19
20 cat("\nIndeksi raznolikosti po lokalitetima:\n")
21 print(data.frame(
22 Lokalitet = names(shannon),
23 Shannon = round(shannon, 3),
24 Simpson = round(simpson, 3)
25 ))
26
27 # 2. NMDS ordinacija
28 nmbs <- metaMDS(zajednica_mat, distance = "bray")
29
30 plot(nmbs$points,
31 pch = 19,
32 cex = 2,
33 col = 1:3,
34 main = "NMDS ordinacija lokaliteta")
35 text(nmbs$points,
36 labels = rownames(zajednica_mat),
37 pos = 3)
38
39 # 3. Rarefaction krivulje
40 rarecurve(zajednica_mat,
41 col = 1:3,
42 main = "Rarefaction krivulje")
43 legend("bottomright",
44 legend = rownames(zajednica_mat),
45 col = 1:3,
46 lty = 1)

```

1.8 Zaključak i daljnji koraci

1.8.1 Što ste naučili

Kroz ovih šest vježbi savladali ste:

1. **Osnove R-a:** varijable, tipovi podataka, operatori
2. **Vektore:** rad s nizovima podataka, statističke funkcije
3. **Data frame-ove:** tablice podataka, filtriranje, sortiranje
4. **Vizualizaciju:** histogrami, scatter plotovi, box plotovi, ggplot2
5. **Učitavanje i čišćenje podataka:** CSV, Excel, rad s NA vrijednostima
6. **Integrativnu analizu:** kompletan workflow od podataka do izvještaja

1.8.2 Vježbovne navike

Za uspješno učenje R-a:

- **Vježbajte redovno** – 30 minuta dnevno bolje je od 5 sati jednom tjedno
- **Pišite kod vlastoručno** – ne samo kopirajte-zalijepite
- **Eksperimentirajte** – mijenjajte brojeve, probajte nove funkcije
- **Čitajte dokumentaciju** – `?function_name` je vaš prijatelj
- **Koristite Google/Stack Overflow** – gotovo sigurno netko je imao istu grešku
- **Radite s pravim podacima** – vaši vlastiti projekti najbolje motiviraju

1.8.3 Korisni resursi

Online tečajevi

- **DataCamp:** Introduction to R (besplatni uvodni tečaj)
- **Coursera:** R Programming (Johns Hopkins University)
- **edX:** Data Science: R Basics (Harvard)

Knjige (besplatne online)

- *R for Data Science* – Hadley Wickham (<https://r4ds.had.co.nz/>)
- *Advanced R* – Hadley Wickham (<https://adv-r.hadley.nz/>)
- *Hands-On Programming with R* – Garrett Grolemund

Ekološki specifični resursi

- *Numerical Ecology with R* – Borcard et al.
- *Mixed Effects Models and Extensions in Ecology with R* – Zuur et al.
- *Ecological Models and Data in R* – Bolker
- **CRAN Task View: Environmetrics** (<https://cran.r-project.org/web/views/Environmetrics.html>)

Pomoć i zajednica

- **Stack Overflow:** <https://stackoverflow.com/questions/tagged/r>
- **RStudio Community:** <https://community.rstudio.com/>
- **R-bloggers:** <https://www.r-bloggers.com/>
- **Twitter #rstats:** aktivna zajednica R korisnika

1.8.4 Sljedeći koraci

Nakon ovih vježbi, spremni ste za:

1. **Naprednije statističke metode:**
 - Linearna i višestruka regresija
 - ANOVA i post-hoc testovi

- Generalizirani linearni modeli (GLM)
- Modeli miješanih efekata (GLMM)

2. Specijalizirane ekološke analize:

- Multivarijatne analize (PCA, NMDS, RDA)
- Analiza raznolikosti
- Prostorne analize
- Modeli distribucije vrsta (SDM)

3. Naprednije programiranje:

- Pisanje vlastitih funkcija
- Automatizacija workflow-a
- Pakiranje koda u R pakete
- Shiny aplikacije za interaktivne analize

1.8.5 Zadnje riječi

Učenje R-a (ili bilo kojeg programskog jezika) je maraton, ne sprint. Bit će frustrirajućih trenutaka kada kod ne radi, greške nemaju smisla, i čini vam se da nikad nećete razumjeti. To je normalno i prođe kroz to svi, čak i iskusni programeri.

Ključ je u ustrajnosti. Svaki put kada riješite problem, naučite nešto novo. Svaki greška koju ispravite učvršćuje vaše razumijevanje. S vremenom, ono što je bilo teško postaje lako, a ono što je bilo nemoguće postaje rutina.

R je moćan alat koji će vam omogućiti da radite analizu koje prije niste mogli, da postavite pitanja koja prije nisu bila moguća, i da doprinesete ekološkoj znanosti na način koji nije bio dostupan prethodnim generacijama ekologa.

Sretno na vašem putovanju kroz kvantitativnu ekologiju!

Povratne informacije: Ako imate pitanja, komentare ili prijedloge za poboljšanje ovih vježbi, kontaktirajte nastavnika ili pošaljite email na [email].

