

```
#####  
# 1. DEO – UVOD #  
#####  
  
### Jednostavne operacije  
  
4+5+6  
sum(4,5,6)  
  
### Vektori i drugi objekti. Uporedite:  
  
o1 = 4+5+6  
v1 = c(4,5,6)  
  
o1*2  
v1*2  
  
## Unos naziva objekta daje njegovu vrednost  
  
o1  
v1  
  
### Matematički i logički operatori  
  
+ - * /  
== > < >= <=  
  
& # i  
| # ili  
! # negacija (!= znači 'nije jednako')  
  
### Učitavanje eksternih dokumenata  
  
## Provera radnog direktorijuma  
  
getwd()  
  
## Tipično se dobija nešto poput:  
## "C:/Documents and Settings/Administrator/My Documents" (Windows)  
## "/Users/Maja" (Mac)  
  
## Napomena za Windows: putanje u Windows Explorer-u sadrže \ (npr. C:  
  \Documents and Settings\Administrator), ali u R se mora uneti / ili \  
  
## Eventualna promena postiže se komandom setwd("putanja/direktorijum")  
## Dodatni način promene radnog direktorijuma je putem menija File (Windows)  
  ili Misc (Mac) > Change dir... / Change Working Directory  
  
### Učitavanje podataka (iz radnog direktorijuma)  
  
reldi1 = read.delim("reldi1.tsv")  
refl = read.csv("reflexives.csv")  
  
## Možete koristiti i opciju koja otvara iskačući prozor za izbor dokumenta:  
  
reldi1 = read.delim(file.choose())  
refl = read.csv(file.choose())  
  
## Pregled podataka i njihovih osobina:
```

```
head(reldi1) # vrh tabele (predefinisana opcija je prikaz 6 redova)
tail(reldi1) # dno tabele
head(refl,n=10) # podešavanje prikaza prvih n redova tabele
head(refl,20)

summary(reldi1) # osnovne statističke mere za svaku varijablu

str(reldi1) # prikaz vrsta varijabli i početka niza njihovih vrednosti

reldi1$duzina.recenica.2=as.factor(reldi1$duzina.recenica.2) # pretvaranje
  varijable iz numeričke u kategoričku (faktor)

names(reldi1) # prikaz naziva varijabli

## Operacije sa pojedinačnim varijablama

levels(reldi1$korpus) # prikaz naziva nivoa varijabli

sum(refl$Cloze)
sum(refl$Cloze[refl$Group=="Control-I"])

## Ponavljanje operacija na više podskupova podataka

tapply(refl$Cloze, refl$Group, sum) # ponavljanje operacije sabiranja za
  svaku grupu posebno
by(refl$Cloze, refl$Group, sum) # takođe ponavljanje operacije, drugačiji
  format rezultata

### Dodavanje, brisanje i kopiranje podataka

## Dodavanje nove kolone u podatke

reldi1$tipovi = reldi1$tokeni*reldi1$ttr # računanje broja tipova
head(reldi1)

## Brisanje kolone iz podataka

reldi1$tipovi = NULL
head(reldi1)

## Izdvajanje varijabli ili redova

reldi1a = subset(reldi1, select="tokeni") # formiranje novog objekta sa jednom
  kolonom
reldi1b = subset(reldi1, select=c("tokeni","ttr")) # formiranje novog objekta
  sa dve kolone

reldi1c = subset(reldi1, tokeni>500) # formiranje novog objekta sa redovima u
  kojima je vrednost varijable tokeni viša od 500

## Izdvajanje slučajnog manjeg uzorka

reldi1mini = reldi1[sample(1:nrow(reldi1), 100, replace=FALSE),] # izdvaja
  slučajni uzorak od ukupno 100 tekstova

## Izvoz objekata van R-a

write.table(reldi1mini, "reldi1mini.txt", sep="\t") # izvoz tabele odvojene
  tabulatorima; ako se ne zada putanja, R će sačuvati dokument u radnom
  direktorijumu
```

```
### "Dugački" i "široki" prikaz podataka
```

```
## Iako se uobičajeno kaže da u R-u jedna kolona predstavlja jednu varijablu,
a jedan red jednog ispitanika/jedan tekst/..., to se zapravo odnosi pre
svega na nezavisne uzorke. Kod zavisnih uzoraka jedan način unosa podataka
podrazumeva da se podaci za istog ispitanika/isti tekst... kodiraju u više
redova. Drugi način unosa podataka jeste da svi podaci istog ispitanika/
teksta... budu u jednom redu, ali tada jedna kolona ne predstavlja nužno
jednu varijablu, već može predstavljati vrednosti zavisne varijable za
jedan nivo nezavisne varijable (N.B. statistički programi tipično nazivaju
kolone varijablama, što ne odgovara nužno pojmu varijable u teorijsko-
metodološkom smislu, gde je bitno razlikovati varijablu i njene nivoe).
Prvi način unosa podataka predstavlja "dugački" (long), a drugi
"široki" (wide) format. Neke analize zahtevaju jedan, a neke drugi format.
Za transformaciju se unutar R-a može koristiti funkcija reshape.
```

```
head(refl) # podaci u objektu refl imaju široki format - za svakog ispitanika
postoji jedan red; nazivi nivoa varijabli koje se ispituju na zavisnim
uzorcima (vrsta glagola / vrsta refleksivnog markera) u trenutnom obliku
nisu čitljivi za funkciju reshape, gde je potrebno da brojevi stoje na
kraju; nazivi se mogu izmeniti u csv dokumentu i ponovo učitati, ili
izmeniti u R-u funkcijom names.
```

```
names(refl)[names(refl)=="Reflc"] = "Reflc1" # primer promene naziva za
jedan nivo
```

```
refl2=reshape(refl, varying=c("Reflc1","Reflc2","Reflc3",
"Reflp1","Reflp2","Reflp3"), direction="long", idvar=c("Subject","Group"),
sep="") # zadaju se kolone koje treba transformisati, ciljni format,
varijable prema kojima će izvršiti grupisanje u dugačkom formatu i
separator koji odvađa brojeve u nazivima varijabli od samih naziva;
detaljnije na https://www.r-bloggers.com/converting-a-dataset-from-wide-to-long/
```

```
head(refl2) # novi raspored podataka, u dugačkom formatu
```

```
names(refl2)[names(refl2)=="time"] = "VerbGroup" # promena automatski
dodeljenog naziva varijable "time" u "VerbGroup"
```

```
refl4$VerbGroup=as.factor(refl4$VerbGroup) # pretvaranje varijable VerbGroup
iz numeričke u kategoričku (faktor)
```

```
#####
# 2a. DEO - DESKRIPTIVNE MERE #
#####
```

```
### Deskriptivne mere
```

```
## Mere učestalosti
```

```
table(reldi1$korpus) # daje učestalost vrednosti kategoričke varijable,
ovde broj tekstova prema korpusima
table(refl$Group)
```

```
table(reldi1$korpus=="srwac.lat")
```

```
table(reldi1$korpus=="srwac.lat" & reldi1$duzina.recenica.2==1) # obratite
pažnju na navodnike - potrebni su za kategorije, za numeričke vrednosti se
ne navode
```

```
table(reldi1$korpus=="srwac.lat" & reldi1$duzina.recenica.2!=1)
```

```
# Crosstabs (kontingencijske ili tabele unakrsne klasifikacije)
```

```

duzina.recenica.2.po.korpusima=table(reldi1$korpus,reldi1$duzina.recenica.2)
leks.gust.2.po.korpusima=xtabs(~reldi1$korpus+reldi1$leks.gust.2)

# "Ručno" formiranje tabele (primer sa slajda)

infinitivi = matrix(c(24484790, 199792951, 4608002, 79296848), ncol=2,
  byrow=TRUE)
colnames(infinitivi) = c("Infinitivi", "Ostalo")
rownames(infinitivi) = c("hrWaC", "srWaC")
infinitivi = as.table(infinitivi)

# Relativne frekvencije

leks.gust.2.po.korpusima.prop = prop.table(leks.gust.2.po.korpusima,1) # 1
  označava računanje procenata prema redovima (2 je prema kolonama, 0 prema
  celoj tabeli)

## Mere centralne tendencije, pozicione mere i mere disperzije

mean(reldi1$tokeni)
sd(reldi1$tokeni)
var(reldi1$tokeni)

median(refl$Cloze)
IQR(refl$Cloze)
quantile(refl$Cloze)

min(refl$Cloze)
max(refl$Cloze)

# R nema predefinisane funkcije za mod

summary(reldi1$tokeni)

library(pastecs)
stat.desc(refl$Cloze)

### Deskriptivne mere izračunate na podskupovima podataka i ponovljene na više
  podskupova

mean(reldi1$tokeni[reldi1$korpus=="hrwac"])
median(refl$Refl3p[refl$Group=="Control-I"])

tapply(reldi1$tokeni, reldi1$korpus, mean)
tapply(refl$Refl3p, refl$Group, median)

#####
# 2b. DEO – GRAFIČKI PRIKAZ #
#####

### Osnovni tipovi grafikona

## Scatterplot

plot(reldi1$tokeni, reldi1$recenice)
plot(reldi1$leks.gust, reldi1$ceste.reci)
plot(reldi1$ceste.reci, reldi1$ttr)

## Histogram

```

```
hist(reldi1$recenice)
hist(reldi1$recenice[reldi1$korpus=="hrwac"])
hist(reldi1$ttr)

histogram1=hist(refl$Cloze[refl$Group=="Control-I"])
histogram1 # pored grafikona dobijaju se i dodatne informacije (npr. o tome
           kako su podaci grupisani u raspone)

## Boxplot

boxplot(reldi1$recenice~reldi1$korpus)
boxplot(reldi1$leks.gust~reldi1$korpus)

boxplot(refl$Cloze~refl$Group)

## Mosaic plot

mosaicplot(duzina.recenica.2.po.korpusima)
mosaicplot(leks.gust.2.po.korpusima)

mosaicplot(infinitivi)

## Bar plot

barplot(duzina.recenica.2.po.korpusima, beside=TRUE) # predefinisana opcija je
           "beside=FALSE", tada su potkategorije jedne iznad drugih (stacked columns)

refl.Cloze = tapply(refl$Cloze, refl$Group, mean) # računanje prosečnog
           broja bodova za svaku grupu i čuvanje rezultata u novom objektu
           "refl.Cloze"
barplot(refl.Cloze)

### Podešavanje opcija za grafikone

## Pregled trenutnih podešavanja

par()

## Primer: promena boje naziva osa grafikona; globalna promena predefinisane
           vrednosti (biće primenjena do kraja sesije ili do sledeće promene)

par(col.lab="red")

## Ista promena može se primeniti i na pojedinačne grafikone unutar funkcije
           za crtanje

hist(reldi1$tokeni, col.lab="red")

## Drugi parametri koji se mogu podesiti iz funkcije za crtanje (ovde je dat
           pregled, nemojte kopirati same parametre u R, potrebno ih je zadati u
           okviru funkcija za crtanje grafikona)

main="Naslov" # naslov celog grafikona

xlab="Naziv X ose"
ylab="Naziv y ose"

xlim=c(xmin, xmax) # min i max vrednosti prikazane na X osi
ylim=c(ymin, ymax) # min i max vrednosti prikazane na Y osi
```

```

## Definisiranje boja centralnog dela grafikona (stubića, linija...); potrebno
  je navesti onoliko boja koliko ima stubića (i sl.), inače će boje početi
  da se ponavljaju; detaljan pregled boja i njihovih naziva: http://
  research.stowers-institute.org/efg/R/Color/Chart/ColorChart.pdf

col=c("orchid","turquoise")

## Izbor simbola za skater i linijske dijagrame, npr. 0=kvadrat, 1=krug
  (predefinisana opcija); više vrednosti može se videti na: http://
  www.statmethods.net/advgraphs/parameters.html

pch=0,1,2,3...

## Izbor vrste linije, 0=bez linije, 1=puna linija (predefinisana opcija),
  2=crtice, 3=tačke, 4=tačke i crtice, 5=duže crtice, 6=dve crtice

lty=0,1,2,3...

### Prikaz više grafikona u jednom panelu

par(mfrow=c(nrows, ncols)) # popunjavanje matrice grafikona po redovima,
  zadaju se i broj redova i broj kolona
par(mfcol=c(nrows, ncols)) # popunjavanje matrice grafikona po kolonama,
  zadaju se i broj redova i broj kolona

### Primena gornjih parametara (plus neke dodatne opcije)

## Scatterplot

plot(reldi1$tokeni, reldi1$recenice, col="turquoise", main="Broj tokena i broj
  rečenica", pch=4)
abline(lm(reldi1$recenice ~ reldi1$tokeni)) # trendline (linija koja najbolje
  opisuje podatke)

## Histogrami

hist(reldi1$recenice, col="lightblue", xlab="Broj rečenica", main="Raspodela
  broja rečenica u tekstu")

# Nekoliko važnih napomena za histograme:
# (1) Koriste se za prikaz učestalosti različitih (raspona) vrednosti
  kontinuiranih varijabli; za frekvencije kategoričkih varijabli koriste se
  grafikoni sa stubićima (gde između stubića postoji razmak)
# (2) Na histogramima se ne prikazuje učestalost pojedinačnih vrednosti, već
  raspona vrednosti (npr. 10-19, 20-29... ili 0.25-0.29, 0.30-0.34...)
# (3) Apsolutna frekvencija se može prikazati samo ukoliko su svi rasponi
  jednake širine; R automatski pravi raspodelu na jednake raspone, ali
  koristeći argument "breaks" mogu mu se zadati i različiti rasponi
# (4) Kada su rasponi različite širine, na Y osi se prikazuje gustina
  frekvencije (frequency density), koja se dobija formulom apsolutna
  frekvencija / ukupan broj opservacija * širina raspona, i čije množenje
  širinom raspona daje relativnu frekvenciju; prikaz gustine verovatnoće
  moguć je i kod jednakih raspona zadavanjem argumenta "freq=FALSE"
# (5) Na Y osi može se prikazati i relativna frekvencija, ali u R-u se mora
  "zaobilazno" računati (ne postoji kao opcija u okviru funkcije hist)
# (6) Sve tri opcije koje se mogu prikazati na Y osi manje ili više direktno
  odražavaju verovatnoću pojavljivanja vrednosti iz određenog raspona
  (relativna frekvencija je najdirektniji pokazatelj)

```

```

hist(reldi1$ttr, freq=FALSE)
lines(density(reldi1$ttr)) # dodavanje krive (density curve)

plot(density(reldi1$ttr)) # prikaz same krive je kernel density plot
  (detaljnije: https://www.r-bloggers.com/exploratory-data-analysis-combining-histograms-and-density-plots-to-examine-the-distribution-of-the-ozone-pollution-data-from-new-york-in-r/)

## Preklapanje više histograma (u svrhu poređenja više distribucija)

hist(reldi1$leks.gust[reldi1$korpus=="srwac.cyr"], col=rgb(0,0,1,0.5))
hist(reldi1$leks.gust[reldi1$korpus=="srwac.lat"], col=rgb(1,0,0,0.5), add=T)
  # definisanje boja pomoću rgb sistema, daje mogućnost definisanja
  transparentnog preklopljenog dela histograma

## Mosaic plots

mosaicplot(duzina.recenica.2.po.korpusima, col=c("tomato3", "gold1",
  "olivedrab3", "plum3", "turquoise"), main="Dužina rečenica po korpusima",
  xlab="Korpus", ylab="Dužina rečenica")

## Box plots

boxplot(reldi1$ceste.rec[reldi1$korpus=="hrwac"], col=c("purple"))
boxplot(reldi1$ceste.rec[reldi1$korpus=="hrwac"]~reldi1$ceste.rec.
  2[reldi1$korpus=="hrwac"], col=c("red","blue","green","yellow"))
boxplot(reldi1$ceste.rec~reldi1$korpus,
  col=c("mediumvioletred","royalblue3","limegreen"), notch=TRUE) #
  "useci" (notches) označavaju interval pouzdanosti medijane na nivou
  pouzdanosti 95% (što znači da očekujemo da u različitim uzorcima iz iste
  populacije dati interval u 95% slučajeva sadrži medijanu populacije); ako
  se useci za različite kategorije ne preklapaju, vrlo verovatno između
  njihovih medijana postoji značajna razlika

boxplot(refl$Cloze~refl$Group, notch=TRUE)

boxplot(refl$Cloze~refl$Group, col=c(rainbow(5)), notch=TRUE)
grid() # dodaje koordinatnu mrežu

## Kombinovanje više grafikona u jedan panel

par(mfrow=c(1,2))
boxplot(refl$Cloze~refl$Group)
grid()
boxplot(refl$Cloze~refl$Group, col=c(rainbow(5)))
grid()
par(mfrow=c(1,1)) # povratak na crtanje pojedinačnih grafikona

#####
# 3. DEO - TESTOVI #
#####

### Chi-square test (hi-kvadrat)

chisq.test(duzina.recenica.2.po.korpusima) # hi-kvadrat na već formiranoj
  tabeli
chisq.test(table(reldi1$korpus, reldi1$duzina.recenica.2)) # formiranje
  tabele unutar funkcije za hi-kvadrat

```

```

summary(duzina.recenica.2.po.korpusima) # ako je input kontingencijska tabela,
funkcija summary daje i hi-kvadrat test
summary(table(reldi1$korpus,reldi1$duzina.recenica.2))

## Za tabele u kojima je neka od frekvencija <5 hi-kvadrat test će prijaviti
da rezultat nije pouzdan; u takvim slučajevima treba koristiti Fisher's
exact test – fisher.test(), pod uslovom da preostale frekvencije nisu
izuzetno visoke – u tom slučaju će R verovatno prijaviti da nema dovoljno
memorije za test

## Veličina efekta (effect size) – kvantifikuje stepen u kome se rezultat
dobijen na uzorku razlikuje od očekivanja zasnovanih na nultoj hipotezi.
Za hi-kvadrat se kao mera veličine efekta obično koristi koeficijent  $\phi$ 
( $\phi$ , za tabele 2x2) ili Kramerov V koeficijent (Cramer's V, za ostale
tabele). Hi-kvadrat test prvo treba sačuvati kao objekat, a zatim na taj
objekat primeniti formulu za  $\phi$ /Cramer's V (formula preuzeta iz Gries 2013:
186).

rezultat.duzina.recenica.2=chisq.test(duzina.recenica.2.po.korpusima)
sqrt(rezultat.duzina.recenica.2$statistic/sum(duzina.recenica.
2.po.korpusima)*(min(dim(duzina.recenica.2.po.korpusima))-1)) # obratite
pažnju na to da su u formuli zastupljeni i objekat u kome je sačuvan
rezultat i početna tabela

## Uobičajena procena dobijenog rezultata jeste da koeficijenti čija je
vrednost niska (približno 0.10) označavaju mali efekat, koeficijenti oko
0.30 osrednji efekat, a koeficijenti oko 0.50 jak efekat. Međutim, u
različitim disciplinama tumačenja se mogu razlikovati, zbog razlika u
vrsti podataka i tipičnim rezultatima koji se u oblasti dobijaju, pa treba
dodatno potražiti tumačenje vrednosti u već objavljenim radovima.

### Test normalnosti raspodele: Shapiro-Wilk test

shapiro.test(reldi1$tokeni) # ako je rezultat statistički značajan, raspodela
NIJE normalna

tapply(reldi1$tokeni,reldi1$korpus,shapiro.test) # kada postoji više
uzoraka, test normalnosti se radi za svaki uzorak posebno
tapply(refl$Cloze,refl$Group,shapiro.test)

### Provera jednakosti varijanse: Ansari-Bradley test

ansari.test(refl$Cloze[refl$Group=="SerLI-I"],refl$Cloze[refl$Group=="SerUI-
I"]) # ako je rezultat statistički značajan, varijanse u poređenim
uzorcima NISU jednake

### Korelacija

cor.test(reldi1$tokeni, reldi1$recenice, method="spearman")

cor.test(reldi1$duzina.recenica, reldi1$leks.gust, method="spearman")
cor.test(reldi1mini$duzina.recenica, reldi1mini$leks.gust, method="spearman")
# uporedite vrednost p za test urađen na 1500 i 100 testova

cor.test(reldi1$duzina.recenica[reldi1$korpus=="hrwac"],
reldi1$leks.gust[reldi1$korpus=="hrwac"], method="spearman")
cor.test(reldi1mini$duzina.recenica[reldi1mini$korpus=="hrwac"], reldi1mini
$leks.gust[reldi1mini$korpus=="hrwac"], method="spearman")

### T-testovi i Wilcoxon testovi

```



```

t.test(refl$Cloze[refl$Group=="SerLI-I"], refl$Cloze[refl$Group=="SerUI-I"], var.equal=TRUE)
t.test(refl$Cloze[refl$Group=="SerLI-I" | refl$Group=="SerUI-I"] ~ refl$Group[refl$Group=="SerLI-I" | refl$Group=="SerUI-I"], var.equal=TRUE)
# predefinisana opcija podrazumeva nejednake varijanse i dvosmerni test

## Veličina efekta za t-testove može se iskazati kao koeficijent korelacije r. Koeficijent se dobija formulom  $\sqrt{t^2/(t^2+df)}$  (Field et al. 2012: 384-385); t i df se mogu preuzeti iz prikaza rezultata testa ili izdvojiti iz objekta u kome je test sačuvan.

ttest.cloze=t.test(refl$Cloze[refl$Group=="SerLI-I"], refl$Cloze[refl$Group=="SerUI-I"], var.equal=TRUE)
t=ttest.cloze$statistic[[1]]
df=ttest.cloze$parameter[[1]]
sqrt(t^2/(t^2+df))

## Rezultat se tumači isto kao Kramerov koeficijent V

wilcox.test(reldi1$tokeni[reldi1$korpus=="srwac.cyr"], reldi1$tokeni[reldi1$korpus=="srwac.lat"])
wilcox.test(reldi1$tokeni[reldi1$korpus=="srwac.cyr" | reldi1$korpus=="srwac.lat"] ~ reldi1$korpus[reldi1$korpus=="srwac.cyr" | reldi1$korpus=="srwac.lat"])

wilcox.test(refl$Refl1c[refl$Group=="SerLI-I"], refl$Refl3c[refl$Group=="SerLI-I"], paired=TRUE)

## Veličina efekta za Vilkoksonove testove takođe se može iskazati kao koeficijent korelacije r. Prvo treba kopirati u R funkciju za rFromWilcox (iz Field et al. 2012: 665), zatim sačuvati Vilkoksonov test kao objekat i izračunati broj opservacija, i najzad primeniti funkciju rFromWilcox.

rFromWilcox = function(wilcoxModel, N) {
  z = qnorm(wilcoxModel$p.value/2)
  r = z/sqrt(N)
  cat("Effect Size, r =", r)}
wilcox_tokeni =
  wilcox.test(reldi1$tokeni[reldi1$korpus=="srwac.cyr"], reldi1$tokeni[reldi1$korpus=="srwac.lat"])
length(reldi1$tokeni[reldi1$korpus=="srwac.cyr"])
+length(reldi1$tokeni[reldi1$korpus=="srwac.lat"])
rFromWilcox(wilcox_tokeni, 1000) # drugi argument je broj opservacija, dobijen u prethodnom koraku

## Rezultat se tumači isto kao Kramerov koeficijent V

### Jednofaktorska ANOVA, Kruskal-Wallis test i Friedman test

## Za nezavisnu jednofaktorsku analizu varijanse mogu se koristiti funkcije oneway.test ili aov; aov je opštija funkcija koja se može koristiti i za druge vrste analize varijanse; detaljnija objašnjenja: https://ww2.coastal.edu/kingw/statistics/R-tutorials/oneway.html

oneway.test(refl$Cloze[refl$Group!="Control-I"] ~ refl$Group[refl$Group!="Control-I"]) # pretpostavlja nejednake varijanse
oneway.test(refl$Cloze[refl$Group!="Control-I"] ~ refl$Group[refl$Group!="Control-I"], var.equal=TRUE) # jednake varijanse

```

```

anova1=aov(refl$Cloze[refl$Group!="Control-I"]~refl$Group[refl$Group!
  ="Control-I"]) # pretpostavlja jednake varijanse
summary(anova1)      # prikaz rezultata
TukeyHSD(anova1)    # posthoc testovi za poređenje parova grupa
summary.lm(anova1)  # izlistavanje dodatnih parametara

## Veličina efekta se može dobiti kao koeficijent r korenovanjem - sqrt() -
vrednosti R^2 ("Multiple R-squared"), koje se dobija funkcijom
summary.lm(). Pouzdanija opcija je računanje koeficijenta omega-kvadrat
( $\omega^2$ ) po formuli (SumSqFaktor - dfFaktor * MeanSqResiduals)/(SumSqFaktor +
SumSqResiduals + MeanSqResiduals), unošenjem vrednosti iz tabele koju daje
summary(anova1), gde bi u ovom primeru faktor bila grupa (refl$Group[refl
$Group != "Control-I"]). Korenovanjem se dobija  $\omega$ , koje se može tumačiti
slično kao r.

## Za ponovljenu analizu varijanse (repeated measures) potreban je dugački
format podataka i neophodno je u formuli za funkciju specificovati da se
radi o zavisnim uzorcima (tako što će biti uneta greška, odnosno rezidual
koji održava činjenicu da postoji više podataka istih ispitanika);
detaljnija objašnjenja: https://ww2.coastal.edu/kingw/statistics/R-
tutorials/repeated.html

## N.B. primer dat ispod predstavlja samo ilustraciju procedure, u pitanju su
ordinalni podaci i na njima NE treba raditi analizu varijanse

anova2 = aov(refl2$Reflc[refl2$Group=="Control-
I"]~refl2$VerbGroup[refl2$Group=="Control-I"] +
Error(refl2$Subject[refl2$Group=="Control-I"]/
refl2$VerbGroup[refl2$Group=="Control-I"])) # poređenje odgovora
kontrolne grupe na različite grupe glagola upotrebljene sa reflektivnim
klitikama
summary(anova2)

## R ne dozvoljava da se u ovom slučaju koristi funkcija TukeyHSD,
potencijalna zamena je funkcija pairwise.t.test()

## treba imati na umu i da zavisna ANOVA zahteva ispunjen uslov sferičnosti
(sphericity), koji se u okviru funkcije aov ne testira (a nekada prisutan
mauchley.test() u novim verzijama ne postoji)

## Za veličinu efekta i ovde se koristi  $\omega^2$ , ali se izračunava po složenijoj
formuli.

kruskal.test(refl$Cloze~refl$Group)
kruskal.test(reldi1$tokeni~reldi1$korpus)

## Veličinu efekta ovde nije jednostavno dobiti, često se ni ne navodi za
Kruskal-Volison test, već samo za poređenja parova.

## Za posthoc poređenja parova koriste se višestruki Vilkoksonovi testovi uz
Bonferonijevu ispravku - podešavanje nivoa statističke značajnosti na
0.05/broj poređenja (npr. da bi se rezultat smatrao statistički značajnim
za 3 poređenja potrebno je 0.05/3, tj.  $p < 0.01666667$ ); drugim rečima,
dobijenu vrednost p treba pomnožiti sa 3.

## Za Fridmanov test potrebno je uneti podatke u drugačijem formatu od formata
za jednofaktorsku analizu varijanse i Kruskal-Volison test; takođe je
potrebno da u podatke budu uključene samo varijable koje će se analizirati

refl1 = subset(refl, Group=="Control-I") # izdvajanje grupe ispitanika koja

```

```
nas zanima
refl2 = subset(refl1, select=c("Refl1c", "Refl2c", "Refl3c")) # izdvajanje
nivoa varijabli koji nas zanimaju
friedman.test(as.matrix(refl2)) # pretvaranje podataka u format matrice

## Za veličinu efekta i posthoc testove važi isto što i za Kruskal–Volisov
test.
```