R programma

Didzis Elferts

Lekcijas tēmas

- Par programmu R
- Komandu veidošana
- Datu importēšana
- Darbs ar datiem
- Pamatstatistiskie testi

R priekšrocības

- Atvērtā koda programma, kas nepārtraukti tiek papildināta un uzlabota
- R var tikt izmantots uz dažādām platformām: Linux, Windows, MacOS
- Lielākā daļa statistisko metožu un aprēķinu ir iekļauts šajā programmā
- Lietotājam ir iespējams kontrolēt visus parametrus veicot aprēķinus
- Ļoti labas grafiskās iespējas, kas ļauj veidot augstas kvalitātes grafikus
- Iespēja veidot pašam savas funkcijas
- Var nodrošināt reproducējamību (reproducible)

R trūkumi

- Lēns "mācīšanās" temps
- Daļējs trūkums - komandu rindas
- Grūti izsekot līdzi visām papildus paketēm
- Ne vienmēr var ātri atrast nepieviešamo

Noderīgas mājaslapas

- Programmas R un ar to saistīto materiālu mājaslapa http://www.r-project.org/
- RStudio IDE mājaslapa http://www.rstudio.com
- Blogu apkopojums par R saistītām lietām http://www.r-bloggers.com/
- R blogs latviešu valodā http://rvide.wordpress.com/
- Pamācības darbā ar R latviešu valodā http://dendro.daba.lv/R
- R grāmatas melnraksts latviešu valodā http://dendro.daba.lv/R/gramata/
- Video pamācības darbā ar R angļu valodā http://www.twotorials.com/
- Jautājumu un atbilžu lapa Stack overflow http://stackoverflow.com/
- Meklēšanas rīks visās R paketēs http://www.rdocumentation.org/

Piemērs

```
data(cars)
cor.test(cars$speed,cars$dist)
```

```
##
##  Pearson's product-moment correlation
##
## data:  cars$speed and cars$dist
## t = 9.464, df = 48, p-value = 1.49e-12
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.6816422 0.8862036
## sample estimates:
##       cor
## 0.8068949
```

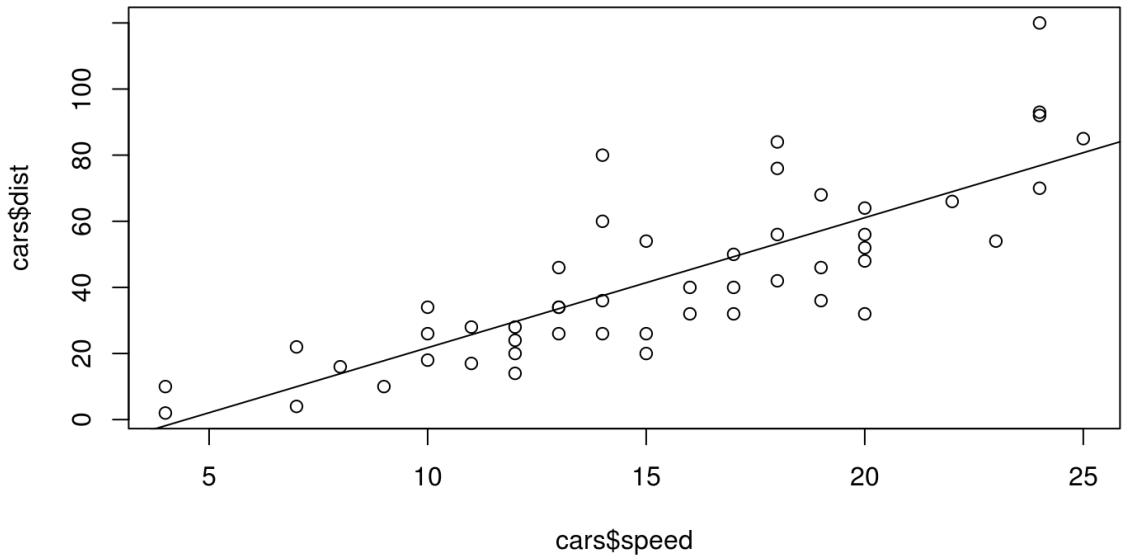
Piemērs

```
summary(lm(dist~speed,data=cars))
```

```
##
## Call:
## lm(formula = dist ~ speed, data = cars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -29.069  -9.525  -2.272   9.215  43.201
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -17.5791     6.7584  -2.601   0.0123 *
## speed         3.9324     0.4155   9.464 1.49e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.38 on 48 degrees of freedom
## Multiple R-squared:  0.6511, Adjusted R-squared:  0.6438
## F-statistic: 89.57 on 1 and 48 DF,  p-value: 1.49e-12
```

Piemērs

```
plot(cars$dist~cars$speed)
abline(lm(dist~speed,data=cars))
```



Komandu veidošana

- Visas komandas tiek rakstītas pēc ">" zīmes
- Lai komandu rindai pievienotu komentāru, pirms tā ir jāraksta “#”
- Atstarpes komandās parasti tiek ignorētas, izņēmums ir rakstot “<-”
- Ja komanda ir pārāk gara, tad to var sadalīt vienkārši ar Enter taustiņu
- Iztrūkstošās vērtības programmā R apzīmē ar NA

Palīdzības iegūšana

```
help.start()
```

```
help(plot)
```

```
args(cor)
```

```
## function (x, y = NULL, use = "everything", method = c("pearson",
##     "kendall", "spearman"))
## NULL
```

```
example(plot)
```

Mājaslapā:

http://www.rdocumentation.org/

R paketes

- Bāzes R programmā ir tikai neliela daļa no iespējām, ko piedāvā R
- Papildus iespējas (funkcijas) pieejamas kā paketes (library), kuras no sākuma jāuzinstalē (ja tas jau nav izdarīts) un tad jāpievieno (tas jādara katrā sesijā)
- Pakešu instalācija notiek ar funkciju `install.packages()`, bet pievienošana darba sesijai ar funkciju `library()`.

```
instal.packages("ggplot2")
library(ggplot2)
```

R kā kalkulators

R var tikt izmantots kā vienkāršs kalkulators

```
4+7
```

```
## [1] 11
```

```
log(8,2)
```

```
## [1] 3
```

```
exp(2)
```

```
## [1] 7.389056
```

Datu importēšana

```
read.table(file="/../faila.nosaukums.txt", header=TRUE, sep="\t",dec=".")
```

Kolonnu atdalītāja `sep=` vērtības var būt: "\t" " " , " " ; "

Decimāldaļu atdalītājs failā `dec=` varbūt "." vai ","

Pēc noklusējuma visas teksta kolonnas tiek pārvērstas par faktoriem. To var novērst pievienojot argumentu `stringsAsFactors=FALSE`.

Garo saiti uz failu var nerakstīt, ja fails atrodas darba direktorijā (Working directory) (var nomainīt ar `File/Change dir...`)

Darba direktorija

```
getwd()
```

```
## [1] "/home/didzis/Dropbox/DidzaDati/2015/Kursi_pasniedzejiem"
```

```
setwd("/home/didzis/Documents")
```

```
getwd()
```

```
## [1] "/home/didzis/Documents"
```

Datu importēšana

```
dati <- read.table(file="niedres.txt",header=T,
                   sep="\t", dec=".")
dati2 <- read.csv2(file="niedres.txt",header=T,
                   sep="\t", dec=".")
```

```
dati
```

```
##    garums platums
## 1    31.6     2.5
## 2    23.2     2.3
## 3    39.2     2.1
## 4    37.4     5.8
## 5    21.1     2.2
## 6    37.0     4.1
## 7    24.7     3.5
## 8    31.3     4.2
## 9    37.4     2.5
## 10   39.7     2.8
## 11   38.0     4.3
## 12   24.9     1.1
## 13   30.8     1.9
## 14   26.7     0.4
```

Datu eksportēšana

Datus, kas izveidoti R kā datu tabulas, visērtāk var eksportēt kā `.txt` vai `.csv` failus.

```
write.table(x=tas.kas.jaeksporte,
  file="/saite/uz/failu/failanosaukums.txt",
  sep="\t",dec=".")
```

```
write.table(x=dati, file="eksports.txt",sep="\t",dec=".")
```

Eksportējot pēc noklusējuma tiek eksportēti arī rindiņu nosaukumi. Ja tos nevajag, tad jāpievieno arguments `row.names=FALSE`

Datu struktūra

Funkcija `str()` ļauj apskatīt kāda veida dati ir objektā, kā arī šo datu pirmās vērtības.

```
str(dati)
```

```
## 'data.frame':    50 obs. of  2 variables:
##  $ garums : num  31.6 23.2 39.2 37.4 21.1 37 24.7 31.3 37.4 39.7 ...
##  $ platums: num  2.5 2.3 2.1 5.8 2.2 4.1 3.5 4.2 2.5 2.8 ...
```

Faktors

- Īpašs datu veids, kas ļauj veidot nominālas un kārtas datus statistiskajām analīzēm un grafikiem
- Pēc noklusējuma faktora līmeņu secība tiek veidota alfabētiski
- Faktoram nevar vienkārši pievienot citus datus, ja tie nav starp nodefinētajiem līmeņiem

Faktors

```
otrais <- c("A","B","C","D","E","F")
otrais
```

```
## [1] "A" "B" "C" "D" "E" "F"
```

```
otrais.fakt<-factor(otrais)
otrais.fakt
```

```
## [1] A B C D E F
## Levels: A B C D E F
```

Faktors

Lai faktora līmeņiem būtu cita secība, tad to maina ar argumentu `levels=`

```
otrais.fakt <- factor(otrais,levels=c("D","A","C","B","E","F"))

otrais.fakt
```

```
## [1] A B C D E F
## Levels: D A C B E F
```

Objekta dimensijas

```
dim(dati) #datu tabula
```

```
## [1] 50  2
```

Nosaukumi

```
names(dati)
```

```
## [1] "garums"  "platums"
```

Datu apskatīšana

```
head(dati)
```

```
##   garums platums
## 1   31.6     2.5
## 2   23.2     2.3
## 3   39.2     2.1
## 4   37.4     5.8
## 5   21.1     2.2
## 6   37.0     4.1
```

```
tail(dati,n=3)
```

```
##    garums platums
## 48   43.1     3.7
## 49   38.8     3.9
## 50   46.9     5.4
```

Datu atlasīšana – datu tabula

Kolonnas atlasīšana

```
dati$garums
```

```
##  [1] 31.6 23.2 39.2 37.4 21.1 37.0 24.7 31.3 37.4 39.7 38.0 24.9 30.8 26.7
## [15] 34.7 33.3 59.0 17.7 24.3 41.4 49.1 46.2 11.9 39.5 45.3 39.6 45.8 31.0
## [29] 39.9 56.4 37.3 41.1 45.3 49.3 24.9 51.7 40.6 43.5 48.9 31.6 53.6 53.5
## [43] 26.9 35.7 46.1 29.1 33.7 43.1 38.8 46.9
```

```
dati["garums"]
```

```
##    garums
## 1    31.6
## 2    23.2
## 3    39.2
## 4    37.4
## 5    21.1
## 6    37.0
## 7    24.7
## 8    31.3
## 9    37.4
## 10   39.7
```

Datu atlasīšana – datu tabula

Kolonnas atlasīšana

```
dati[,1]
```

```
##  [1] 31.6 23.2 39.2 37.4 21.1 37.0 24.7 31.3 37.4 39.7 38.0 24.9 30.8 26.7
## [15] 34.7 33.3 59.0 17.7 24.3 41.4 49.1 46.2 11.9 39.5 45.3 39.6 45.8 31.0
## [29] 39.9 56.4 37.3 41.1 45.3 49.3 24.9 51.7 40.6 43.5 48.9 31.6 53.6 53.5
## [43] 26.9 35.7 46.1 29.1 33.7 43.1 38.8 46.9
```

```
dati[,1,drop=FALSE]
```

```
##    garums
## 1    31.6
## 2    23.2
## 3    39.2
## 4    37.4
## 5    21.1
## 6    37.0
## 7    24.7
## 8    31.3
## 9    37.4
```

Datu atlasīšana - rindiņa

Rindiņas atlasīšana

```
dati[1,]
```

```
##   garums platums
## 1   31.6     2.5
```

Datu atlasīšana - viens ieraksts

Viena elementa atlasīšana

```
dati[1,2]
```

```
## [1] 2.5
```

Datu atlasīšana - vektors

```
dati$garums[1]
```

```
## [1] 31.6
```

```
dati$garums[c(1,4)]
```

```
## [1] 31.6 37.4
```

```
dati$garums[-3]
```

```
##  [1] 31.6 23.2 37.4 21.1 37.0 24.7 31.3 37.4 39.7 38.0 24.9 30.8 26.7 34.7
## [15] 33.3 59.0 17.7 24.3 41.4 49.1 46.2 11.9 39.5 45.3 39.6 45.8 31.0 39.9
## [29] 56.4 37.3 41.1 45.3 49.3 24.9 51.7 40.6 43.5 48.9 31.6 53.6 53.5 26.9
## [43] 35.7 46.1 29.1 33.7 43.1 38.8 46.9
```

Darbs ar datiem

```
garums
```

```
Error: object 'garums' not found
```

Programmā R datu tabulu kolonnas automātiski netiek uzskatītas par atsevišķiem mainīgiem

Darba vides sakārtošana

Atmiņā esošo objektu apskatīšana

```
ls()
```

```
## [1] "bietes"      "cars"        "dati"        "dati2"       "otrais"
## [6] "otrais.fakt"
```

Objekta dzēšana

```
rm(otrais)
```

Paraugkopas grafiskā analīze

Paraugkopas grafiskā analīze

Pirms datu analīzes vienmēr ieteicams veikt datu grafisko analīzi, jo:

- tas ļauj atrast kļūdainas vai ekstrēmas vērtības, kuras nav redzamas vienkārši apskatot datus
- ļauj novērtēt vai datos redzamas kāda grupēšanās
- ļauj novērtēt saistības starp mainīgajiem

Dati

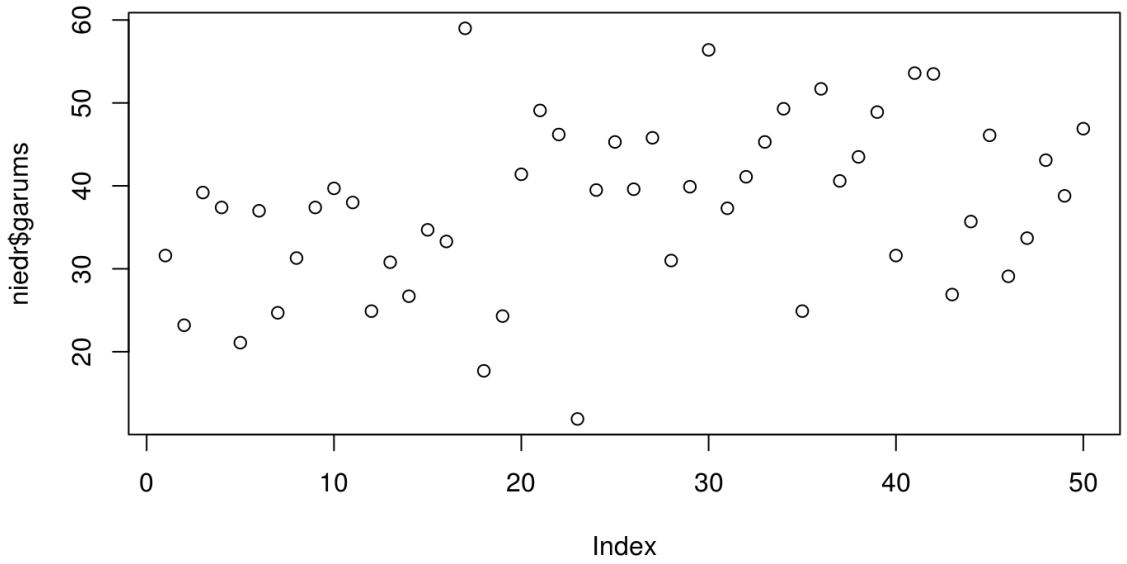
```
niedr<-read.table(file="niedres2.txt",
          header=TRUE,sep="\t",dec=".")
```

```
str(niedr)
```

```
## 'data.frame':    50 obs. of  3 variables:
##  $ garums : num  31.6 23.2 39.2 37.4 21.1 37 24.7 31.3 37.4 39.7 ...
##  $ platums: num  2.5 2.3 2.1 5.8 2.2 4.1 3.5 4.2 2.5 2.8 ...
##  $ paraug : Factor w/ 3 levels "Austr","Riet",..: 1 1 1 1 1 1 1 1 1 1 ...
```

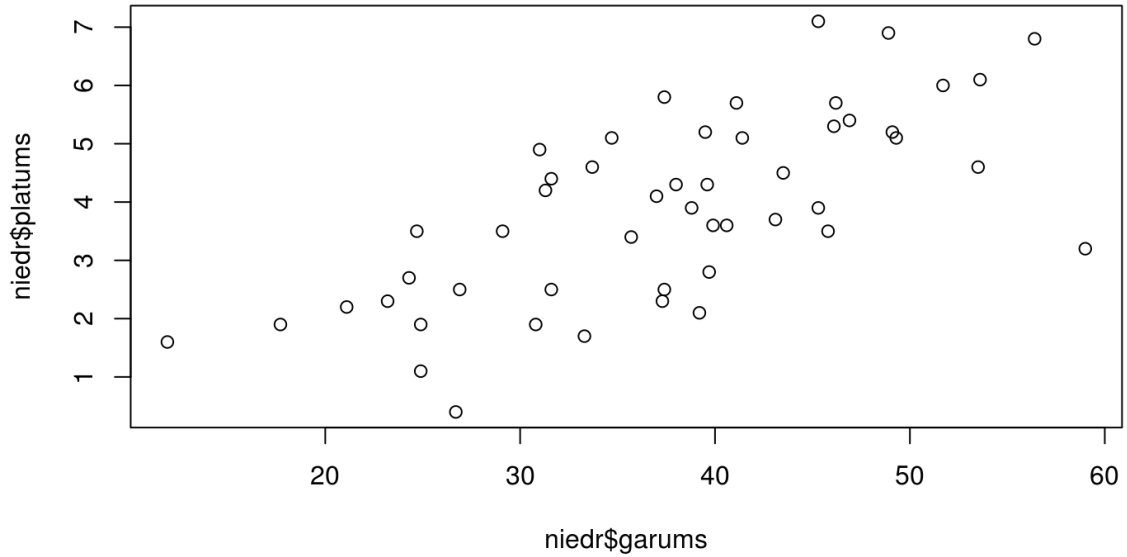
Grafiskā analīze

```
plot(niedr$garums)
```



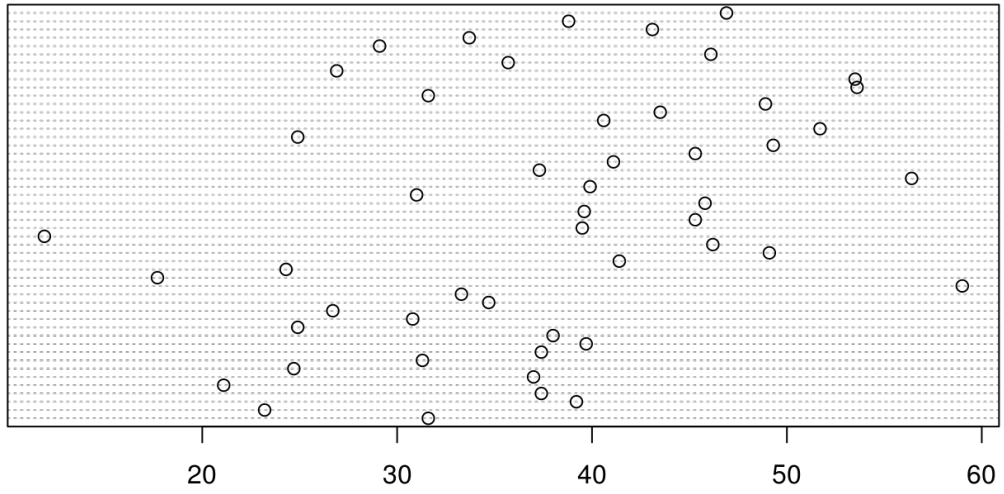
Grafiskā analīze

```
plot(niedr$garums,niedr$platums)
```



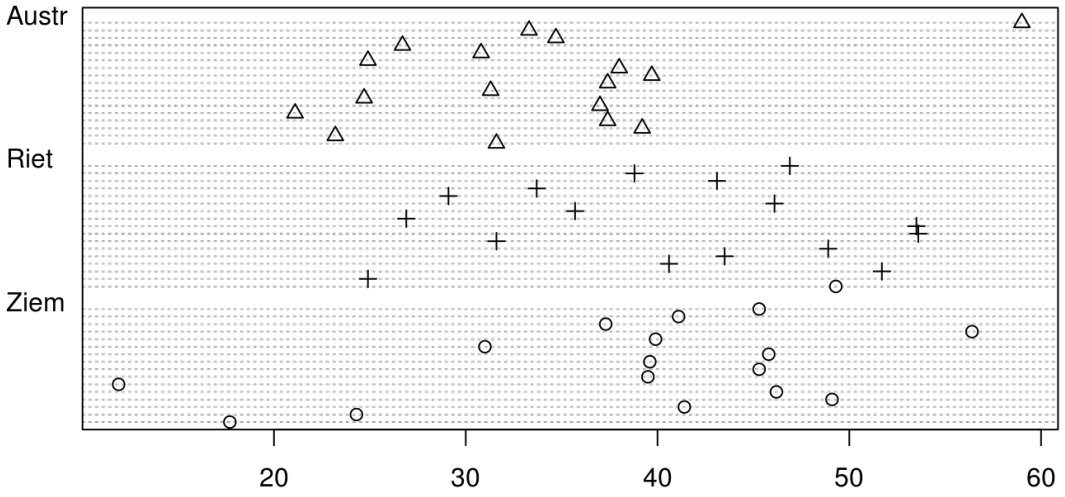
Grafiskā analīze

```
dotchart(niedr$garums)
```



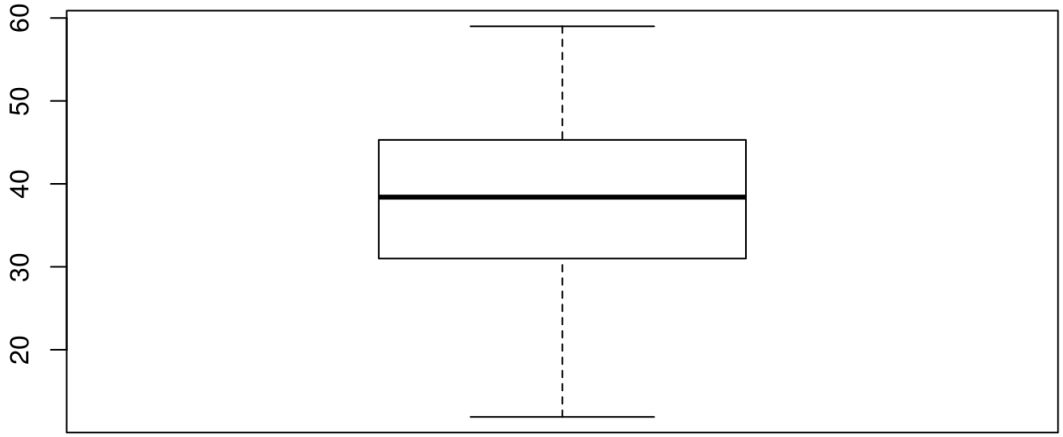
Grafiskā analīze

```
dotchart(niedr$garums,groups=niedr$paraug,pch=as.numeric(niedr$paraug))
```



Grafiskā analīze

```
boxplot(niedr$garums)
```



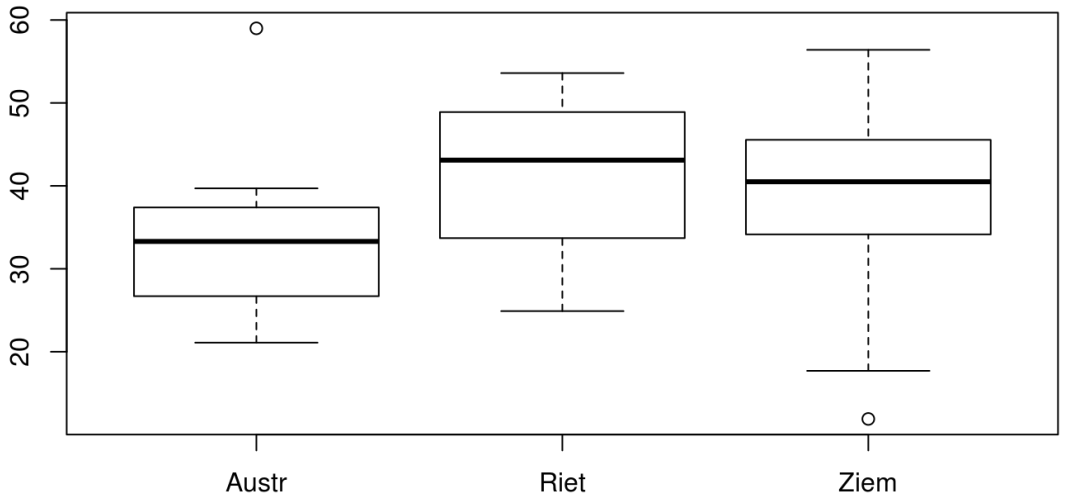
Grafiskā analīze

Boxplot attēlā ir parādīts:

- mediāna - vidējā tumšā līnija;
- 1. un 3. kvartile - taisnstūra apakšējā un augšējā mala;
- minimālā un maksimālā vērtība - apakšējais un augšējais nogrieznis (ja nav ekstrēmu vērtību);
- ja kāda vērtība ir tālāk nekā 1,5 reizes par 3. un 1. kvartiles starpību (taisnstūra augstums), tad šīs vērtības parādās kā atsevišķi punkti.

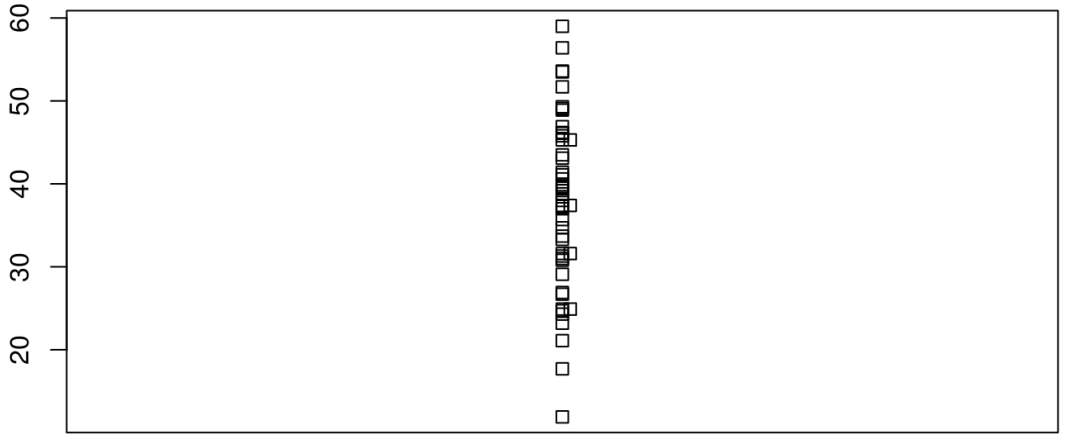
Grafiskā analīze

```
boxplot(niedr$garums~niedr$paraug)
```



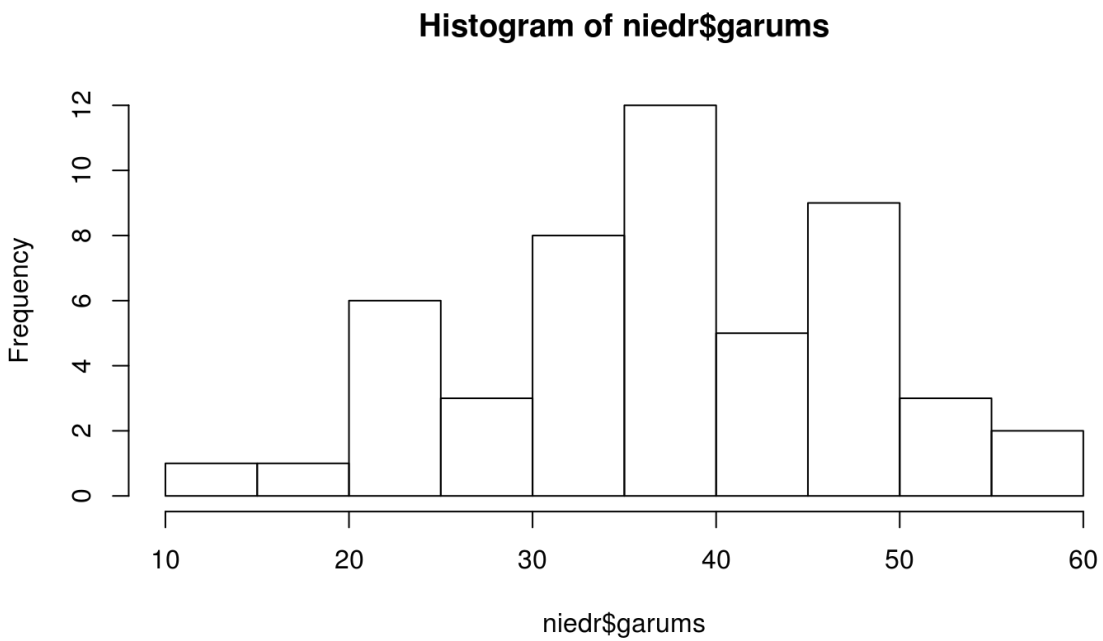
Grafiskā analīze

```
stripchart(niedr$garums,method="stack",vert=TRUE)
```



Grafiskā analīze

```
hist(niedr$garums)
```



"Izlēcošās" vērtības

Ja datos tiek konstatēta "izlēcoša" vērtība:

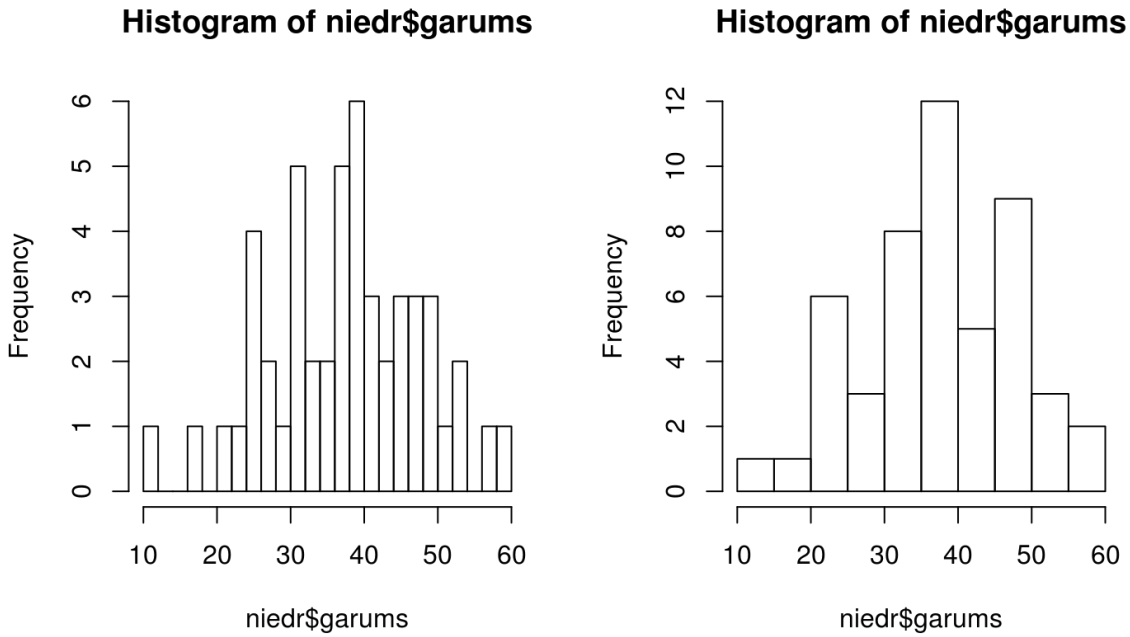
- Jāpārliecinās, ka tā nav ievadīšanas kļūda (piemēram, **1.80** vietā uzrakstīts **18.0**)
- Varbūt tas tiešām ir tikai ekstrēms, bet reāls novērojums. Šādā gadījumā ir vērts analīzes uztaisīt ar un bez šīs vērtības
- Ja vērtība tiešām ir kļūdaina ("izlēcoša") un to ir jāizdzēšs, tad šis process ir arī jāpiemin metodēs pie datu apstrādes

Atbilstība normālajam sadalījumam

- Grafiskā pārbaude
- Analītiskā pārbaude

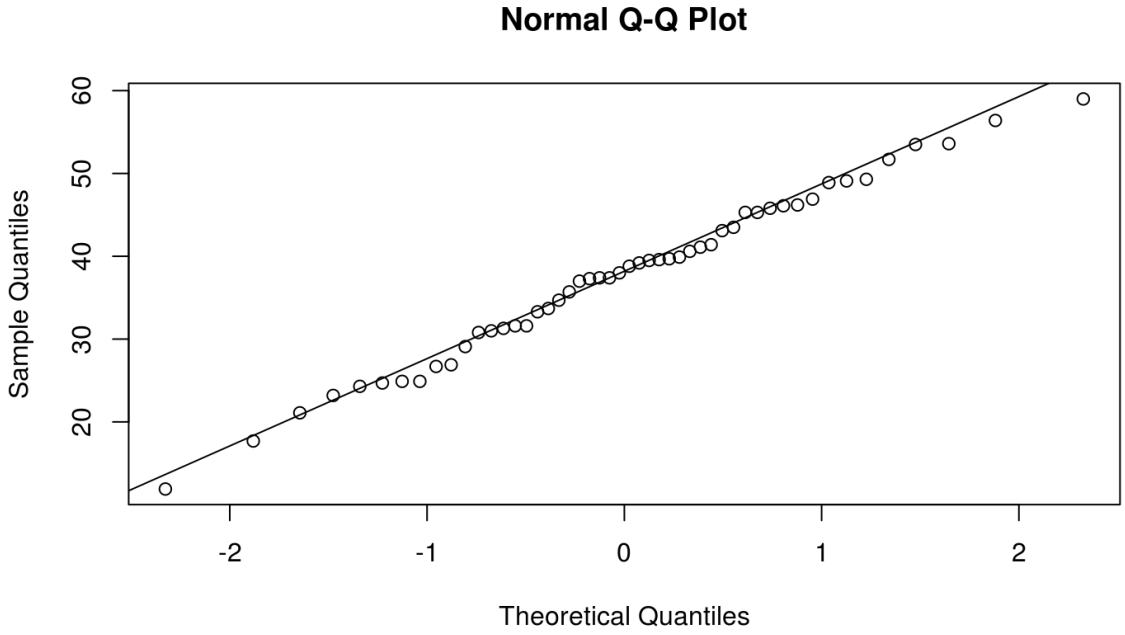
Grafiskā pārbaude - histogramma

```
par(mfrow=c(1,2))
hist(niedr$garums,breaks=20)
hist(niedr$garums,breaks=10)
```



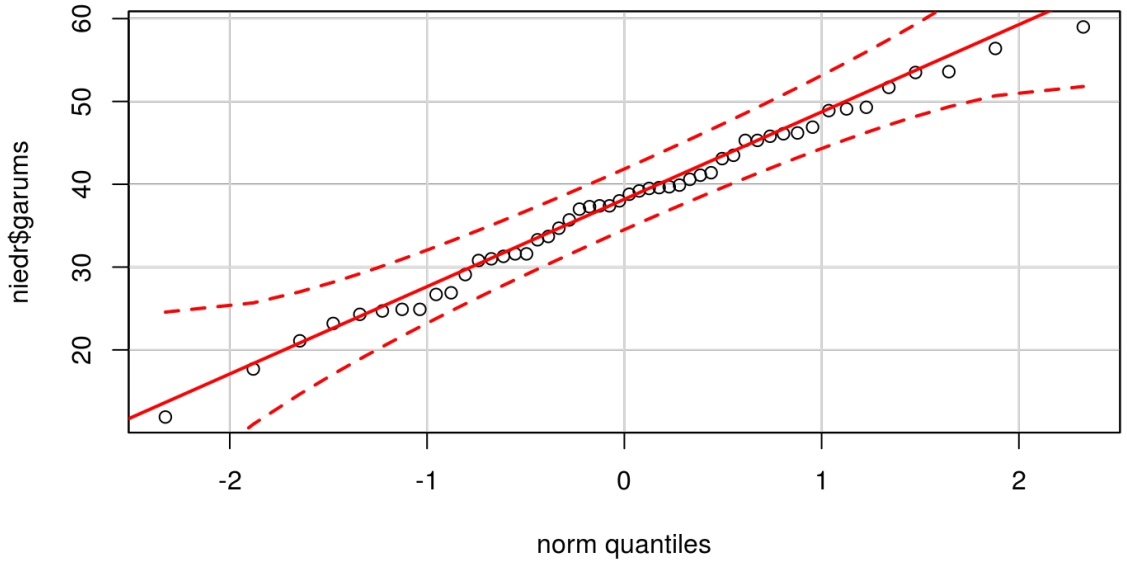
Grafiskā pārbaude - QQplot 1

```
qqnorm(niedr$garums)
qqline(niedr$garums)
```



Grafiskā pārbaude - QQplot 2

```
library(car)
qqPlot(niedr$garums)
```



Analītiskā pārbaude - Šapīro tests 1

```
shapiro.test(niedr$garums)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  niedr$garums
## W = 0.991, p-value = 0.9668
```

Secinājums: pie būtiskuma līmeņa $\alpha = 0,05$ niedru garuma datu sadalījums būtiski neatšķiras no normālā sadalījuma, jo iegūtā p vērtība ir lielāka par būtiskuma līmeni.

Statistiskie rādītāji

Statistiskie rādītāji 1

Vidējo aritmētisko, standartnovirzi, dispersiju un mediānu aprēķina attiecīgi ar funkcijām `mean()`, `sd()`, `var()` un `median()`. Visām šīm funkcijām kā arguments jānorāda skaitļu vektors/viena kolonna.

```
mean(niedr$garums)
```

```
[1] 37.594
```

```
sd(niedr$garums)
```

```
[1] 10.31332
```

```
var(niedr$garums)
```

```
[1] 106.3647
```

```
median(niedr$garums)
```

```
[1] 38.4
```

Statistiskie rādītāji 2

Skaitļu noapaļošanai izmanto funkciju `round()`, kurā norāda vienu skaitli, vai objektu ar vairākiem skaitļiem un decimāldaļu skaitu aiz komata.

```
round(mean(niedr$garums),1)
```

```
[1] 37.6
```

Statistiskie rādītāji 3

```
x <- c(1:20,NA)
mean(x)
```

```
[1] NA
```

```
mean(x,na.rm=TRUE)
```

```
[1] 10.5
```

Daudzas funkcijas R dod rezultātu NA, ja kāda no apstrādājamām vērtībām arī ir NA. Tāpēc jānorāda, ko darīt ar šīm NA vērtībām.

Statistiskie rādītāji 4

Minimālo un maksimālo vērtību aprēķina ar funkcijām `min()` un `max()`, vai arī ar funkciju `range()` var aprēķināt abas vērtības uzreiz.

```
min(niedr$garums)
```

```
[1] 11.9
```

```
max(niedr$garums)
```

```
[1] 59
```

```
range(niedr$garums)
```

```
[1] 11.9 59.0
```

Statistiskie rādītāji 5

Kvartiles aprēķina ar funkciju `quantile()`. Ja nepieciešams aprēķināt procentiles, vai arī tikai kādu no kvartilēm, tad papildus norāda argumentu `probs=` un vajadzīgo rādītāju izteiktu decimāldaļās.

```
quantile(niedr$garums)
```

```
    0%    25%    50%    75%   100% 
11.900 31.075 38.400 45.300 59.000 
```

```
quantile(niedr$garums,probs=c(0.025,0.975))
```

```
  2.5%  97.5% 
18.465 55.770 
```

Statistiskie rādītāji 6

Ja funkcijā `summary()` ievieto skaiļu vektoru, rezultātā iegūst sešus statistiskos rādītājus, kas to raksturo.

```
summary(niedr$garums)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  11.90   31.08   38.40   37.59   45.30   59.00 
```

Statistiskie rādītāji 7

Ja funkcijā `summary()` ievieto datu tabulu, rezultātā katrai skaitliskajai kolonnai iegūst sešus statistiskos rādītājus, bet faktoram/rakstu zīmju kolonnai iegūst vērtību atkārošanās biežumu.

```
summary(niedr)
```

```
     garums          platums         paraug
 Min.   :11.90   Min.   :0.400   Austr:17
 1st Qu.:31.07   1st Qu.:2.500   Riet :17
 Median :38.40   Median :3.900   Ziem :16
 Mean   :37.59   Mean   :3.892
 3rd Qu.:45.30   3rd Qu.:5.100
 Max.   :59.00   Max.   :7.100
```

Paraugkopu salīdzināšana

Dati 1

```
niedr2<-read.table(file="lapas.txt",
                   header=TRUE,sep="\t",dec=".")
```

```
str(niedr2)
```

```
## 'data.frame':    34 obs. of  3 variables:
##  $ garums : num  31.6 23.2 39.2 37.4 21.1 37 24.7 31.3 37.4 39.7 ...
##  $ platums: num  2.5 2.3 2.1 5.8 2.2 4.1 3.5 4.2 2.5 2.8 ...
##  $ paraug : Factor w/ 2 levels "Austr","Riet": 1 1 1 1 1 1 1 1 1 1 ...
```

Dati 2

```
summary(niedr2)
```

```
     garums          platums        paraug
 Min.   :21.10   Min.   :0.400   Austr:17
 1st Qu.:30.93   1st Qu.:2.500   Riet :17
 Median :37.20   Median :3.650
 Mean   :37.29   Mean   :3.679
 3rd Qu.:43.40   3rd Qu.:4.600
 Max.   :59.00   Max.   :6.900
```

Dispersiju salīdzināšana 1

Dispersiju salīdzināšanu ar F testu veic izmantojot funkciju `var.test()`.

Funkcijai kā argumentus var nordātīs divas datu kolonnas, kas atdalītas ar komatu, vai arī datu kolonnu un grupu kolonnu, kas atdalītas ar tildes zīmi.

```
var.test(pirmā.paraugkopa,otrā.paraugkopa)
```

```
var.test(datu.kolonna~grupu.kolonna)
```

Dispersiju salīdzināšana 2

```
var.test(niedr2$garums~niedr2$paraug)


    F test to compare two variances

data:  niedr2$garums by niedr2$paraug
F = 0.8949, num df = 16, denom df = 16, p-value = 0.827
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.3240865 2.4711930
sample estimates:
ratio of variances
         0.8949191
```

Dispersiju salīdzināšana 3

Secinājums: Pie būtiskuma līmeņa $\alpha = 0,05$ divu paraugkopu dispersijas ir homogēnas (neatšķiras būtiski), jo iegūtā F-vērtība ir 0,89 un p-vērtība ir 0,827 (lielāka par 0,05). Ja skatās uz F vērtības ticamības intervālu, tad tas satur skaitli viens (0,32 līdz 2,47), tātad dispersijas ir līdzīgas (homogēnas).

Vidējo aritmētisko salīdzināšana 1

Vidējo aritmētisko salīdzināšanai izmanto funkciju `t.test()`. Ja pirms tam pierādīts, ka dispersijas neatšķiras, tad jāpievieno arguments `var.equal=TRUE`.

Funkcijai kā argumentus var nordātīs divas datu kolonnas, kas atdalītas ar komatu, vai arī datu kolonnu un grupu kolonnu, kas atdalītas ar tildes zīmi.

```
t.test(datu.kolonna~grupu.kolonna,var.equal=TRUE)
```

```
t.test(paraugkopa1,paraugkopa2,var.equal=TRUE)
```

Vidējo aritmētisko salīdzināšana 2

```
t.test(niedr2$garums~niedr2$paraug,var.equal=TRUE)
```

```
    Two Sample t-test

data:  niedr2$garums by niedr2$paraug
t = -2.4034, df = 32, p-value = 0.02221
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -13.899897  -1.147161
sample estimates:
mean in group Austr  mean in group Riet
           33.52941            41.05294
```

Vidējo aritmētisko salīdzināšana 3

Secinājums: Pie būtiskuma līmeņa $\alpha = 0,05$ pastāv statistiski būtiska atšķirība starp Autrumu un Rietumu parauglaukumu vidējiem niedru lapu garumiem (p-vērtība=0,022). Vidējo aritmētisko starpības 95% ticamības intervāls ir no -13,89 līdz -1,15. Intervāls nesatur 0 (ja satūrētu 0, tad tas norādītu, ka atšķirība nav būtiska).

Saistītu paraugkopu vidējo aritmētisko salīdzināšana 1

Saistītu vai atkarīgu paraugkopu vidējo aritmētisko salīdzināšanu veic ar funkciju `t.test()`, kurai papildus iekļauj argumentu `paired=TRUE`. Šī testa veikšanai abu paraugkopu datiem obligāti jāatrodas katrai savā kolonnā.

```
t.test(x,y,paired=TRUE)
```

Saistītu paraugkopu vidējo aritmētisko salīdzināšana 2

```
rokas <- read.table(file="rokas.txt",header=TRUE,
                    sep="\t",dec=".")
```

```
str(rokas)
```

```
## 'data.frame':    25 obs. of  2 variables:
##  $ laba  : num  32.9 39.7 26.9 38.8 58 50.6 31.4 68.7 65.2 31.8 ...
##  $ kreisa: num  21.9 39.2 29.9 34.8 48.7 51.4 34.5 56.4 57.5 27.8 ...
```

Saistītu paraugkopu vidējo aritmētisko salīdzināšana 3

```
t.test(rokas$laba,rokas$kreisa,paired=TRUE)
```

```
    Paired t-test

data:  rokas$laba and rokas$kreisa
t = 4.4743, df = 24, p-value = 0.0001581
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 2.413454 6.546546
sample estimates:
mean of the differences
                   4.48
```

Saistītu paraugkopu vidējo aritmētisko salīdzināšana 4

Secinājums: Pie būtiskuma līmeņa $\alpha = 0,05$ pastāv statistiski būtiska atšķirība starp labās un kreisās rokas spēku (p-vērtība <0,001). Vidējā pāru starpība ir 4,48 un tās 95% ticamības intervāls ir no 2,41 līdz 6,54.

T-tests vienai paraugkopai 1

Lai pārbaudītu vienas paraugkopas vērtību atbilstību zināmai vērtībai (ģenerālkopas vidējam aritmētiskajam), izmanto funkciju `t.test()`, kurai kā papildus argumentu norāda zināmo vērtību `mu=`.

```
t.test(analizējamā.paraugkopa,mu=zināmā.vērtība)
```

T-tests vienai paraugkopai 2

```
t.test(niedr2$platums,mu=3.0)
```

```
    One Sample t-test

data:  niedr2$platums
t = 2.5592, df = 33, p-value = 0.01526
alternative hypothesis: true mean is not equal to 3
95 percent confidence interval:
 3.139297 4.219526
sample estimates:
mean of x
 3.679412
```

T-tests vienai paraugkopai 3

Secinājums: pie būtiskuma līmeņa $\alpha = 0,05$ niedru lapu platums pētītajā teritorijā būtiski atšķiras no vērtības 3 mm (p vērtība ir 0,0156). Niedru lapu platumu vidējais aritmētiskais ir 3,68 un tā 95% ticamības intervāls ir no 3,14 līdz 4,22 (nesatur vērtību 3,0).

Neparametriskās metodes - neatkarīgas paraugkopas 1

Neatkarīgu paraugkopu salīdzināšanai ar neparametriskajām metodēm izmanto funkciju `wilcox.test()`.

Funkcijai kā argumentus var nordātīs divas datu kolonnas, kas atdalītas ar komatu, vai arī datu kolonnu un grupu kolonnu, kas atdalītas ar tildes zīmi.

```
wilcox.test(datu.kolonna~grupu.kolonna)
```

```
wilcox.test(pirma.paraugkopa,otra.paraugkopa)
```

Neparametriskās metodes - neatkarīgas paraugkopas 2

```
wilcox.test(niedr2$garums~niedr2$paraug)
```

```
Warning in wilcox.test.default(x = c(31.6, 23.2, 39.2, 37.4, 21.1, 37,
24.7, : cannot compute exact p-value with ties
```

```
    Wilcoxon rank sum test with continuity correction

data:  niedr2$garums by niedr2$paraug
W = 75, p-value = 0.01745
alternative hypothesis: true location shift is not equal to 0
```

Neparametriskās metodes - neatkarīgas paraugkopas 3

Secinājums: Pie būtiskuma līmeņa $\alpha = 0,05$ starp Austrumu un Rietumu parauglaukumu niedru lapu garumu vērtību sadalījumiem pastāv statistiski būtiska atšķirība (p vērtība 0,017).

Neparametriskās metodes - atkarīgas paraugkopas 1

Saistītu vai atkarīgu paraugkopu vērtību sadalījuma salīdzināšanu veic ar funkciju `wilcox.test()`, kurai papildus iekļauj argumentu `paired=TRUE`. Šī testa veikšanai abu paraugkopu datiem obligāti jāatrodas katrai savā kolonnā.

```
wilcox.test(pirma.paraugkopa,otra.paraugkopa, paired=TRUE)
```

Neparametriskās metodes - atkarīgas paraugkopas 2

```
wilcox.test(rokas$laba,rokas$kreisa,paired=TRUE)
```

```
Warning in wilcox.test.default(rokas$laba, rokas$kreisa, paired = TRUE):
cannot compute exact p-value with ties
```

```
    Wilcoxon signed rank test with continuity correction

data:  rokas$laba and rokas$kreisa
V = 291.5, p-value = 0.0005445
alternative hypothesis: true location shift is not equal to 0
```

Neparametriskās metodes - atkarīgas paraugkopas 3

Secinājums: Pie būtiskuma līmeņa $\alpha = 0,05$ pastāv statistiski būtiska atšķirība starp labās un kreisās rokas spēka vērtību sadalījumu (p-vērtība 0,0005).

χ^2 tests 1

χ^2 testa veikšanai izmanto funkciju `chisq.test()`, kurai kā argumentus norāda empīrisko vērtību sadalījumu un sagaidāmo vērtību sadalījumu (kā iespējamības vērtības), ja jāsalīdzina ar teorētisko sadalījmu, vai arī matrici, kas satur divu paraugkopu vērtību sadalījumus, ja jāsalīdzina divas paraugkopas.

```
chisq.test(empir.sad.vektors,teor.sad.vektors)
chisq.test(vertibu.matrice)
```

χ^2 tests 2

```
koki<-matrix(c(12,34,56,23,8,27,33,47,14,11),ncol=2)

rownames(koki) <- c("Priede","Egle","Bērzs","Ozols","Kļava")
colnames(koki) <- c("Paraug A","Paraug B")

koki
```

```
       Paraug A Paraug B
Priede       12       27
Egle         34       33
Bērzs        56       47
Ozols        23       14
Kļava         8       11
```

χ^2 tests 3

```
chisq.test(koki)
```

```
    Pearson's Chi-squared test

data:  koki
X-squared = 9.2298, df = 4, p-value = 0.05561
```

Secinājums: Pie būtiskuma līmeņa $\alpha = 0,05$ nav statistiski būtiskas atšķirības starp koku sugu sadalījumu divos parauglaukumos (p-vērtība 0,056). Tomēr būtu jāņem vērā, ka p-vērtība ir tuvu kritiskajai robežai. Iespējams, ka palielinot datu apjomu, atšķirība jau būtu būtiska.

Korelācijas analīze

Funkcijas analīzes veikšanai

Pamatfunkcijas korelācijas analīzes veikšanai ir `cor()` un `cor.test()`.

Pirmajā funkcijā var likt gan atsevišķus vektorus (kolonnas), gan vienu vai divas vienāda garuma datu tabulas. Otrajā funkcijā var likt tikai divus vektorus (kolonnas), starp kurām jāaprēķina korelācijas koeficients.

Pēc noklusējuma abas funkcijas aprēķina Pīrsona korelācijas koeficientu. Ja nepieciešams cits, tad attiecīgi jāpievieno arguments `method="kendall"` vai `method="spearman"`.

Ja datos ir iztrūkstošās vērtības (NA), pievieno argumentu `use="pairwise.complete.obs"`.

Dati

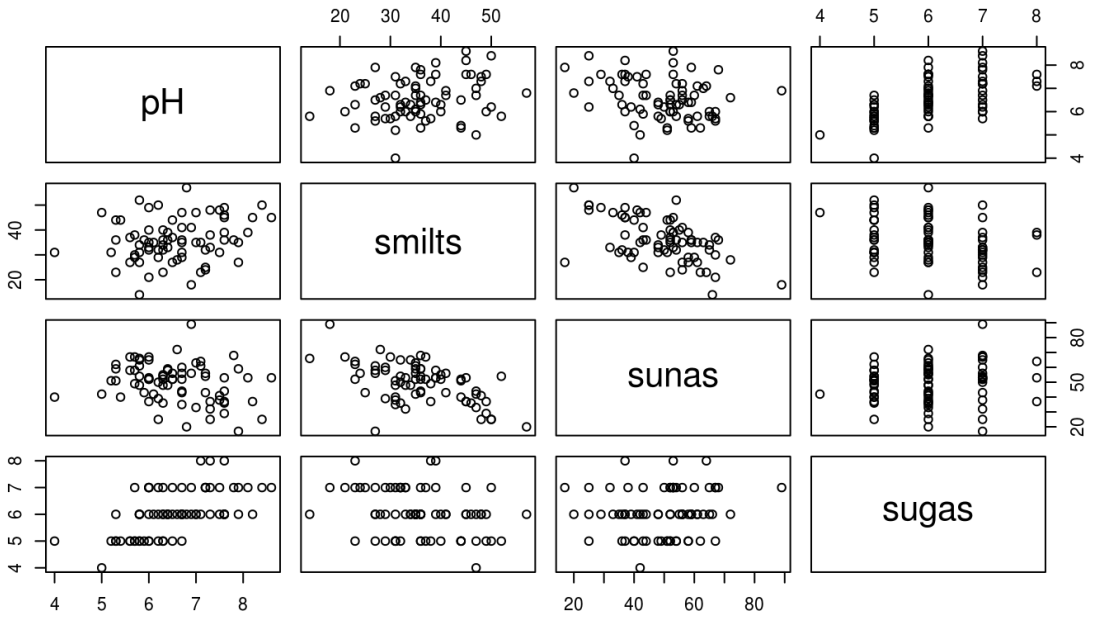
```
smiltaji <- read.table(file="smiltaji.txt",header=TRUE,
            sep="\t",dec=".")
```

```
summary(smiltaji)
```

```
##        pH            smilts          sunas           sugas
##  Min.   :4.000   Min.   :14.00   Min.   :17.00   Min.   :4.000
##  1st Qu.:6.000   1st Qu.:31.00   1st Qu.:40.00   1st Qu.:5.000
##  Median :6.500   Median :35.00   Median :52.00   Median :6.000
##  Mean   :6.556   Mean   :35.92   Mean   :49.56   Mean   :6.055
##  3rd Qu.:7.200   3rd Qu.:41.00   3rd Qu.:58.00   3rd Qu.:7.000
##  Max.   :8.600   Max.   :57.00   Max.   :89.00   Max.   :8.000
```

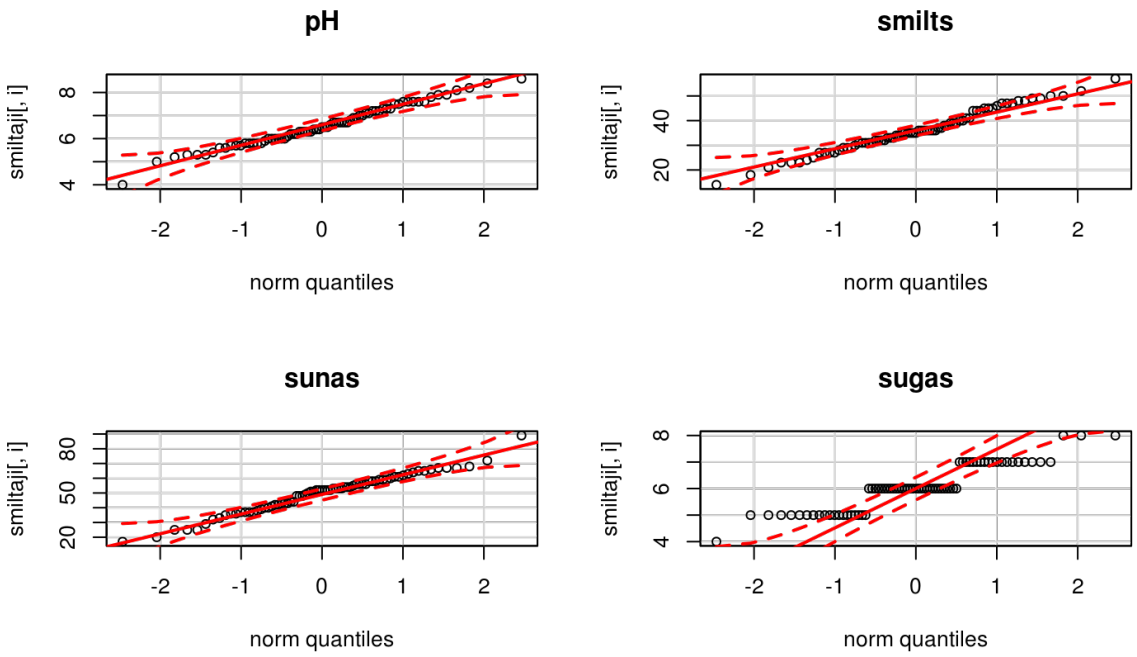
Grafiskā analīze

```
pairs(smiltaji)
```



Normalitātes tests

```
par(mfrow=c(2,2))
library(car)
for(i in 1:4){
 qqPlot(smiltaji[,i],main=names(smiltaji)[i])
}
```



Pīrsona korelācijas koeficients 1

```
cor(smiltaji[,1:3])
```

```
               pH     smilts      sunas
pH      1.0000000  0.2213315 -0.2517885
smilts  0.2213315  1.0000000 -0.5576408
sunas  -0.2517885 -0.5576408  1.0000000
```

Funkcijas `cor()` rezultātos ir redzama korelācijas koeficientu matrica, bet neparādās korelācijas koeficientu būtiskumi (p vērtības).

Pīrsona korelācijas koeficients 2

```
library(ltm)
rcor.test(smiltaji[,1:3])
```

```
       pH     smilts sunas
pH      *****  0.221 -0.252
smilts  0.060  ***** -0.558
sunas   0.032 <0.001  *****

upper diagonal part contains correlation coefficient estimates
lower diagonal part contains corresponding p-values
```

Secinājums: statistiski būtiska korelācija pie būtiskumu līmeņa $\alpha = 0,05$ pastāv starp pH un sūnu segumu un sūnu un smilts segumime, jo atbilstošās p vērtības ir mazākas par būtiskuma līmeni.

Pīrsona korelācijas koeficients 3

```
cor.test(smiltaji$pH,smiltaji$sunas)


    Pearson's product-moment correlation

data:  smiltaji$pH and smiltaji$sunas
t = -2.1922, df = 71, p-value = 0.03164
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.45547107 -0.02305682
sample estimates:
       cor
-0.2517885
```

Secinājums: Pastāv statistiski būtiska negatīva korelācija starp pH un sūnu segumu (-0,25), jo p vērtība ir mazāka par būtiskuma līmeni (0,032<0,05).

Spirmena korelācijas koeficients

```
cor.test(smiltaji$sugas,smiltaji$smilts,method="spearman")
```

```
Warning in cor.test.default(smiltaji$sugas, smiltaji$smilts, method =
"spearman"): Cannot compute exact p-value with ties
```

```
    Spearman's rank correlation rho

data:  smiltaji$sugas and smiltaji$smilts
S = 83657.35, p-value = 0.01265
alternative hypothesis: true rho is not equal to 0
sample estimates:
       rho
-0.2905304
```

Secinājums: Pastāv statistiski būtiska negatīva korelācija starp sugu skaitu un smilts segumu (-0,29), jo p vērtība ir mazāka par būtiskuma līmeni (0,013<0,05).

Kendela korelācijas koeficients

```
cor.test(smiltaji$sugas,smiltaji$smilts,method="kendall")
```

```
    Kendall's rank correlation tau

data:  smiltaji$sugas and smiltaji$smilts
z = -2.5457, p-value = 0.0109
alternative hypothesis: true tau is not equal to 0
sample estimates:
       tau
-0.2335632
```

Secinājums: Pastāv statistiski būtiska negatīva korelācija starp sugu skaitu un smilts segumu (-0,23), jo p vērtība ir mazāka par būtiskuma līmeni (0,011<0,05).

Regresijas analīze

Dati

```
bietes <- read.table(file = "bietes.txt", header = TRUE)
```

```
str(bietes)
```

```
## 'data.frame':    28 obs. of  2 variables:
##  $ udens: int  0 0 0 0 48 50 48 50 88 88 ...
##  $ svars: num  9 10.3 11.5 14.2 12.2 13.8 14 16.2 14 14.5 ...
```

```
summary(bietes)
```

```
##      udens           svars      
##  Min.   :  0.0   Min.   : 9.00  
##  1st Qu.: 50.0   1st Qu.:14.15  
##  Median :147.5   Median :16.85  
##  Mean   :131.8   Mean   :15.80  
##  3rd Qu.:209.2   3rd Qu.:17.65  
##  Max.   :239.0   Max.   :19.20
```

Pāru regresijas analīze

```
modelis <- lm(svars ~ udens, data = bietes)
summary(modelis)
```

```

Call:
lm(formula = svars ~ udens, data = bietes)

Residuals:
    Min      1Q  Median      3Q     Max 
-3.2541 -0.8541 -0.0317  0.6903  2.6002 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 12.254121   0.506327  24.202  < 2e-16 ***
udens        0.026914   0.003269   8.232 1.03e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.408 on 26 degrees of freedom
Multiple R-squared:  0.7227,    Adjusted R-squared:  0.7121 
F-statistic: 67.77 on 1 and 26 DF,  p-value: 1.031e-08
```

Pāru regresijas analīzes secinājumi

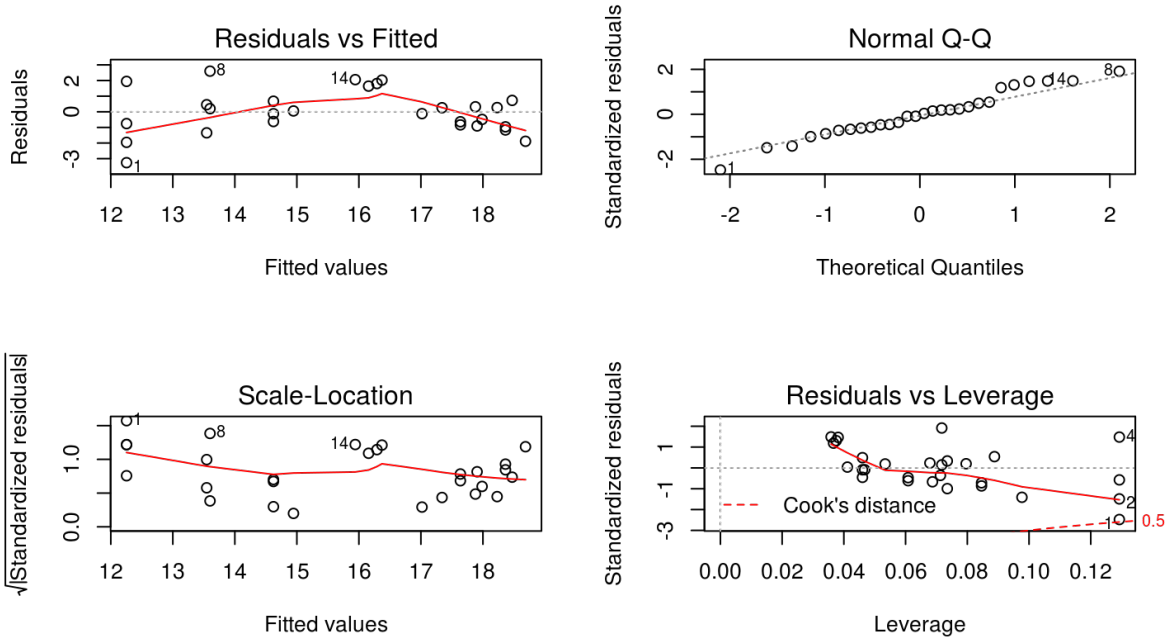
- Determinācijas koeficients Adjusted R-Squared = 0,712, tas ir, ūdens daudzums izskaidro 71,2% no svara vērtību variēšanas
- Gan viss lineārās regresijas modelis kopumā, gan arī atsevišķi šī modeļa koeficienti ir būtiski, jo atbilstošās p-vērtības ir mazākas par būtiskuma līmeni
- Lineārās regresijas vienādojums ir svars = 12,25+0,0269 x udens

Modeļa diagnostika 1

Diagnosticējošos attēlos redzams atlikuma vērtības pret prognozētajām vērtībām, atlikuma vērtību QQ attēls, kā arī attēls, kas parāda katra novērojuma ietekmi uz modeli (apakšējais labais attēls). Ja ietekmes attēlā kāds punkts atrodas aiz raustītās līnijas (Cook's distance virs 0,5), novērojums uzskatāms par ietekmīgu.

```
par(mfrow = c(2, 2))
plot(modelis)
par(mfrow = c(1, 1))
```

Modeļa diagnostika 1



Vērtību prognozēšana 1

Funkcija `coefficients()` no regresijas analīzes objekta paņem regresijas vienādojuma koeficientus.

```
koef <- coefficients(modelis)
koef
```

```
(Intercept)       udens
12.25412112  0.02691369
```

```
udens2 <- 301
raza2 <- koef[1] + koef[2] * udens2
raza2
```

```
(Intercept)
   20.35514
```

Vērtību prognozēšana 2

Vērtību prognozēšanai var izmantot funkciju `predict()`, kurai kā argumentus norāda izveidoto modeli, kā arī datu tabulu ar jaunajām vērtībām. Šajā tabulā jābūt identiskiem kolonnu nosaukumiem kā modelī izmantotajiem regresoriem.

```
jaunidati <- data.frame(udens = 301)
predict(modelis, jaunidati, interval = "prediction")
```

```
       fit      lwr      upr
1 20.35514 17.19722 23.51307
```

Daudzfaktoru regresijas analīzes funkcijas

Regresijas analīzes veikšanai izmanto funkciju `lm()`, kurai kā argumentus norāda pārbaudāmo formulu (regresents~regresors1+regresors2+..), kā arī datu tabulu, kurā atrodas mainīgie.

Regresijas analīzes rezultātu apskata ar funkciju `summary()`.

```
modelis <- lm(regresents~regresors1+regresors2+regresors3+
                ...+regresorsK,data=datu.tabula)

summary(modelis)
```

Dati 1

```
dati <- read.table(file = "renda.txt", header = TRUE)
```

```
str(dati)
```

```
## 'data.frame':    81 obs. of  6 variables:
##  $ gads: int  1924 1925 1926 1927 1928 1929 1930 1931 1932 1933 ...
##  $ hron: num  0.894 1.099 0.891 0.96 0.835 ...
##  $ dec : num  -3.2 -5.8 -3.4 -3.1 -5.5 -3 1.7 -2.8 -1.5 1.7 ...
##  $ jan : num  -8.2 1.5 -5 -5.2 -2.6 -7.1 0.5 -4.5 0.3 -6.5 ...
##  $ feb : num  -7.1 1.3 -4.7 -3.7 -3.6 -14 -3.7 -6.2 -6.5 -4.3 ...
##  $ mar : num  -3.3 -1.5 -0.6 1.7 -2.2 -2.3 0.4 -4.9 -5.2 0.1 ...
```

Dati 2

```
summary(dati)
```

```
      gads           hron             dec              jan         
 Min.   :1924   Min.   :0.6280   Min.   :-8.800   Min.   :-12.900  
 1st Qu.:1944   1st Qu.:0.9100   1st Qu.:-5.500   1st Qu.: -6.500  
 Median :1964   Median :1.0090   Median :-3.100   Median : -3.600  
 Mean   :1964   Mean   :0.9923   Mean   :-2.499   Mean   : -4.136  
 3rd Qu.:1984   3rd Qu.:1.0860   3rd Qu.: 1.100   3rd Qu.: -1.300  
 Max.   :2004   Max.   :1.1910   Max.   : 3.800   Max.   :  2.400  
      feb               mar         
 Min.   :-14.000   Min.   :-10.800  
 1st Qu.: -6.600   1st Qu.: -3.100  
 Median : -4.000   Median : -0.900  
 Mean   : -4.257   Mean   : -1.259  
 3rd Qu.: -1.000   3rd Qu.:  0.500  
 Max.   :  3.700   Max.   :  3.700  
```

Modeļa definēšana

```
modelis <- lm(hron ~ dec + jan + feb + mar, data = dati)

summary(modelis)
```

```
##
## Call:
## lm(formula = hron ~ dec + jan + feb + mar, data = dati)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.234333 -0.066903  0.007898  0.072541  0.187510
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.031433   0.018172  56.759  < 2e-16 ***
## dec         -0.004039   0.003105  -1.301  0.19733
## jan         -0.005548   0.003789  -1.464  0.14728
## feb          0.012333   0.003700   3.333  0.00133 **
## mar          0.015654   0.004704   3.328  0.00135 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.09739 on 76 degrees of freedom
## Multiple R-squared:  0.3508, Adjusted R-squared:  0.3167
## F-statistic: 10.27 on 4 and 76 DF,  p-value: 1.061e-06
```

Regresijas analīze - secinājumi

- Regresijas modelis izskaidro 31,7% no vērtību variēšanas.
- Modelis kopumā ir būtisks (p vērtība <0,001), bet atsevišķu faktoru (decembra un janvāra temperatūras) ietekme nav būtiska (p vērtības attiecīgi 0,1973 un 0,1473).
- Tā kā atsevišķu regresoru ietekme nav būtiska, var veidot vienkāršāku modeli.

Kolinearitātes pārbaude

```
library(car)
vif(modelis)
```

```
##      dec      jan      feb      mar
## 1.073597 1.532734 1.766607 1.390537
```

Secinājums: tā kā visiem mainīgajiem VIF vērtības ir mazas, var uzskatīt, ka šajā modelī nav problēmas ar kolinearitāti.

Modeļa definēšana

```
modelis1 <- lm(hron ~ jan + feb + mar, data = dati)

summary(modelis1)
```

```
##
## Call:
## lm(formula = hron ~ jan + feb + mar, data = dati)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.229956 -0.072878 -0.002383  0.078372  0.187456
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.036591   0.017814  58.191  < 2e-16 ***
## jan         -0.006298   0.003762  -1.674  0.09818 .
## feb          0.011864   0.003699   3.207  0.00195 **
## mar          0.015783   0.004724   3.341  0.00129 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.09783 on 77 degrees of freedom
## Multiple R-squared:  0.3364, Adjusted R-squared:  0.3105
## F-statistic: 13.01 on 3 and 77 DF,  p-value: 5.849e-07
```

Regresijas analīze - secinājumi

- Regresijas modelis izskaidro 31,1% no vērtību variēšanas.
- Modelis kopumā ir būtisks (p vērtība <0,001), bet janvāra temperatūru ietekme joprojām nav būtiska (p vērtība ir 0,0982).
- Tā kā atsevišķu regresoru ietekme nav būtiska, var veidot vienkāršāku modeli.

Modeļa definēšana

```
modelis2 <- lm(hron ~ feb + mar, data = dati)

summary(modelis2)
```

```
##
## Call:
## lm(formula = hron ~ feb + mar, data = dati)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.25327 -0.08051  0.00321  0.07134  0.17581
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.049088   0.016360  64.126  < 2e-16 ***
## feb         0.008973   0.003309   2.712  0.00823 **
## mar         0.014798   0.004741   3.121  0.00253 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.09895 on 78 degrees of freedom
## Multiple R-squared:  0.3122, Adjusted R-squared:  0.2946
## F-statistic: 17.71 on 2 and 78 DF,  p-value: 4.573e-07
```

Regresijas analīze - secinājumi

- 29,46% kopējās regresenta (hron) izkliedes izskaidrojama ar regresoru (feb un mar) lineāro ietekmi;
- pēc Fišera kritērija F =17,7, p-vērtība<0.0001 lineārās regresijas modelis ir statistiski būtisks;
- lineārās regresijas vienādojums ir hron = 1,049088+0,008973 x feb+0,014798 x mar
- pēc Stjūdenta kritērija visi koeficients ir būtisks pie $\alpha = 0,05$.

Modeļu salīdzinājums

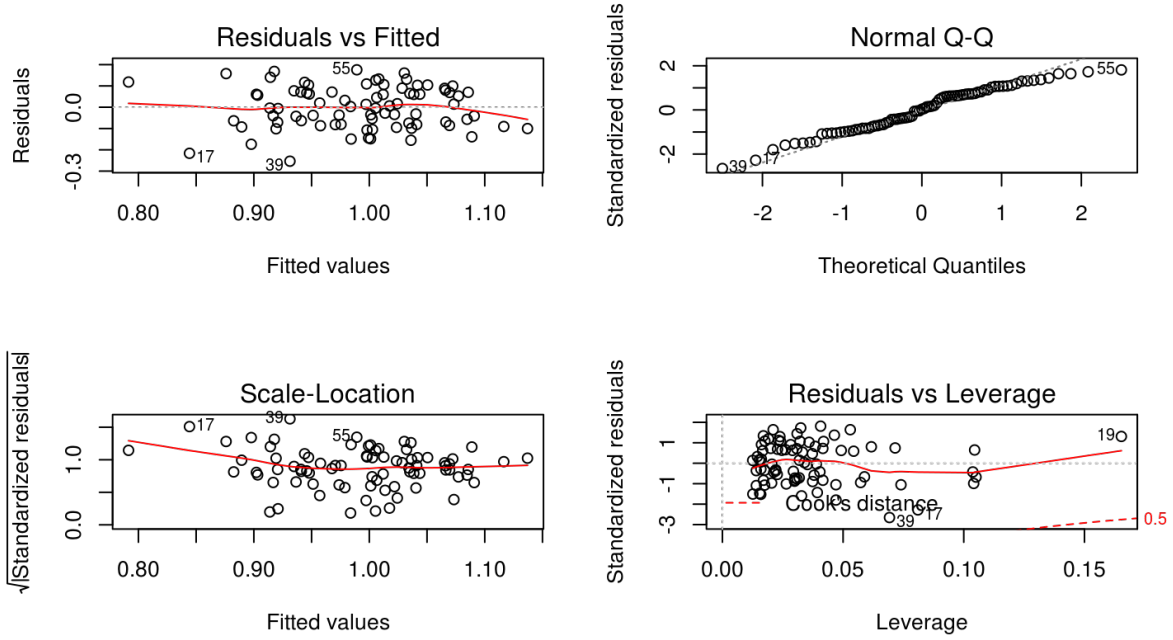
```
anova(modelis,modelis2)
```

```
Analysis of Variance Table

Model 1: hron ~ dec + jan + feb + mar
Model 2: hron ~ feb + mar
  Res.Df     RSS Df Sum of Sq      F Pr(>F)
1     76 0.72089
2     78 0.76375 -2 -0.042865 2.2595 0.1114
```

Secinājums: starp modeļiem nav statistiski būtiskas atšķirības (p vērtība 0,1114 lielāka par būtiskuma līmeni), tāpēc var izvēlēties vienkāršāko modeli.

Modeļa diagnostika

```
par(mfrow = c(2, 2))
plot(modelis2)
```



Dispersijas analīze

Dispersijas analīzes funkcijas

Dispersijas analīzi programmā R veic ar funkciju `aov()`, vai arī kombinējot funkcijas `anova()` un `lm()`.

```
aov(pētāmā.pazīmē~faktors,data=datu.tabula)
```

```
anova(lm(pētāmā.pazīmē~faktors,data=datu.tabula))
```

Lai analīze tiktu veikta pareizi, regresoram (x) ir jābūt izteiktam kā faktoram.

Dati 1

```
miezi <- read.table(file = "miezi.txt",
  header = TRUE,sep="\t",dec=".")
```

```
str(miezi)
```

```
## 'data.frame':    30 obs. of  2 variables:
##  $ skirne: int  1 1 1 1 1 1 1 1 1 1 ...
##  $ raza  : num  66.6 70 63.7 72.1 74.2 67.3 73.4 76.2 76.3 66.4 ...
```

```
head(miezi)
```

```
##   skirne raza
## 1      1 66.6
## 2      1 70.0
## 3      1 63.7
## 4      1 72.1
## 5      1 74.2
## 6      1 67.3
```

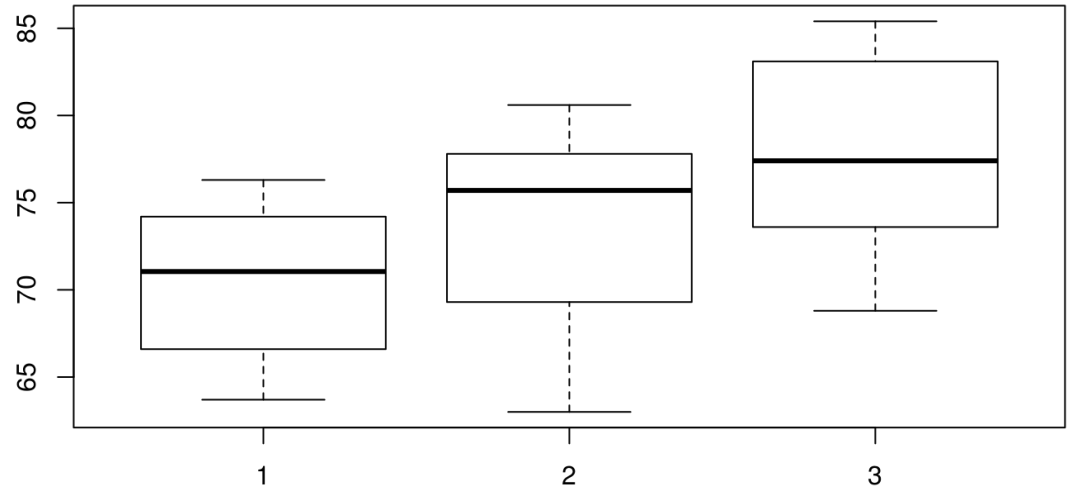
Dati 2

```
summary(miezi)
```

```
     skirne       raza      
 Min.   :1   Min.   :63.00  
 1st Qu.:1   1st Qu.:69.47  
 Median :2   Median :74.60  
 Mean   :2   Mean   :74.04  
 3rd Qu.:3   3rd Qu.:77.45  
 Max.   :3   Max.   :85.40  
```

Vizuālais salīdzinājums

```
boxplot(miezi$raza~miezi$skirne)
```



Pārveidošana par faktoru

Lai pārbaudītu vai mainīgais tiek uztverts kā faktors, izmanto funkciju `is.factor()`. Lai mainīgo pārvērstu par faktoru, izmanto funkciju `as.factor()`.

```
is.factor(miezi$skirne)
```

```
[1] FALSE
```

```
miezi$skirne<-as.factor(miezi$skirne)
is.factor(miezi$skirne)
```

```
[1] TRUE
```

Dispersiju salīdzināšana

Pirms dispersijas analīzes veikšanas grupu dispersijas salīdzina ar funkciju `leveneTest()`, kas atrodas paketē `car`.

```
library(car)
leveneTest(y = miezi$raza, group = miezi$skirne)
```

```
Levene's Test for Homogeneity of Variance (center = median)
      Df F value Pr(>F)
group  2  0.1785 0.8375
      27
```

Secinājums: pie būtiskuma līmenā $\alpha = 0,05$ starp atsevišķi gradācijas klašu dispersijām nepastāv statistiski būtiska atšķirība jeb dispersijas ir homogēnas, jo iegūtā p vērtība (0,84) ir lielāka par būtiskuma līmeni.

Vienfaktora dispersijas analīze

```
anov.miezi <- aov(raza~skirne,data=miezi)
summary(anov.miezi)
```

```
            Df Sum Sq Mean Sq F value Pr(>F)
skirne       2  274.9  137.45   4.596 0.0192 *
Residuals   27  807.6   29.91
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Secinājums: pie būtiskuma līmenā $\alpha = 0,05$ starp gradācijas klašu vidējiem aritmētiskajiem pastāv statistiski būtiska atšķirība jeb faktora šķirne ietekme uz ražu ir būtiska, jo iegūtā p vērtība (0,019) ir mazāka par būtiskuma līmeni.

Faktora ietekmes īpatsvars

Faktora ietekmes īpatsvaru aprēķina dalot faktora kvadrātu summas (Sum Sq) vērtību ar kopējo kvadrātu summu.

```
275/(275+808)
```

```
[1] 0.2539243
```

Secinājums: Faktora šķirne ietekmes īpatsvars uz ražas lielumu ir 25% un nekontrolēto (atlikuma) faktoru summārā iedarbība ir 75% (1-0.25).

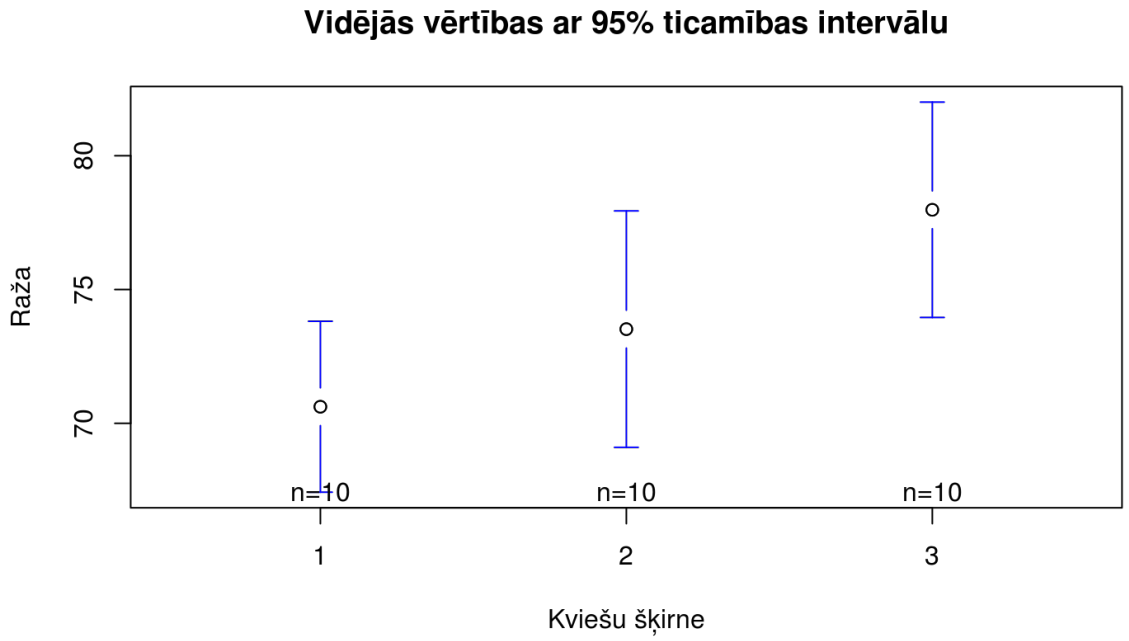
Gradācijas klašu salīdzināšana

```
TukeyHSD(anov.miezi)


  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = raza ~ skirne, data = miezi)

$skirne
    diff       lwr       upr     p adj
2-1 2.90 -3.164184  8.964184 0.4716580
3-1 7.36  1.295816 13.424184 0.0150186
3-2 4.46 -1.604184 10.524184 0.1810875
```

Secinājums: pie būtiskuma līmenā $\alpha = 0,05$ statistiski būtiska atšķirība pastāv starp 3. un 1. šķirnes ražu (p vērtība 0,015), bet starp 1. un 2., kā arī 2. un 3. šķirni atšķirība nav statistiski būtiska (p vērtības lielākas par 0,05).

Vidējo vērtību grafiskais attēlojums

```
library(gplots)
plotmeans(miezi$raza ~ miezi$skirne,
 connect=FALSE,xlab = "Kviešu šķirne",ylab = "Raža",
 main = "Vidējās vērtības ar 95% ticamības intervālu")
```



Dati

```
soja <- read.table(file="soja.txt",header=TRUE)
```

```
str(soja)
```

```
## 'data.frame':    52 obs. of  3 variables:
##  $ gaisma: Factor w/ 2 levels "normala","zema": 2 2 2 2 2 2 2 2 2 2 ...
##  $ stress: Factor w/ 2 levels "ir","nav": 2 2 2 2 2 2 2 2 2 2 ...
##  $ lapas : int  264 200 225 268 215 241 232 256 229 288 ...
```

```
head(soja)
```

```
##   gaisma stress lapas
## 1   zema    nav   264
## 2   zema    nav   200
## 3   zema    nav   225
## 4   zema    nav   268
## 5   zema    nav   215
## 6   zema    nav   241
```

Datu pārbaude

```
names(soja)
```

```
[1] "gaisma" "stress" "lapas"
```

```
is.factor(soja$gaisma)
```

```
[1] TRUE
```

```
is.factor(soja$stress)
```

```
[1] TRUE
```

Dispersiju salīdzināšana

```
library(car)
leveneTest(y=soja$lapas,group=soja$gaisma:soja$stress)
```

```
Levene's Test for Homogeneity of Variance (center = median)
      Df F value Pr(>F)
group  3  0.1963 0.8984
      48
```

Secinājums: pie būtiskuma līmenā $\alpha = 0,05$ starp atsevišķi gradācijas klašu dispersijām nepastāv statistiski būtiska atšķirība jeb dispersijas ir homogēnas, jo iegūtā p vērtība (0,9) ir lielāka par būtiskuma līmeni.

Daudzfaktoru dispersijas analīze 1

Pieraksts faktors1*faktors2 nozīmē, ka pārbauda katra faktora ietekmi, kā arī faktoru kombinācijas ietekmi.

```
modelis <- aov(lapas~gaisma*stress,data=soja)
summary(modelis)
```

```
             Df Sum Sq Mean Sq F value   Pr(>F)
gaisma        1  42752   42752  47.749 1.01e-08 ***
stress        1  14858   14858  16.595 0.000173 ***
gaisma:stress 1     26      26   0.029 0.864570
Residuals    48  42976     895
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Daudzfaktoru dispersijas analīze 2

Secinājums: pie būtiskuma līmenā $\alpha = 0,05$ sojas lapu laukumu būtiski ietekmē gaismas daudzums (p vērtība <0,0001) un stresa līmenis (p vērtība 0,0002), bet faktoru kombinācijas ietekme nav būtiska (p vērtība 0,86).

Ietekmes īpatsvara noteikšana

```
kv.sum <- 42752 + 14858 + 26 + 42976
42752/kv.sum  #gaisma
```

```
[1] 0.4249195
```

```
14858/kv.sum  #stress
```

```
[1] 0.1476762
```

**Secinājums:* Faktora gaisma ietekmes īpatsvars uz lapu virsmas laukumu ir 42,5%, stresa ietekmes īpatsvars ir 14,8%.

Gradācijas klašu salīdzināšana

Šajā gadījumā Post-Hoc testu var arī neveikt, jo faktoram ir tikai divas gradācijas klases - ja faktora ietekme ir būtiska, tad, attiecīgi, šīs klases būtiski atšķiras.

```
TukeyHSD(modelis,"gaisma")
```

```
  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = lapas ~ gaisma * stress, data = soja)

$gaisma
                  diff       lwr       upr p adj
zema-normala -57.34615 -74.03228 -40.66003     0
```

Kruskal-Wallis tests

```
kruskal.test(raza~skirne,data=miezi)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  raza by skirne
## Kruskal-Wallis chi-squared = 6.7123, df = 2, p-value = 0.03487
```

Secinājums: pie būtiskuma līmeņa $\alpha = 0.05$ varam secināt, ka miežu sķirņu dati nenāk no vienas populācijas (p=0.035) (to vērtību sadalījums nav vienāds).