Kodutöö I: vihjeid R-is tegemiseks

Mark Gimbutas

15.11.2019

Table of Contents

[2. Keskmine, mediaan, histogramm lisanditega 1](#_Toc24750466)

[3. t-test 2](#_Toc24750467)

[4. ANOVA 4](#_Toc24750468)

[5. Mitteparameetriline ANOVA (Kruskal-Wallise test) 7](#_Toc24750469)

[6. Korrelatsioonanalüüs 8](#_Toc24750470)

[7. Korrelatsiooni tugevus ja olulisus erinevate valimimahtude puhul 9](#_Toc24750471)

[8. Regressioonanalüüs 12](#_Toc24750472)

[9. Kahefaktoriline ANOVA koosmõjuga 13](#_Toc24750473)

[10. Kovariatsioonanalüüs (ANCOVA) 14](#_Toc24750474)

Järgnevad esimese kodutöö ülesannete R-i abil lahendamiseks vajalikud vihjed.

# 2. Keskmine, mediaan, histogramm lisanditega

x <- c(1,2,3,4,10)  
mean(x)

## [1] 4

median(x)

## [1] 3

var(x) # dispersioon

## [1] 12.5

sd(x) / mean(x) # variatsioonikoefitsient

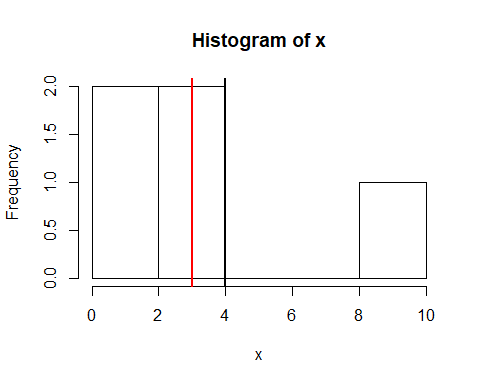
## [1] 0.8838835

quantile(x, 0.25) # 25% kvantiil (ehk alumine kvartiil)

## 25%   
## 2

Pildi joonistamiseks kasutame R-i baas-võimalusi. Esialgu teeme histogrammi. Seejärel lisame sellele histogrammile kaks sirget joont: üks keskmisele teine mediaanile

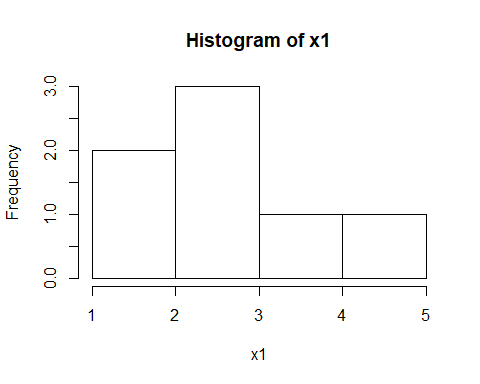
hist(x) # histogramm  
abline(v = mean(x), lwd=2) # v nagu vertical, lwd nagu line width  
abline(v = median(x), col="red", lwd=2) # col nagu colour



Üldiselt, R põhi-funktsioonidega graafikute tegemise kohta on palju häid näiteid teinud ja üles laadinud Märt Möls lehele <http://www-1.ms.ut.ee/mart/R/Rgraafika.html>.

# 3. t-test

x1 <- c(1,2,3,3,3,4,5)  
x2 <- c(4,5,5,6,7)  
hist(x1)



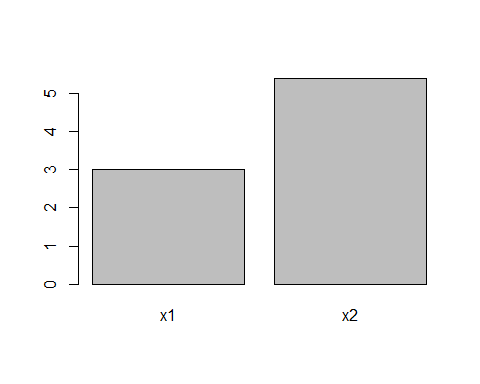
hist(x2)



t.test(x1, x2, var.equal = T)

##   
## Two Sample t-test  
##   
## data: x1 and x2  
## t = -3.3245, df = 10, p-value = 0.007688  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -4.0084984 -0.7915016  
## sample estimates:  
## mean of x mean of y   
## 3.0 5.4

barplot(c(x1 = mean(x1), x2 = mean(x2)))



Testi väljund peaks olema küllalt ilmne. Real t = -3.3245, df = 10, p-value = 0.007688 on kirjas t-statistik, tema vabadusastmed ja vastav p-väärtus. Paar rida allpool näeme keskmiste *vahe* usaldusintervalli ja valimikeskmisi endid.

T-testi muude variantide jaoks (sõltuvad või sõltumatud valimid, rühmade varieeruvused võrdsed või mitte) tuleks uurida abifaili ?t.test.

# 4. ANOVA

Esialgu tekitame andmestiku.

faktor <- rep(c("A", "B", "C"), each = 5)  
x <- c(1,3,4,5,7,  
 4,6,6,7,8,  
 7,8,8,9,9)  
andmed <- data.frame(faktor, x)

Kõiki lineaarseid mudeleid (ANOVA, ANCOVA, lineaarne regressioon, üldine lineaarne mudel) saab R-is hinnata funktsiooniga lm. Selle funktsiooni tuumaks on nn regressiooni valem:

* vasakul pool lainekest on sõltuva tunnuse nimi andmetabelis
* keskel on laineke ehk tilde, mille vähemasti minu klaviatuuril saab Shift+(see nupp number ühest vasakul) ja seejärel tuleb tühikut vajutada (alles siis ilmub laineke nähtavale)
* paremal pool lainekest on argumenttunnus(ed)

Nagu kõigi muude asjadega R-is, tuleks vaevaga välja arvutatud objekt salvestada. Antud juhul siis nimega m nagu mudel.

m <- lm(x ~ faktor, data = andmed)

Tüüp I ANOVA tabeli saab kätte funktsiooniga anova:

anova(m)

## Analysis of Variance Table  
##   
## Response: x  
## Df Sum Sq Mean Sq F value Pr(>F)   
## faktor 2 44.133 22.0667 8.3797 0.005277 \*\*  
## Residuals 12 31.600 2.6333   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Mõjude hinnangud ja palju muud huvitavat (sealhulgas R-ruut, kogu mudeli olulisuse F-test, jne) saab kätte funktsiooniga summary:

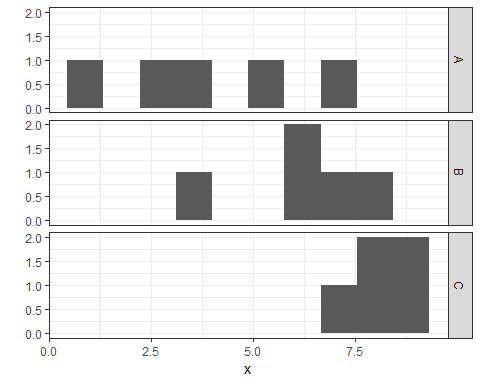
summary(m)

##   
## Call:  
## lm(formula = x ~ faktor, data = andmed)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3.0 -0.6 -0.2 0.8 3.0   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 4.0000 0.7257 5.512 0.000134 \*\*\*  
## faktorB 2.2000 1.0263 2.144 0.053250 .   
## faktorC 4.2000 1.0263 4.092 0.001493 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.623 on 12 degrees of freedom  
## Multiple R-squared: 0.5827, Adjusted R-squared: 0.5132   
## F-statistic: 8.38 on 2 and 12 DF, p-value: 0.005277

Pane tähele, et faktori võrdlus-tasemeks võetakse faktori kõige esimene tase (tähestikulises järjekorras). Antud juhul siis A.

Vahelduse mõttes teeme graafiku ühe R-i väga kasuliku lisapaketiga.

#install.packages("ggplot2") # kui see juba tehtud ei ole (1 kord arvuti kohta)  
library(ggplot2)  
qplot(x, # mis tunnus läheb x-teljele   
 data = andmed, # andmetabeli nimi  
 geom = "histogram", # graafiku tüüp  
 facets = faktor~., # tee iga faktori "faktor" taseme kohta üks graafik ja pane ritta  
 bins = 10) + # histogrammi lisaparameeter: mitu tulpa histogrammile teha  
 theme\_bw() # graafik üldiselt must-valge muljega



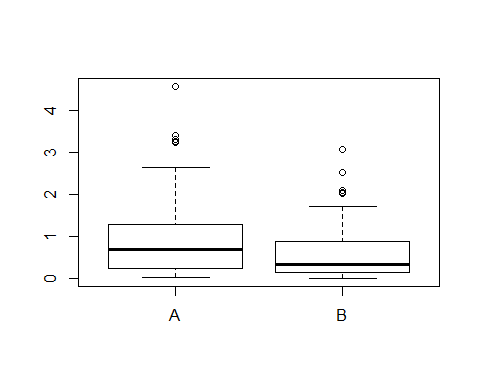
# 5. Mitteparameetriline ANOVA (Kruskal-Wallise test)

Tekitame andmestiku. Kasutame sealjuures juhuslike arvude tekitamise võimalust: R oskab nimelt küsimise peale meile genereerida (pseudo)juhuslikke arve õige mitmest jaotusest.

faktor <- rep(c("A", "B"), each=100)  
x <- c(rexp(100, 1),   
 rexp(100, 2)) # 100 arvu eksponentjaotusest parameetriga 2  
andmed5 <- data.frame(faktor, x)

Esitame andmed karpdiagrammil. Ühel karpdiagrammil võime kõrvuti kuvada mitu karpi, kui määrame selle faktori mille järgi need karbid joonistada.

boxplot(x~faktor, data=andmed5)



Kasutame rühmade A ja B võrdlemiseks mitteparameetrilist ANOVA-t, täpsemalt siis Kruskal-Wallise testi.

kruskal.test(x~faktor, data=andmed5)

##   
## Kruskal-Wallis rank sum test  
##   
## data: x by faktor  
## Kruskal-Wallis chi-squared = 11.091, df = 1, p-value = 0.0008674

# 6. Korrelatsioonanalüüs

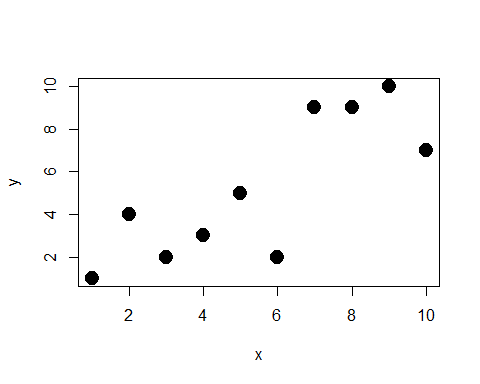
x <- 1:10  
y <- c(1,4,2,3,5,2,9,9,10,7)  
cor(x,y) # arvutab Pearsoni korrelatsioonikordaja

## [1] 0.8053158

cor.test(x,y) # test kas korrelatsioonikordaja on null? (Pearsoni oma vaikimisi)

##   
## Pearson's product-moment correlation  
##   
## data: x and y  
## t = 3.8421, df = 8, p-value = 0.004932  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.3564028 0.9521542  
## sample estimates:  
## cor   
## 0.8053158

plot(x,y, # andmed graafikule  
 pch=20, # täpikesed seest täidetud   
 cex=3) # täpikesed 3 korda suuremad



cor(x,y, method="spearman") # arvutab Spearmani korrelatsioonikordaja

## [1] 0.7866

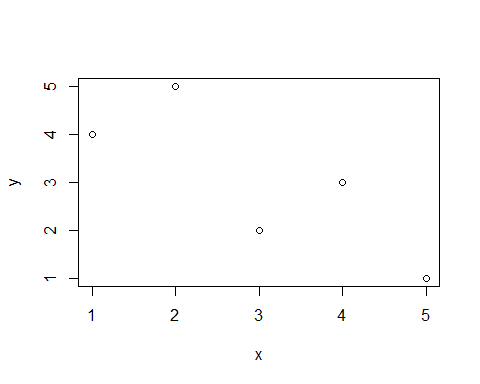
cor.test(x,y, method = "spearman")

## Warning in cor.test.default(x, y, method = "spearman"): Cannot compute  
## exact p-value with ties

##   
## Spearman's rank correlation rho  
##   
## data: x and y  
## S = 35.211, p-value = 0.00695  
## alternative hypothesis: true rho is not equal to 0  
## sample estimates:  
## rho   
## 0.7866

# 7. Korrelatsiooni tugevus ja olulisus erinevate valimimahtude puhul

x <- 1:5  
y <- c(4,5,2,3,1)  
plot(x,y)

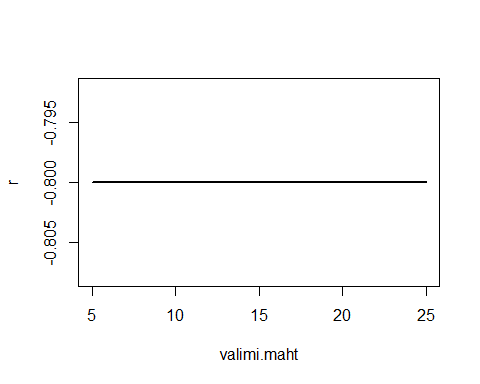


cor.test(x,y)

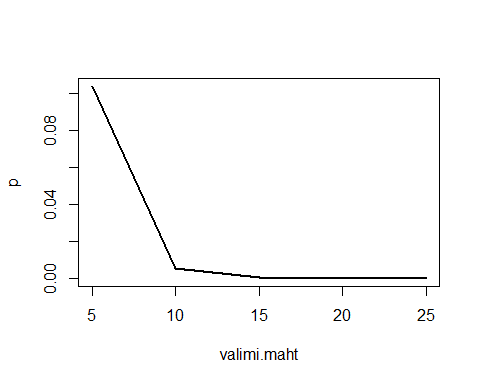
##   
## Pearson's product-moment correlation  
##   
## data: x and y  
## t = -2.3094, df = 3, p-value = 0.1041  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## -0.9861962 0.2796400  
## sample estimates:  
## cor   
## -0.8

Nüüd hakkame oma andmestikku lihtsa copy-paste-ga paljundama ja korrelatsioonikordaja ning selle olulisuse muutust jälgima. Teeme seda for-tsükli abil.

valimi.maht <- rep(NA, 5) # teeme tühjad pesad tulemuste korjamiseks  
r <- rep(NA, 5)   
p <- rep(NA, 5)  
for(i in 1:5){  
 # teeme testi ja salvestame kogu selle testi  
 korrelatsiooni.test <- cor.test(rep(x, i), rep(y, i))  
 # korjame välja korrelatsioonikordaja väärtuse ja topime ta vektori r i-ndale kohale  
 r[i] <- korrelatsiooni.test$estimate  
 # korjame välja testi p-väärtuse ja topime ta vektori p i-ndale kohale  
 p[i] <- korrelatsiooni.test$p.value  
 # Valimi mahu oskame ise ära arvata, teades et alguses oli 5 ja siis me kopeerisime  
 valimi.maht[i] <- 5\*i  
}  
plot(valimi.maht, r, type="l", lwd=2) # lwd nagu line width



plot(valimi.maht, p, type="l", lwd=2)

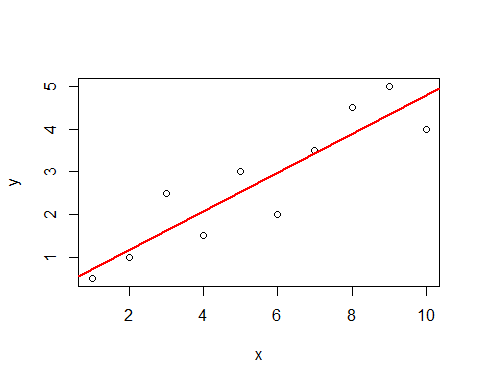


# 8. Regressioonanalüüs

x <- 1:10  
y <- c(1,2,5,3,6,4,7,9,10,8)/2  
m8 <- lm(y~x)  
summary(m8)

##   
## Call:  
## lm(formula = y ~ x)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.97576 -0.48409 -0.04848 0.58485 0.87879   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.26667 0.47119 0.566 0.586952   
## x 0.45152 0.07594 5.946 0.000344 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.6898 on 8 degrees of freedom  
## Multiple R-squared: 0.8155, Adjusted R-squared: 0.7924   
## F-statistic: 35.35 on 1 and 8 DF, p-value: 0.0003436

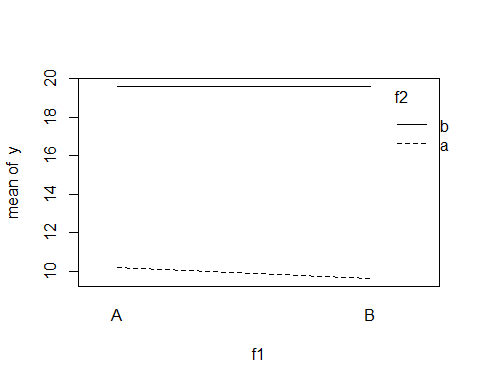
plot(x,y)  
abline(a=.2667, b=.45152, col="red", lwd=2) # a on intercept ja b on slope



# 9. Kahefaktoriline ANOVA koosmõjuga

Loome andmed ja kanname rühmade keskmised graafikule spetsiaalse funktsiooni interaction.plot abil. Koosmõju puudumine tähendab seda, et ühe faktori mõju ei sõltu teise faktori konkreetsest tasemest. Peamõju on olemas, kui faktori kõigi tasemete keskmised *ei ole* võrdsed. Alumisel graafikul faktoril f2 on peamõju olemas, sest tasemete a ja b keskmised on erinevad. Peamõju puudumine tähendab et faktori kõigi tasemete keskmised on võrdsed. Faktoril f1 peamõju puudub.

set.seed(123456789)  
f1 <- rep(c("A", "B"), each = 10)  
f2 <- rep(c("a", "b", "a", "b"), each = 5)  
y <- rep(c(10,20,10,20), each = 5) + # grupikeskmised pluss  
 rnorm(20) # juhuslik viga  
interaction.plot(f1, f2, y)



Vaatame kas ANOVA tulemused ühtivad meie oletustega.

m <- lm(y ~ f1 \* f2)  
anova(m)

## Analysis of Variance Table  
##   
## Response: y  
## Df Sum Sq Mean Sq F value Pr(>F)   
## f1 1 0.42 0.42 0.3189 0.5801   
## f2 1 469.90 469.90 359.2826 2.186e-12 \*\*\*  
## f1:f2 1 0.39 0.39 0.2965 0.5936   
## Residuals 16 20.93 1.31   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Nagu näha, oluline koosmõju puudub (p=0.59), samuti faktori f1 peamõju (p=0.58), aga faktoril f2 on peamõju (p<0.0001).

# 10. Kovariatsioonanalüüs (ANCOVA)

Andmete tekitamine.

f <- rep(c("A", "B"), each = 10) # faktor  
x1 <- rnorm(20) # üks kovariaat  
x2 <- rnorm(20) # teine kovariaat  
y <- x1 + x2 + 1\*(f == "A") + rnorm(20) # sõltuv muutuja

Nüüd teeme kovariatsioonanalüüsi.

m <- lm(y ~ x1 + x2 + f)  
anova(m)

## Analysis of Variance Table  
##   
## Response: y  
## Df Sum Sq Mean Sq F value Pr(>F)   
## x1 1 31.854 31.854 47.389 3.675e-06 \*\*\*  
## x2 1 18.824 18.824 28.004 7.300e-05 \*\*\*  
## f 1 8.338 8.338 12.404 0.002829 \*\*   
## Residuals 16 10.755 0.672   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1