MD1

Pēteris Račinskis pr20015

4/7/2021

1. mājas darbs Mate6029

1. uzdevums

a) Varbūtību telpas

Bibliotēka darbam ar varbūtību telpām:

```
library(prob)
```

2 kauliņa metienu modelēšana ar varbūtību telpu

```
roll2 = iidspace(c(1:6), ntrials=2, probs=rep(1/6, times=6))
A = subset(roll2, X1 == X2)
B = subset(roll2, X1+X2 >= 7 & X1+X2 <= 10)
C = subset(roll2, X1+X2 == 2 | X1+X2 == 7 | X1+X2 == 8)
test1 = subset(roll2, X1 == 1)
test2 = subset(roll2, X2 == 2)
```

Notikumu varbūtības:

```
Prob(A)
```

```
## [1] 0.1666667
```

```
Prob(B)
```

```
## [1] 0.5
```

```
Prob(C)
```

```
## [1] 0.3333333
```

Notikumu šķēlums un varbūtību reizinājums:

```
(Prob(intersect(A,B,C)) == Prob(A) * Prob(B) * Prob(C))
```

```
## [1] FALSE
```

Neatkarības pārbaudes: A un B; B un C:

```
(Prob(intersect(A,B)) == Prob(A) * Prob(B))
```

```
## [1] FALSE
```

```
(Prob(intersect(B,C)) == Prob(B) * Prob(C))
```

```
## [1] FALSE
```

3 kauliņa metienu modelēšana:

```
roll3 = iidspace(c(1:6), ntrials=3, probs=rep(1/6, times=6))
```

Notikumi - 3 dažādi skaitļi un visi vienlīdzīgi:

```
A = subset(roll3, !((X1 == X2) | (X1 == X3) | (X2 == X3)))
B = subset(roll3, X1 == 1 | X2 == 1 | X3 == 1)
```

Nosacītā varbūtība:

```
Prob(B, given = A)
```

```
## [1] 0.5
```

b) Diskrēti gadījuma lielumi

Palīgfunkciju definīcija (koda pārskatāmības nolūkos):

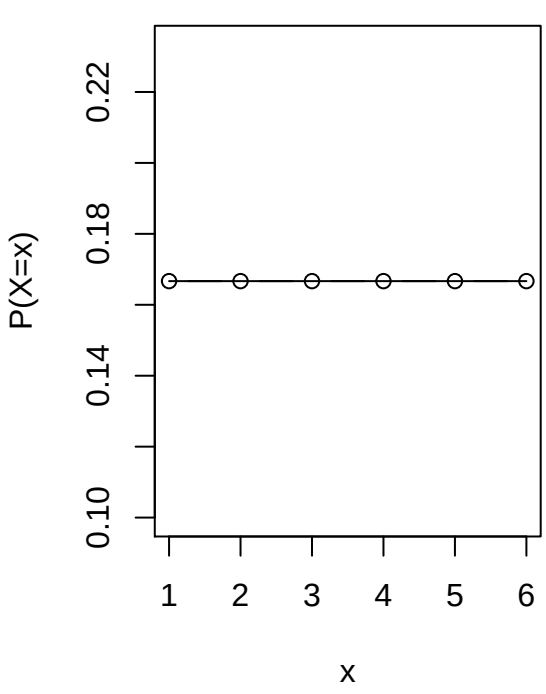
```
dist_mean <- function(dist) {
  sum(dist[1,]*dist[2,])
}
dist_var <- function(dist) {
  mu <- dist_mean(dist)
  sum((dist[1,]^2)*dist[2,])-mu^2
}
dist_sigma <- function(dist) {
  sqrt(dist_var(dist))
}
interval_prob <- function(dist, a, b) {
  sum(dist[2, dist[1,] > a & dist[1,] < b])
}
single_prob <- function(dist, x) {
  dist[2, dist[1,] == x]
}
```

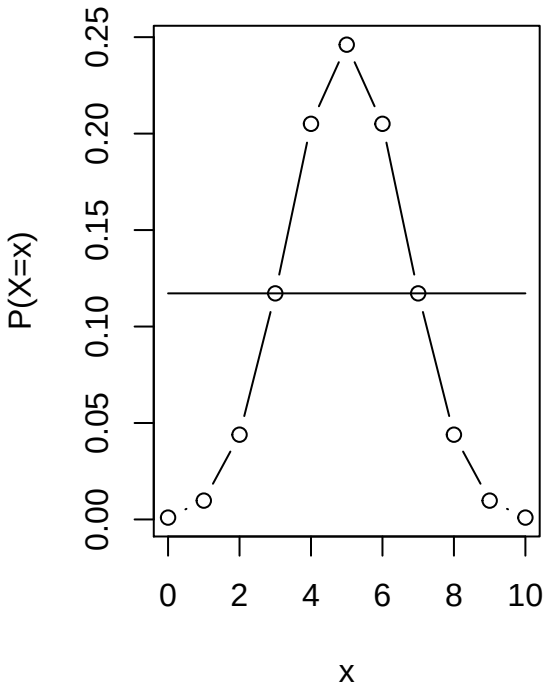
Apvienota funkcija, kas izpilda uzdevuma nosacījumus:

```
combined <- function(dist, x, a, b) {
  plot(dist[1,], dist[2,], type="b",xlab="x",ylab="P(X=x)")
  flatline <- rep(single_prob(dist,x),times=length(dist[1,]))
  lines(dist[1,],flatline)
  list(mu=dist_mean(dist),
       var=dist_var(dist),
       sig=dist_sigma(dist),
       interval=interval_prob(dist,a,b))
}
```

Pielietojums kauliņa metienam un binomiālajam sadalījumam:

```
d <- t(rolldie(1,makespace = TRUE))
n <- 10
k <- 0:n
p <- 0.5
b <- rbind(k,dbinom(k,n,p))
par(mfrow = c(1,2))
r1 <- combined(d, 4, 1, 3)
r2 <- combined(b, 3, 2, 7)
```





Kauliņa sadalījums:

```
r1
```

```
## $mu
## [1] 3.5
##
## $var
## [1] 2.916667
##
## $sig
## [1] 1.707825
##
## $interval
## [1] 0.1666667
```

Binomiālais sadalījums:

```
r2
```

```
## $mu
## [1] 5
##
## $var
## [1] 2.5
##
## $sig
## [1] 1.581139
##
## $interval
## [1] 0.7734375
```

2. uzdevums

Iterācijas funkcija:

```
nsize_repeat <- function(nsizes, params, times) {
  sapply(nsizes, function(i) {
    params$n <- i
    sapply(1:times, function(j) mean(params$cb(params)))
  })
}
```

Izlasi veidojošās funkcijas:

```
chi_callback <- function(params) {
  rchisq(params$n, params$df)
}
exp_callback <- function(params) {
  rexp(params$n, params$rate)
}
uni_callback <- function(params) {
  runif(params$n, params$min, params$max)
}
lnm_callback <- function(params) {
  rlnorm(params$n, params$meanlog, params$sdlog)
}
```

Izlašu konfigurācijas:

```
config = list(
  chi=list(cb=chi_callback,df=1),
  exp=list(cb=exp_callback,rate=1),
  uni=list(cb=uni_callback,min=-1,max=1),
  lnm=list(cb=lnm_callback,meanlog=1,sdlog=1)
)
```

Izlašu veidošana:

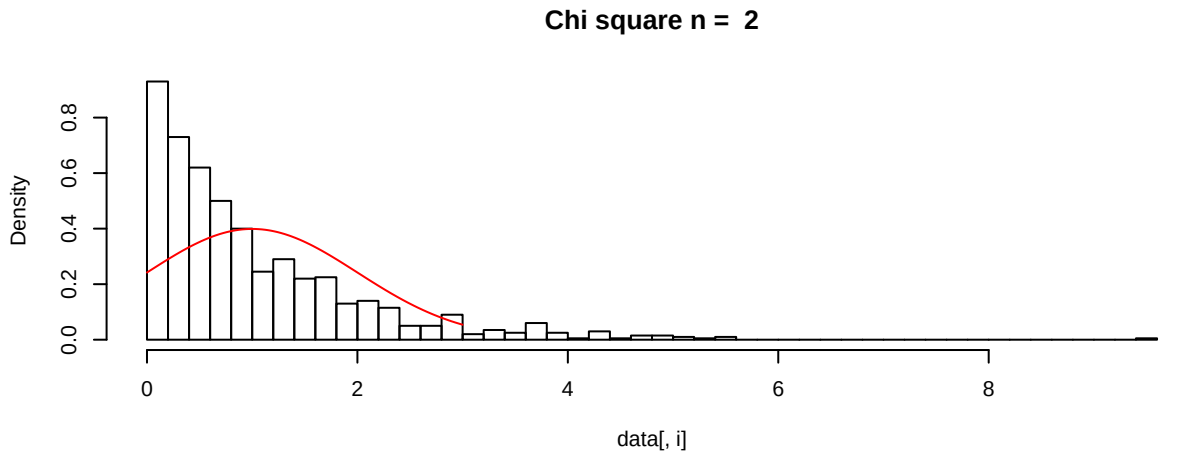
```
nsizes = c(2,10,50)
times = 1000
results <- lapply(config, function(i) nsize_repeat(nsizes, i, times))
```

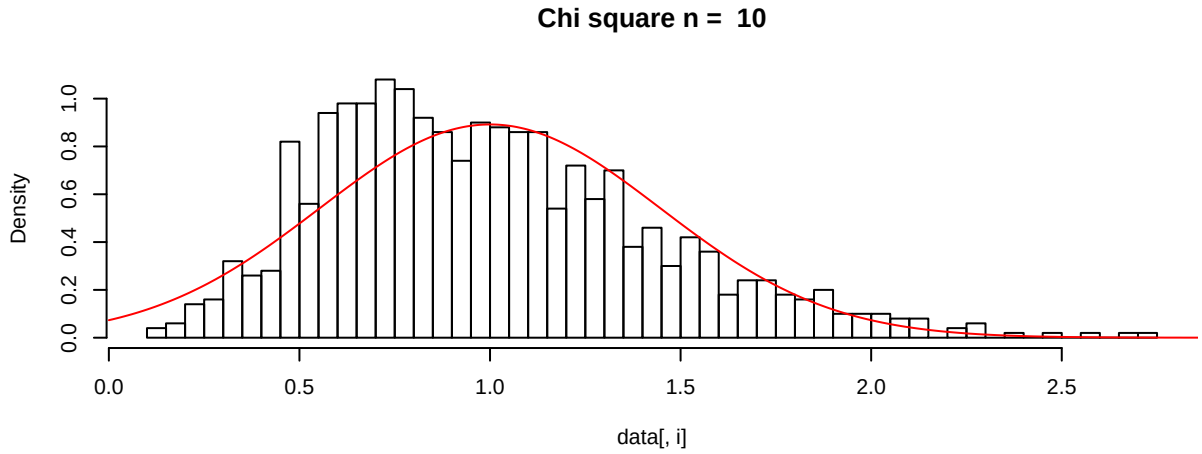
Grafiku zīmēšanas funkcija un reizinātāji:

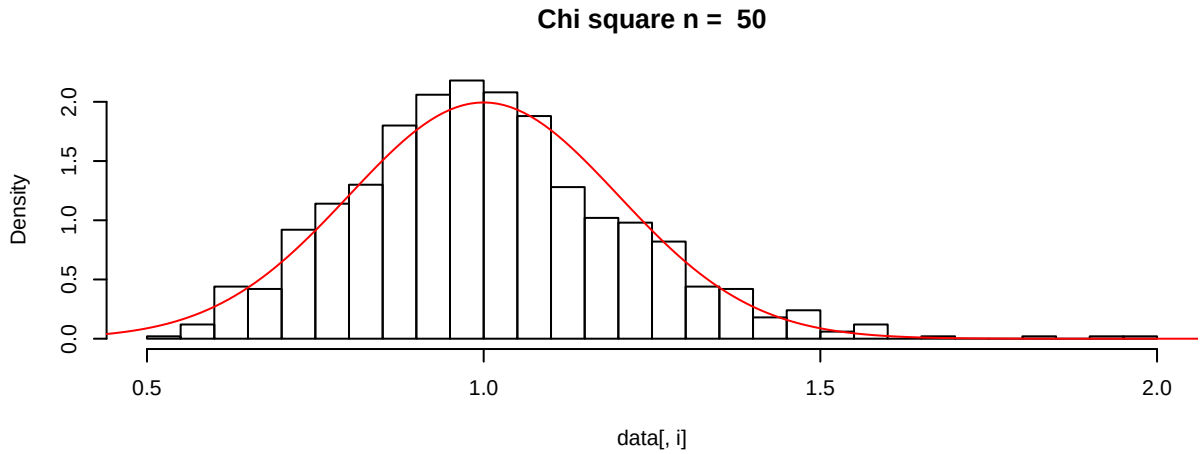
```
compare <- function(arg, data, mu, var, nvals, name) {
  par(mfrow = c(length(nvals),1))
  sapply(1:length(nvals), function(i) {
    y <- dnorm(arg, mu, sqrt(var*nvals[i]))
    hist(data[,i],freq=FALSE,breaks=50,main=paste(name,1/nvals[i]))
    lines(arg,y,col="red")
  })
  par(mfrow = c(1,1))
}

ns <- c(1/2, 1/10, 1/50)
```

Chi-square grafiki:

```
xax <- seq(0,3,0.01)
compare(xax,results$chi,1,2,ns,"Chi square n = ")
```

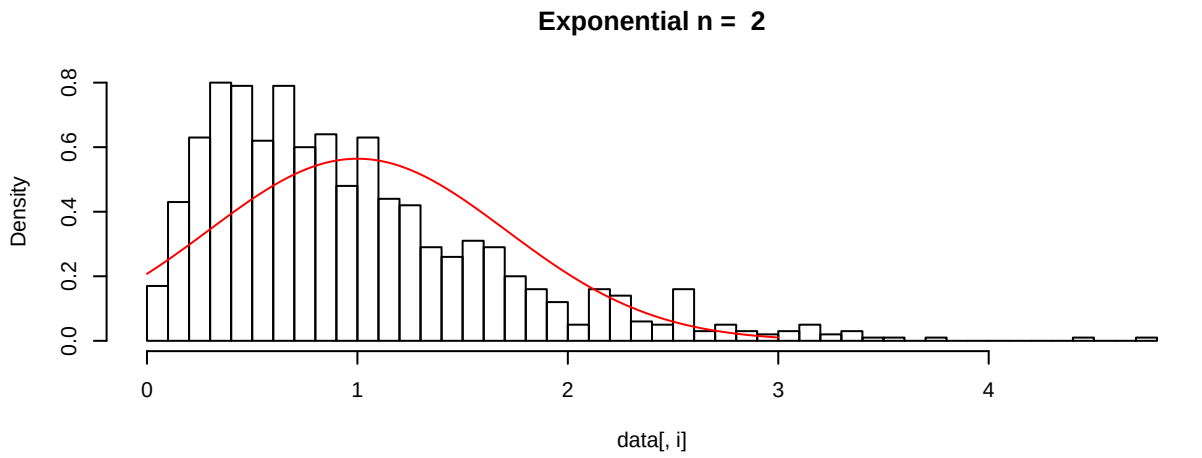


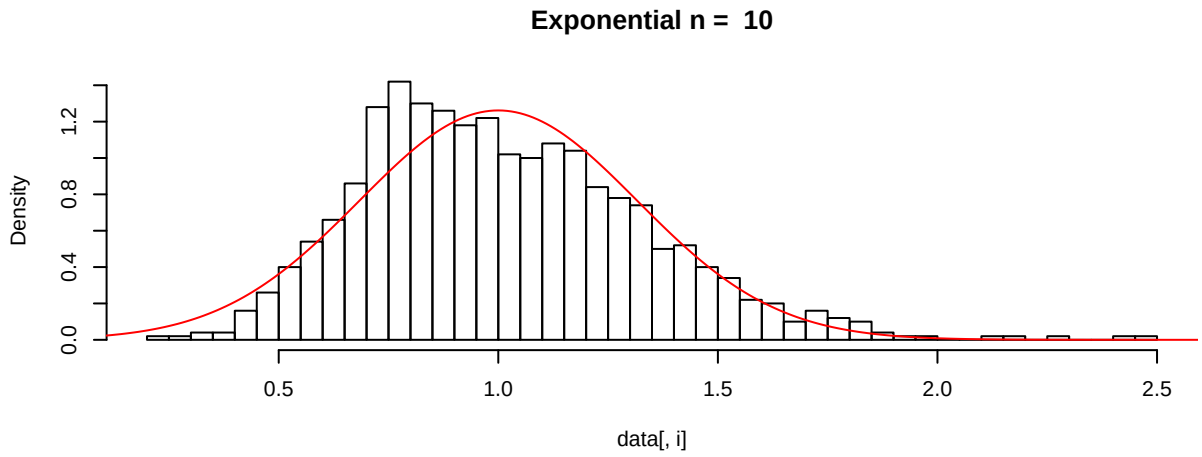


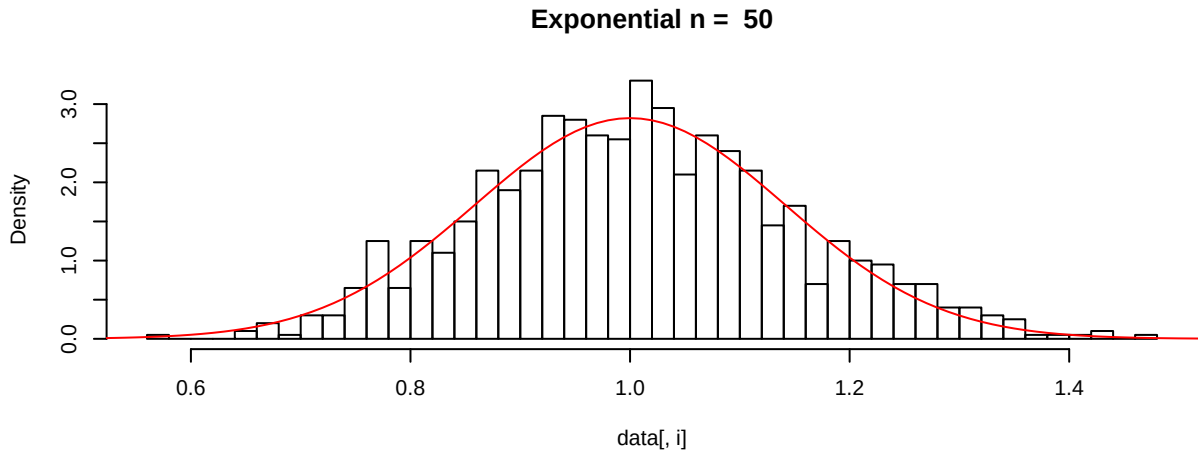


Exp grafiki:

```
compare(xax,results$exp,1,1,ns,"Exponential n = ")
```

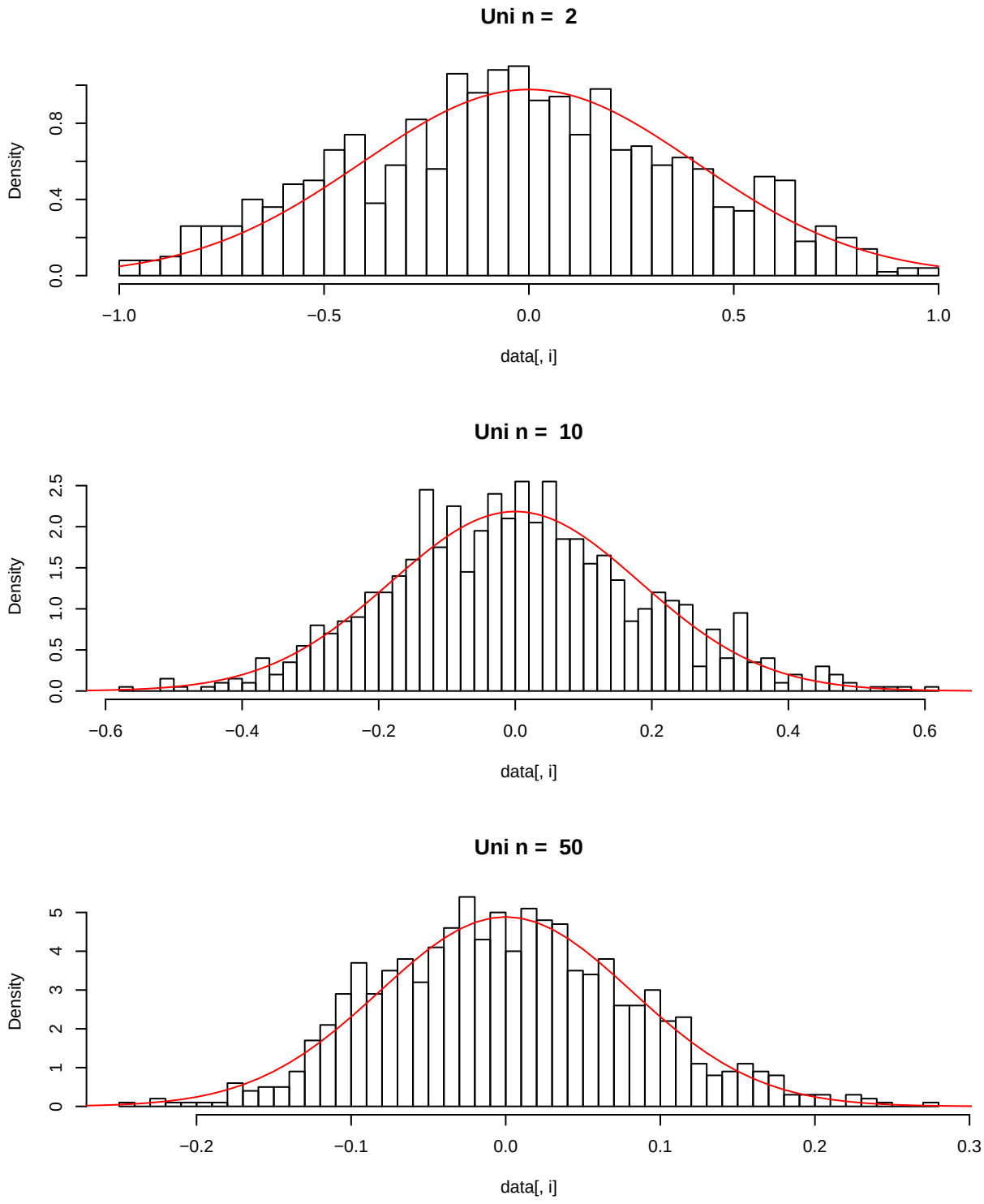






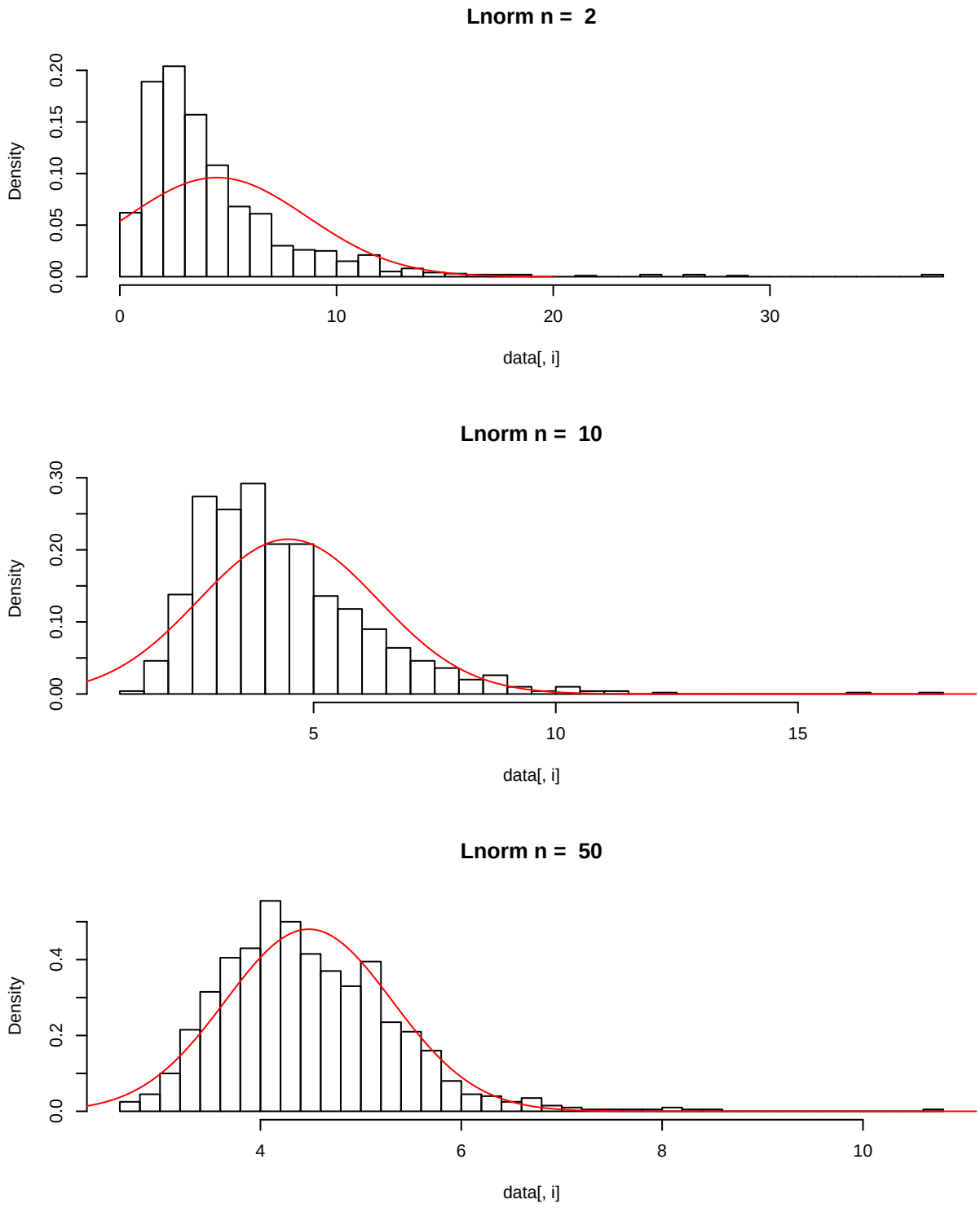
Uniform grafiki:

```
xax <- seq(-1,1,0.01)
compare(xax,results$uni,0,1/3,ns,"Uni n = ")
```



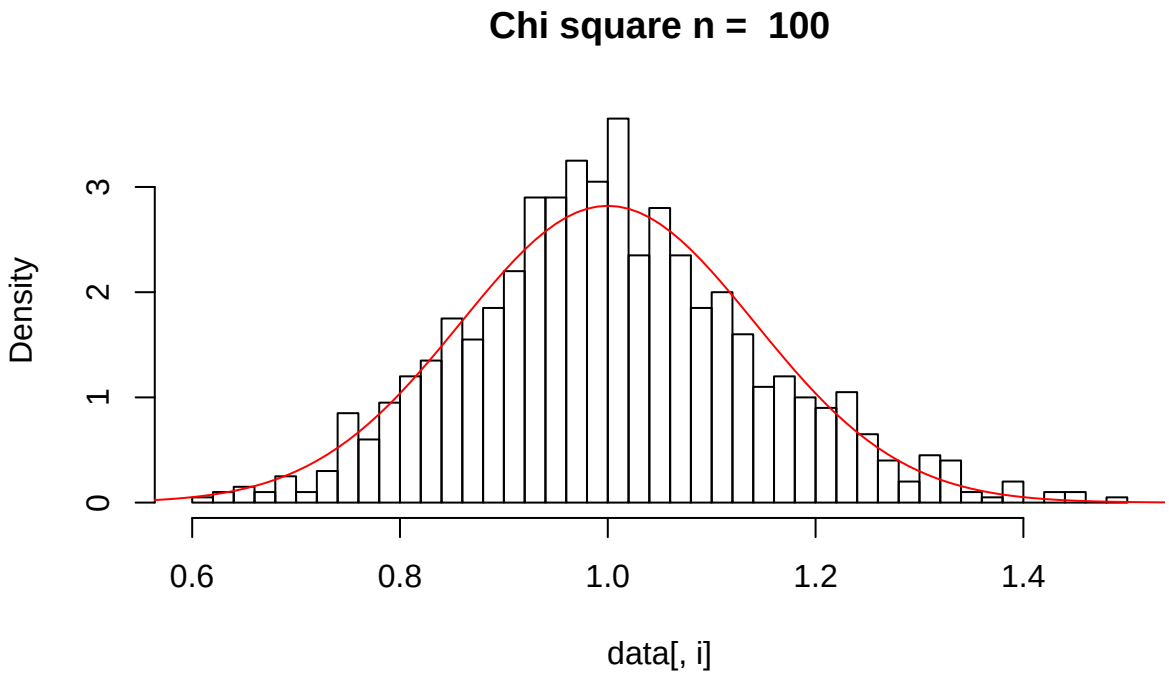
Lognorm grafiki:

```
lnm_mean <- exp(1+1/2)
lnm_var <- (exp(1) - 1) * exp(3)
xax <- seq(0,20,0.01)
compare(xax,results$lnm,lnm_mean,lnm_var,ns,"Lnorm n = ")
```

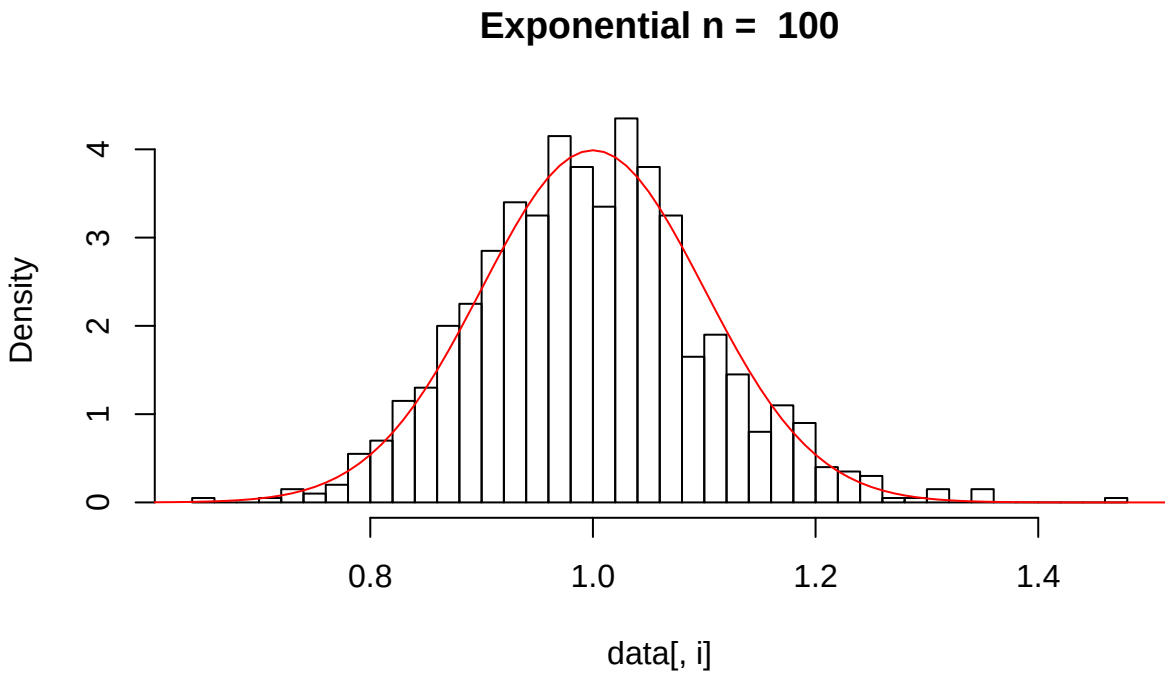


Pārrēķinātie rezultāti pie n = 100 un atbilstošie grafiki:

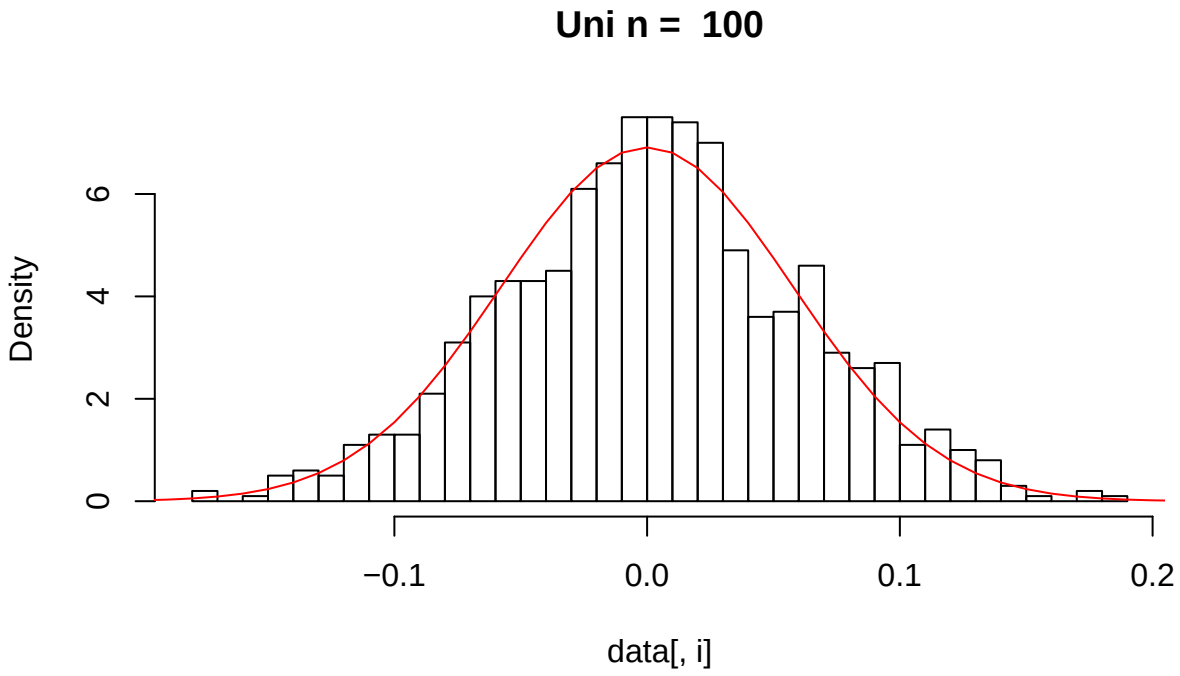
```
nsizes <- c(100)
ns <- c(1/100)
times <- 1000
new_data <- lapply(config, function(i) nsize_repeat(nsizes, i, times))
xax <- seq(0,3,0.01)
compare(xax,new_data$chi,1,2,ns,"Chi square n = ")
```



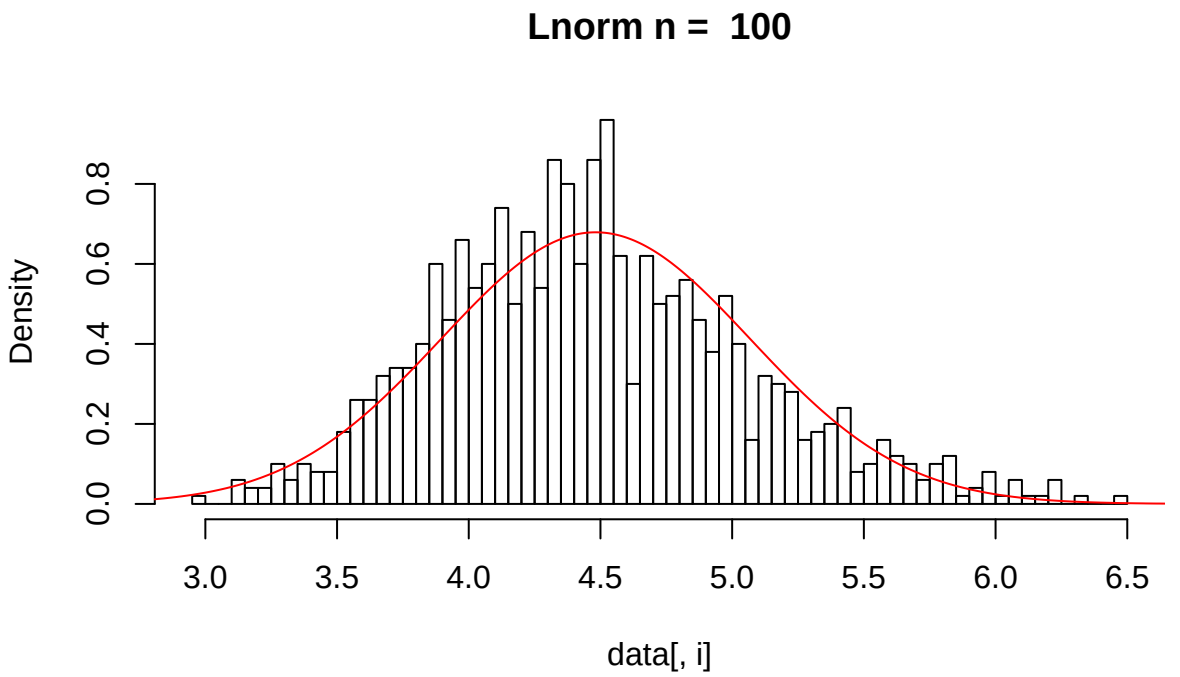

```
compare(xax,new_data$exp,1,1,ns,"Exponential n = ")
```



```
xax <- seq(-1,1,0.01)
compare(xax,new_data$uni,0,1/3,ns,"Uni n = ")
```




```
xax <- seq(0,10,0.01)
compare(xax,new_data$lnm,lnm_mean,lnm_var,ns,"Lnorm n = ")
```



Secinājumi: ar izlases izmēru n = 100 pietiek, lai redzētu konverģenci pie visiem sadalījumiem.

3. uzdevums

a) Datu kopa "quakes", apraksts

Datu kopas vispārīgs apraksts:

```
summary(quakes)
```

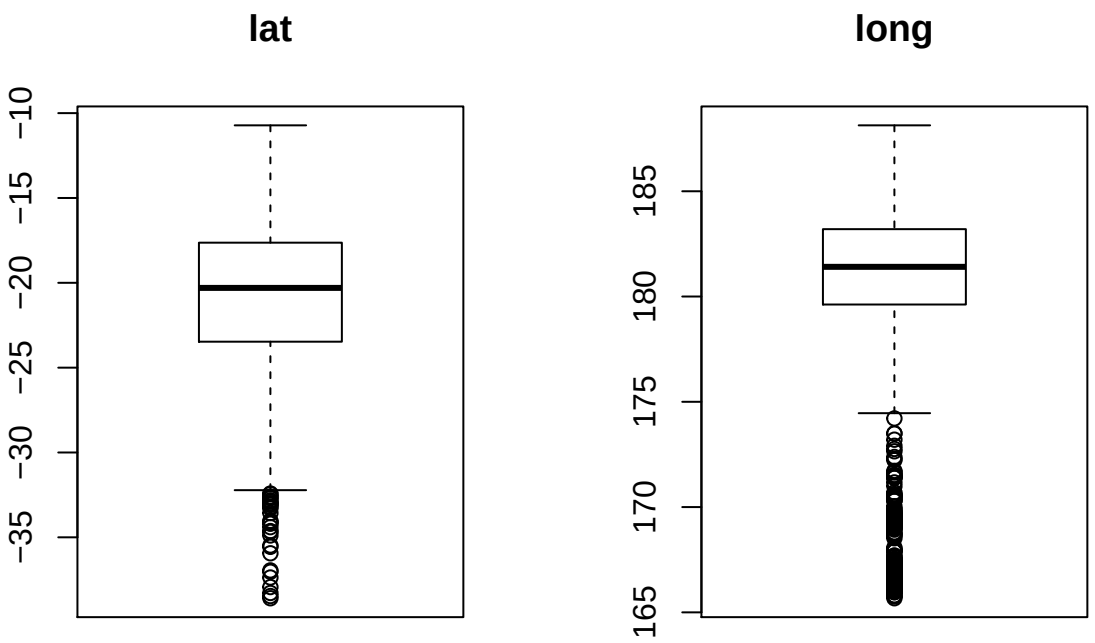
```
##       lat              long           depth            mag
##  Min.   :-38.59   Min.   :165.7   Min.   : 40.0   Min.   :4.00
##  1st Qu.:-23.47   1st Qu.:179.6   1st Qu.: 99.0   1st Qu.:4.30
##  Median :-20.30   Median :181.4   Median :247.0   Median :4.60
##  Mean   :-20.64   Mean   :179.5   Mean   :311.4   Mean   :4.62
##  3rd Qu.:-17.64   3rd Qu.:183.2   3rd Qu.:543.0   3rd Qu.:4.90
##  Max.   :-10.72   Max.   :188.1   Max.   :680.0   Max.   :6.40
##     stations
##  Min.   : 10.00
##  1st Qu.: 18.00
##  Median : 27.00
##  Mean   : 33.42
##  3rd Qu.: 42.00
##  Max.   :132.00
```

Satur mainīgos:

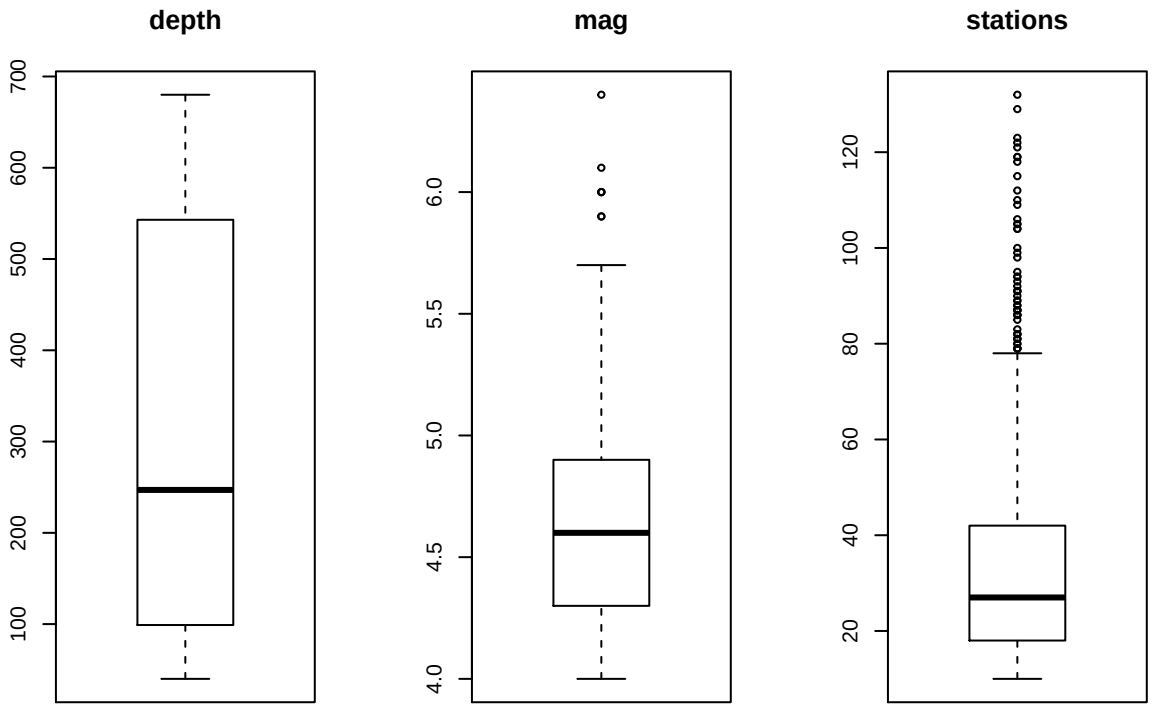
- lat,long: ģeogrāfiskās koordinātes
- depth(km): zemestrīces fokusa dziļums
- mag: magnitūda
- stations: notikumu reģistrējušo novērošanas staciju skaits

b) Kastu grafiki

```
attach(quakes)
par(mfrow = c(1,2))
boxplot(lat,main="lat")
boxplot(long,main="long")
```




```
par(mfrow = c(1,3))
boxplot(depth,main="depth")
boxplot(mag,main="mag")
boxplot(stations,main="stations")
```



Asimetriski sadalījumi:

- lat, long: it kā liels skaits izlēcēju un nobīde, taču jāņem vērā, ka koordinātes ir mākslīgi "nogrieztas", jo ņemtas no kuba formas telpas;
- mag, stations: diezgan skaidri redzami exponenciāli vai log-normāli sadalījumi;
- depth: grūti spriest, taču manāma asimetrija boxplot.

Ko nozīmē liels skaits izlēcēju: sadalījums ar "treknām astēm" un/vai izteiktu asimetriju - pēc noklusējuma "whisker" garumu nosaka kā 1.5 * (Q3-Q1), t.i., pusotru reizi "kastes" platuma.

c) InsectSprays

Bibliotēka aprakstošo statistiku iegūšanai:

```
library(psych)

require(ggplot2)
require(qqplotr)
require(gridExtra)
```

Vispārīgi: kukaiņu skaiti eksperimentos ar dažādiem pesticīdiem.

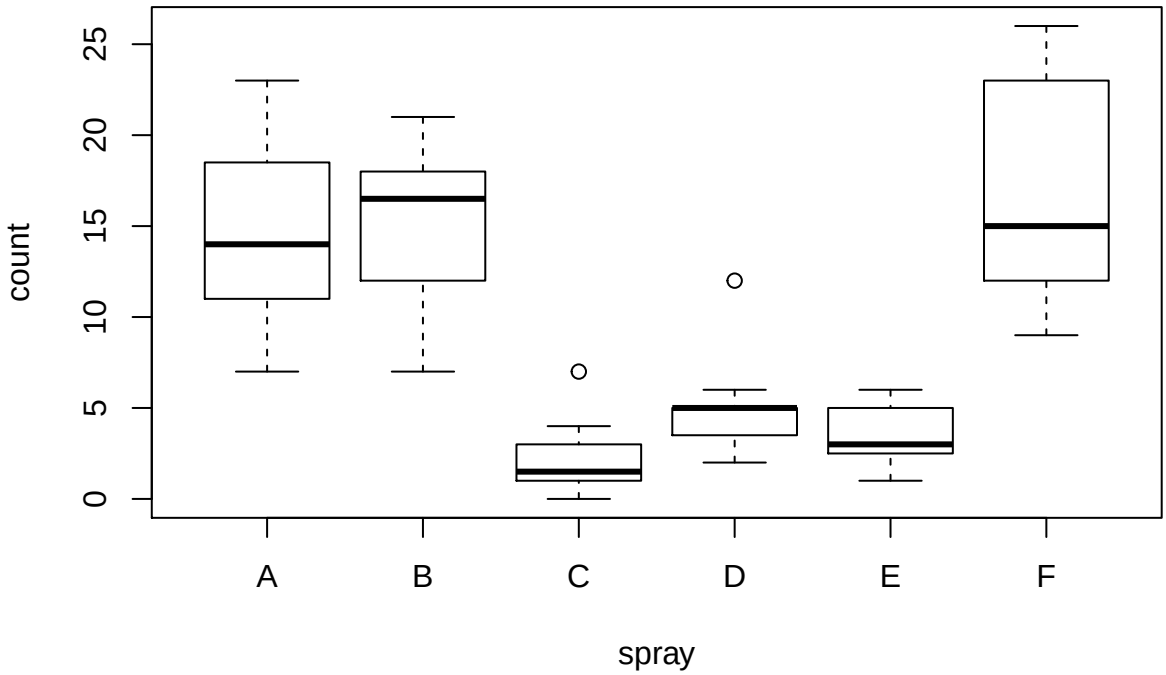
```
summary(InsectSprays)
```

```
##      count       spray
##  Min.   : 0.00   A:12
##  1st Qu.: 3.00   B:12
##  Median : 7.00   C:12
##  Mean   : 9.50   D:12
##  3rd Qu.:14.25   E:12
##  Max.   :26.00   F:12
```

```
attach(InsectSprays)
```

Kastu grafiks un aprakstošās statistikas:

```
boxplot(count~spray)
```




```
describe(count)
```

```
##    vars  n mean  sd median trimmed  mad min max range skew kurtosis   se
## X1    1 72  9.5 7.2      7     8.9 7.41   0  26    26 0.56    -0.84 0.85
```

```
describe(count[spray=="A"])
```

```
##    vars  n mean   sd median trimmed  mad min max range skew kurtosis   se
## X1    1 12 14.5 4.72     14    14.4 5.19   7  23    16 0.27    -1.13 1.36
```

```
describe(count[spray=="B"])
```

```
##    vars  n  mean   sd median trimmed  mad min max range  skew kurtosis   se
## X1    1 12 15.33 4.27   16.5    15.6 4.45   7  21    14 -0.35    -1.04 1.23
```

```
describe(count[spray=="C"])
```

```
##    vars  n mean   sd median trimmed  mad min max range skew kurtosis   se
## X1    1 12 2.08 1.98    1.5     1.8 1.48   0   7     7 1.13     0.52 0.57
```

```
describe(count[spray=="D"])
```

```
##    vars  n mean  sd median trimmed  mad min max range skew kurtosis   se
## X1    1 12 4.92 2.5      5     4.5 1.48   2  12    10 1.68     2.56 0.72
```

```
describe(count[spray=="E"])
```

```
##    vars  n mean   sd median trimmed  mad min max range skew kurtosis  se
## X1    1 12  3.5 1.73      3     3.5 2.22   1   6     5 0.05    -1.41 0.5
```

```
describe(count[spray=="F"])
```

```
##    vars  n  mean   sd median trimmed  mad min max range skew kurtosis   se
## X1    1 12 16.67 6.21     15    16.5 6.67   9  26    17 0.39    -1.56 1.79
```

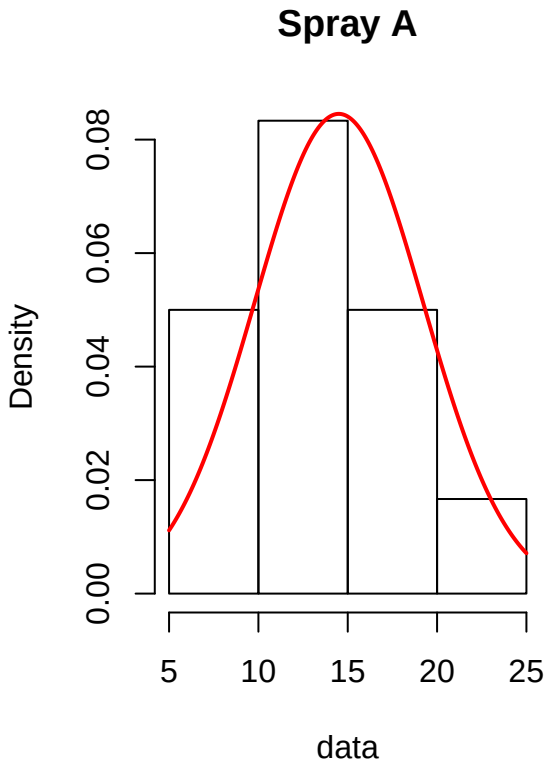
Līdzīgās izlases:

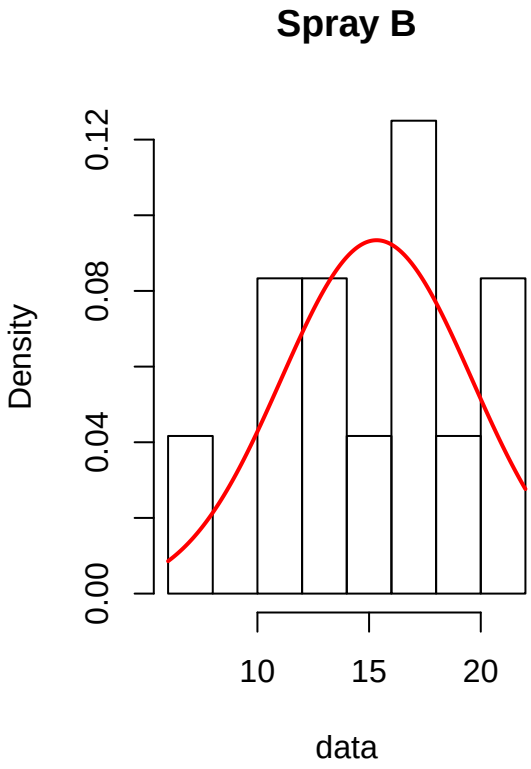
- A,B,F
- D,E

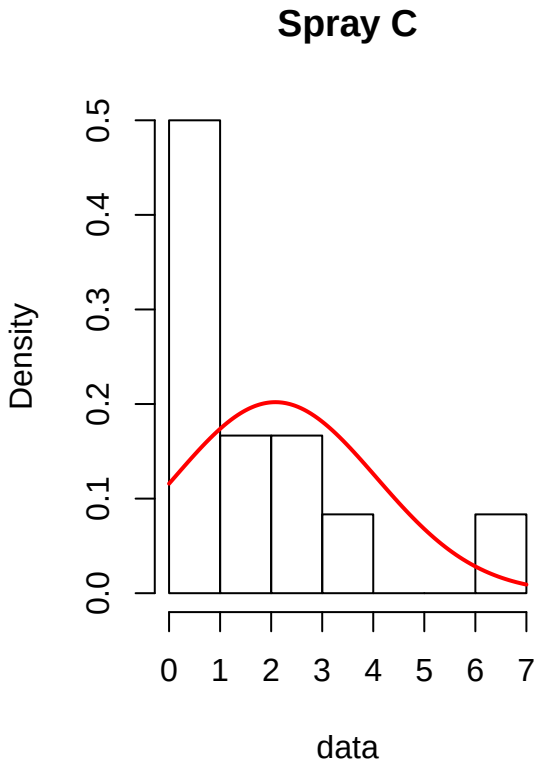
Histogrammu zīmēšanas funkcija:

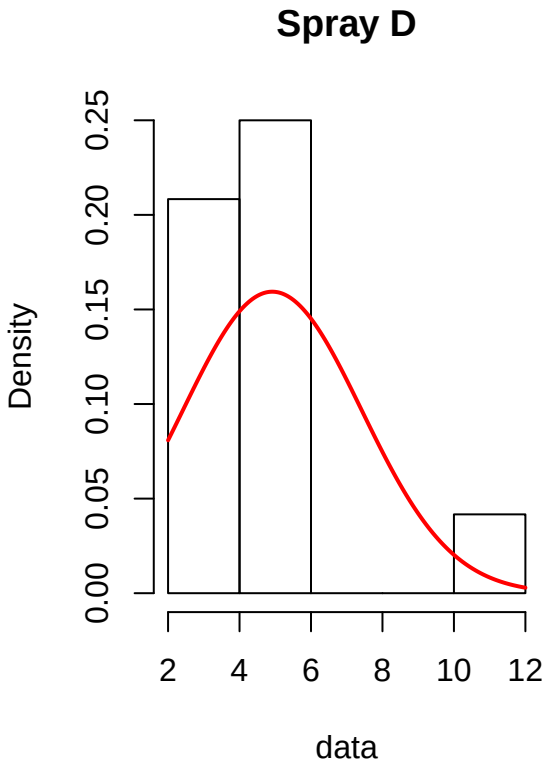
```
draw_near <- function(data, name) {
  h <- hist(data,prob=T,main = name)
  xax <- seq(min(h$breaks),max(h$breaks),0.01)
  lines(xax,dnorm(xax,mean(data),sd(data)),col="red",lwd=2)
}
```

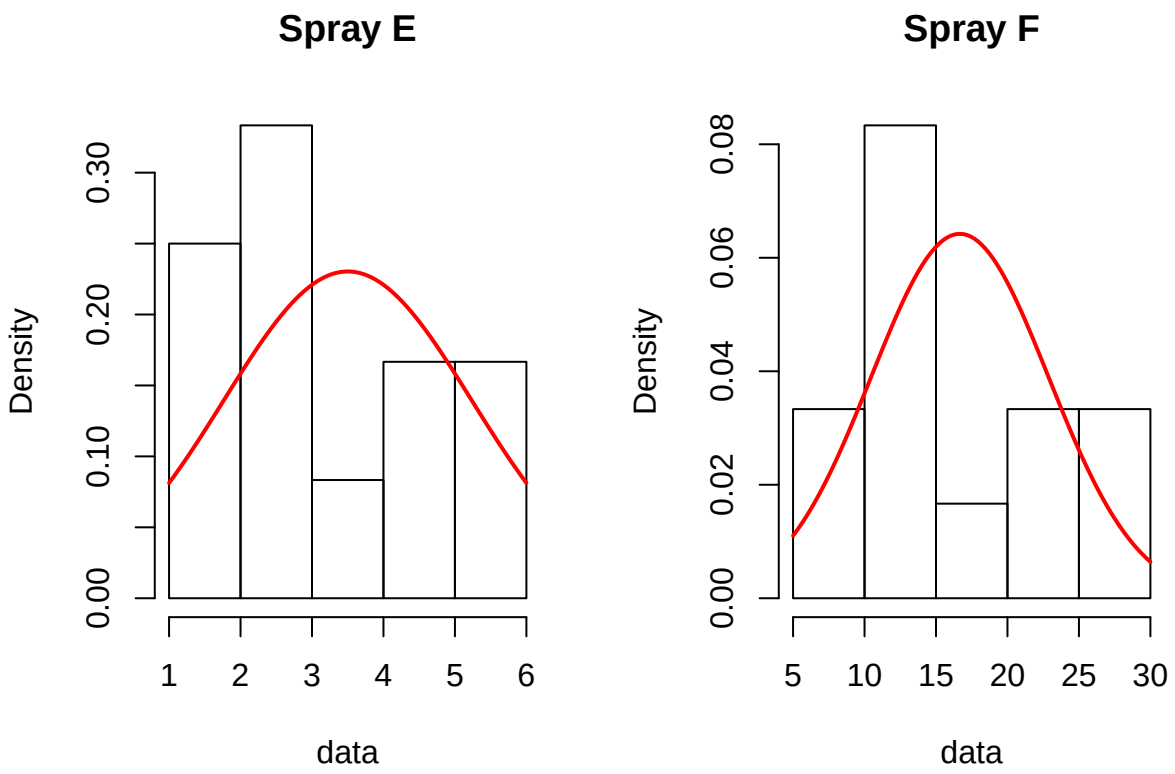
Histogrammas:









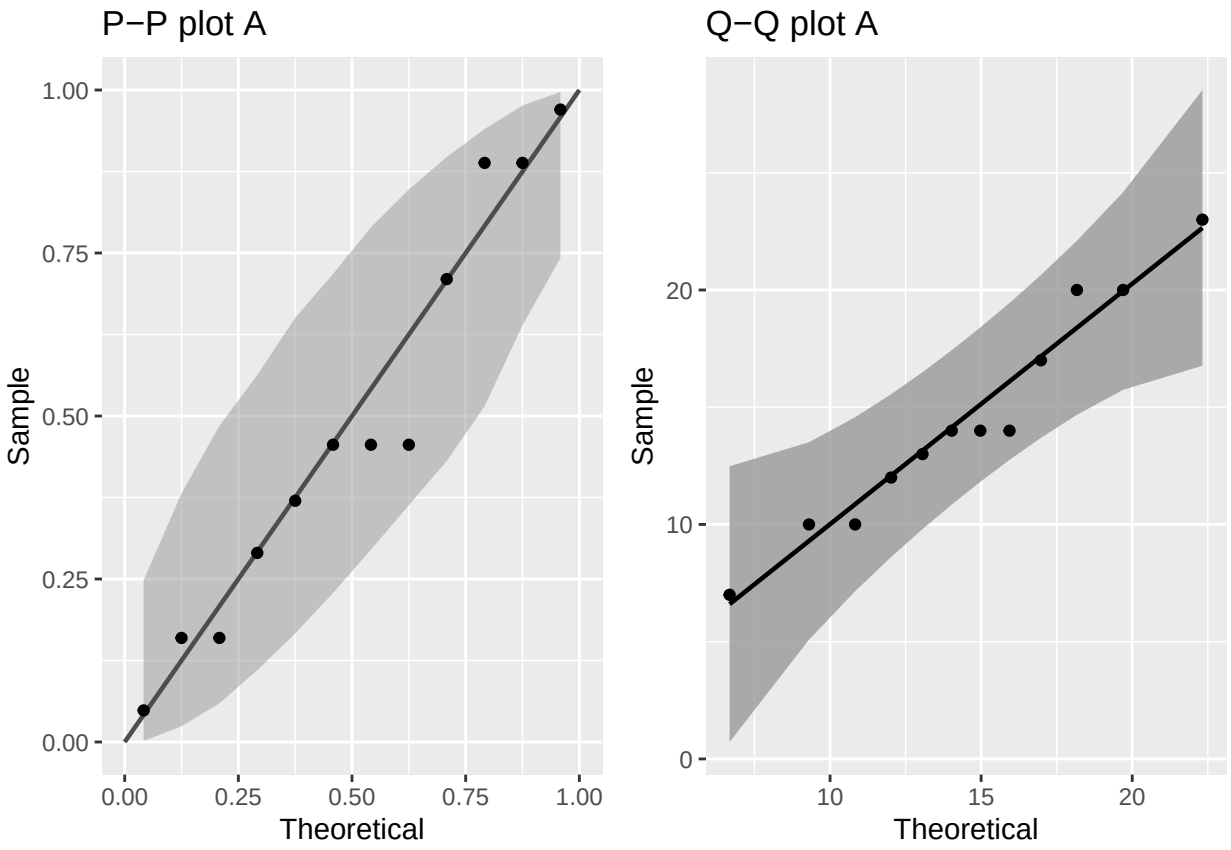


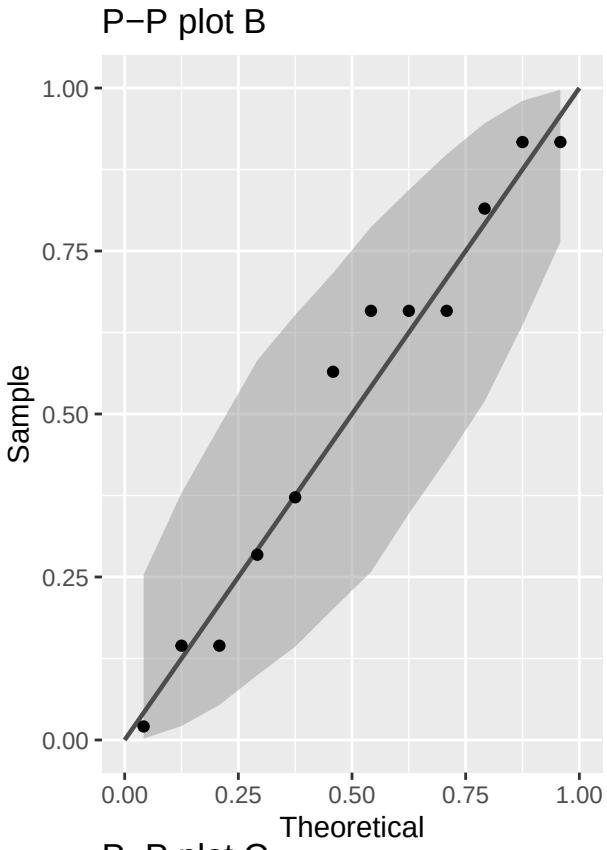
Funkcijas QQ,PP grafiku zīmēšanai (+- copy&paste no kursa materiāliem):

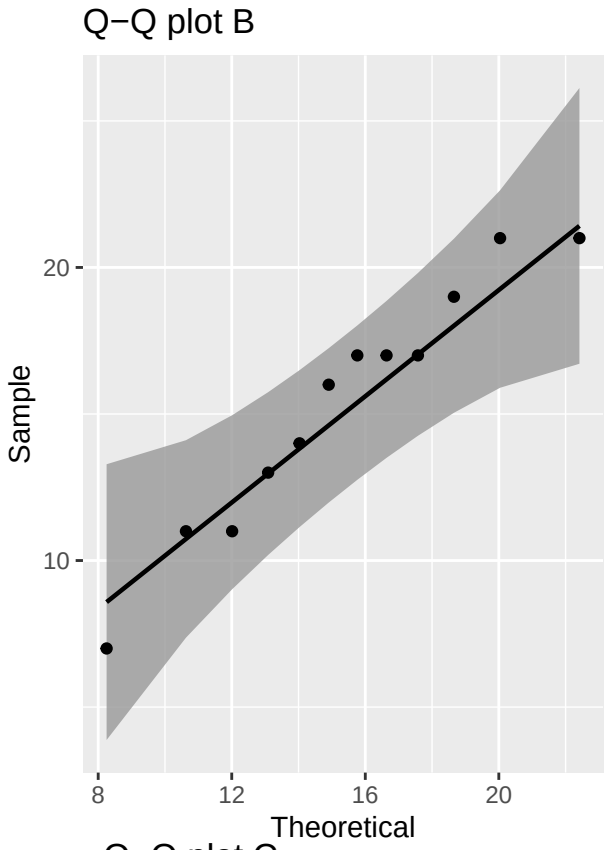
```
plot_PP_norm <- function(data) {
  g <- ggplot(data = data, mapping = aes(sample = count)) +
    stat_pp_band() +
    stat_pp_line() +
    stat_pp_point()+
    labs(title = paste("P-P plot",data$spray[1]), x = "Theoretical", y = "Sample")
  g
}
plot_QQ_norm <- function(data) {
  g <- ggplot(data = data, mapping = aes(sample = count)) +
    stat_qq_band() +
    stat_qq_line() +
    stat_qq_point()+
    labs(title = paste("Q-Q plot",data$spray[1]), x = "Theoretical", y = "Sample")
  g
}
plots <- function(data) {
  grid.arrange(plot_PP_norm(data),plot_QQ_norm(data),ncol=2)
}
compare_QQ <- function(d1,d2){
  x <- d1$count
  y <- d2$count
  sx <- sort(x)
  sy <- sort(y)
  lenx <- length(sx)
  leny <- length(sy)
  if (leny < lenx) sx <- approx(1L:lenx, sx, n = leny)$y
  if (leny > lenx) sy <- approx(1L:leny, sy, n = lenx)$y
```

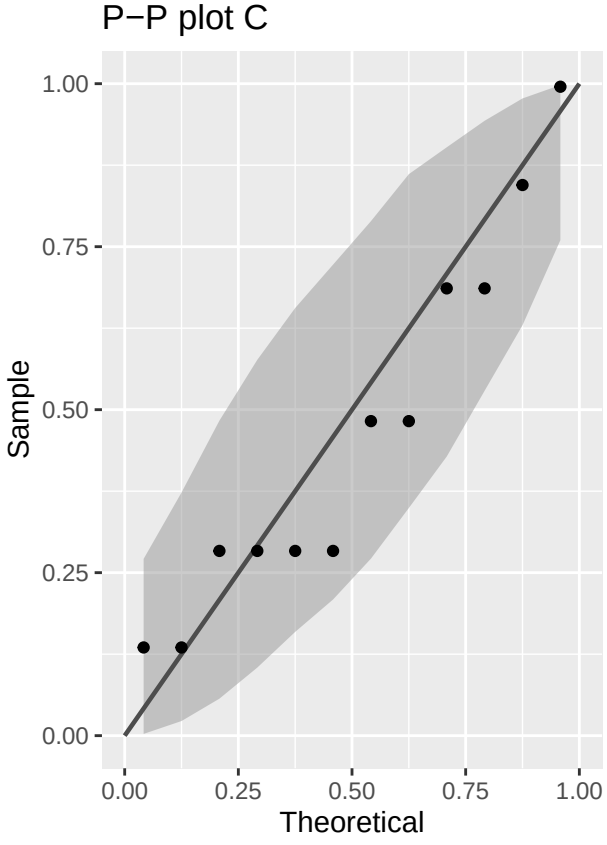
```
  require(ggplot2)
  g = ggplot() + geom_point(aes(x=sx, y=sy))+
    geom_abline(intercept =0, slope = 1)+
    labs(title="QQ compare",x=d1$spray[1],y=d2$spray[1])
  g
}
compare_PP <- function(d1,d2,min_c,max_c) {
  x <- d1$count
  y <- d2$count
  fn1<-ecdf(x)
  fn2<-ecdf(y)
  xx<-seq(min_c,max_c,0.1)
  dd<-data.frame(y=fn1(xx),x=fn2(xx))
  g <- ggplot(dd, aes(x=x,y=y))+geom_point()+
    geom_abline(intercept =0, slope = 1)+
    labs(title="PP compare",x=d1$spray[1],y=d2$spray[1])
  g
}
compare_two <- function(d1,d2,min_c,max_c) {
  #par(mfrow = c(2,1))
  grid.arrange(compare_PP(d1,d2,min_c,max_c),compare_QQ(d1,d2),ncol=2)
  #par(mfrow = c(1,1))
}
```

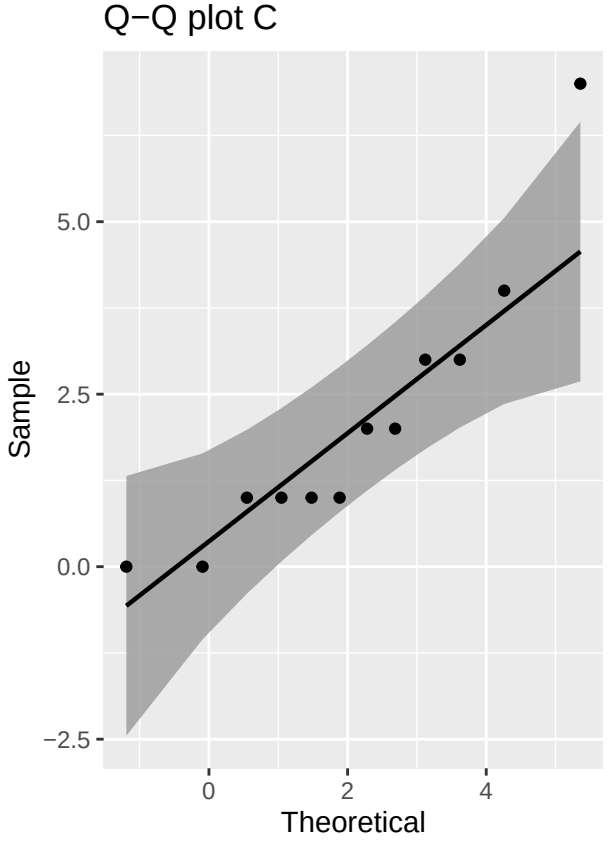
Grafiki:

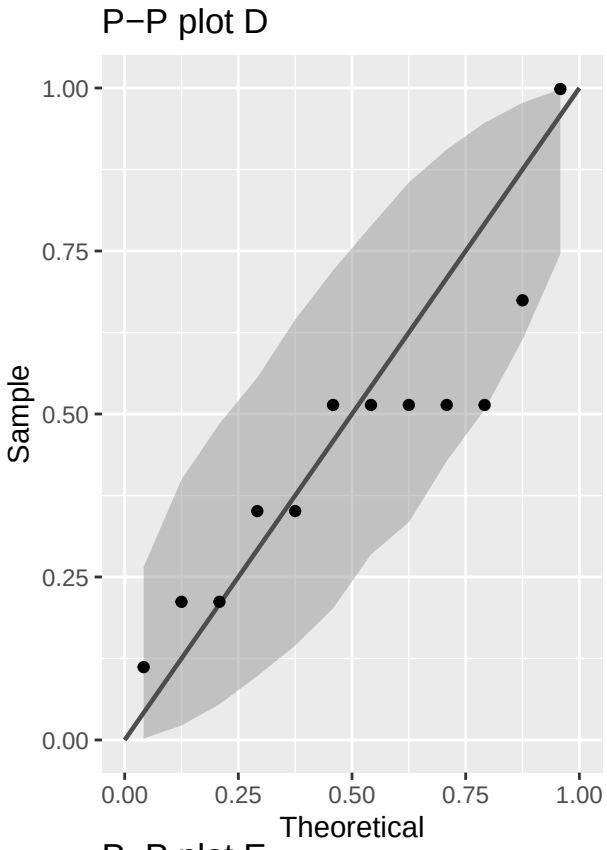


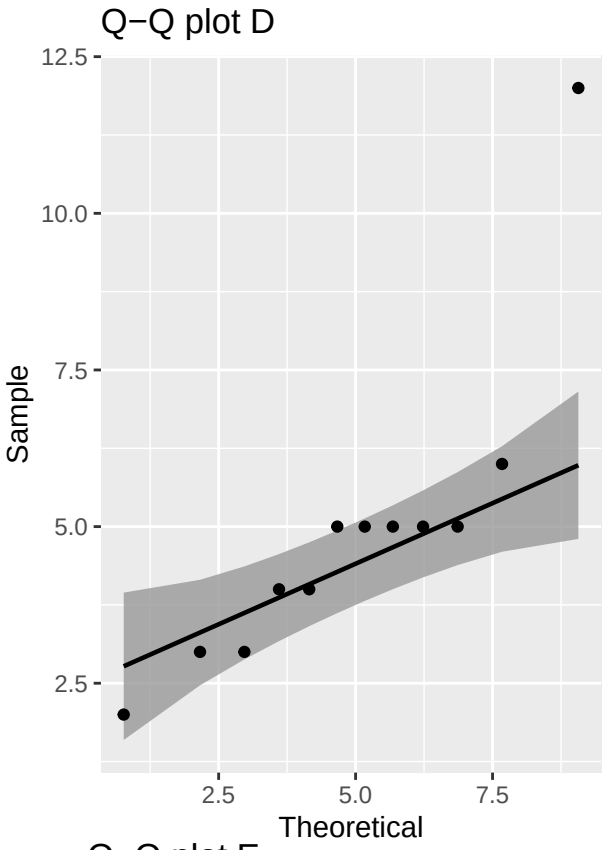

P−P plot B
Sample
Theoretical
0.00
0.25
0.50
0.75
1.00

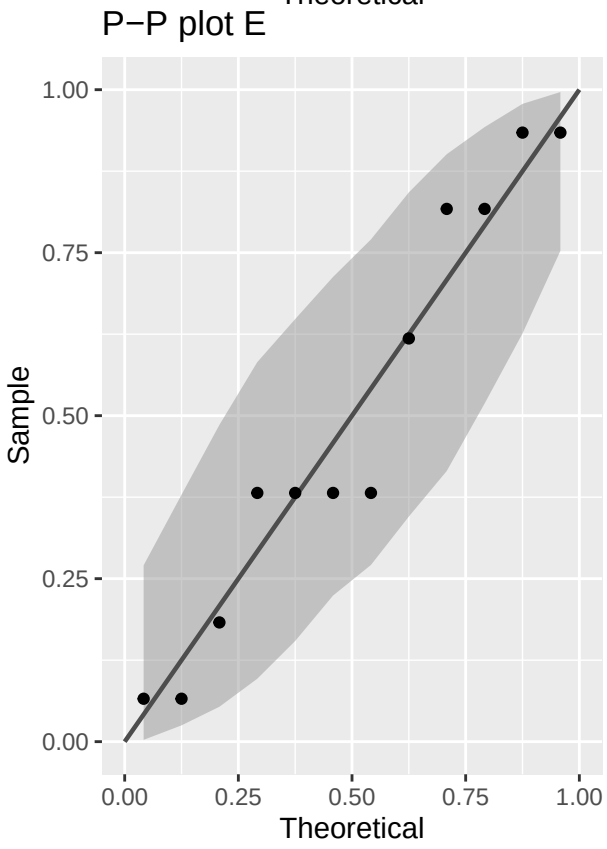

Q−Q plot B
Sample
Theoretical
8
12
16
20
10
20

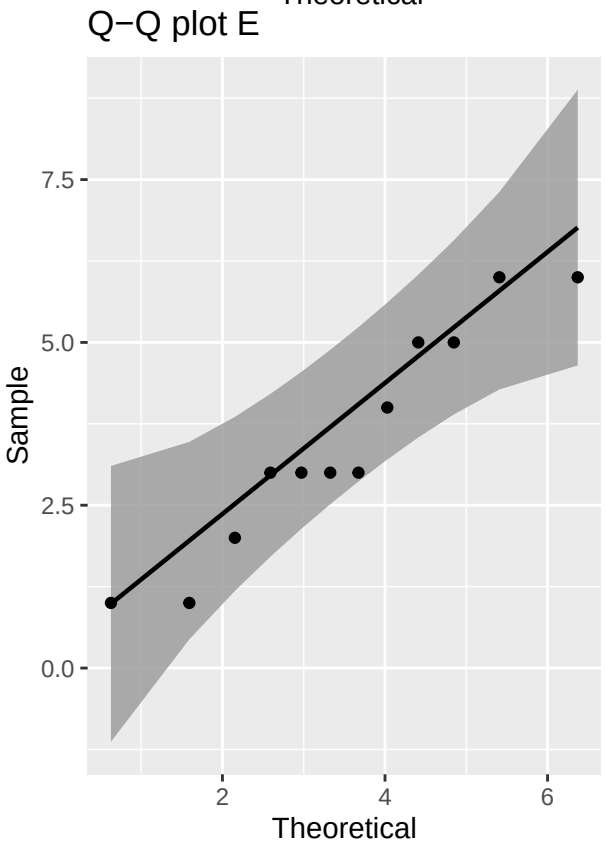

P−P plot C
Sample
Theoretical
0.00
0.25
0.50
0.75
1.00

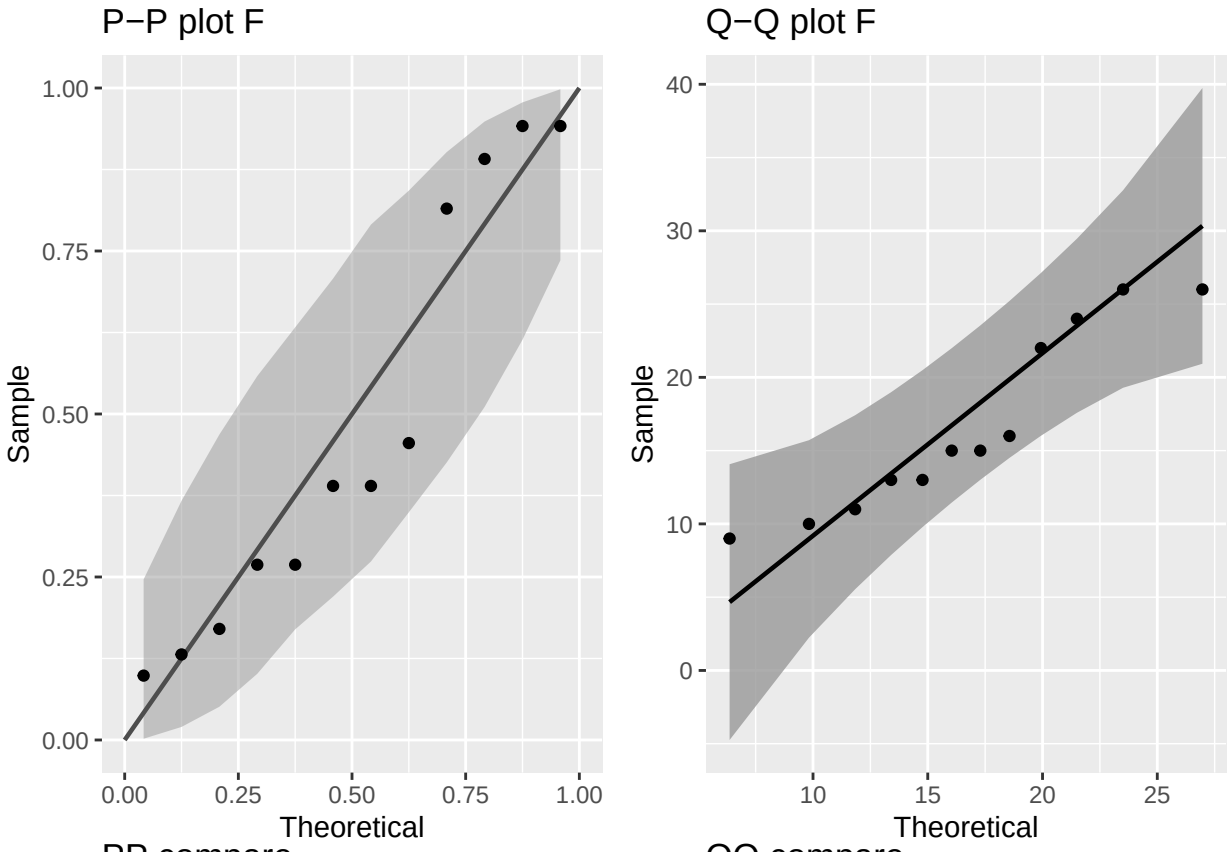

Q−Q plot C
Sample
Theoretical
−2.5
0.0
2.5
5.0
0
2
4


P−P plot D
Sample
Theoretical
0.00
0.25
0.50
0.75
1.00

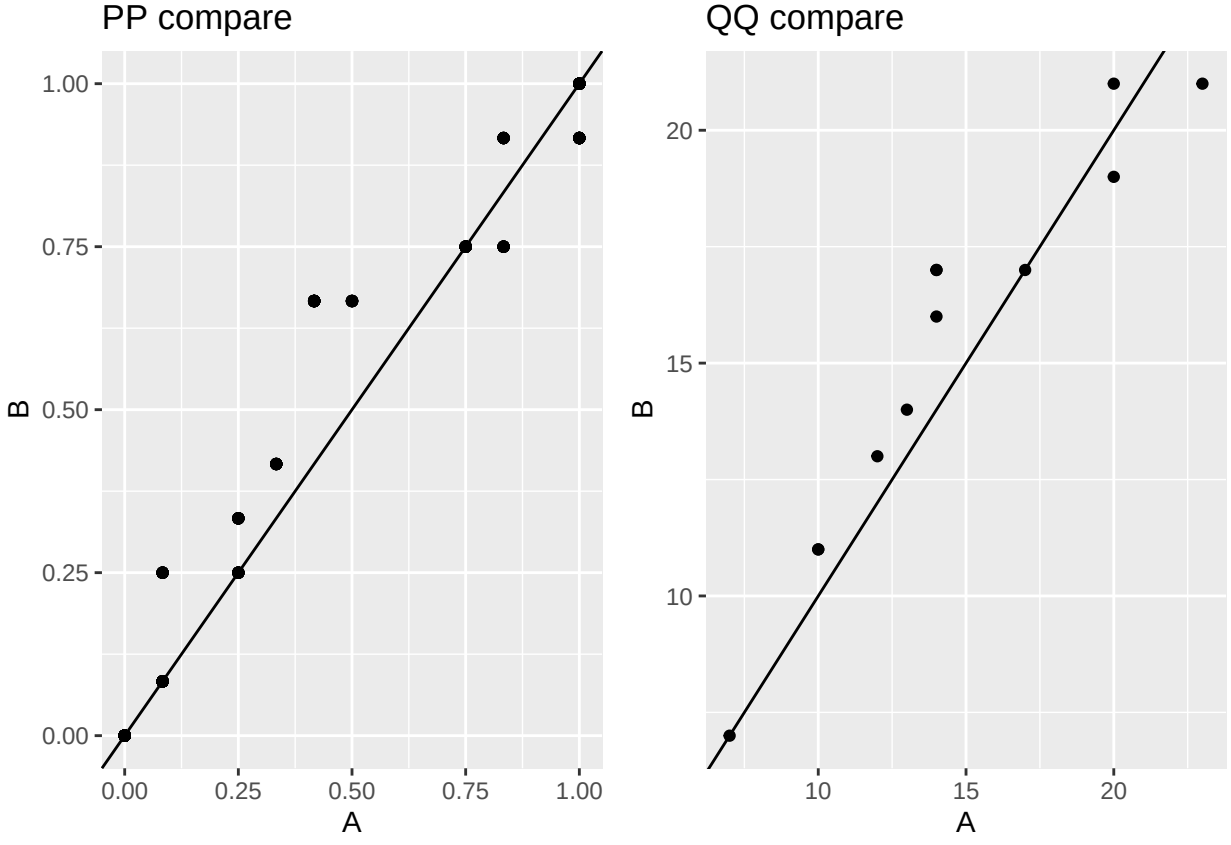

Q−Q plot D
Sample
Theoretical
2.5
5.0
7.5
10.0
12.5


P−P plot E
Sample
Theoretical
0.00
0.25
0.50
0.75
1.00

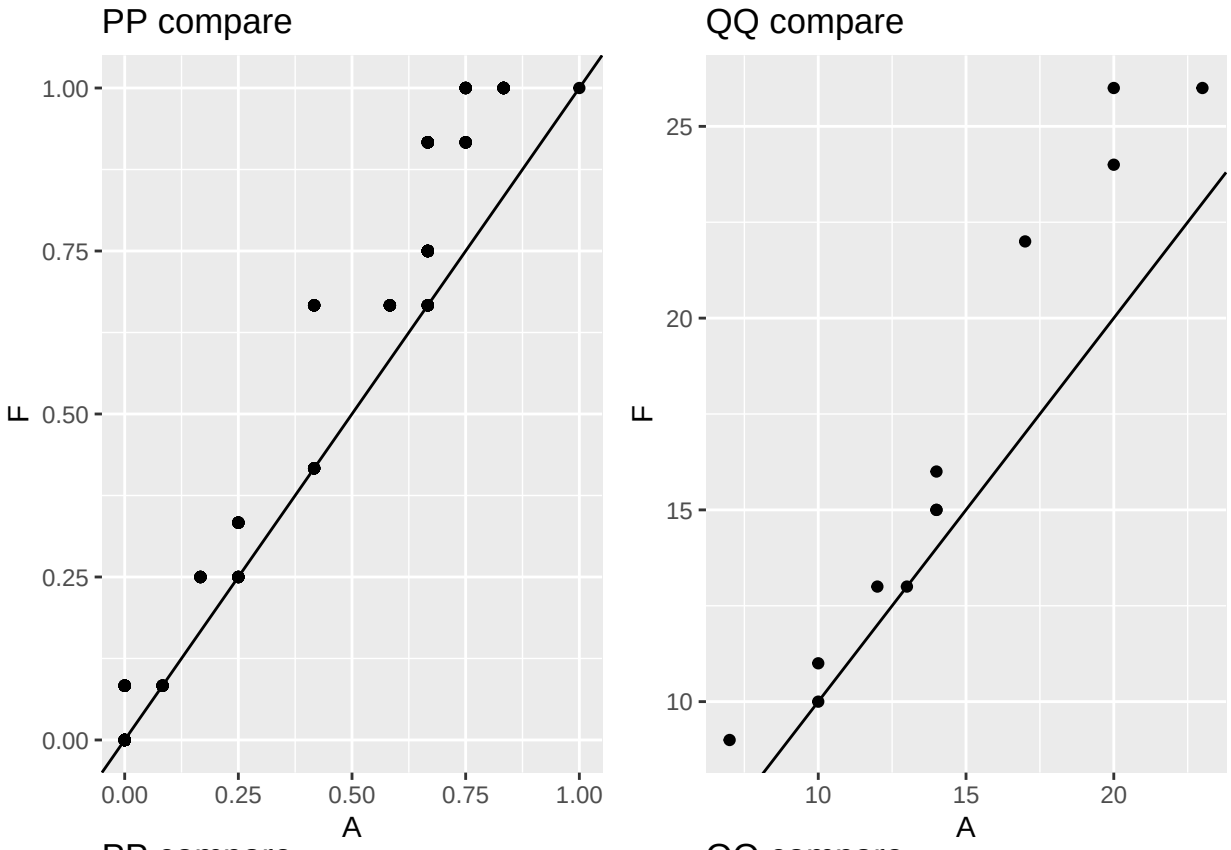

Q−Q plot E
Sample
Theoretical
0.0
2.5
5.0
7.5
2
4
6


P−P plot F
Sample
Theoretical
0.00
0.25
0.50
0.75
1.00

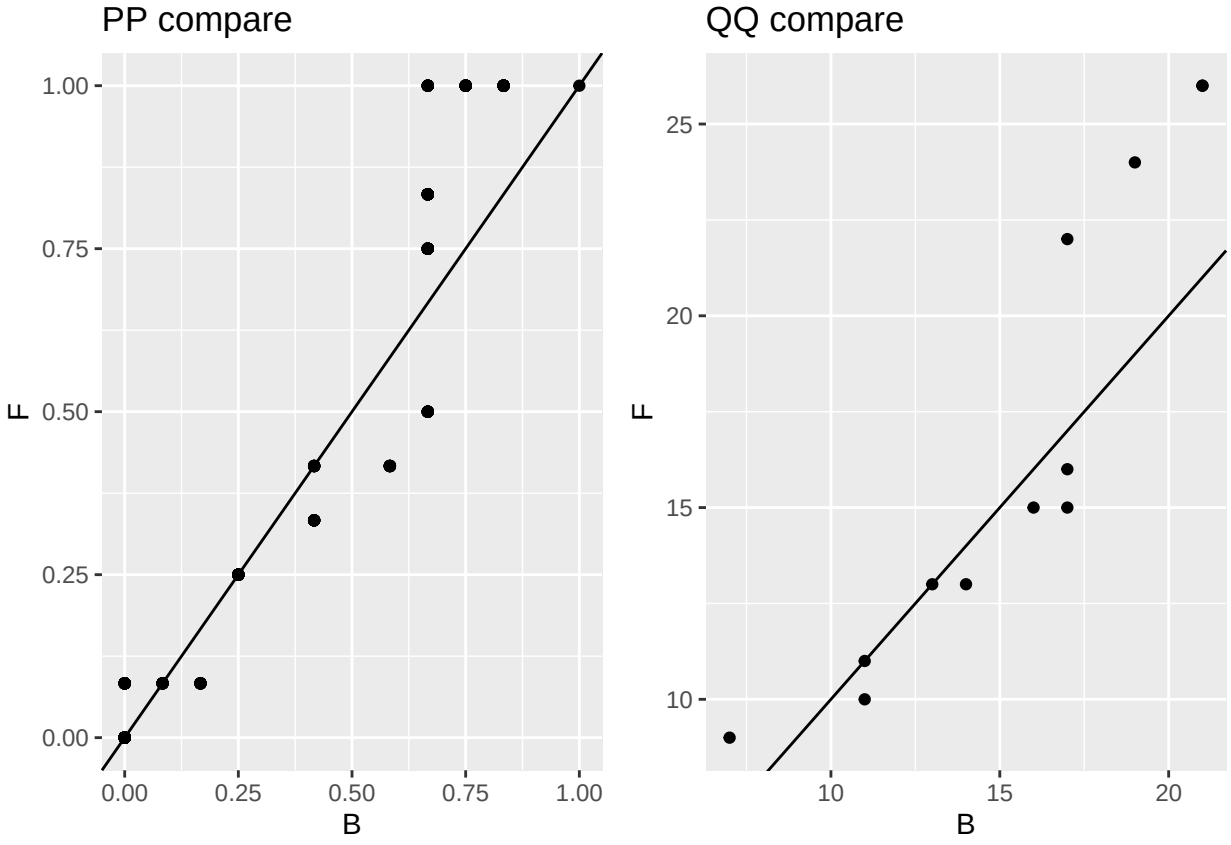
Q−Q plot F
Sample
Theoretical
0
10
20
30
40
10
15
20
25


PP compare
B
A
0.00
0.25
0.50
0.75
1.00

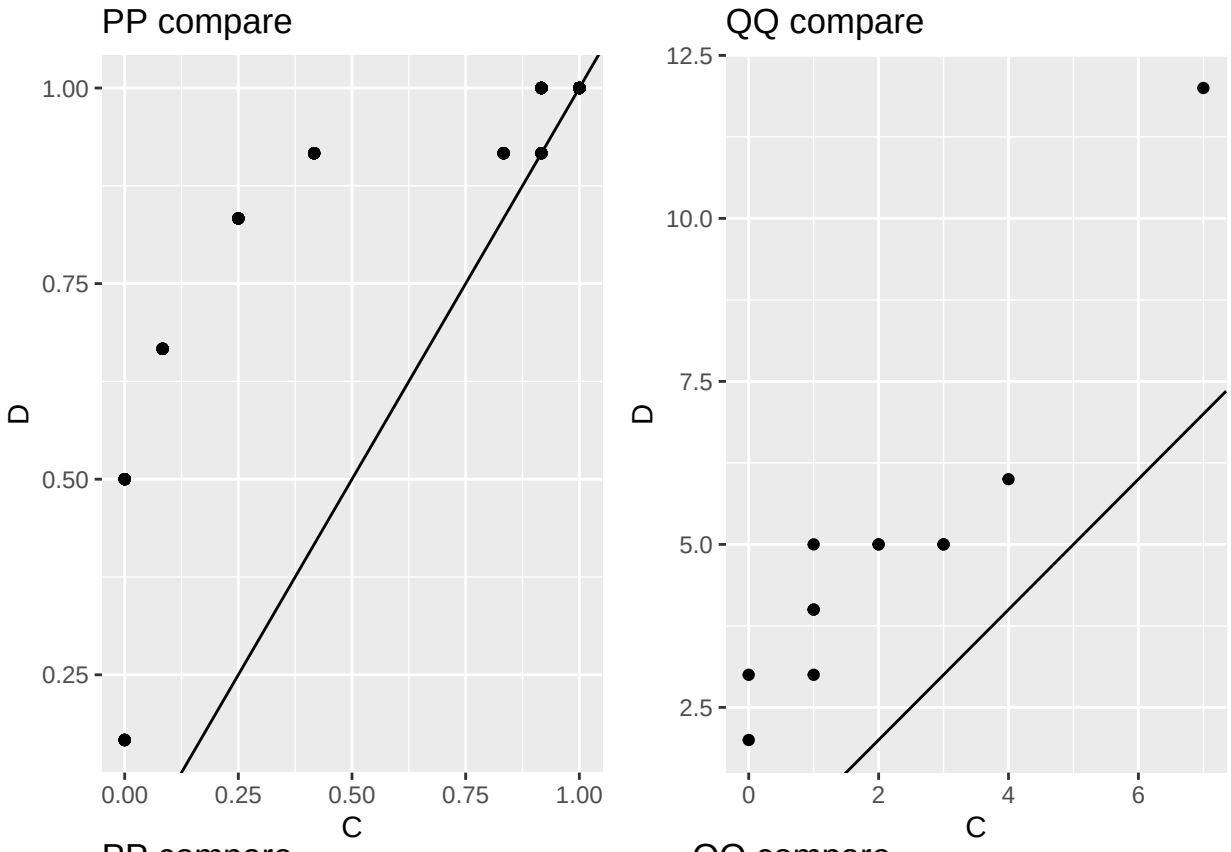
QQ compare
B
A
10
15
20


PP compare
1.00
0.75
0.50
0.25
0.00
F
0.00
0.25
0.50
0.75
1.00
A

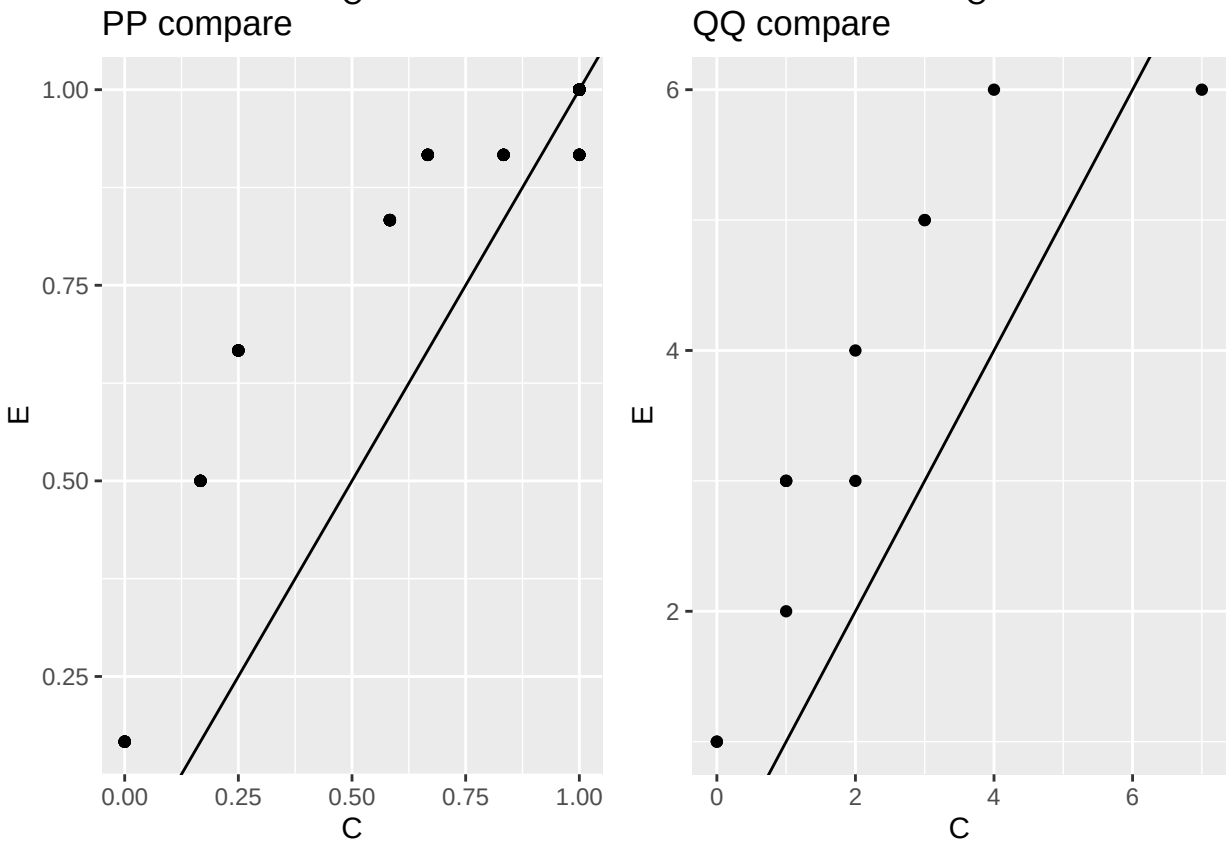
QQ compare
25
20
15
10
F
10
15
20
A


PP compare
1.00
0.75
0.50
0.25
0.00
F
0.00
0.25
0.50
0.75
1.00
B

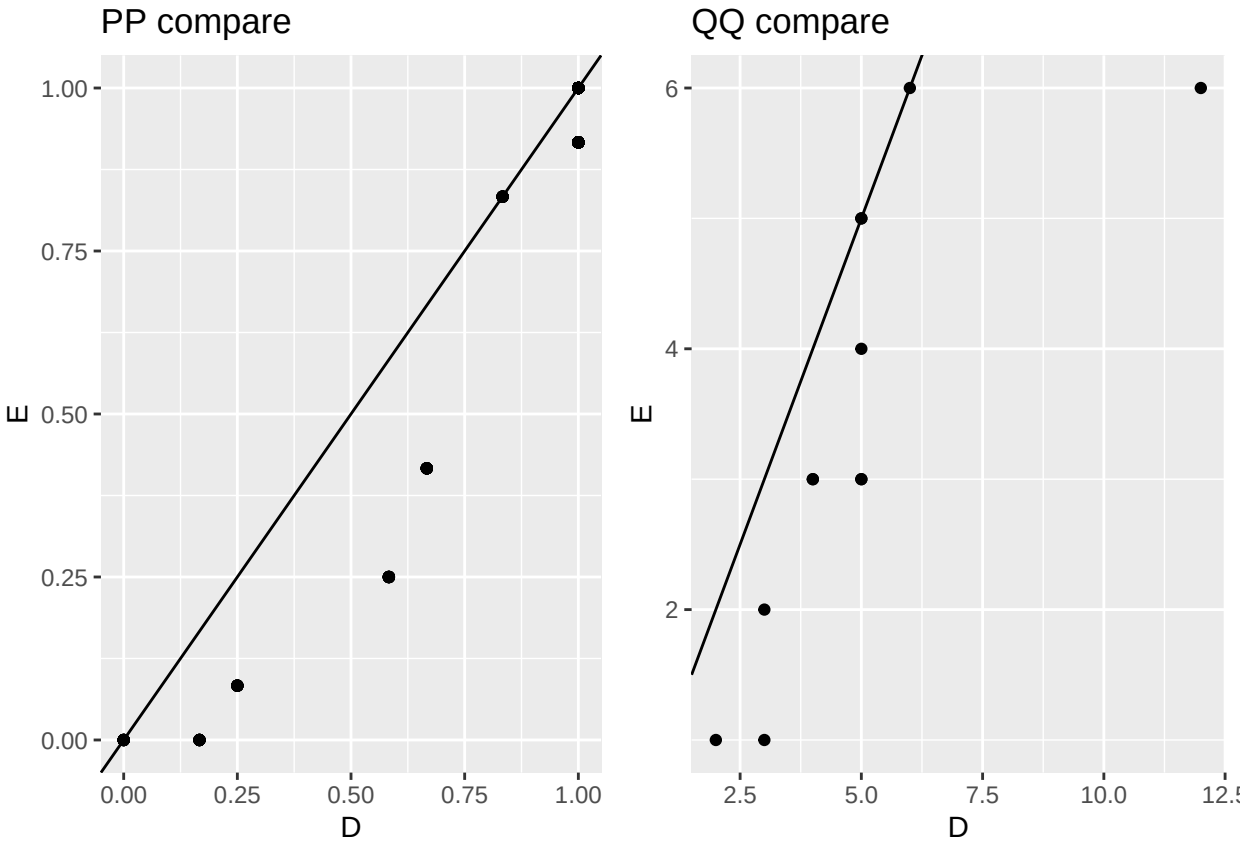
QQ compare
25
20
15
10
F
10
15
20
B


PP compare
1.00
0.75
0.50
0.25
D
0.00
0.25
0.50
0.75
1.00
C

QQ compare
12.5
10.0
7.5
5.0
2.5
D
0
2
4
6
C


PP compare
1.00
0.75
0.50
0.25
E
0.00
0.25
0.50
0.75
1.00
C

QQ compare
6
4
2
E
0
2
4
6
C



Secinājumi:

- Pēc QQ/PP gandrīz visas izlases varētu būt normali sadalītas, taču histogrammās tikai A un varbūt B izskatās pēc normālā sadalījuma.
- A,B,F ir diezgan līdzīgas, D,E - mazāk. Grūti spriest, vai kādas no izlasēm ņemtas no tā paša sadalījuma.

4. uzdevums

Bibliotēka MLE optimizācijas uzdevumu risināšanai:

```
library(maxLik)
```

Momentu metode:

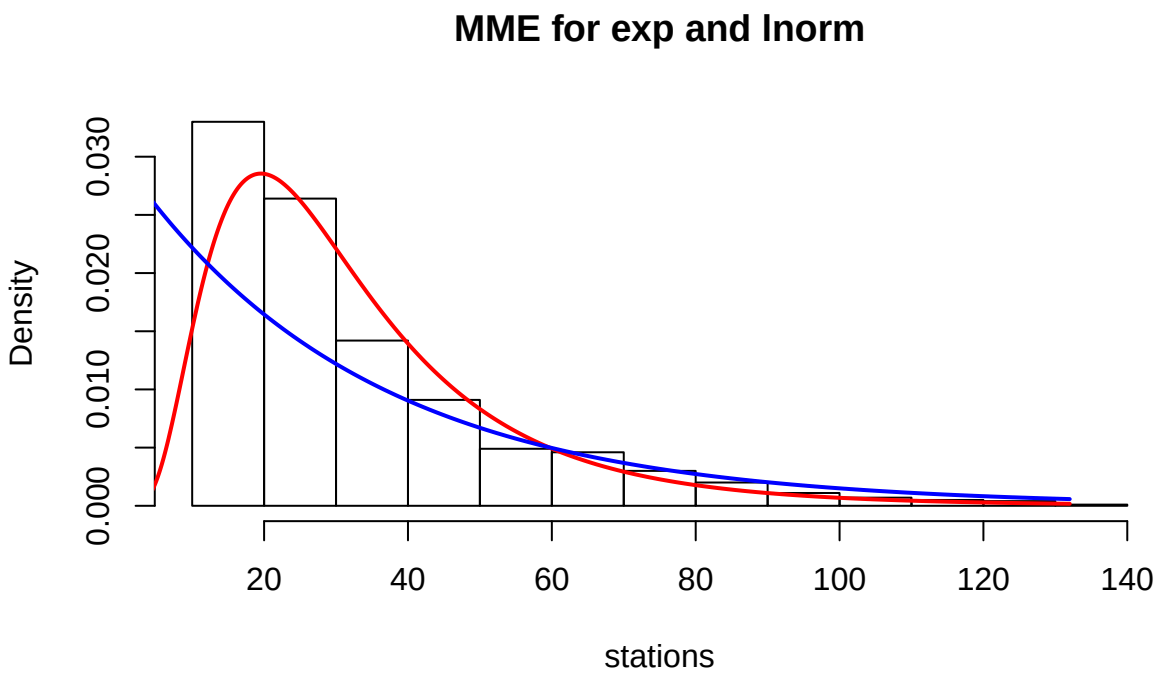
```
lnorm_pdf <-function(x,mu,sigma)
{
  location <- log(mu^2 / sqrt(sigma^2 + mu^2))
  shape <- sqrt(log(1 + (sigma^2 / mu^2)))
  dlnorm(x,location,shape)
}
# moment method for exp <- rate = 1/mean <-> mean = 1/rate
exp_pdf <- function(x,mu) {
  dexp(x,1/mu)
}
```

Maksimālās ticamības metode:

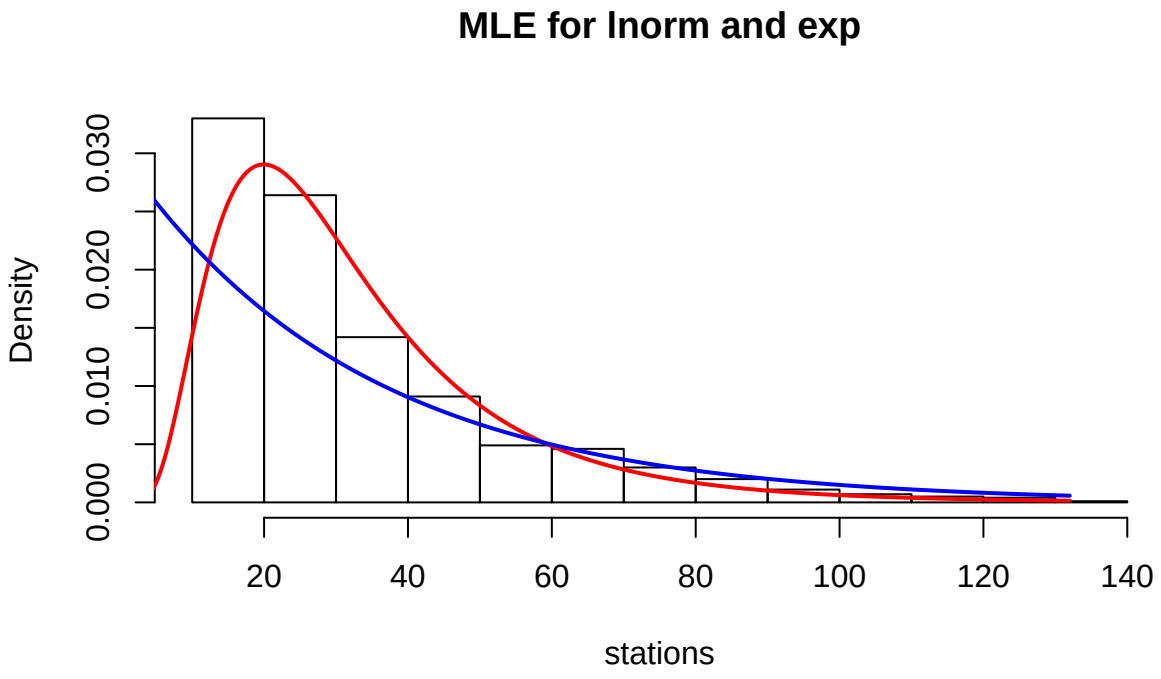
```
llf_lnorm <- function(param) {
  mu <- param[1]
  sd <- param[2]
  x <- stations
  llValue <- dlnorm(x, mean=mu, sd=sd, log=TRUE)
  sum(llValue)
}
llf_exp <- function(param) {
  lambda <- param[1]
  x <- stations
  n <- length(x)
  n*log(lambda)-lambda*sum(x)
}

mu_init = 3.5
sd_init = 0.6
mllnm <- maxLik(llf_lnorm, start=c(mu_init,sd_init))
lambda_init <- 1/20
mlexp <- maxLik(llf_exp, start=c(lambda_init))
```

Salīdzinājuma grafiki:

```
attach(quakes)
hist(stations,prob=T,main="MME for exp and lnorm")
xax <- seq(0,max(stations),0.1)
x <- stations
lines(xax,lnorm_pdf(xax,mean(x),sd(x)),col="red",lwd=2)
lines(xax,exp_pdf(xax,mean(x)),col="blue",lwd=2)
```




```
hist(stations,prob=T,main="MLE for lnorm and exp")
lines(xax,dlnorm(xax,mllnm$estimate[1],mllnm$estimate[2]),col="red",lwd=2)
lines(xax,dexp(xax,mlexp$estimate[1]),col="blue",lwd=2)
```



Eksponenciālais sadalījums izskatās tuvāks izlases formai un labāk atspoguļo lielākau vērtību varbūtības. Log-normālais sadalījums tuvāk seko datiem pie nelielām vērtībām.

5. uzdevums

Datu nolasīšana no faila (iepriekš sagatavota, apstrādājot copy-paste datus ar Python skriptu)

```
df <- read.csv("anorex.csv")[,2:4]
attach(df)
```

a) Datu apraksts

Dati doti tabulas formā ar 3 saistītām ("paired") kolonnām:

- A - Svars pirms terapijas;
- B - Svars pēc terapijas;
- C - Starpība.

Lai ar datiem pierādītu metodes efektivitāti, mean(A) < mean(B) un mean(C) > 0

b) Kāpēc nulles hipotēze ir svarīga?

Nulles hipotēze ietver sākotnējo pieņēmumu par rezultātu (šai gadījumā - metode svaru neietekme), un kalpo kā atskaites punkts, pret ko salīdzināt iegūtos datus. Uzdodot problēmu nulles hipotēzes noraidīšanas formā var iepriekš noteikt kādu konkrētu pārliecības slieksni, pēc kā izdara secinājumus par eksperimenta rezultātiem un to nozīmi.

c) t-testa aprēķini

t-tests ar iebūvēto funkciju:

```
res <- t.test(change,alternative="g",conf.level=0.95); res
```

```
##
##  One Sample t-test
##
## data:  change
## t = 2.2069, df = 28, p-value = 0.01784
## alternative hypothesis: true mean is greater than 0
## 95 percent confidence interval:
##  0.6875368       Inf
## sample estimates:
## mean of x
##         3
```

t-testa rezultāts, kritiskā vērtība c, p-vērtība.

```
res$statistic
```

```
##        t
## 2.206908
```

```
alpha <- 0.05
cutoff <- qt(1-alpha,n-1); cutoff
```

```
## [1] 1.833113
```

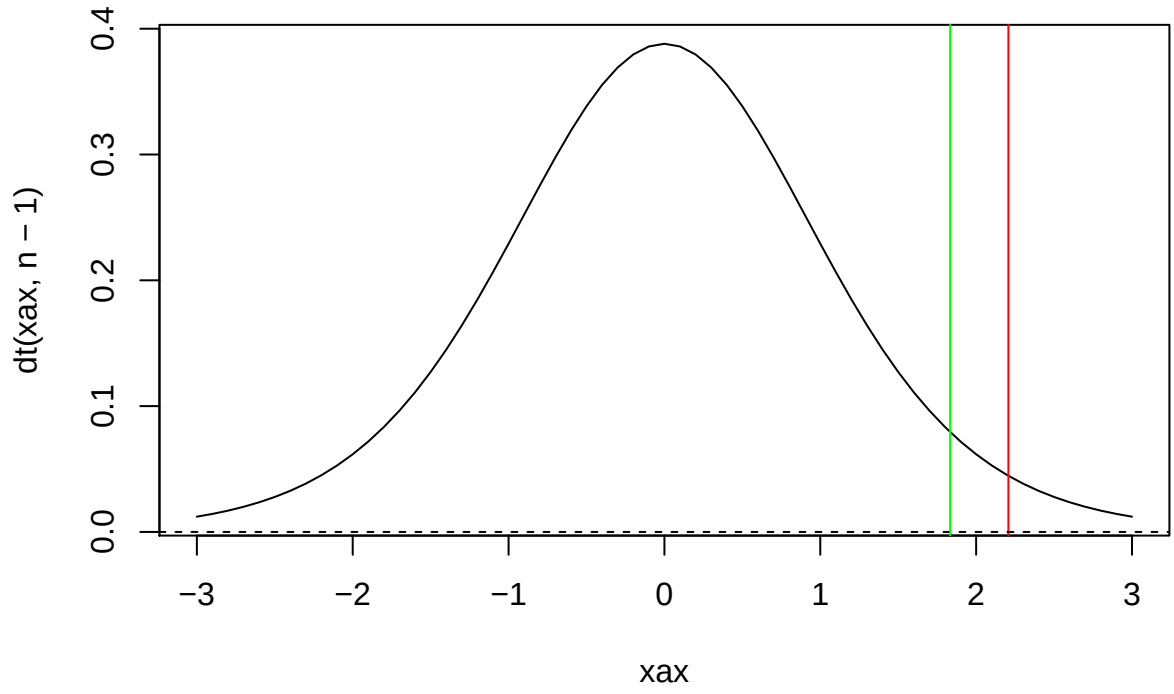
```
res$p.value
```

```
## [1] 0.01784396
```

Kritiskais apgabals: c > t; pieņemšanas apbgabals: c < t

d) t-testa grafisks attēlojums

```
xax <- seq(-3,3,0.1)
plot(xax,dt(xax,n-1),type="l")
abline(v=res$statistic,col="red")
abline(h=0,lty=2)
abline(v=cutoff,col="green")
```



e) Secinājumi - hipotēzi pieņem vai noraida?

```
conclusion <- (res$p.value < 0.05)
conclusion
```

```
## [1] TRUE
```

f) Vai var izmantot Wilkoksona testu? Vai p-vērtība stipri atšķiras?

Testu izmantot var, palīdz fakts, ka dati doti arī pa pāriem. p-vērtība ir nedaudz sliktāka - ap 3% nevis ~1% kā t-testam.

```
wilcox.test(before,after,alternative="l",paired=T,exact=F)
```

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  before and after
## V = 131.5, p-value = 0.03223
## alternative hypothesis: true location shift is less than 0
```

6. uzdevums

a) Datu ielasīšana un apraksts.

Datu ielasīšana no faila. Dati sastāv no trīs kolonnām - urbšanas dziļumiem, vidējā urbšanas laika līdz katram dziļumam sausā režīmā, vidējā urbšanas laika līdz katram dziļumam slapjā režīmā. Dati ievākti sešos urbumos.

```
df <- read.csv('cleaned.csv')
attach(df)
str(df)
```

```
## 'data.frame':    80 obs. of  3 variables:
##  $ Depth: int  5 205 10 210 15 215 20 220 25 225 ...
##  $ Dry  : num  641 803 675 794 708 ...
##  $ Wet  : num  830 962 800 865 711 ...
```

```
summary(df)
```

```
##      Depth            Dry              Wet
##  Min.   :  5.0   Min.   : 584.0   Min.   : 697.3
##  1st Qu.:103.8   1st Qu.: 687.5   1st Qu.: 851.5
##  Median :202.5   Median : 757.5   Median : 917.5
##  Mean   :202.5   Mean   : 805.5   Mean   : 943.8
##  3rd Qu.:301.2   3rd Qu.: 912.4   3rd Qu.:1029.9
##  Max.   :400.0   Max.   :1238.0   Max.   :1238.5
```

Q: Kāpēc svarīgi salīdzināt urbšanas ilgumu atkarībā no režīma?

A: Pastāv iespēja, ka var ātrāk veikt urbumus sausajā režīmā.

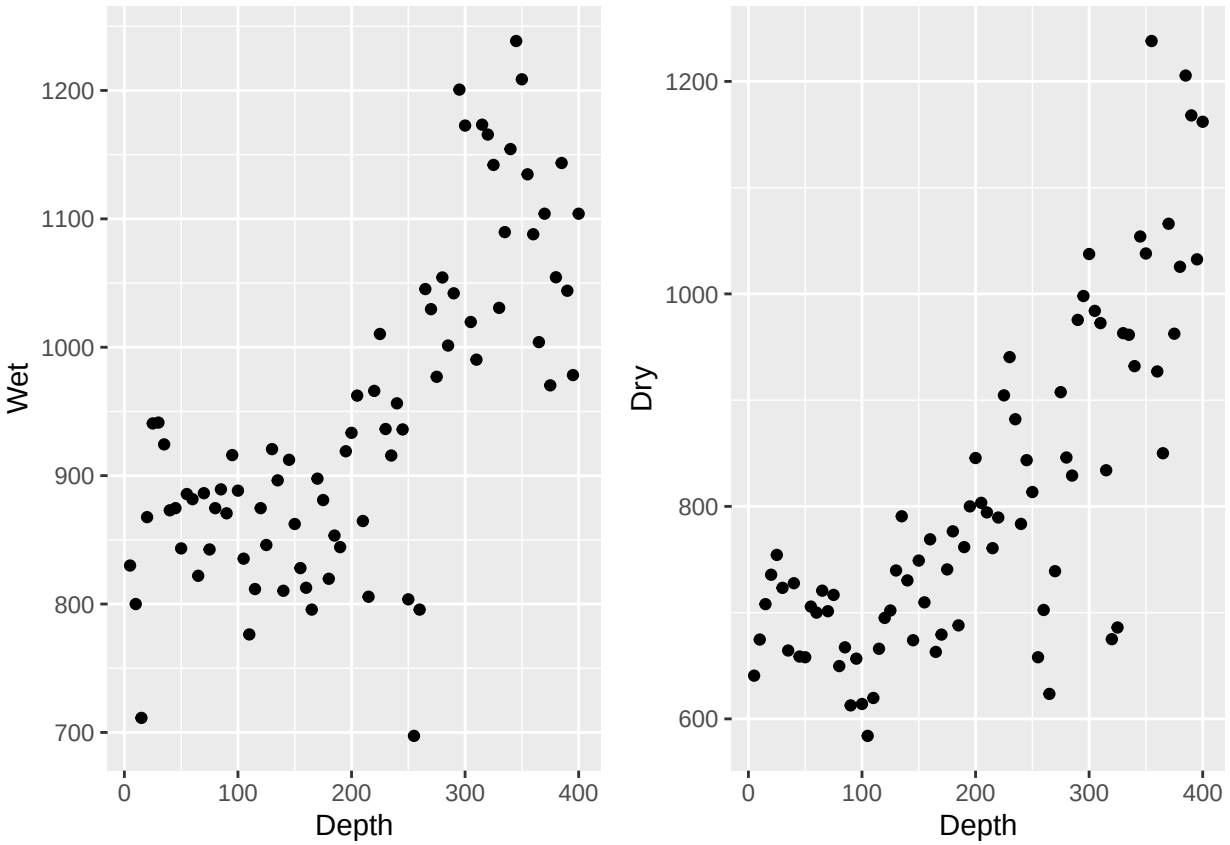
b) Punktu grafiks urbšanas laikiem un dziļumam

Bibliotēkas:

```
require(ggplot2)
require(gridExtra)
```

Grafiku konstruēšana:

```
ap <- ggplot(data=df,aes(x=Depth, y=Wet)) + geom_point()
bp <- ggplot(data=df,aes(x=Depth, y=Dry)) + geom_point()
grid.arrange(ap,bp,ncol=2)
```



Q: Vai urbšanas ilgums ir atkarīgs no urbuma dziļuma?

A: Pie nelieliem dziļumiem, izskatās, ka nē - droši vien dominē citi faktori. Pie lielākiem dziļumiem - izskatās, ka jā.

Q: Pie kāda dziļuma notiek pārmaiņa?

A: Pēc grafika spriežot, aptuveni 200m.

Q: Kādi varētu būt iemesli?

A: Kā viena no iespējām grāmatā tiek minēta cietāka klints lielākos dziļumos. Cita iespēja varētu būt fakts, ka dziļākā urbumā lielāku daļu no kopējā urbšanas ilguma sastāda tīri mehāniskais urbšanas process, nevis dažādi nesaistīti kavēkļi.

c) Kastu grafiks slapjai un sausai urbšanai

Aprakstošas statistikas:

```
require(psych)
describe(Dry)
```

```
##    vars  n   mean     sd median trimmed    mad min  max range skew kurtosis
## X1    1 80 805.53 154.17  757.5  788.64 130.84 584 1238   654 0.91    -0.01
##       se
## X1 17.24
```

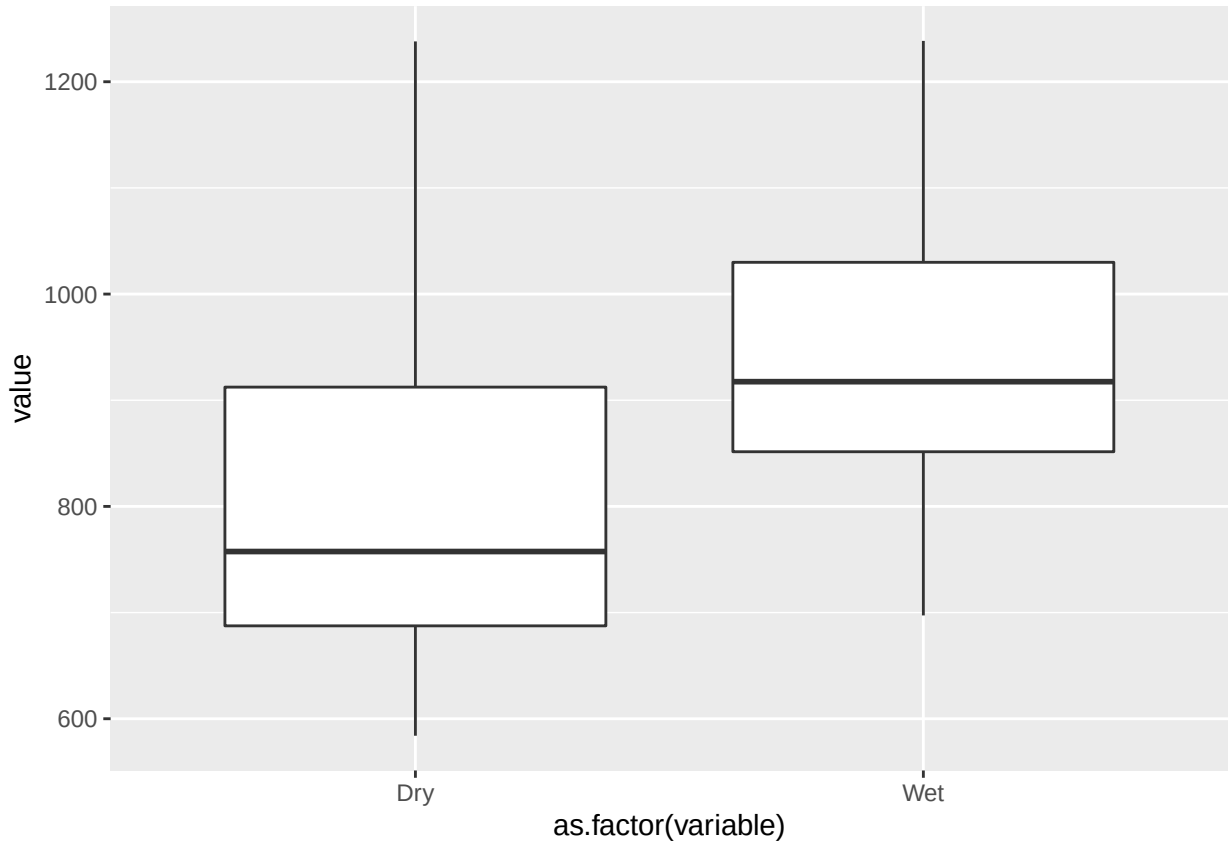
```
describe(Wet)
```

```
##    vars  n   mean    sd median trimmed    mad    min    max  range skew
## X1    1 80 943.82 123.8  917.5  935.35 123.06 697.33 1238.5 541.17 0.52
```

```
##    kurtosis    se
## X1    -0.53 13.84
```

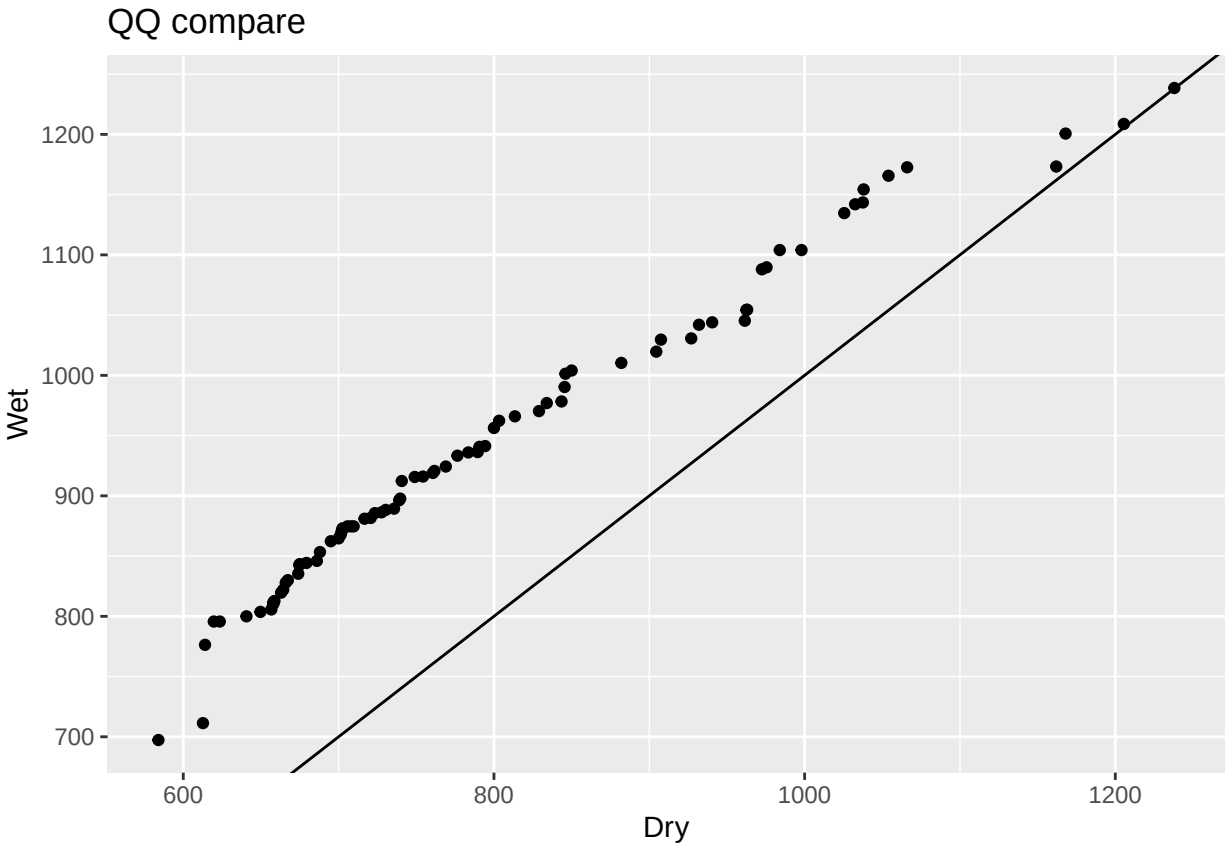
Kastu grafiks:

```
require(reshape)
md <- melt(df[,2:3])
ggplot(data=md,aes(x=as.factor(variable),y=value)) + geom_boxplot()
```



Izmainīta funkcija QQ salīdzinājumam, QQ salīdzinājums

```
compare_QQ <- function(d,v1,v2){
  x <- subset(d, variable==v1)$value
  y <- subset(d, variable==v2)$value
  sx <- sort(x)
  sy <- sort(y)
  lenx <- length(sx)
  leny <- length(sy)
  if (leny < lenx) sx <- approx(1L:lenx, sx, n = leny)$y
  if (leny > lenx) sy <- approx(1L:leny, sy, n = lenx)$y
  require(ggplot2)
  g = ggplot() + geom_point(aes(x=sx, y=sy))+
    geom_abline(intercept =0, slope = 1)+
    labs(title="QQ compare",x=v1,y=v2)
  g
}

compare_QQ(melt(df[,2:3]),"Dry","Wet")
```



Q: Vai dispersijas var būt vienādas?

A: Dati izskatās diezgan līdzīgi tikai nedaudz nobīdīti mazāko vērtību apgabalā, taču izskatās, ka parādās lielākas atšķirības pie lielākām vērtībām. Vizuāli dispersiju nav viegli novērtēt.

d) Salīdzinājums

Pieņemot vienādas dispersijas:

```
t.test(Dry,Wet,alternative="two.sided",var.equal=T,conf.level=0.95)
```

```
##
##  Two Sample t-test
##
## data:  Dry and Wet
## t = -6.2555, df = 158, p-value = 3.556e-09
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -181.94698  -94.62302
## sample estimates:
## mean of x mean of y
##  805.5316  943.8166
```

Pieņemot dažādas dispersijas:

```
t.test(Dry,Wet,alternative="two.sided",var.equal=F,conf.level=0.95)
```

```
##
##  Welch Two Sample t-test
##
```

```
## data:  Dry and Wet
## t = -6.2555, df = 150.96, p-value = 3.883e-09
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -181.9627  -94.6073
## sample estimates:
## mean of x mean of y
##  805.5316  943.8166
```

Dispersiju vienādības pārbaude:

```
dv <- var.test(Dry,Wet,conf.level=0.99); dv
```

```
##
##  F test to compare two variances
##
## data:  Dry and Wet
## F = 1.5507, num df = 79, denom df = 79, p-value = 0.05288
## alternative hypothesis: true ratio of variances is not equal to 1
## 99 percent confidence interval:
##  0.863461 2.785040
## sample estimates:
## ratio of variances
##           1.550733
```

```
conf <- 0.05
dv$p.value
```

```
## [1] 0.05287956
```

```
dv$estimate
```

```
## ratio of variances
##           1.550733
```

```
(dv$p.value < conf)
```

```
## [1] FALSE
```

Stingri runājot, dispersiju vienādības hipotēze netiek noraidīta, taču tā ir uz robežas.

Salīdzinājums "plakanajā" reģionā, 0-200ft, vispirms nosakot, vai dispersijas ir dažādas:

```
flat <- subset(df,Depth<200)
dv <- var.test(flat$Dry,flat$Wet,conf.level=0.95); dv
```

```
##
##  F test to compare two variances
##
## data:  flat$Dry and flat$Wet
## F = 1.1141, num df = 38, denom df = 38, p-value = 0.7408
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.5842283 2.1246415
## sample estimates:
## ratio of variances
##           1.114126
```

```
(dv$p.value < conf)
```

```
## [1] FALSE
```

Tā kā dispersijas diezgan pārliecinoši ir vienādas, tālāku pārbaudi veic, pieņemot vienādu dispersiju:

```
res <- t.test(flat$Dry,flat$Wet,alternative="two.sided",var.equal=T,conf.level=0.95)
```

e) Vai noraida hipotēzi H0: mu(Wet) = mu(Dry) ?

```
!(res$conf.int[1] < 0 & res$conf.int[2] > 0)
```

```
## [1] TRUE
```