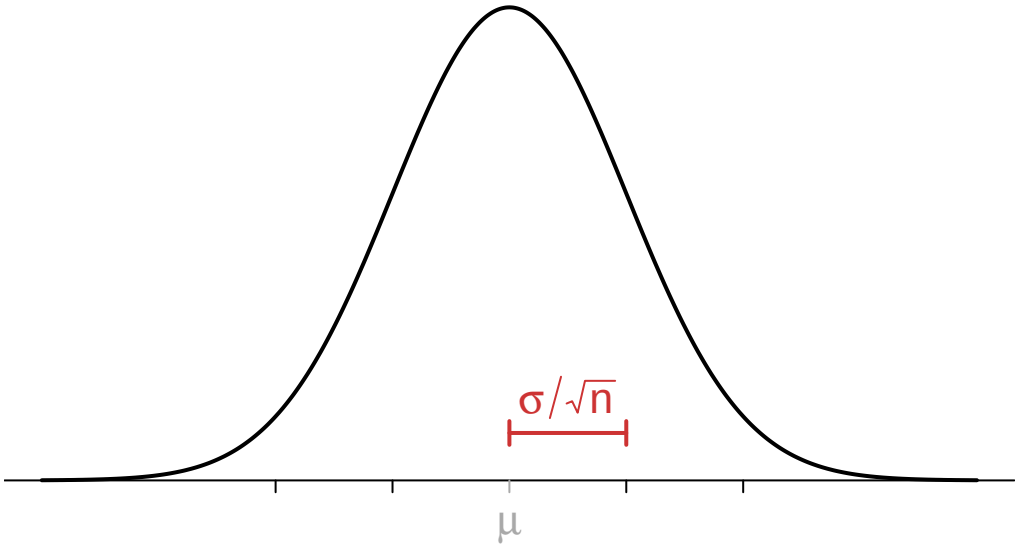
Curs Biostatistica 2017 - Laborator 1 & 2

Contents

1 Intervale de încredere

1.1 Densitatea normală

```
par(bty="n")
x <- seq(-4,4,length=501)
plot(x,dnorm(x),type="l",xaxt="n",yaxt="n",xlab="",ylab="",lwd=2)
abline(h=0)
x <- c(-2,-1,1,2)
segments(x,0,x,-0.01,xpd=TRUE)
segments(0,0,0,-0.01,xpd=TRUE,col="darkgray")
text(0,-0.04,expression(mu),xpd=TRUE,cex=1.3,col="darkgray")
segments(c(0,0,1),c(0.04,0.03,0.03),c(1,0,1),c(0.04,0.05,0.05),lwd=2,col="brown3")
text(0.5,0.07,expression(sigma/sqrt(n)),cex=1.3,col="brown3")
```



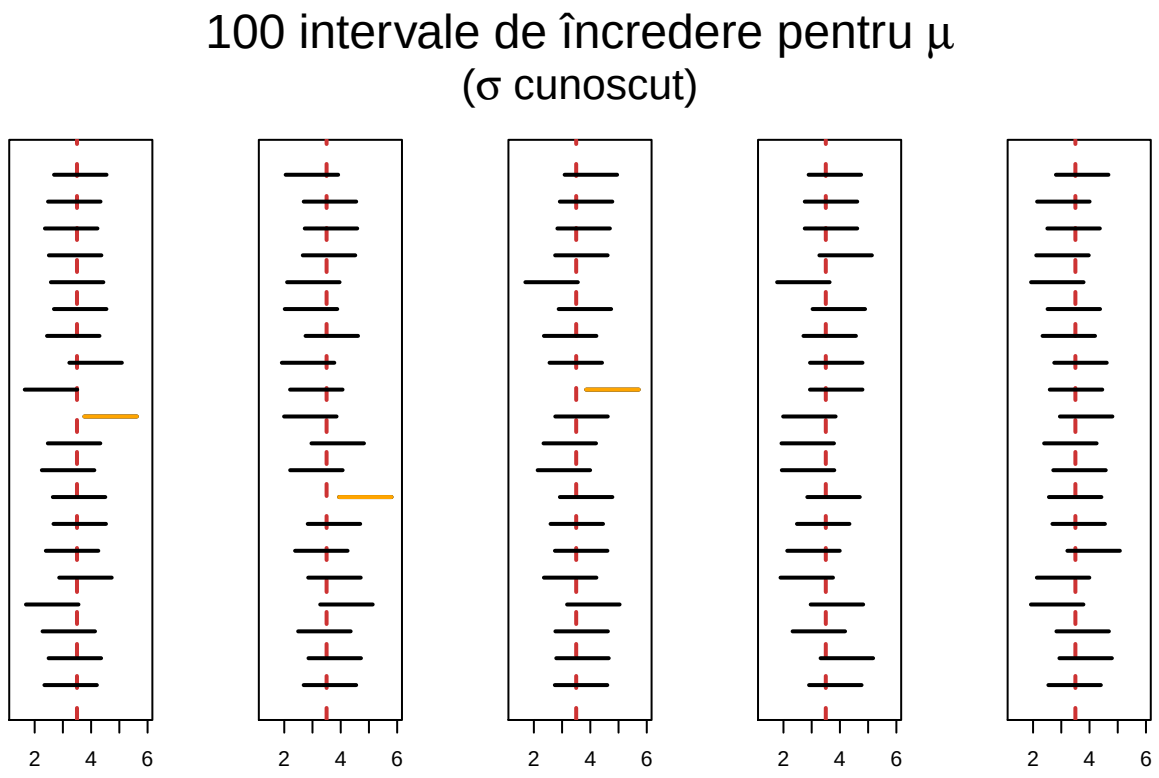
1.2 Intervale de încredere pentru medie

Generarea intervalelor de încredere:

```
p <- 5
n <- 20

lo3 <- hi3 <- lo2 <- hi2 <- lo <- hi <- vector("list",p)

for(i in 1:p) {
  dat <- matrix(rnorm(n*10,3.5,sd=1.5),ncol=10)

  m <- apply(dat,1,mean)
  s <- apply(dat,1,sd)

  lo[[i]] <- m-qnorm(0.975)*1.5/sqrt(10)
  hi[[i]] <- m+qnorm(0.975)*1.5/sqrt(10)

  lo2[[i]] <- m-qnorm(0.975)*s/sqrt(10)
  hi2[[i]] <- m+qnorm(0.975)*s/sqrt(10)

  lo3[[i]] <- m-qt(0.975,9)*s/sqrt(10)
  hi3[[i]] <- m+qt(0.975,9)*s/sqrt(10)
}
```

Intervale de încredere atunci când σ este cunoscut:

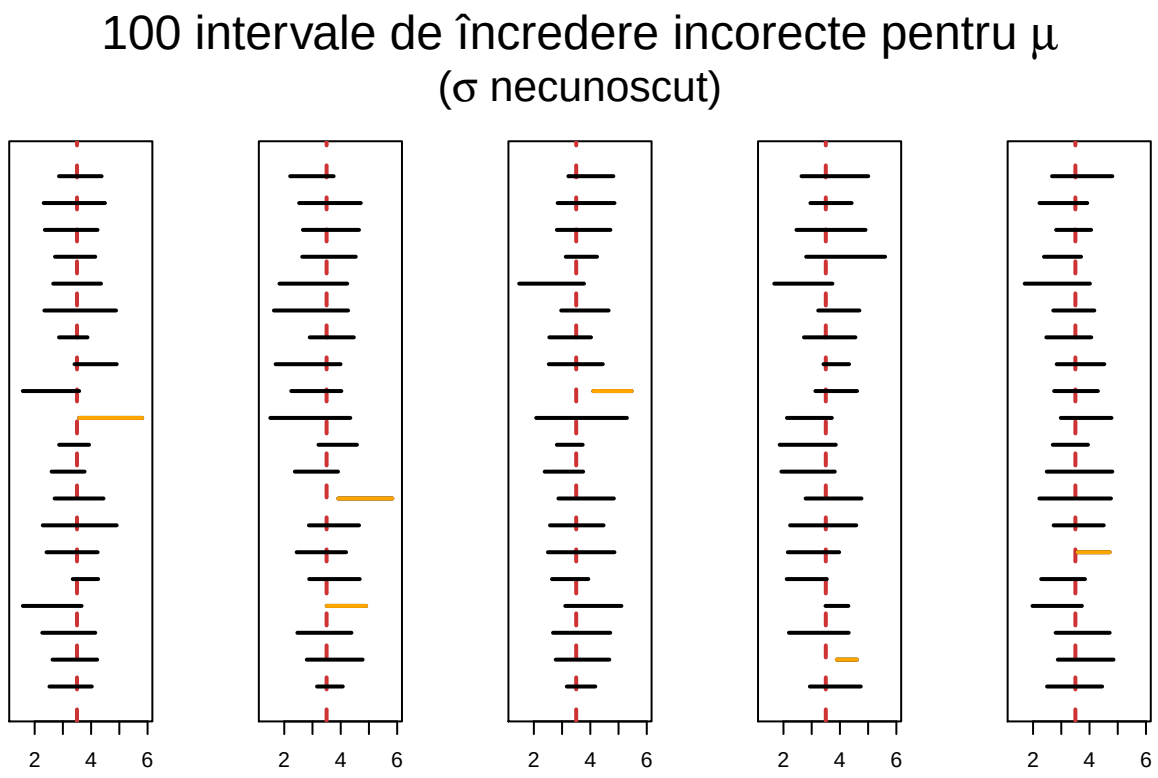
```
r <- range(unlist(c(lo,hi,lo2,hi2,lo3,hi3)))

par(mfrow=c(1,5), las=1, mar=c(5.1,2.1,6.1,2.1))

for(i in 1:p) {
  plot(0,0,type="n",ylim=0.5+c(0,n),xlim=r,ylab="",xlab="",yaxt="n")

  abline(v=3.5,lty=2,col="brown3",lwd=2)

  segments(lo[[i]],1:n,hi[[i]],1:n,lwd=2)

  o <- (1:n)[lo[[i]] > 3.5 | hi[[i]] < 3.5]

  segments(lo[[i]][o],o,hi[[i]][o],o,lwd=2,col="orange")
}

par(mfrow=c(1,1))

mtext(expression(paste("100 intervale de încredere pentru ",mu)),side=3,cex=1.5,xpd=TRUE,line=4)
mtext(expression(paste("(",sigma," cunoscut)")),side=3,cex=1.3,xpd=TRUE,line=2.7)
```



Intervale de încredere **incorecte** atunci când σ nu este cunoscut:

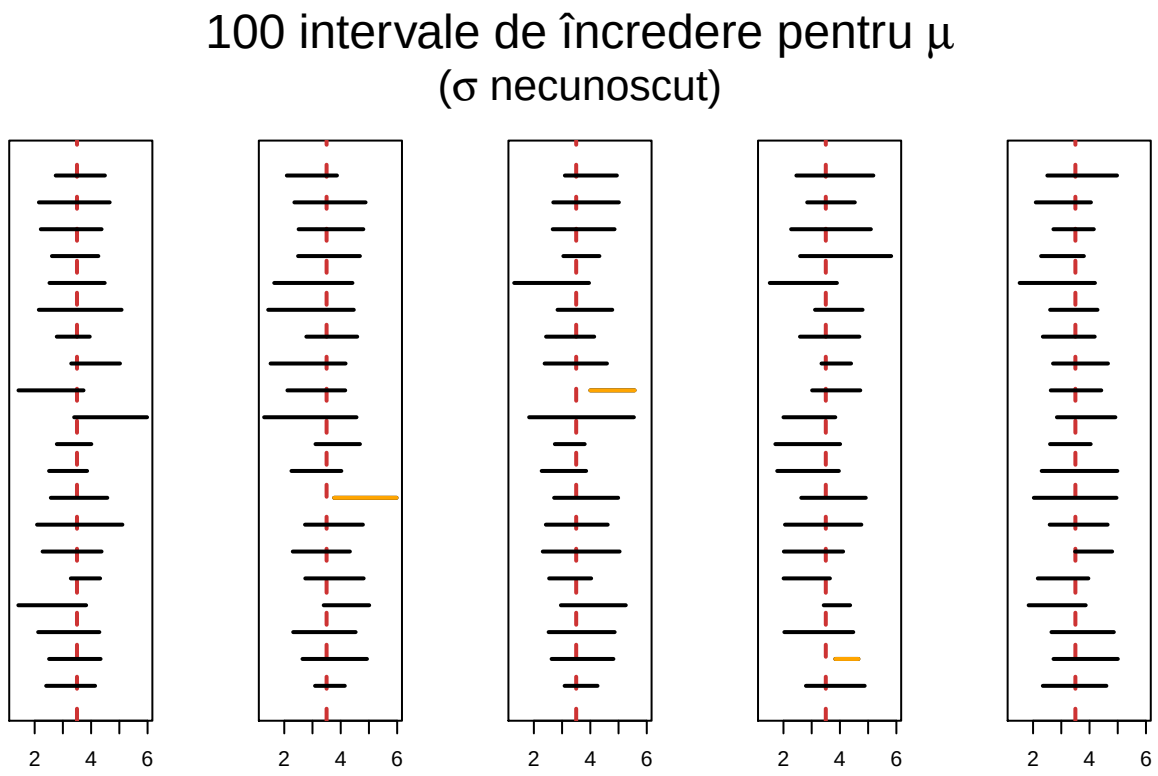
```
par(mfrow=c(1,5), las=1, mar=c(5.1,2.1,6.1,2.1))
for(i in 1:p) {
  plot(0,0,type="n",ylim=0.5+c(0,n),xlim=r,ylab="",xlab="",yaxt="n")
```

```
  abline(v=3.5,lty=2,col="brown3",lwd=2)
  segments(lo2[[i]],1:n,hi2[[i]],1:n,lwd=2)
  o <- (1:n)[lo2[[i]] > 3.5 | hi2[[i]] < 3.5]
  segments(lo2[[i]][o],o,hi2[[i]][o],o,lwd=2,col="orange")
}
par(mfrow=c(1,1))
mtext(expression(paste("100 intervale de încredere incorecte pentru ",mu)),side=3,cex=1.5,xpd=TRUE,line=
mtext(expression(paste("(",sigma," necunoscut)")),side=3,cex=1.3,xpd=TRUE,line=2.7)
```



Intervale de încredere **corecte** atunci când σ nu este cunoscut:

```
par(mfrow=c(1,5), las=1, mar=c(5.1,2.1,6.1,2.1))
for(i in 1:p) {
  plot(0,0,type="n",ylim=0.5+c(0,n),xlim=r,ylab="",xlab="",yaxt="n")
  abline(v=3.5,lty=2,col="brown3",lwd=2)
  segments(lo3[[i]],1:n,hi3[[i]],1:n,lwd=2)
  o <- (1:n)[lo3[[i]] > 3.5 | hi3[[i]] < 3.5]
  segments(lo3[[i]][o],o,hi3[[i]][o],o,lwd=2,col="orange")
}
par(mfrow=c(1,1))
mtext(expression(paste("100 intervale de încredere pentru ",mu)),side=3,cex=1.5,xpd=TRUE,line=4)
mtext(expression(paste("(",sigma," necunoscut)")),side=3,cex=1.3,xpd=TRUE,line=2.7)
```



2 Testarea ipotezelor statistice: inferență asupra unui eșantion

---

2.1 Exemplul 1

Care este temperatura normală a corpului uman ? (vezi articol) Ne dorim să testăm din punct de vedere statistic dacă temperatura medie a corpului uman este de $37°C$ plecând de la următorul set de date descarcă (sursa originală a datelor este *Mackowiak, P. A., Wasserman, S. S., and Levine, M. M. (1992). A Critical Appraisal of 98.6 Degrees F, the Upper Limit of the Normal Body Temperature, and Other Legacies of Carl Reinhold August Wunderlich. Journal of the American Medical Association, 268, 1578-1580*).

Pentru a citi datele putem folosi două metode: sau să le citim direct din pagina de internet (prin comanda `read.table`)

```
file = "https://alexamarioarei.github.io/Teaching/Biostatistics/labs/data/normtemp.txt"
normtemp = read.table(file, header=F, col.names=c("temp","sex","hr"))

head(normtemp)
```

```
##   temp sex hr
## 1 96.3   1 70
## 2 96.7   1 71
```

```
## 3 96.9   1 74
## 4 97.0   1 80
## 5 97.1   1 73
## 6 97.1   1 75
```

sau descărcând local fișierul cu date și înlocuind adresa de internet din `file` cu cea locală.

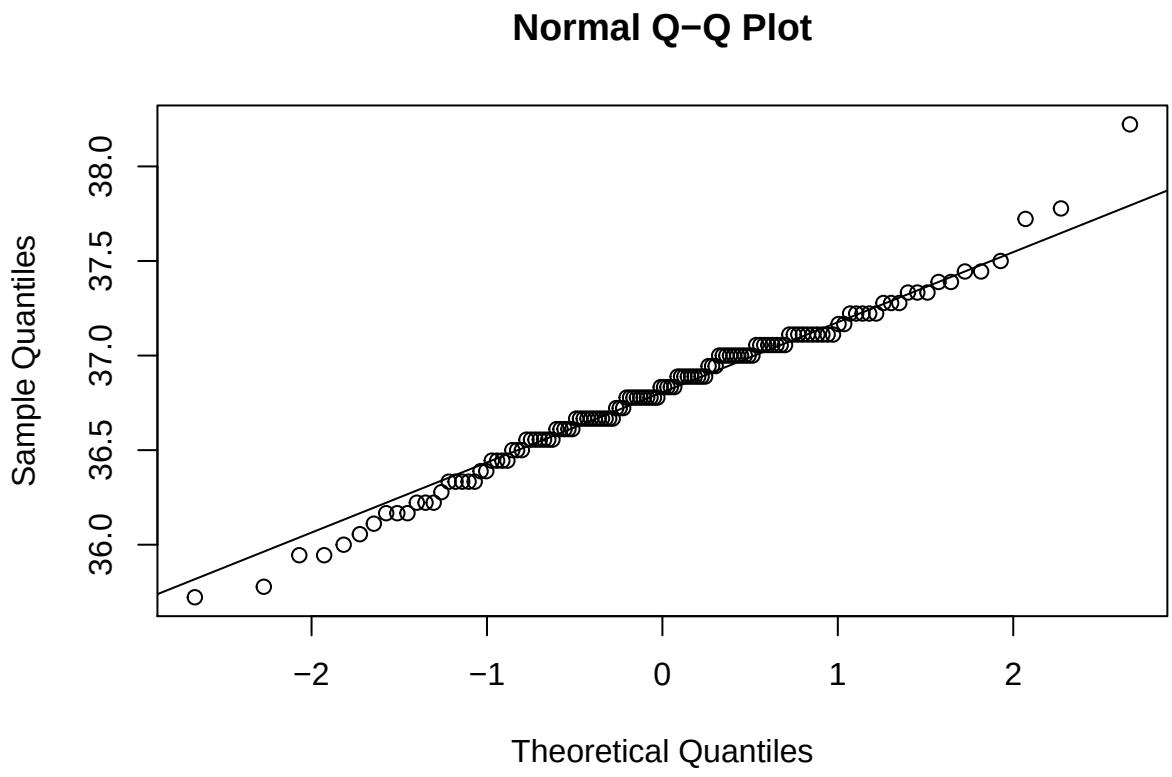
Temperatura apare în grade Fahrenheit și am dori să transformăm în grade Celsius folosind formula:

$$T_C = 5(T_F - 32)/9$$

```
normtemp$tempC = (normtemp$temp - 32)*5/9
degreesC = normtemp$tempC
```

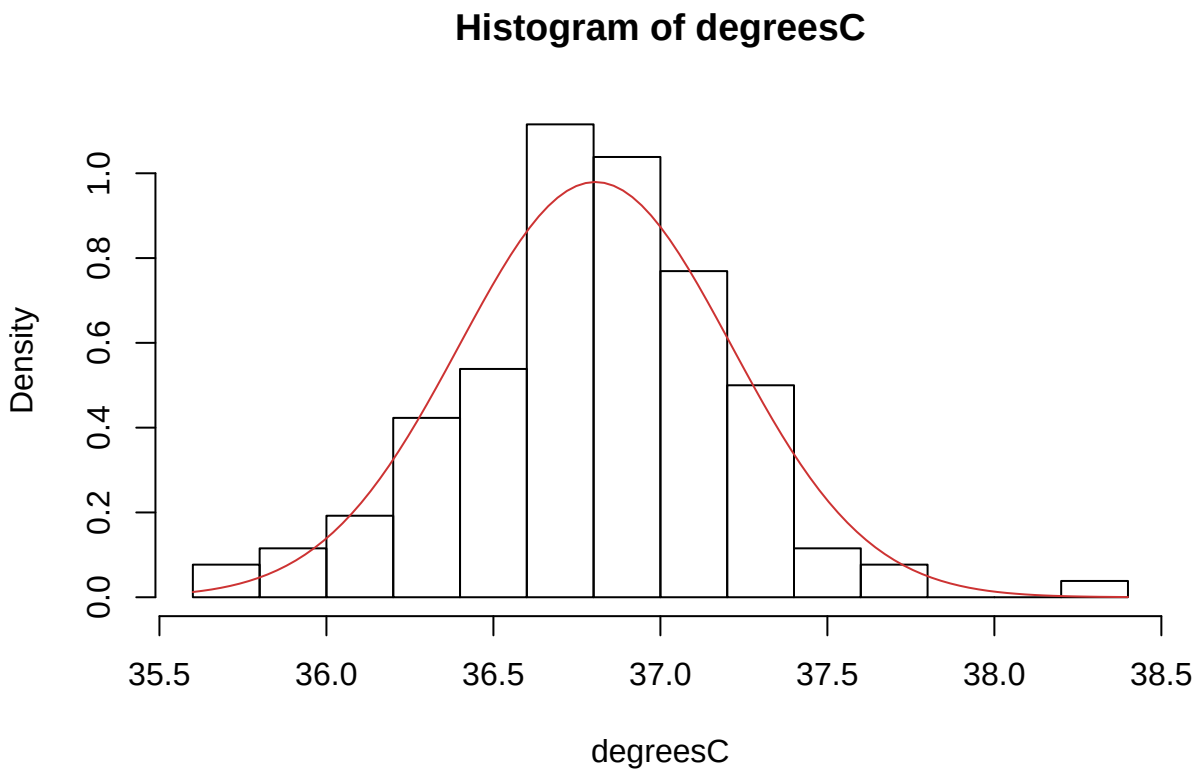
Testul t-student presupune că eșantionul (independent) a provenit dintr-o populație normală și pentru aceasta putem verifica ipoteza de normalitate (`QQ plot`):

```
qqnorm(degreesC)
qqline(degreesC)
```



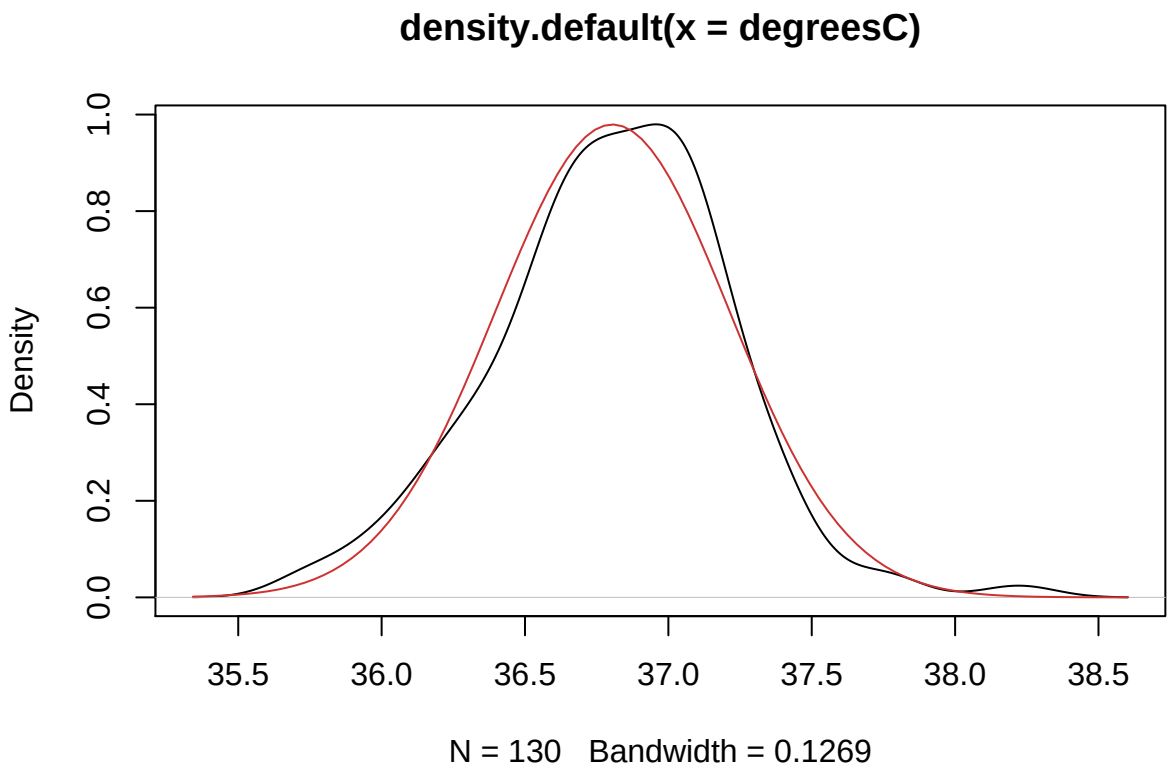
Trasăm histograma:

```
hist(degreesC, probability = T)
degM = mean(degreesC)
degSD = sd(degreesC)
curve(dnorm(x, degM, degSD), add = T, col = "brown3")
```



Trasăm densitatea:

```
plot(density(degreesC))
curve(dnorm(x, degM, degSD), add = T, col = "brown3")
```



Testăm ipoteza de normalitate (folosind testul `Shapiro-Wilk`):

```
shapiro.test(degreesC)# distributia pare sa fie aproape de normala si testul nu detecteaza
```

```
##
##  Shapiro-Wilk normality test
##
## data:  degreesC
## W = 0.98658, p-value = 0.2332
```

```
                      # o abatere semnificativa fata de normala
```

Distribuția pare să fie aproape de normală, testul Shapiro-Wilk nu detectează o deviație semnificantă de la normalitate.

```
t.test(degreesC, mu = 37, alternative = "two.sided") # respingem H0
```

```
##
##  One Sample t-test
##
## data:  degreesC
## t = -5.4548, df = 129, p-value = 2.411e-07
## alternative hypothesis: true mean is not equal to 37
## 95 percent confidence interval:
##  36.73445 36.87581
## sample estimates:
## mean of x
##  36.80513
```

```r
ttest_deg = t.test(degreesC, mu = 37)

ttest_deg$statistic
```

```
##         t
## -5.454823
```

```r
ttest_deg$p.value
```

```
## [1] 2.410632e-07
```

```r
ttest_deg$conf.int
```

```
## [1] 36.73445 36.87581
## attr(,"conf.level")
## [1] 0.95
```

Dacă nu avem datele și avem o problemă de tipul: un eșantion de 130 de persoane a fost selectionat și temperatura corpului a fost masurată. Media eșantionului a fost `36.805` iar abaterea standard `0.4073`. Testati ipoteza nulă că media temperaturii corpului uman este de `37` grade Celsius.

În acest caz avem:

```r
t.obt = (36.805 - 37)/(0.4073/sqrt(130))
t.obt
```

```
## [1] -5.458733
```

```r
qt(c(0.25, 0.975), df = 129) # valorile critice pentru alpha = 0.05
```

```
## [1] -0.6763963  1.9785245
```

```r
2*pt(t.obt, df = 129) # p valoarea pentru testul two-tailed
```

```
## [1] 2.367923e-07
```

Ca să automatizăm aceste calcule putem crea o funcție:

```r
t.single = function(obs.mean, mu, SD, n) {
  t.obt = (obs.mean - mu) / (SD / sqrt(n))
  p.value = pt(abs(t.obt), df=n-1, lower.tail=F)
  print(c(t.obt = t.obt, p.value = p.value))
  warning("P-value pentru one-sided. Dubleaza pentru two-sided.")
}

t.single(36.805, mu = 37, SD = 0.4073, n = 130)
```

```
##         t.obt       p.value
## -5.458733e+00  1.183961e-07
```

```
## Warning in t.single(36.805, mu = 37, SD = 0.4073, n = 130): P-value pentru
## one-sided. Dubleaza pentru two-sided.
```

3 Testarea ipotezelor statistice: inferență asupra a două eșantioane

3.1 Exemplul 1

În contextul exemplului anterior, să presupunem că vrem să vedem dacă există vreo diferență între temperatura medie la bărbați și temperatura medie la femei.

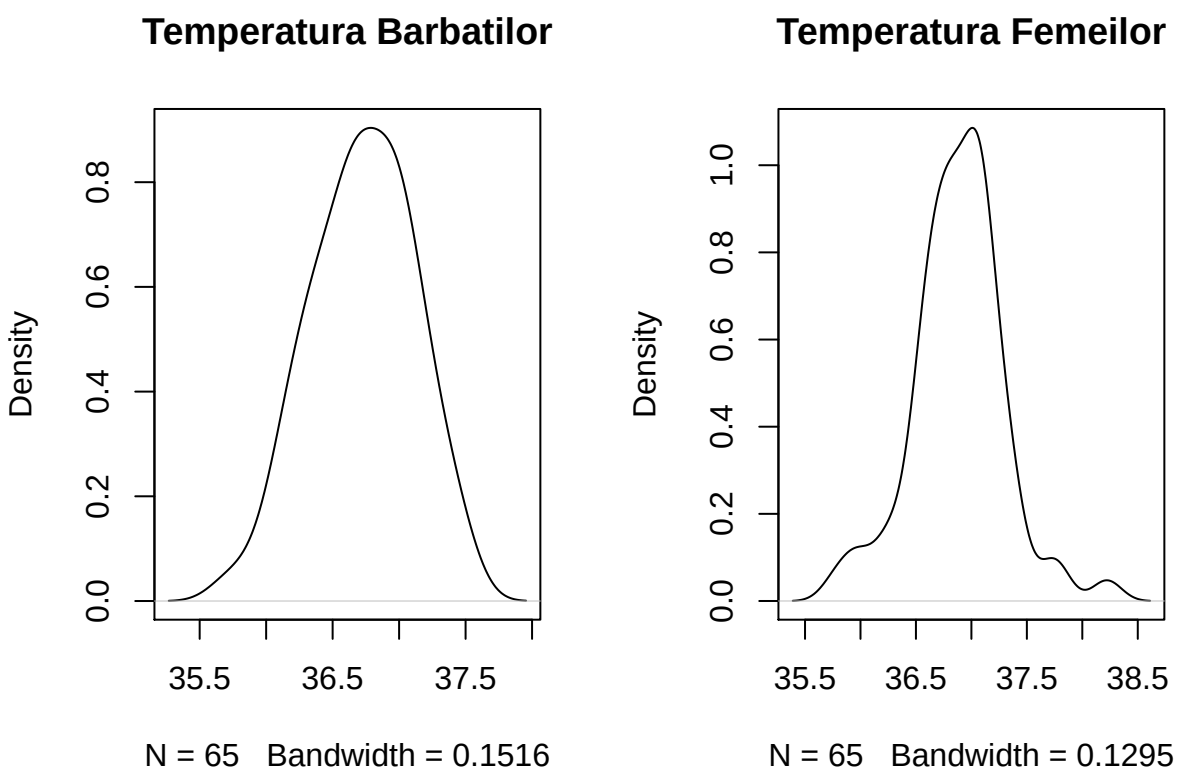
```
str(normtemp)
```

```
## 'data.frame':    130 obs. of  4 variables:
##  $ temp : num  96.3 96.7 96.9 97 97.1 97.1 97.1 97.2 97.3 97.4 ...
##  $ sex  : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ hr   : int  70 71 74 80 73 75 82 64 69 70 ...
##  $ tempC: num  35.7 35.9 36.1 36.1 36.2 ...
```

```
tempB = normtemp$tempC[which(normtemp$sex == 1)]
tempF = normtemp$tempC[which(normtemp$sex == 2)]
```

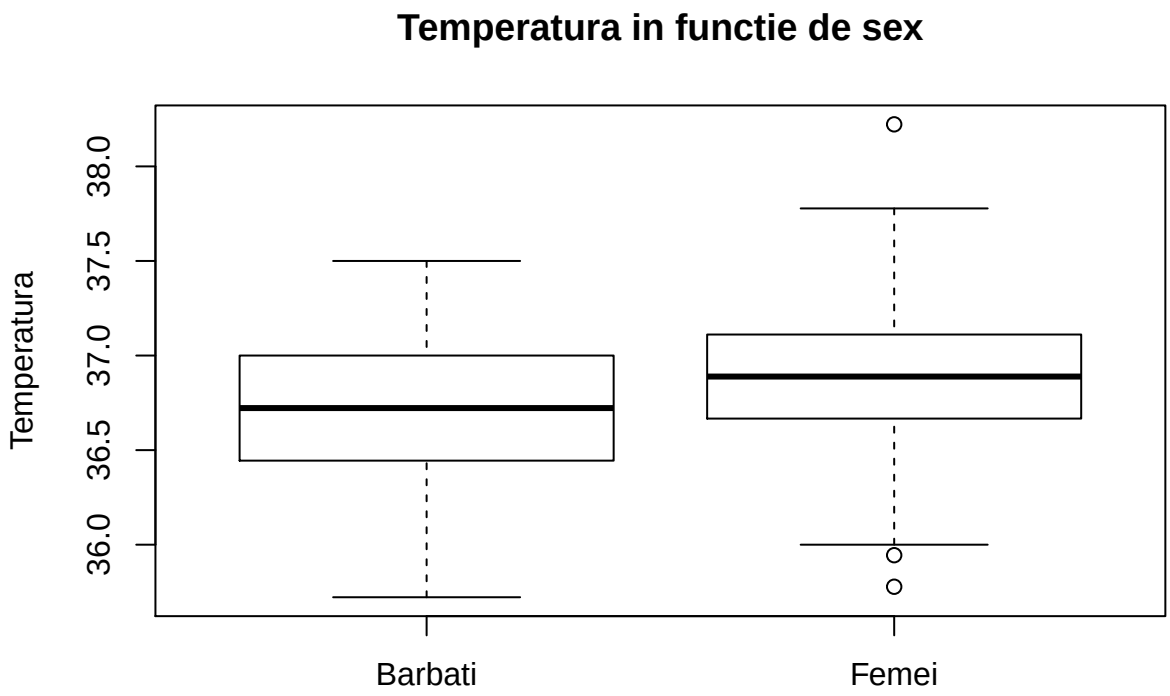
Ilustrare a temperaturii bărbaților și a femeilor:

```
par(mfrow=c(1,2))
plot(density(tempB), main="Temperatura Barbatilor")
plot(density(tempF), main="Temperatura Femeilor")
```



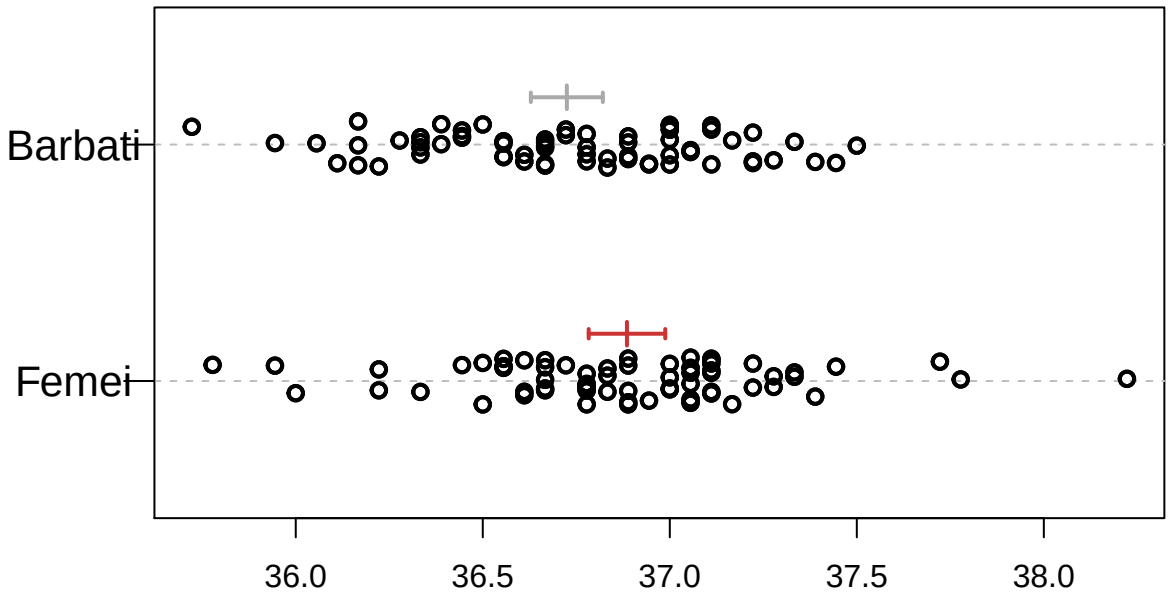
Sub formă de `boxplot`:

```
par(mfrow = c(1,1))
boxplot(tempB, tempF, ylab="Temperatura",     # plot and label y-axis
                names=c("Barbati","Femei"),                          # group names on x-axis
                main="Temperatura in functie de sex")         # main title
```



Trasarea datelor împreună cu intervalele de încredere:

```
source("functions/dotplot.R")

dotplot(tempB, tempF, labels=c("Barbati","Femei"))
```



Testarea ipotezelor statistice cu ajutorul testului t-student (corecția lui `Welch`):

```
t.test(tempB, tempF) # Welch correction
```

```
##
##  Welch Two Sample t-test
##
## data:  tempB and tempF
## t = -2.2854, df = 127.51, p-value = 0.02394
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.29980476 -0.02156277
## sample estimates:
## mean of x mean of y
##  36.72479  36.88547
```

Verificăm dacă cele două eșantioane au variațe egale (folosim testul lui `Fisher`):

```
var.test(tempB, tempF)
```

```
##
##  F test to compare two variances
##
## data:  tempB and tempF
## F = 0.88329, num df = 64, denom df = 64, p-value = 0.6211
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.5387604 1.4481404
```

```
## sample estimates:
## ratio of variances
##          0.8832897
```

Aplicăm acum testul t-student cu opțiunea de varianțe egale (`pooled variance`):

```
t.test(tempB, tempF, var.equal = T) # without Welch correction
```

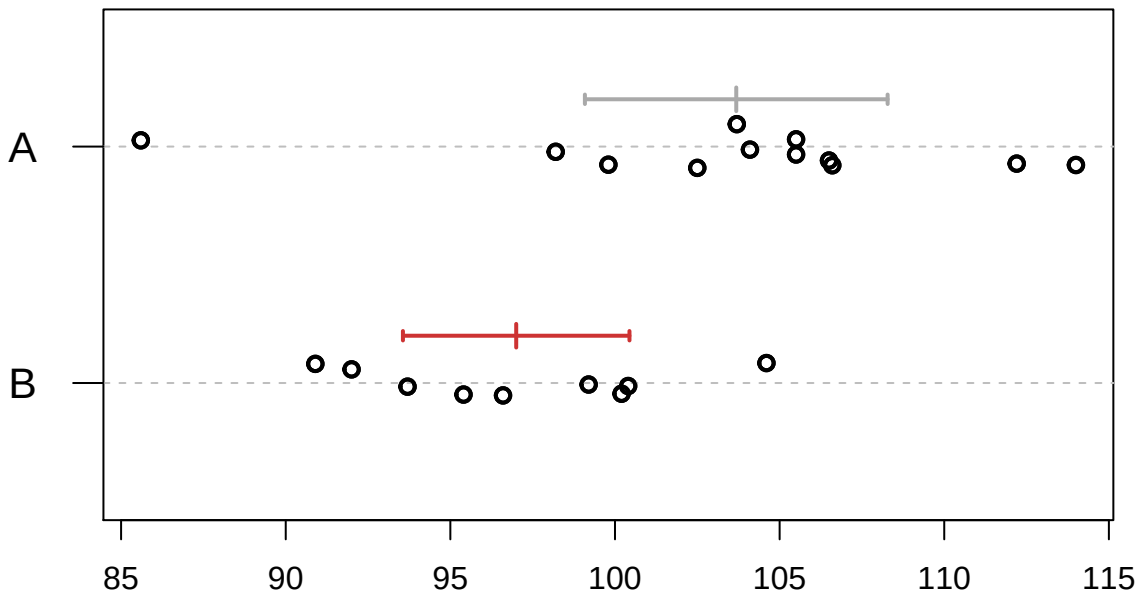
```
##
##  Two Sample t-test
##
## data:  tempB and tempF
## t = -2.2854, df = 128, p-value = 0.02393
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.29979966 -0.02156786
## sample estimates:
## mean of x mean of y
##  36.72479  36.88547
```

3.2 Exemplul 2

```
# Example data
x <- c(102.5, 106.6,  99.8, 106.5, 103.7, 105.5, 98.2, 104.1,  85.6, 105.5, 114.0, 112.2)
y <- c( 93.7,  90.9, 100.4,  92.0, 100.2, 104.6, 95.4,  96.6,  99.2)

# Two-sided t-test allowing un-equal population SDs
t.test(x,y)
```

```
##
##  Welch Two Sample t-test
##
## data:  x and y
## t = 2.6041, df = 18.475, p-value = 0.01769
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##   1.30124 12.06543
## sample estimates:
## mean of x mean of y
##  103.6833   97.0000
```

```
dotplot(x,y)
```

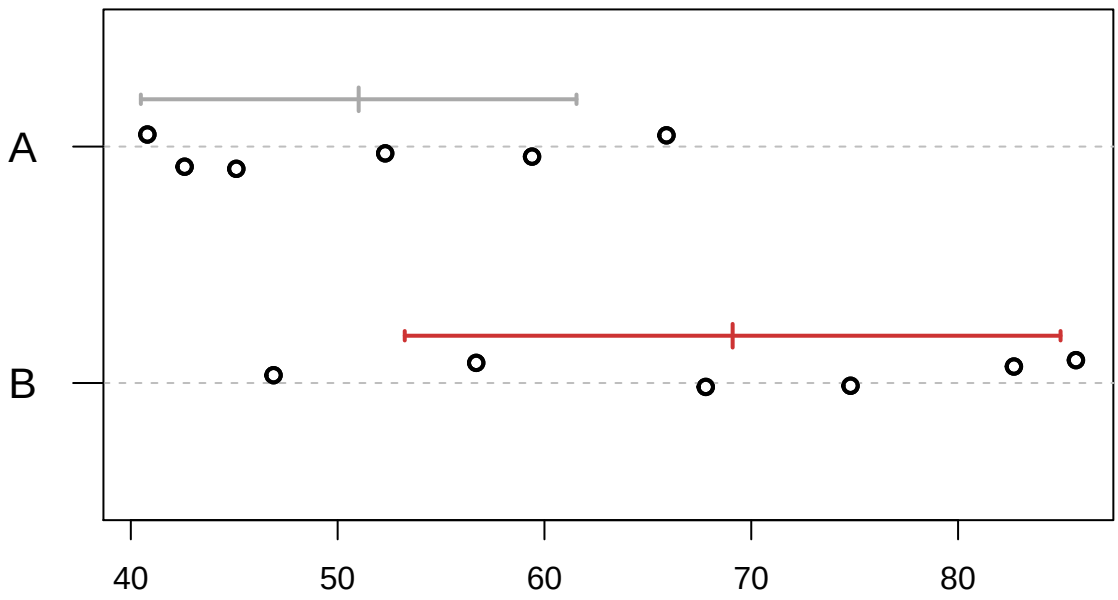


3.3 Exemplul 3

```
# One-tailed test example
x <- c(59.4, 52.3, 42.6, 45.1, 65.9, 40.8)
y <- c(82.7, 56.7, 46.9, 67.8, 74.8, 85.7)

# One-tailed t-test
t.test(x,y,alt="less")
```

```
##
##  Welch Two Sample t-test
##
## data:  x and y
## t = -2.4421, df = 8.6937, p-value = 0.01907
## alternative hypothesis: true difference in means is less than 0
## 95 percent confidence interval:
##       -Inf -4.454703
## sample estimates:
## mean of x mean of y
##  51.01667  69.10000
```

```
# The dotplot
dotplot(x,y)
```

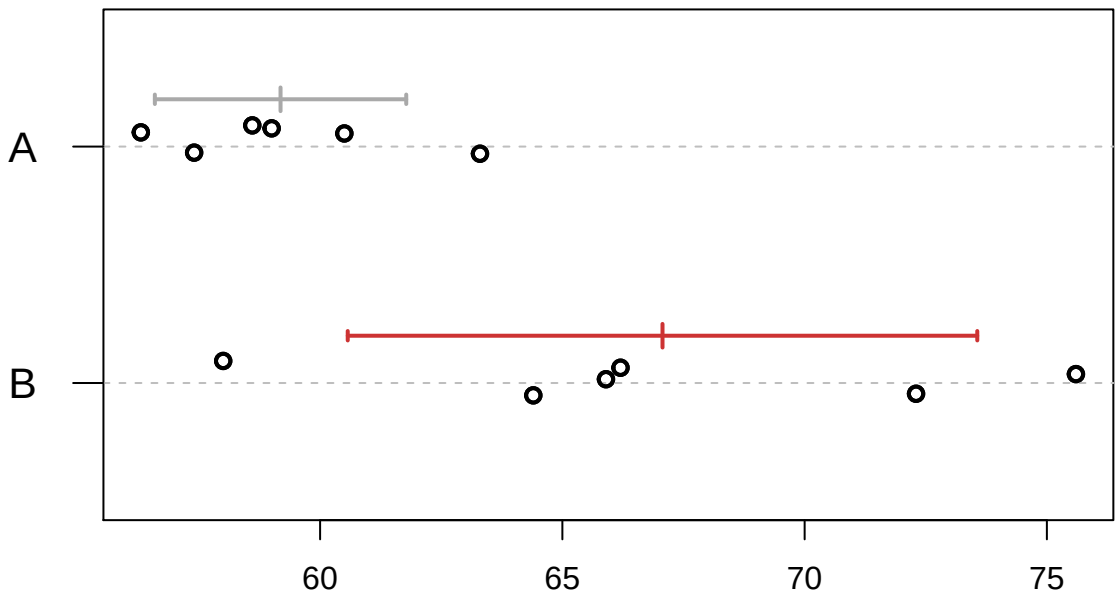


3.4 Exemplul 4

```
# another one-tailed test example
x <- c(63.3, 58.6, 59.0, 60.5, 56.3, 57.4)
y <- c(75.6, 65.9, 72.3, 58.0, 64.4, 66.2)
t.test(x,y,alt="less")
```

```
##
##  Welch Two Sample t-test
##
## data:  x and y
## t = -2.8968, df = 6.5546, p-value = 0.01242
## alternative hypothesis: true difference in means is less than 0
## 95 percent confidence interval:
##       -Inf -2.674212
## sample estimates:
## mean of x mean of y
##  59.18333  67.06667
```

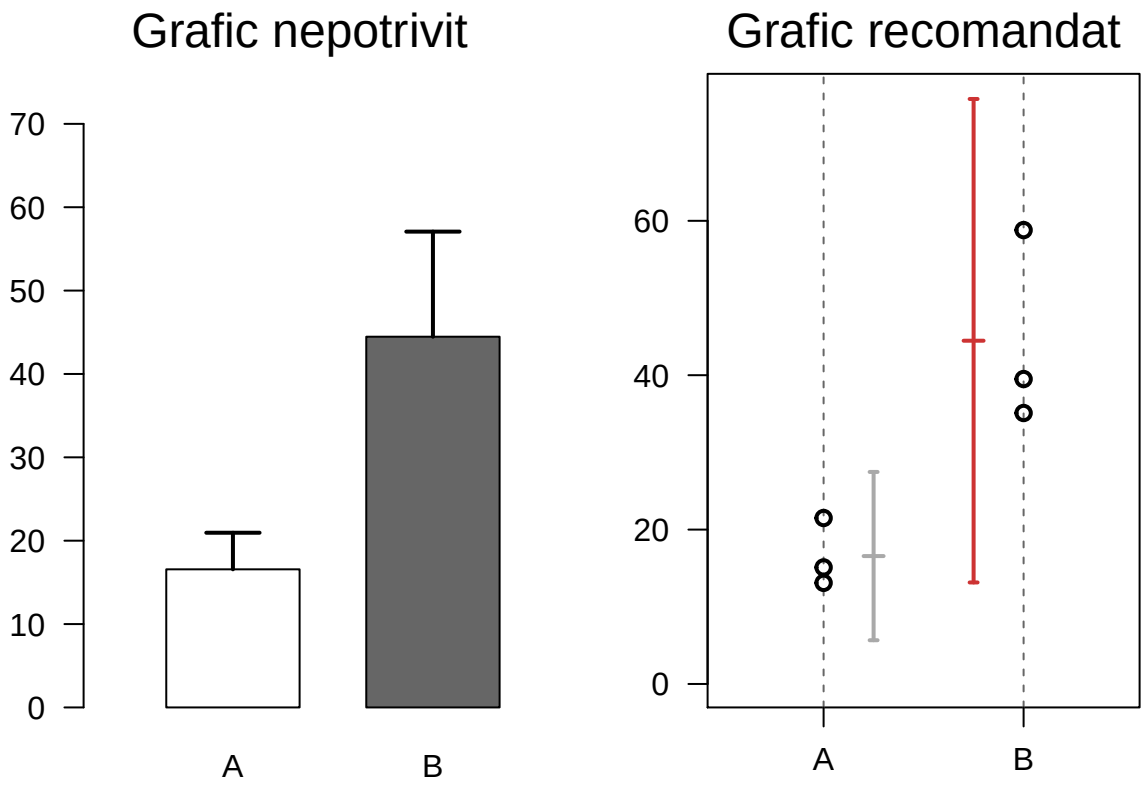
```
dotplot(x,y)
```



3.5 Grafic bun / Grafic rau

```
x <- c(15.1, 13.1, 21.5)
y <- c(35.1, 39.5, 58.8)

par(mar=c(4,4,2,1),mfrow=c(1,2),las=1)

barplot(c(mean(x),mean(y)),width=1,space=c(0.5,0.5),
        col=c("white","gray40"),xlim=c(0,3),names=c("A","B"),
        ylim=c(0,76))
segments(1,mean(x),1,mean(x)+sd(x),lwd=2)
segments(0.8,mean(x)+sd(x),1.2,mean(x)+sd(x),lwd=2)
segments(2.5,mean(y),2.5,mean(y)+sd(y),lwd=2)
segments(2.3,mean(y)+sd(y),2.7,mean(y)+sd(y),lwd=2)
mtext("Grafic nepotrivit",cex=1.5,line=0.5)

plot(rep(0:1,c(3,3)),c(x,y),xaxt="n",ylim=c(0,76),xlim=c(-0.5,1.5),ylab="",xlab="")
abline(v=0:1,col="gray40",lty=2)
points(rep(0:1,c(3,3)),c(x,y),lwd=2)
mtext("Grafic recomandat",cex=1.5,line=0.5)
xci <- t.test(x)$conf.int
yci <- t.test(y)$conf.int
segments(0.25,xci[1],0.25,xci[2],lwd=2,col="darkgray")
segments(c(0.23,0.23,0.2),c(xci,mean(x)),c(0.27,0.27,0.3),c(xci,mean(x)),lwd=2,col="darkgray")
```

```r
segments(1-0.25,yci[1],1-0.25,yci[2],lwd=2,col="brown3")
segments(1-c(0.23,0.23,0.2),c(yci,mean(y)),1-c(0.27,0.27,0.3),c(yci,mean(y)),lwd=2,col="brown3")
u <- par("usr")
segments(0:1,u[3],0:1,u[3]-diff(u[3:4])*0.03,xpd=TRUE)
text(0:1,u[3]-diff(u[3:4])*0.08,c("A","B"),xpd=TRUE)
```



4 Testarea ipotezelor statistice: inferență asupra a două eșantioane dependente (perechi)

Considerăm următorul set de date din pachetul `MASS` (luarea in greutate de catre femei anorexice):

```r
data(anorexia, package="MASS")
attach(anorexia)
str(anorexia)
```

```
## 'data.frame':    72 obs. of  3 variables:
##  $ Treat : Factor w/ 3 levels "CBT","Cont","FT": 2 2 2 2 2 2 2 2 2 2 ...
##  $ Prewt : num  80.7 89.4 91.8 74 78.1 88.3 87.3 75.1 80.6 78.4 ...
##  $ Postwt: num  80.2 80.1 86.4 86.3 76.1 78.1 75.1 86.7 73.5 84.6 ...
```

```r
ft=subset(anorexia,Treat="FT") # family treatment
```

Testăm dacă există diferențe între luarea în greutate înainte de tratament și după tratament:

```r
with(ft, t.test(Postwt-Prewt, mu=0, alternative="greater"))
```

```
##
##  One Sample t-test
##
## data:  Postwt - Prewt
## t = 2.9376, df = 71, p-value = 0.002229
## alternative hypothesis: true mean is greater than 0
## 95 percent confidence interval:
##  1.195825      Inf
## sample estimates:
## mean of x
##  2.763889
```

sau

```r
with(ft, t.test(Postwt, Prewt, paired=T, alternative="greater"))
```

```
##
##  Paired t-test
##
## data:  Postwt and Prewt
## t = 2.9376, df = 71, p-value = 0.002229
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  1.195825      Inf
## sample estimates:
## mean of the differences
##                2.763889
```

Curs Biostatistica 2017 - Laborator 3 & 4

Contents

1 Compararea proporțiilor, tabele de contingență 2×2

1.1 Aproximarea normală

> Un studiu clinic a investigat efectele metodelor contraceptive orale (OC) asupra bolilor de inimă la femeile cu vârste între 40 și 44 de ani. Cercetătorii au găsit că printre 5000 de femei care utilizau metode contraceptive orale la momentul studiului (cazuri), 13 dintre acestea au dezvoltat un infarct miocardic (MI) (pe o perioadă de 3 ani) pe când printre 10000 de femei care nu au folosit niciodată OC (grupul de control) doar 7 au dezvoltat MI (pe aceeași perioadă). Vrem să vedem dacă există vreo asociere între consumul de anticoncepționale pe cale orală și incidența infarctului miocardic (pe această perioadă).

Dacă notăm cu $p_1 = \mathbb{P}(MI \,|\, OC)$ și $p_2 = \mathbb{P}(MI \,|\, non - OC)$ atunci vrem să testăm:

$$H_0 : \; p_1 = p_2$$
$$H_1 : \; p_1 \neq p_2$$

```
n1 = 5000 # nr total cazuri OC
n11 = 13 # nr cazuri cu MI

n2 = 10000 # nr total control non-OC
n21 = 7 # nr control cu MI

p1 = n11/n1
p2 = n21/n2

p = (n11+n21)/(n1+n2) # proportia comuna - pooled p

# Verificam daca putem aplica aproximarea normala
n1*p*(1-p)>5
```

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

```
n2*p*(1-p)>5
```

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

```
# Calculam statistica de test cu corectia de continuitate
z = (abs(p1-p2)-0.5*(1/n1+1/n2))/sqrt(p*(1-p)*(1/n1+1/n2))
z
```

```
## [1] 2.768839
```

```
# Calcul de p-valoare: test bilateral
pval = min(2*(1-pnorm(z)),1)
pval
```

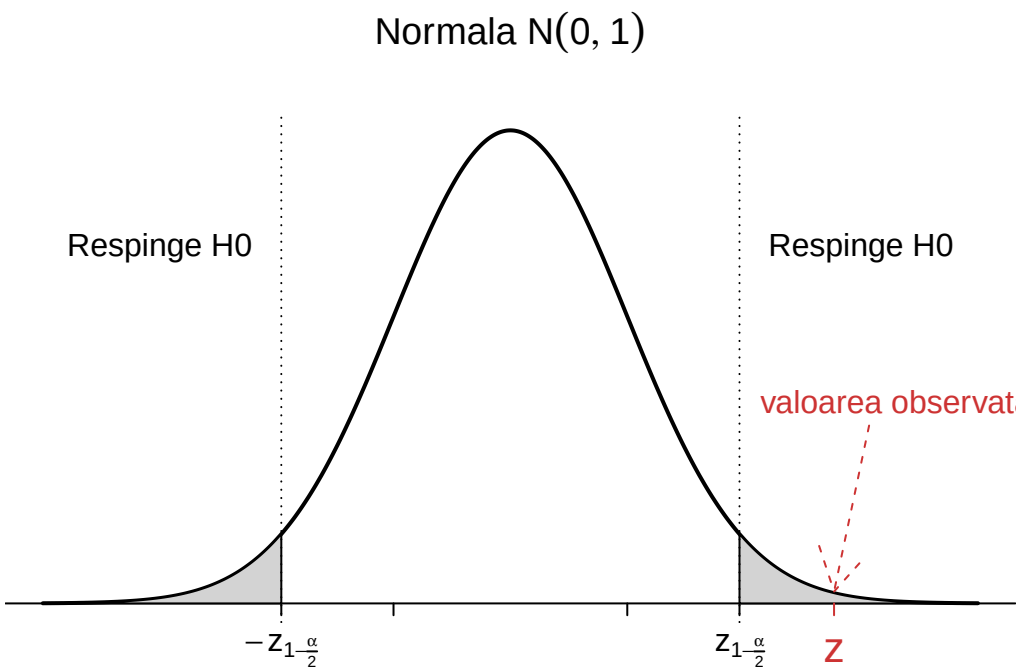
```
## [1] 0.005625635
```

```
# Intervalul de incredere

cat("Intervalul de incredere pentru p1-p2 la pragul de semnificatie 95% este ","IC = [", p1-p2 - qnorm(
```

```
## Intervalul de incredere pentru p1-p2 la pragul de semnificatie 95% este  IC = [ 0.0006612366 , 0.003
```

```
# Intervalul de incredere Agresti & Caffo 2000

p1b = (n11+1)/(n1+2)
p2b = (n21+1)/(n2+2)

cat("Intervalul de incredere (Agresti-Caffo) pentru p1-p2 la pragul de semnificatie 95% este ","IC = ["
```

```
## Intervalul de incredere (Agresti-Caffo) pentru p1-p2 la pragul de semnificatie 95% este  IC = [ 0.000
```



Concluzionăm că folosirea de anticoncepționale pe cale orală este semnificativ asociat cu incidența crescută de cazuri de MI pe perioada de 3 ani. Puteți crea o funcție care să automatizeze procesul ?

1.2 Pearson χ^2

Considerăm aceeași problemă de mai sus dar o scriem sub formă de tabel de contingență 2×2 (tabelul observat):

	MI	non-MI	Total
OC	13	4987	5000
non-OC	7	9993	10000
Total	20	14980	15000

Calculul tabelului de pe care ne așteptăm să-l observăm ($E_{ij} = \frac{n_{i\cdot} n_{\cdot j}}{n}$):

```
# Observat
n11 = 13
n1o = 5000
n12 = n1o-n11

n21 = 7
n2o = 10000
n22 = n2o-n21
```

```r
no1 = n11+n21
no2 = n12+n22

n = n1o+n2o

#Asteptat
e11 = n1o*no1/n
e12 = n1o*no2/n
e21 = n2o*no1/n
e22 = n2o*no2/n

Mobs = matrix(c(n11,n12,n21,n22),ncol = 2, byrow = T, dimnames = list(c("OC","non-OC"), c("MI", "non-MI

Mexp = matrix(c(e11,e12,e21,e22),ncol = 2, byrow = T, dimnames = list(c("OC","non-OC"), c("MI", "non-MI
Mexp
```

```
##               MI   non-MI
## OC      6.666667 4993.333
## non-OC 13.333333 9986.667
```

	MI	non-MI
OC	6.666667	4993.333
non-OC	13.333333	9986.667

Calculul statisticii de test cu corecția lui Yates:

$$X^2 = \sum_{i=1}^{2}\sum_{j=1}^{2} \frac{(|O_{ij} - E_{ij}| - 0.5)^2}{E_{ij}} \sim_{H_0} \chi^2_1$$

```r
X2 = (abs(n11-e11)-0.5)^2/e11 + (abs(n12-e12)-0.5)^2/e12 + (abs(n21-e21)-0.5)^2/e21 + (abs(n22-e22)-0.5
X2
```

```
## [1] 7.666472
```

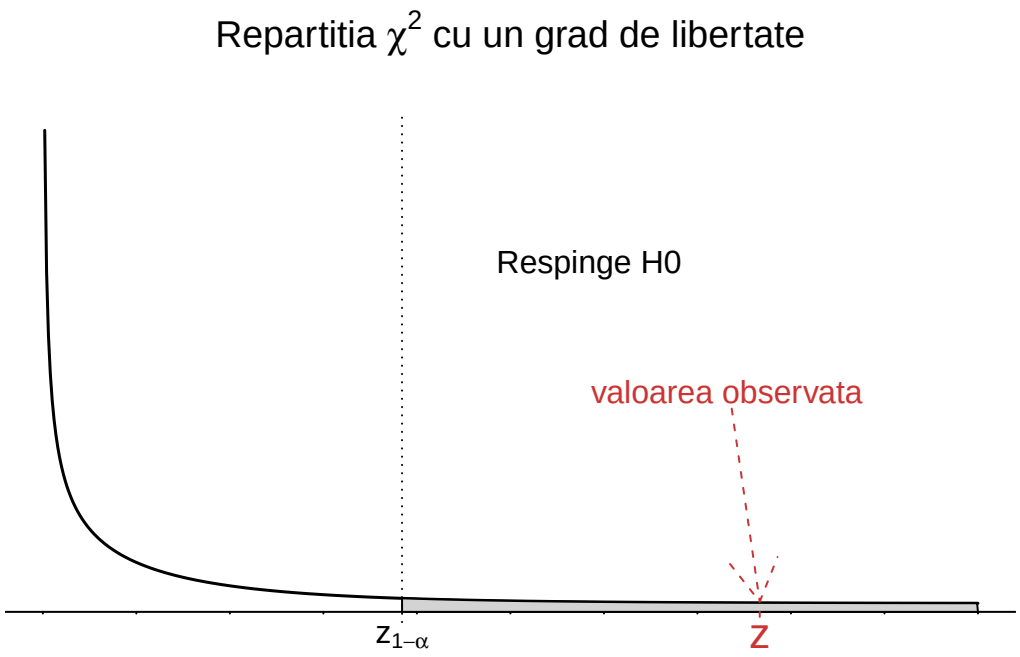
```r
pval = 1-pchisq(X2,1) #df = 1
pval
```

```
## [1] 0.005625635
```

Sau folosind testul lui Pearson cu corecția lui Yates `chisq.test` avem:

```r
chisq.test(Mobs)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  Mobs
## X-squared = 7.6665, df = 1, p-value = 0.005626
```



Același rezultat se obține și dacă folosim testul `prop.test`, acesta fiind un caz particular al testului hi-pătrat:

```
prop.test(Mobs)
```

```
##
##  2-sample test for equality of proportions with continuity
##  correction
##
## data:  Mobs
## X-squared = 7.6665, df = 1, p-value = 0.005626
## alternative hypothesis: two.sided
## 95 percent confidence interval:
##  0.0002463116 0.0035536884
## sample estimates:
## prop 1 prop 2
## 0.0026 0.0007
```

1.3 Raportul de verosimilitate maximă

În contextul exemplului de mai sus vrem să vedem testul bazat pe raportul de verosimilitate. Considerând modelul multinomial $(n_{11}, n_{12}, n_{21}, n_{22}) \sim \mathcal{M}(n; p_{11}, p_{12}, p_{21}, p_{22})$, obținem raportul de verosimilitate

$$\Lambda(x) = \frac{\sup_{\theta \in \Theta_0} L(\theta|x)}{\sup_{\theta \in \Theta} L(\theta|x)} = \prod_{i=1}^{2} \prod_{j=1}^{2} \left(\frac{n_{i\cdot} \times n_{\cdot j}}{n \times n_{ij}} \right)^{n_{ij}}$$

și din teorema lui Wilks (cazul multidimensional) avem $-2\log\Lambda \to \chi^2(d-d_0)$ unde $d=\dim(\Theta)$ și $d_0=\dim(\Theta_0)$. În cazul nostru

$$\Theta = \left\{(p_{11},p_{12},p_{21},p_{22}) \,|\, p_{ij}\in(0,1),\ \sum_{i=1}^2\sum_{j=1}^2 p_{ij}=1\right\}$$
$$\Theta_0 = \left\{(p_1q_1,p_1q_2,p_2q_1,p_2q_2) \,|\, p_i,q_j\in(0,1),\ \sum_{i=1}^2 p_i=1,\ \sum_{j=1}^2 q_j=1\right\}$$

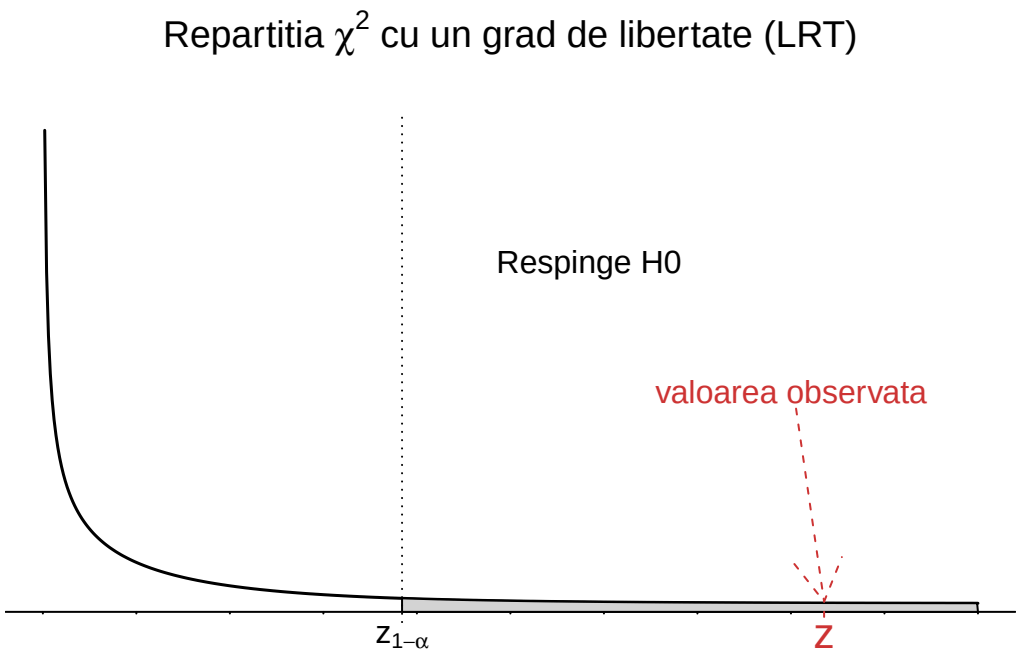
unde p_i și q_j sunt repartițiile marginale. Obținem că $\dim(\Theta)=4-1$ iar $\dim(\Theta_0)=4-2$, deci $-2\log\Lambda\to\chi^2(1)$.

```
# Observat
n11 = 13
n1o = 5000
n12 = n1o-n11

n21 = 7
n2o = 10000
n22 = n2o-n21

no1 = n11+n21
no2 = n12+n22

LRT = n11*log((n1o*no1)/(n*n11)) + n12*log((n1o*no2)/(n*n12)) + n21*log((n2o*no1)/(n*n21)) + n22*log((n
LRT = -2*LRT
LRT
```

```
## [1] 8.354617
```

```
pval = 1-pchisq(LRT,1) #df = 1
pval
```

```
## [1] 0.003847085
```



Să creăm o funcție care automatizează procesul:

```
LRT1 = function(dat){
  # dat este sub forma de matrice
  rs = rowSums(dat) # apply(dat, 1, sum)
  cs = colSums(dat) # apply(dat, 2, sum)

  n = sum(dat)

  expected <- outer(rs,cs,"*")/n

  lrt <- -2*sum(dat * log(expected/dat))

  dm = dim(dat) # dimensiunea tabloului pentru a calcula gradele de libertate
  pval = 1-pchisq(lrt,(dm[1]-1)*(dm[2]-1))

  cat("Statistica LRT este ", lrt, "\n")
  cat("P-valoarea testului bazat pe raportul de verosimilitate este ", pval)

  return(list(statistic = lrt, pvalue = pval))
}

Mobs = matrix(c(n11,n12,n21,n22),ncol = 2, byrow = T, dimnames = list(c("OC","non-OC"), c("MI", "non-MI

LRT1(Mobs)
```

```
## Statistica LRT este  8.354617
```

```
## P-valoarea testului bazat pe raportul de verosimilitate este  0.003847085
```

```
## $statistic
## [1] 8.354617
##
## $pvalue
## [1] 0.003847085
```

1.4 Testul exact al lui Fisher

Să presupunem că vrem să investigăm legătura dintre regimul bogat în sare și decesul datorat unei boli cardiovasculare (CVD). Să presupunem că suntem în contextul unui studiu retrospectiv efectuat pe un grup de bărbați cu vârste cuprinse între 50 și 54 de ani dintr-o anumită regiune geografică care au decedat pe parcursul unui luni. S-a încercat introducerea în studiu a unui grup cât mai omogen (s-a încercat includerea în studiu a unui număr egal de persoane care au decedat din cauză de CVD și care au decedat din alte cauze). S-a obținut următorul tabel:

	Ridicat Sare	Scazut Sare	Total
non-CVD	2	23	25
CVD	5	30	35
Total	7	53	60

Tabelul pe care ne așteptam să-l obținem (H_0) este:

```
# Observat
n11 = 2
n1o = 25
n12 = n1o-n11

n21 = 5
n2o = 35
n22 = n2o-n21

no1 = n11+n21
no2 = n12+n22

n = n1o+n2o

#Asteptat
e11 = n1o*no1/n
e12 = n1o*no2/n
e21 = n2o*no1/n
e22 = n2o*no2/n

MobsF = matrix(c(n11,n12,n21,n22),ncol = 2, byrow = T, dimnames = list(c("non-CVD", "CVD"), c("Ridicat S

MexpF = matrix(c(e11,e12,e21,e22),ncol = 2, byrow = T, dimnames = list(c("non-CVD", "CVD"), c("Ridicat S
MexpF
```

```
##         Ridicat Sare Scazut Sare
## non-CVD     2.916667    22.08333
## CVD         4.083333    30.91667
```

	Ridicat Sare	Scazut Sare
non-CVD	2.916667	22.08333
CVD	4.083333	30.91667

Observăm că avem două celule în tabelul așteptat care conțin mai puțin de 5 observații prin urmare nu putem folosi metodele de mai sus (aproximarea normală, testul lui Pearson sau testul bazat pe raportul de verosimilitate). Dacă am încerca am obține:

```
# Testul lui Pearson (Hi patrat)

chisq.test(MobsF)
```

```
## Warning in chisq.test(MobsF): Chi-squared approximation may be incorrect
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  MobsF
## X-squared = 0.11552, df = 1, p-value = 0.7339
```

```
# Testul bazat pe raportul de verosimilitate

LRT1(MobsF)
```

```
## Statistica LRT este  0.5810517
## P-valoarea testului bazat pe raportul de verosimilitate este  0.4459004
```

```
## $statistic
## [1] 0.5810517
##
## $pvalue
## [1] 0.4459004
```

Enumerăm tabelele și probabilitățile lor de apariție:

```
# Fixez marginalele

n1o = 25
n2o = 35

no1 = 7
no2 = 53

for (i in 0:7){
  cat("---------------------------------------\n")
  cat("Tabelul ", i+1, " :\n")

  # calculez valorile din tabel
  n11 = i
  n12 = n1o - n11
  n21 = no1 - n11
  n22 = no2 - n12

  MobsF1 = matrix(c(n11,n12,n21,n22),ncol = 2, byrow = T, dimnames = list(c("non-CVD", "CVD"), c("Ridica

  print(MobsF1)
```

```
  cat("Probabilitatea de a obtine tabelul ", i+1, " este ", dhyper(i, no1, no2, n1o), "\n")
  cat("-------------------------------------\n")
}
```

```
## -------------------------------------
## Tabelul  1  :
##         Ridicat Sare Scazut Sare
## non-CVD            0          25
## CVD                7          28
## Probabilitatea de a obtine tabelul  1  este  0.0174117
## -------------------------------------
## -------------------------------------
## Tabelul  2  :
##         Ridicat Sare Scazut Sare
## non-CVD            1          24
## CVD                6          29
## Probabilitatea de a obtine tabelul  2  este  0.1050706
## -------------------------------------
## -------------------------------------
## Tabelul  3  :
##         Ridicat Sare Scazut Sare
## non-CVD            2          23
## CVD                5          30
## Probabilitatea de a obtine tabelul  3  este  0.2521695
## -------------------------------------
## -------------------------------------
## Tabelul  4  :
##         Ridicat Sare Scazut Sare
## non-CVD            3          22
## CVD                4          31
## Probabilitatea de a obtine tabelul  4  este  0.3118225
## -------------------------------------
## -------------------------------------
## Tabelul  5  :
##         Ridicat Sare Scazut Sare
## non-CVD            4          21
## CVD                3          32
## Probabilitatea de a obtine tabelul  5  este  0.214378
## -------------------------------------
## -------------------------------------
## Tabelul  6  :
##         Ridicat Sare Scazut Sare
## non-CVD            5          20
## CVD                2          33
## Probabilitatea de a obtine tabelul  6  este  0.0818534
## -------------------------------------
## -------------------------------------
## Tabelul  7  :
##         Ridicat Sare Scazut Sare
## non-CVD            6          19
## CVD                1          34
## Probabilitatea de a obtine tabelul  7  este  0.01604969
## -------------------------------------
```

```
## ------------------------------------
## Tabelul  8  :
##         Ridicat Sare Scazut Sare
## non-CVD            7          18
## CVD                0          35
## Probabilitatea de a obtine tabelul  8  este  0.00124467
## ------------------------------------
```

Aplicăm testul exact al lui Fisher `fisher.test`:

```
fisher.test(MobsF)
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  MobsF
## p-value = 0.6882
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##  0.04625243 3.58478157
## sample estimates:
## odds ratio
##   0.527113
```

P-valoarea în `R` este calculată după formula:

$$p_{value} = \sum_{\{i:\mathbb{P}(i)\leq\mathbb{P}(obs)\}} \mathbb{P}(i)$$

care în cazul nostru devine

```
n1o = 25
n2o = 35

no1 = 7
no2 = 53

n11 = 2

ps = dhyper(0:no1, no1, no2, n1o)
pobs = dhyper(n11, no1, no2, n1o)

pval = sum(ps[ps<=pobs])
pval
```

```
## [1] 0.6881775
```

1.5 Date pereche - Testul lui McNemar

Ne propunem să comparăm două regimuri de chimioterapie pentru pacienții cu cancer la sân care au efectuat operația de mastectomie. Cele două grupuri de tratament investigate ar trebui să fie cât mai comparabile din punct de vedere al celorlalți factori. Presupunem că un studiu de potrivire (matched study) a fost pregătit așa încât din fiecare pereche (potrivită din punct de vedere al vârstei și a condițiilor clinice) s-a selectat aleator un membru căruia i-a fost administrat

tratamentul A iar celuilalt membru tratamentul B. Pacienții au fost urmăriți pe o perioadă de 5 ani, iar variabila de interes a fost supraviețuirea în această perioadă. S-au obșinut următoarele date:

	Supravietuit	Decedat	Total
A	526	95	621
B	515	106	621
Total	1041	201	1242

Observăm că nu putem folosi testul lui Pearson (cu corecția lui Yates) deoarece datele nu sunt *independente*. Dacă am folosi am obține:

```
M1csq = matrix(c(526,95,515,106),ncol = 2, byrow = T)
chisq.test(M1csq)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  M1csq
## X-squared = 0.59357, df = 1, p-value = 0.441
```

Construim următorul tabel, în care unitatea de analiză nu mai este *pacientul* ci *perechea* iar perechile sunt clasificate după cum membrii acelei perechi au supraviețuit sau nu o perioadă post-operatorie de 5 ani (liniile tabelului sunt rezultatele pacientului care a urmat tratamentul A iar coloanele sunt rezultatele pacientului care a urmat tratamentul B):

	Supravietuit	Decedat	Total
Supravietuit	510	16	526
Decedat	5	90	95
Total	515	106	621

Observăm că 600 (510+90) de perechi au avut același rezultat (perechi concordante) și doar 21 de perechi au avut rezultate diferite (perechi neconcordante).

Aplicăm testul lui McNemar `mcnemar.test` :

```
M1 = matrix(c(510,16,5,90),ncol = 2, byrow = T,
            dimnames = list(c("Supravietuit", "Decedat"), c("Supravietuit", "Decedat")))
mcnemar.test(M1)
```

```
##
##  McNemar's Chi-squared test with continuity correction
##
## data:  M1
## McNemar's chi-squared = 4.7619, df = 1, p-value = 0.0291
```

2 Tabele de contingență $r \times c$

Următorul tabel prezintă repartiția grupelor de sânge (A, B, AB și O) în trei eșantioane de cetățeni afro-americani care trăiesc în trei state diferite (Florida, Iowa și Missouri). Vrem să

testăm la un nivel de semnificație $\alpha = 0.5$ dacă repartiția grupelor de sânge pentru cetățenii afro-americani diferă de-a lungul celor trei state.

	A	B	AB	O	Total
Florida	122	117	19	244	502
Iowa	1781	1351	288	3301	6721
Missouri	353	269	60	713	1395
Total	2256	1737	367	4258	8618

2.1 Testul χ^2 al lui Pearson

Tabelul pe care ne așteptăm să-l observat atunci când ipoteza nulă este adevărată:

```
matAA_observed = rbind(c(122, 117, 19, 244),
          c(1781, 1351, 288, 3301),
          c(353, 269, 60, 713))

rs = rowSums(matAA_observed)
cs = colSums(matAA_observed)

n = sum(matAA_observed)

matAA_expected <- outer(rs,cs,"*")/n
```

	A	B	AB	O	Total
Florida	131.4124	101.1806	21.37781	248.0292	502
Iowa	1759.4078	1354.6504	286.21571	3320.7262	6721
Missouri	365.1799	281.1691	59.40647	689.2446	1395
Total	2256.0000	1737.0000	367.00000	4258.0000	8618

Aplicând funcția `chisq.test` obținem:

```
chisq.test(matAA_observed)
```

```
##
##  Pearson's Chi-squared test
##
## data:  matAA_observed
## X-squared = 5.6382, df = 6, p-value = 0.4649
```

2.2 Testul bazat pe raportul de verosimilitate

Aplicând funcția `LRT1` construită anterior obținem p-valoarea testului bazat pe raportul de verosimilitate:

```
LRT1(matAA_observed)
```

```
## Statistica LRT este  5.548169
## P-valoarea testului bazat pe raportul de verosimilitate este  0.475654
```

```
## $statistic
## [1] 5.548169
##
## $pvalue
## [1] 0.475654
```

2.3 Testul aproximat al lui Fisher

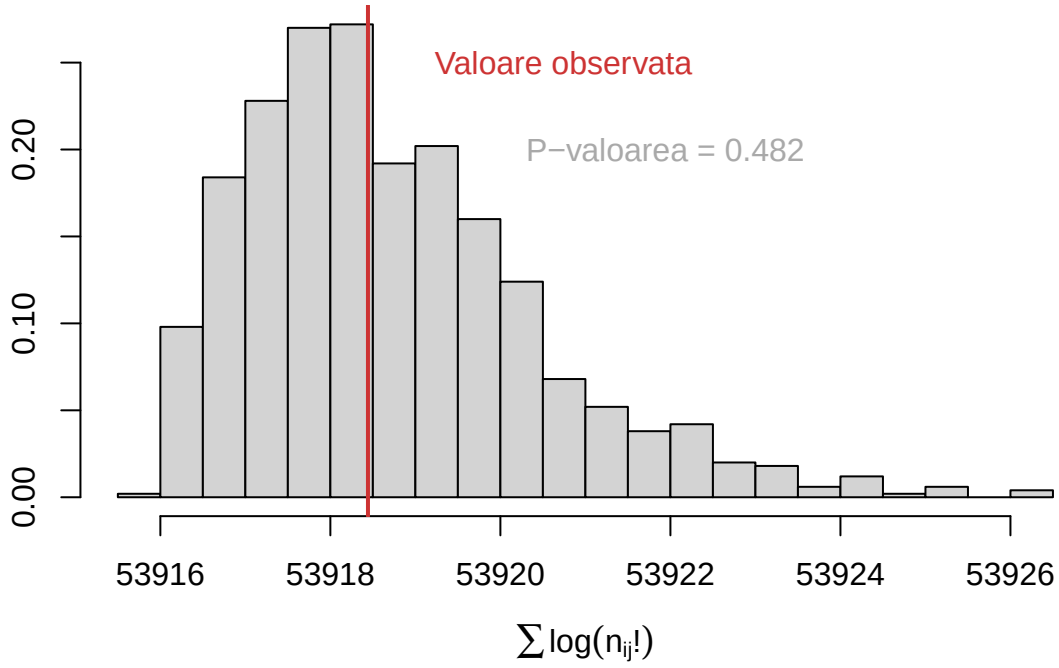
Testul exact al lui Fisher poate fi aplicat și în cazul tabelelor de tip $r \times c$ (pentru o generalizare a testului prezentat la curs puteți consulta http://mathworld.wolfram.com/FishersExactTest.html) numai că numărul de tabele pe care trebuie să le generăm devine prohibitiv. În acest caz putem aproxima p-valoarea testului cu ajutorul metodelor de tip Monte-Carlo.

Generalizând raționamentul din cazul 2×2 obținem că probabilitatea (condiționată) de a observa un tabel dat fiind marginalele (pe rânduri și pe coloane) este dată de:

$$\mathbb{P}(\, tabel\,) = \frac{\prod_{i=1}^{r} n_{i\cdot}! \prod_{j=1}^{c} n_{\cdot j}!}{n! \prod_{i=1}^{r} \prod_{j=1}^{c} n_{ij}!} \propto \frac{1}{\prod_{j=1}^{c} n_{ij}!}$$

```
fisher <- function(tab, n.sim=1000, return.all=FALSE, prnt=FALSE){
  bot0 <- sum(lgamma(tab+1))# lgamma - logaritm natural din gamma - logaritm din factorial

  bot <- 1:n.sim
  a <- list(rep(row(tab),tab), rep(col(tab),tab))
  for(i in 1:n.sim) {
    a[[1]] <- sample(a[[1]])
    bot[i] <- sum(lgamma(table(a)+1))
    if(prnt) { if(i == round(i/10)*10) cat(i,"\n") }
  }
  if(return.all) return(list(bot0, bot, mean(bot0 <= bot)))
  cat("P-valoarea aproximata cu Monte Carlo este ", mean(bot0 <= bot))
}

set.seed(5)
fisher(matAA_observed)
```

```
## P-valoarea aproximata cu Monte Carlo este  0.482
```



Același rezultat îl obținem și dacă folosim funcția `fisher.test` (care este mai rapidă):

```
fisher.test(matAA_observed, simulate.p.value = TRUE, B = 1000)
```

```
##
##  Fisher's Exact Test for Count Data with simulated p-value (based
##  on 1000 replicates)
##
## data:  matAA_observed
## p-value = 0.4975
## alternative hypothesis: two.sided
```

Curs Biostatistică 2017 - Laborator 5 & 6

Analiză de varianță - ANOVA

Contents

1 Analiză de varianță cu un factor (one-way ANOVA)

1.1 Exemplul 1

Vom analiza setul de date `Cushings` din pachetul `MASS`. Sindromul *Cushing* reprezintă o serie de semne și simptome ca urmare a expunerii organismului pentru o perioadă îndelungată de timp la o concentrație ridicată de cortizon (mai multe detalii aici și aici). Pentru fiecare individ din eșantion, ratele de excreție urinară a doi metaboliți steroizi sunt înregistrate: *Tetrahydrocortisone* și *Pregnanetriol*. Variabila *Type* arată tipul de sindrom Cushing, acesta putând lua una din următoarele patru categorii: *adenom* (a), *hiperplazia bilaterală* (b), *carcinom* (c) și *necunoscut* (u). Obiectivul este să investigăm dacă cele patru tipuri de sindrom sunt diferite în raport cu excreția urinară de *Tetrahydrocortisone*.

Începem prin a atașa setul de date `Cushings`:

```
library(MASS)
data("Cushings")
attach(Cushings)
```

Tetrahydrocortisone	Pregnanetriol	Type
3.1	11.70	a
3.0	1.30	a
1.9	0.10	a
3.8	0.04	a
4.1	1.10	a
1.9	0.40	a
8.3	1.00	b
3.8	0.20	b
3.9	0.60	b
7.8	1.20	b
9.1	0.60	b
15.4	3.60	b

Tetrahydrocortisone	Pregnanetriol	Type
7.7	1.60	b
6.5	0.40	b
5.7	0.40	b
13.6	1.60	b
10.2	6.40	c
9.2	7.90	c
9.6	3.10	c
53.8	2.50	c
15.8	7.60	c
5.1	0.40	u
12.9	5.00	u
13.0	0.80	u
2.6	0.10	u
30.0	0.10	u
20.5	0.80	u

Notăm cu Y excreția urinară de *Tetrahydrocortisone* (variabila răspuns) și cu X variabila *Type* (variabila factor), cu $X \in \{1, 2, 3, 4\}$ după cum $Type \in \{a, b, c, u\}$. Astfel obiectivul este de a investiga dacă media variabilei răspuns Y diferă pentru valori diferite ale nivelelor variabilei factor X. Dacă notăm observațiile individuale cu y_{ij} (excreția urinară de *Tetrahydrocortisone* a individului j cu tipul de sindrom i) atunci putem determina

- numărul de observații din fiecare grup (n_i)

```
n = length(Cushings$Tetrahydrocortisone)

# varianta 1 - nr de observatii pe grup
ng = table(Cushings$Type)
ng
```

```
##
##  a  b  c  u
##  6 10  5  6
```

```
# varianta 2 - nr de observatii pe grup
ng2 = tapply(Cushings$Tetrahydrocortisone, Cushings$Type, length)
ng2
```

```
##  a  b  c  u
##  6 10  5  6
```

- media fiecărui grup ($\bar{y}_i$)

```
# media globala
my = mean(Cushings$Tetrahydrocortisone)

# varianta 1 - media pe grup
myg = tapply(Cushings$Tetrahydrocortisone, Cushings$Type, mean)
myg
```

```
##         a         b         c         u
##  2.966667  8.180000 19.720000 14.016667
```

```
# varianta 2 - media pe grup
myg2 = aggregate(Cushings$Tetrahydrocortisone, by = list(Cushings$Type), mean)
```

```
myg2
```

```
##   Group.1         x
## 1       a  2.966667
## 2       b  8.180000
## 3       c 19.720000
## 4       u 14.016667
```

- deviația standard a fiecărui grup

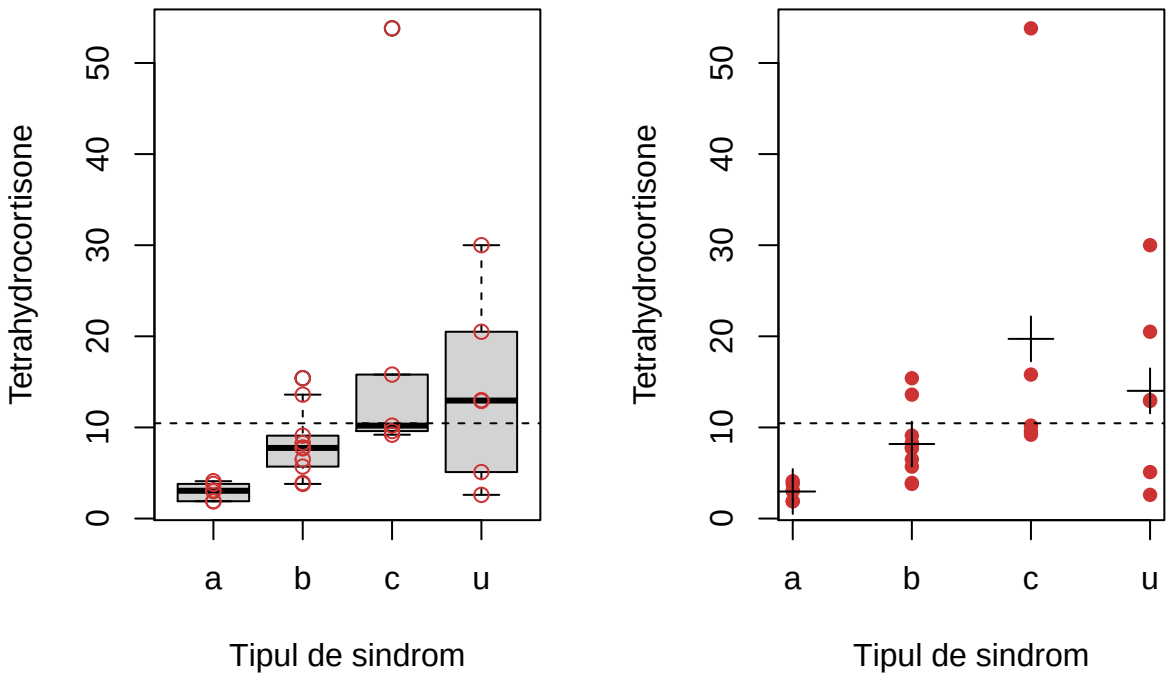
```
# varianta 1 - media pe grup
syg = tapply(Cushings$Tetrahydrocortisone, Cushings$Type, sd)
syg
```

```
##          a          b          c          u
##  0.9244818  3.7891072 19.2388149 10.0958242
```

```
# varianta 2 - media pe grup
syg2 = aggregate(Cushings$Tetrahydrocortisone, by = list(Cushings$Type), sd)
syg2
```

```
##   Group.1          x
## 1       a  0.9244818
## 2       b  3.7891072
## 3       c 19.2388149
## 4       u 10.0958242
```

Considerăm următorul grafic unde fiecare observație este reprezentată printr-un punct (gol în figura din stânga și plin în cea din dreapta) iar media globală este ilustrată printr-o linie punctată. În figura din stânga avem *boxplot*-ul pentru fiecare categorie a lui X iar în figura din dreapta (*stripchart*) mediile eșantioanelor din fiecare grup sunt ilustrate cu o cruce de culoare neagră:



Din figura de mai sus putem observa că avem o variație considerabilă între mediile grupurilor de-a lungul celor 4 categorii de sindrom *Cushing*. De asemenea, în interiorul grupurilor, avem grade diferite de variație a observațiilor (vezi figura din stânga). Ambele surse de variabilitate contribuie la variabilitatea totală a observațiilor în jurul mediei globale (linia punctată).

Calculăm **variabilitatea dintre grupuri** (r este numărul de grupuri):

$$SS_B = \sum_{i=1}^{r} n_i(\bar{y}_i - \bar{y})^2$$

```
# avem ng nr de observatii din fiecare grup, myg media lui y din fiecare grup si my media totala

SS_B = ng%*%(myg-my)^2 # unde %*% este produs de matrice
SS_B
```

```
##          [,1]
## [1,] 893.521
```

Calculăm **variabilitatea reziduală** (din grupuri):

$$SS_W = \sum_{i=1}^{r} \sum_{j=1}^{n_i} (y_{ij} - \bar{y}_i)^2$$

```
y = Cushings$Tetrahydrocortisone # y_{ij}
ryi = rep(myg, ng)
```

```r
SS_W = sum((y-ryi)^2)
SS_W
```

```
## [1] 2123.646
```

Calculăm **variabilitatea totală**:

$$SS_T = \sum_{i=1}^{r}\sum_{j=1}^{n_i}(y_{ij} - \bar{y})^2 = SS_B + SS_W$$

```r
# calculat cu SS_B+SS_W
SS_T = SS_B + SS_W
SS_T
```

```
##          [,1]
## [1,] 3017.167
```

```r
# calculat cu sume (verificam formula)
SS_T2 = sum((y-my)^2)
SS_T2
```

```
## [1] 3017.167
```

Observăm că *variabilitatea totală poate fi atribuită parțial variabilității dintre grupuri și parțial variabilității din interiorul grupurilor.*

Considerăm ipoteza nulă:

$$H_0 : \mu_1 = \cdots = \mu_i = \mu$$

unde μ este media populației Y iar $\mu_1, \ldots, \mu_i$ sunt mediile populațiilor din fiecare grup.

Statistica de test este:

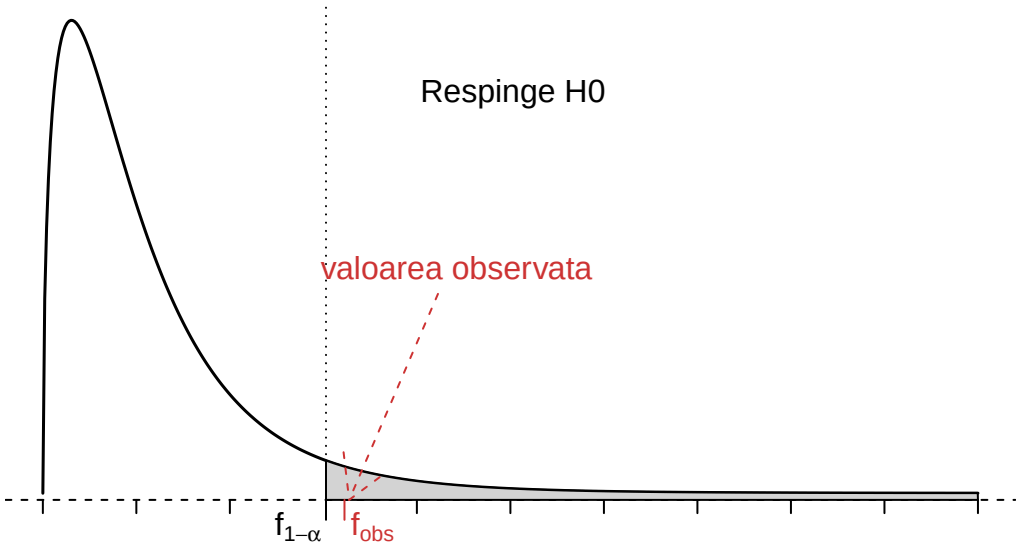
$$F = \frac{\frac{SS_B}{r-1}}{\frac{SS_W}{n-r}}$$

unde $\frac{SS_B}{r-1}$ și $\frac{SS_W}{n-r}$ sunt mediile pătrate pentru grupuri (mean square) și respectiv reziduri. Dacă condițiile *ANOVA* (datele din fiecare grup sunt i.i.d. și sunt normal distribuite) sunt satisfăcute și presupunând că H_0 este adevărată avem că $F \sim F(r-1, n-r)$.

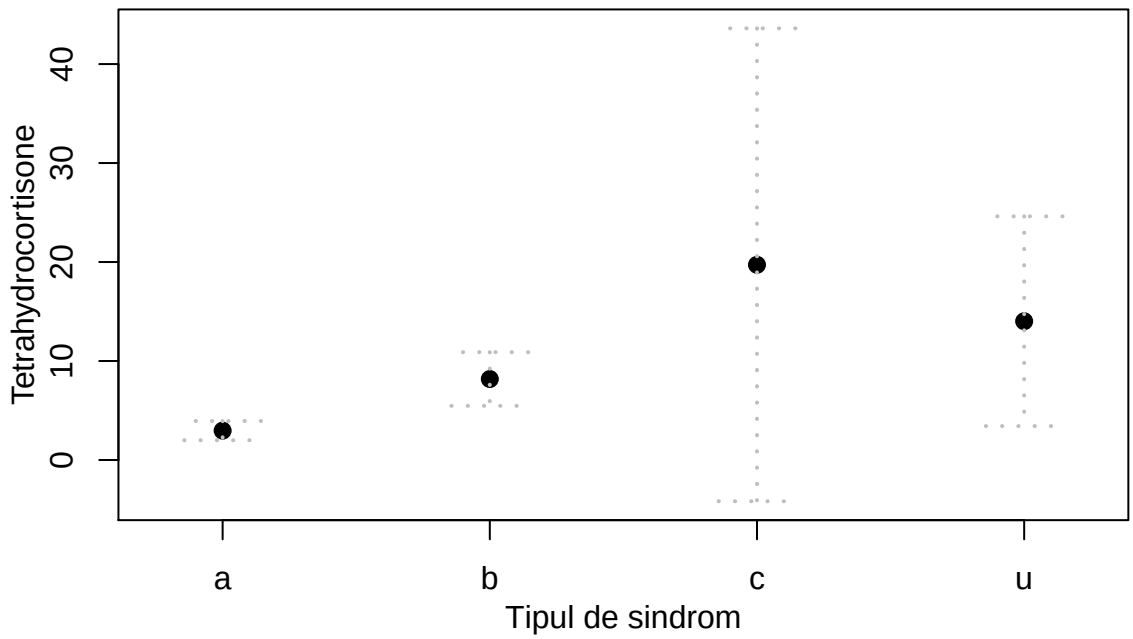
Avem modelul *ANOVA*:

```r
anova_model = aov(Tetrahydrocortisone~Type, data = Cushings)

summary(anova_model)
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## Type         3  893.5  297.84   3.226 0.0412 *
## Residuals   23 2123.6   92.33
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```



Verificarea ipotezelor ANOVA



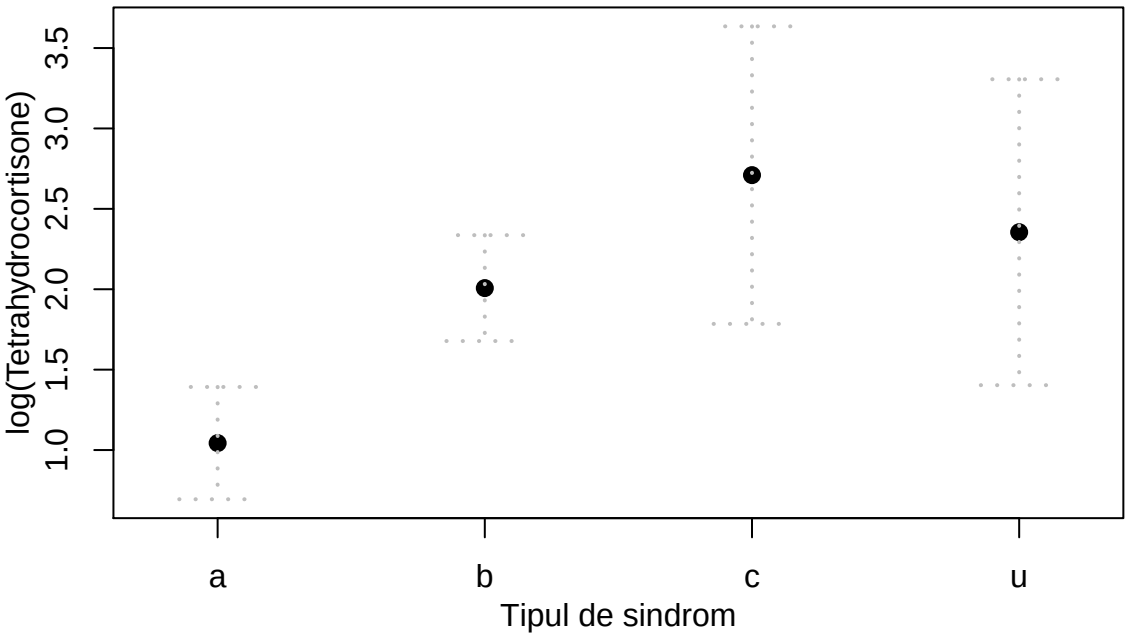
Aplicăm *testul lui Bartlett* pentru a testa homoscedasticitatea modelului (i.e. verificăm $H_0 : \sigma_1 = \cdots = \sigma_r$):

```
bartlett.test(Tetrahydrocortisone~Type, data = Cushings)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  Tetrahydrocortisone by Type
## Bartlett's K-squared = 31.595, df = 3, p-value = 6.37e-07
```

Observăm că ipoteza de omogenitate este respinsă în favoarea alternativei prin urmare ipoteza de omogenitate din ANOVA este invalidată.

Transformăm variabila răspuns ($\log(Y) = \log(Tetrahydrocortisone)$):



Verificăm ipoteza de omogenitate (homoscedasticitatea):

```
bartlett.test(log(Tetrahydrocortisone)~Type, data = Cushings)
```

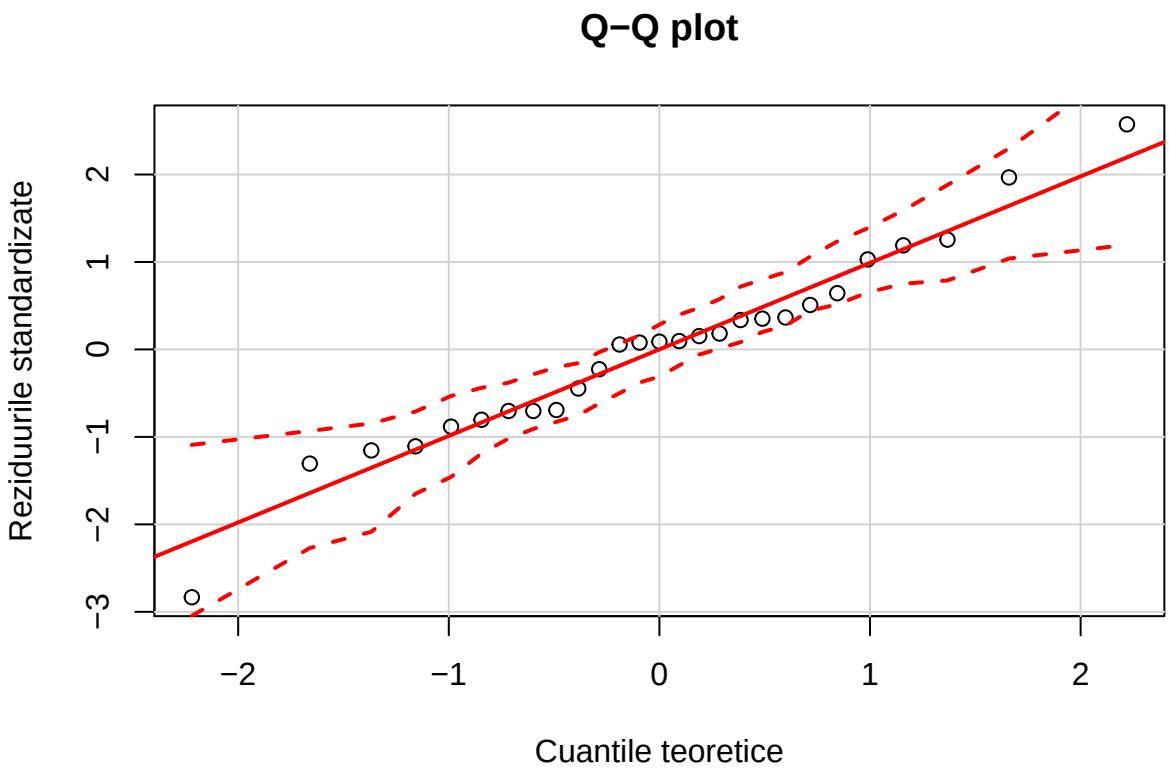
```
##
##  Bartlett test of homogeneity of variances
##
## data:  log(Tetrahydrocortisone) by Type
## Bartlett's K-squared = 5.7249, df = 3, p-value = 0.1258
```

Testăm normalitatea modelului transformat (*testul lui Shapiro-Wilks* sau *Shapiro-Francia*):

```
anova_model_tr = aov(log(Tetrahydrocortisone)~Type, data = Cushings)
shapiro.test(residuals(anova_model_tr))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(anova_model_tr)
## W = 0.97953, p-value = 0.8515
```

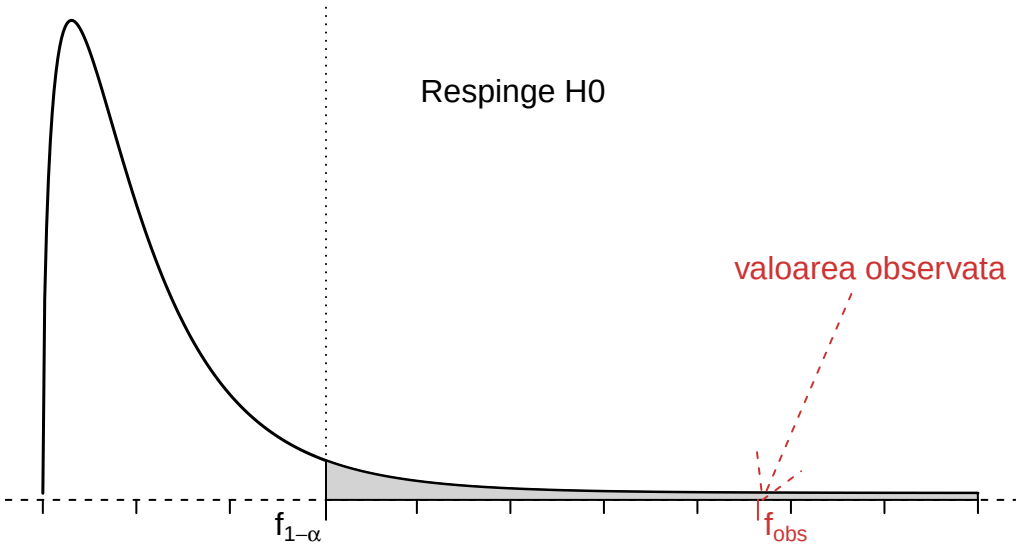
Verificăm normalitatea și grafic cu `Q-Q Plot`:



ANOVA pentru modelul transformat:

```
summary(anova_model_tr)
```

```
##             Df Sum Sq Mean Sq F value  Pr(>F)
## Type         3  8.766  2.9220   7.647 0.00102 **
## Residuals   23  8.789  0.3821
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```



1.2 Exemplul 2

În acest exemplu vom folosi setul de date *Cholesterol* din pachetul `multcomp` (datele se pot descărca de aici). Datele prezintă cu cât s-a redus nivelul de colesterol (variabila *response*) la 50 de pacienți ce au urmat 5 tratamente de reducere a colesterolului. Trei dintre tratamente au implicat același medicament administrat în moduri diferite: 20 mg o dată pe zi (*1time*), 10 mg de două ori pe zi (*2time*) sau 5 mg de patru ori pe zi (*4time*). Celelalte două tratamente au constat din medicamente alternative diferite (*drugD* și *drugE*). Care tratament a produs cea mai mare reducere a colesterolului ?

Începem prin a citi setul de date:

```
cholesterol = read.csv("data/cholesterol.csv", stringsAsFactors = FALSE)
head(cholesterol)
```

```
##     trt response
## 1 1time   3.8612
## 2 1time  10.3868
## 3 1time   5.9059
## 4 1time   3.0609
## 5 1time   7.7204
## 6 1time   2.7139
```

Vedem câte observații avem pentru fiecare tratament:

```r
table(cholesterol$trt)
```

```
##
##  1time 2times 4times  drugD  drugE
##     10     10     10     10     10
```

Observăm că fiecare tratament a fost administrat la câte 10 pacienți (suntem în contextul unui *plan de experiență echilibrat*).

Calculăm:

- numărul total de observații (n) și numărul de observații din fiecare grup (n_i)

```r
n = length(cholesterol$trt) # nr total de observații

# nr de observatii pe grup
ng = table(cholesterol$trt)
```

- media fiecărui grup ($\bar{y}_i$)

```r
# media globala
my = mean(cholesterol$response)

# media pe grup
myg = tapply(cholesterol$response, cholesterol$trt, mean)
myg
```

```
##    1time   2times   4times    drugD    drugE
##  5.78197  9.22497 12.37478 15.36117 20.94752
```

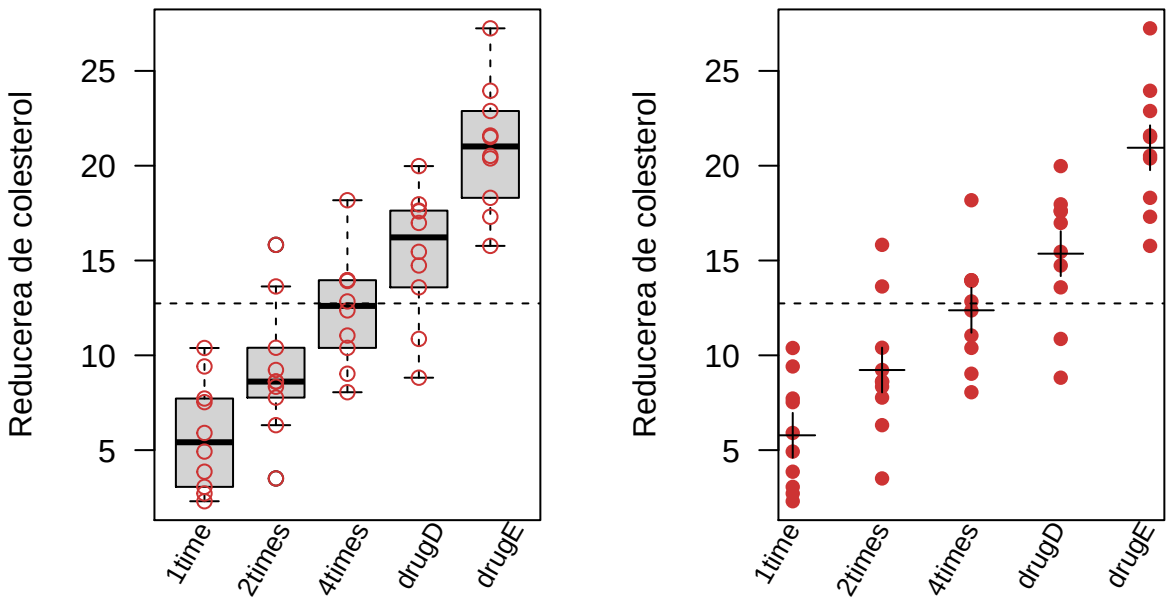
Se observă că drugE a produs (în medie) cea mai mare reducere a colesterolului pe când 1time a produs-o pe cea mai mică.

- abaterea standard a fiecărui grup

```r
# sd pe grup
syg = tapply(cholesterol$response, cholesterol$trt, sd)
syg
```

```
##    1time   2times   4times    drugD    drugE
## 2.878113 3.483054 2.923119 3.454636 3.345003
```

Se observă că abaterile standard sunt relativ constante pentru cele 5 tratamente, luând valori între 2.9 și 3.5.



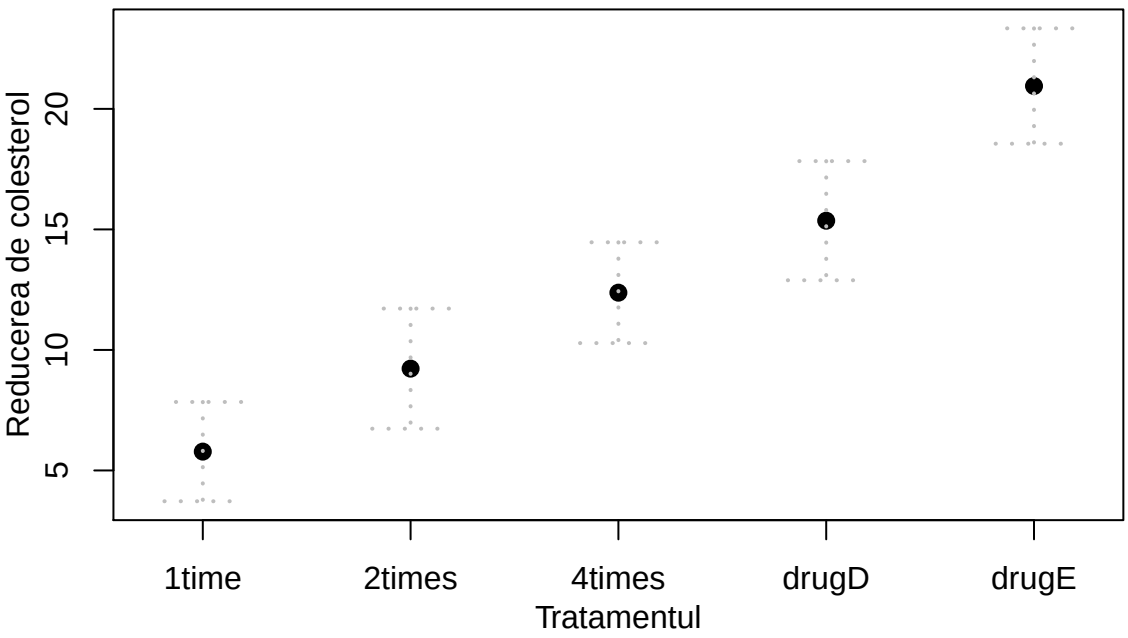
Avem tabelul *ANOVA*:

```
anova_model = aov(response~trt, data = cholesterol)

summary(anova_model)
```

```
##             Df Sum Sq Mean Sq F value   Pr(>F)
## trt          4 1351.4   337.8   32.43 9.82e-13 ***
## Residuals   45  468.8    10.4
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

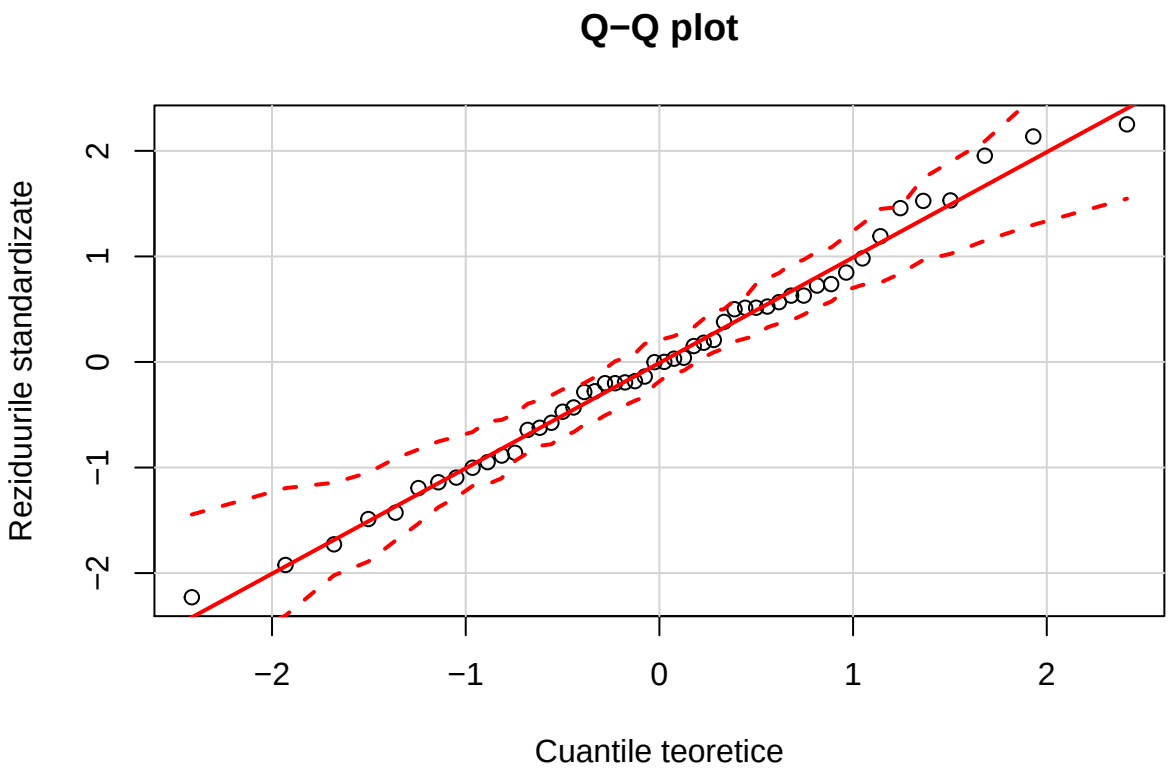
Testul ANOVA (F) pentru tratament (trt) este semnificativ ($p < 0.001$), ilustrând că cele 5 tratamente nu sunt la fel de eficiente.

Reducerea medie de colesterol pentru cele 5 tratamente împreună cu intervalele de încredere de nivel de încredere de 95% corespunzătoare:



Verificarea ipotezelor ANOVA

În ANOVA cu un factor, se presupune că variabila răspuns este repartizată normal cu aceeași variantă în fiecare grup. Pentru testarea normalității putem folosi ca metodă grafică `Q-Q plot`-ul:



De asemenea ipoteza de normalitate poate fi testată și cu testul *Shapiro-Wilks* sau *Shapiro-Francia*:

```
anova_model_chol = aov(response~trt, data = cholesterol)
shapiro.test(residuals(anova_model_chol))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(anova_model_chol)
## W = 0.98864, p-value = 0.9094
```

Pentru testarea ipotezei de homoscedasticitate aplicăm *testul lui Bartlett*(i.e. verificăm $H_0 : \sigma_1 = \cdots = \sigma_r$):

```
bartlett.test(response~trt, data = cholesterol)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  response by trt
## Bartlett's K-squared = 0.57975, df = 4, p-value = 0.9653
```

Testul lui Bartlett ne indică faptul că variațele în cele 5 grupuri nu diferă semnificativ ($p = 0.97$). Pentru testarea ipotezei de omogenitate se mai pot folosi și alte teste printre care includem *testul lui Fligner-Killeen* (`fligner.test`) și *testul Brown-Forsythe* (funcția `hov()` din pachetul `HH`). Ambele teste întorc același rezultat:

```
fligner.test(response~as.factor(trt), data = cholesterol)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
```

```
##
## data:  response by as.factor(trt)
## Fligner-Killeen:med chi-squared = 0.74277, df = 4, p-value = 0.946
```

```
hov(response~trt, data = cholesterol) # hov = homogeneity of variance
```

```
##
##  hov: Brown-Forsyth
##
## data:  response
## F = 0.075477, df:trt = 4, df:Residuals = 45, p-value = 0.9893
## alternative hypothesis: variances are not identical
```

Comparări multiple

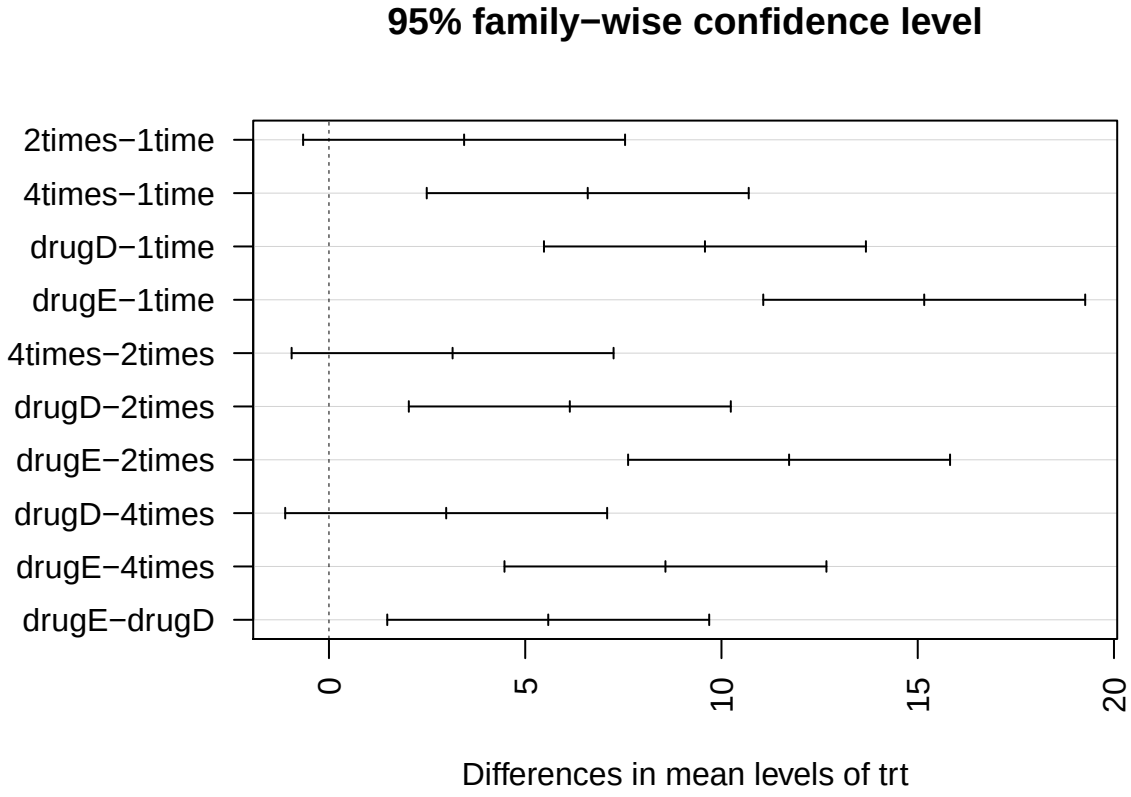
Testul F din ANOVA pentru tratamente ne spune că cele 5 tipuri de medicamente nu sunt la fel de eficiente, însă nu ne spune care dintre ele diferă față de celelalte. Pentru a răspunde la această întrebare vom folosi metodologia testării multiple. Ca exemplu vom folosi *Testul lui Tukey HSD* (Honestly Significant Difference), test care permite compararea tuturor perechilor de diferențe dintre mediile grupurilor:

```
TukeyHSD(anova_model_chol)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = response ~ trt, data = cholesterol)
##
## $trt
##                   diff        lwr       upr     p adj
## 2times-1time   3.44300 -0.6582817  7.544282 0.1380949
## 4times-1time   6.59281  2.4915283 10.694092 0.0003542
## drugD-1time    9.57920  5.4779183 13.680482 0.0000003
## drugE-1time   15.16555 11.0642683 19.266832 0.0000000
## 4times-2times  3.14981 -0.9514717  7.251092 0.2050382
## drugD-2times   6.13620  2.0349183 10.237482 0.0009611
## drugE-2times  11.72255  7.6212683 15.823832 0.0000000
## drugD-4times   2.98639 -1.1148917  7.087672 0.2512446
## drugE-4times   8.57274  4.4714583 12.674022 0.0000037
## drugE-drugD    5.58635  1.4850683  9.687632 0.0030633
```

Observăm că reducerea medie a colesterolului pentru tratamentele *1time* și *2times* nu este semnificativă ($p = 0.138$) pe când reducerea medie a colesterolului pentru tratamentele *1time* și *4times* este semnificativă ($p < 0.001$).

Aceste diferențe se pot observa și grafic:



Trebuie menționat că sunt mai multe metode pentru comparări multiple: *metoda Bonferroni*, *metoda contrastelor liniare*, *metoda bazată pe statistici de rang*, *metoda Newman Keuls* etc.

2 Analiză de varianță cu doi factori (two-way ANOVA)

Analiza de varianță cu doi factori poate fi văzută ca o generalizare a analizei de varianță cu un factor, în acest model subiecții fiind distribuiți în grupe rezultate din încrucișarea modalităților celor doi factori.

Ca și în cazul one-way ANOVA, condițiile de aplicare rămân aceleași: populații normale de aceeași varianță și eșantioane independente.

2.1 Exemplul 1

În acest exemplu vom folosi setul de date `ToothGrowth` din pachetul de bază. Datele fac referire la 60 de porcușori de guinea care sunt repartizați aleator să primească unul din cele trei nivele de Vitamina C (0.5, 1 și 2 mg) prin una din cele două modalități propuse (suc de portocale - OJ sau o soluție apoasă de acid ascorbic - VC), cu restricția ca fiecare combinație de tratament să fie atribuită la 10 porcușori. Vrem să investigăm efectul Vitaminei C asupra creșterii dinților porcușorilor de guinea prin cele două metode de livrare.

Atașăm setul de date `ToothGrowth`:

```
data("ToothGrowth")
attach(ToothGrowth)

head(ToothGrowth)
```

```
##    len supp dose
## 1  4.2   VC  0.5
## 2 11.5   VC  0.5
## 3  7.3   VC  0.5
## 4  5.8   VC  0.5
## 5  6.4   VC  0.5
## 6 10.0   VC  0.5
```

Structura setului de date este:

```
str(ToothGrowth)
```

```
## 'data.frame':    60 obs. of  3 variables:
##  $ len : num  4.2 11.5 7.3 5.8 6.4 10 11.2 11.2 5.2 7 ...
##  $ supp: Factor w/ 2 levels "OJ","VC": 2 2 2 2 2 2 2 2 2 2 ...
##  $ dose: num  0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 ...
```

unde `len` este variabila dependentă (variabila răspuns) iar `supp` și `dose` sunt variabilele explicative (cei doi factori).

Descompunerea erorii în modelul ANOVA cu doi factori este:

$$\underbrace{\sum_{i=1}^{r}\sum_{j=1}^{c}\sum_{k=1}^{s}(Y_{ijk}-\bar{Y}_{...})^2}_{SS_T} = \underbrace{sc\sum_{i=1}^{r}(\bar{Y}_{i..}-\bar{Y}_{...})^2}_{SS_A} + \underbrace{sr\sum_{j=1}^{c}(\bar{Y}_{.j.}-\bar{Y}_{...})^2}_{SS_B}$$
$$+ \underbrace{s\sum_{i=1}^{r}\sum_{j=1}^{c}(\bar{Y}_{ij.}-\bar{Y}_{i..}-\bar{Y}_{.j.}+\bar{Y}_{...})^2}_{SS_{A\times B}} + \underbrace{\sum_{i=1}^{r}\sum_{j=1}^{c}\sum_{k=1}^{s}(Y_{ijk}-\bar{Y}_{ij.})^2}_{SS_W}$$

Tabelul ANOVA devine:

Sursa	DF	SS	MS	F_test
A	$r-1$	SS_A	MS_A	$\frac{MS_A}{MS_W}$
B	$c-1$	SS_B	MS_B	$\frac{MS_B}{MS_W}$
$A\times B$	$(r-1)(c-1)$	$SS_{A\times B}$	$MS_{A\times B}$	$\frac{MS_{A\times B}}{MS_W}$
W	$rc(s-1)$	SS_W	MS_W	
$Total$	$rcs-1$	SS_T		

Pentru a calcula numărul de observații din fiecare încrucișare de categorii vom folosi funcția `table`:

```
# nr de categorii pentru fiecare factor
r = 2
c = 3
s = 10

# nr de observatii pentru factorul A
n_i = table(supp)
```

```
n_i
```

```
## supp
## OJ VC
## 30 30
```

```
# nr de observatii pentru factorul B
n_j = table(dose)
n_j
```

```
## dose
## 0.5   1   2
##  20  20  20
```

```
# nr de observatii pentru fiecare incrucisare a factorilor A si B
n_ij = table(supp, dose)
n_ij
```

```
##     dose
## supp 0.5  1  2
##   OJ  10 10 10
##   VC  10 10 10
```

de unde observăm că suntem în contextul unui plan de experiență echilibrat ($r = 2$, $c = 3$ și $s = 10$).

Vom calcula mediile $\bar{Y}_{ij\cdot}$, $\bar{Y}_{i\cdot\cdot}$, $\bar{Y}_{\cdot j\cdot}$ și $\bar{Y}_{\cdot\cdot\cdot}$:

- pentru $\bar{Y}_{\cdot\cdot\cdot}$

```
m_T = mean(len)
m_T
```

```
## [1] 18.81333
```

- pentru $\bar{Y}_{i\cdot\cdot}$

```
m_i = tapply(len, supp, mean)
m_i
```

```
##       OJ       VC
## 20.66333 16.96333
```

- pentru $\bar{Y}_{\cdot j\cdot}$

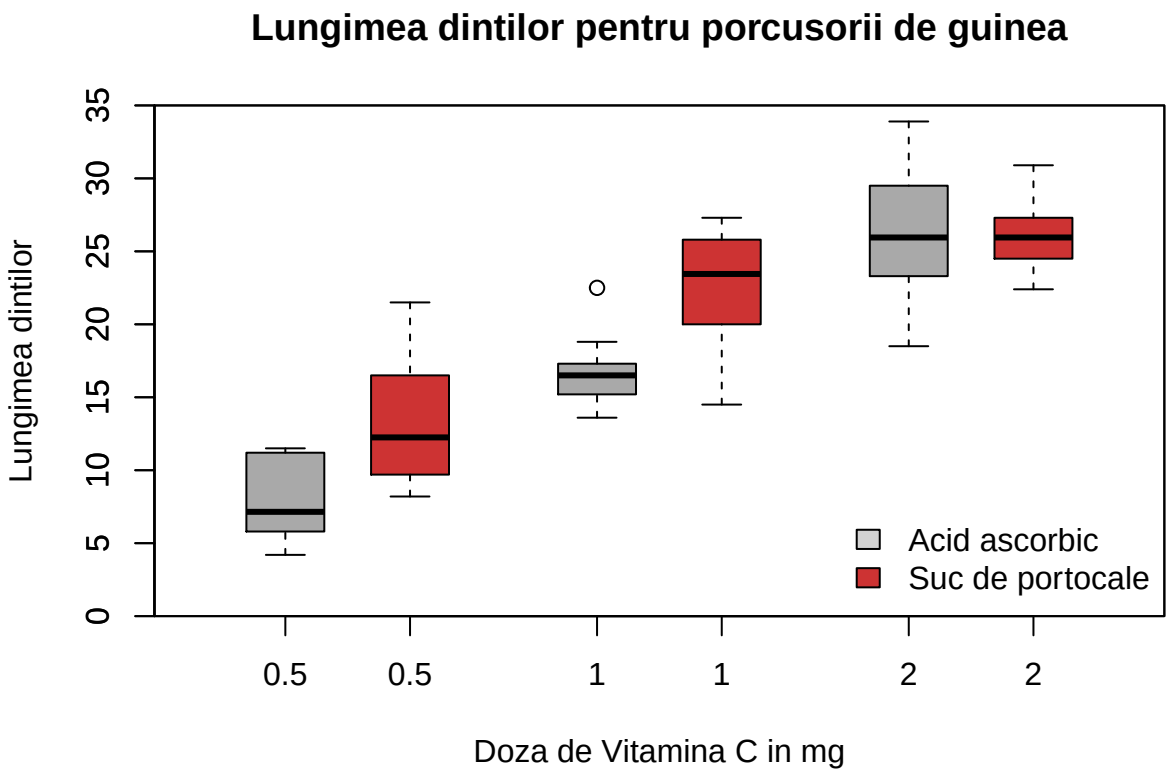
```
m_j = tapply(len, dose, mean)
m_j
```

```
##    0.5      1      2
## 10.605 19.735 26.100
```

- pentru $\bar{Y}_{ij\cdot}$

```
m_ij = tapply(len, list(supp, dose), mean)
m_ij
```

```
##      0.5     1     2
## OJ 13.23 22.70 26.06
## VC  7.98 16.77 26.14
```



Calculăm SS_W, df_W și MS_W:

```
y = len

rep_ij = c(rep(m_ij["VC",], n_ij["VC",]), rep(m_ij["OJ",], n_ij["OJ",]))

SS_W = sum((y-rep_ij)^2)
cat("SS_W = ", SS_W,"\n")
```

```
## SS_W =  712.106
```

```
df_W = r*c*(s-1)
cat("df_W = ", df_W,"\n")
```

```
## df_W =  54
```

```
MS_W = SS_W/df_W
cat("MS_W = ", MS_W,"\n")
```

```
## MS_W =  13.18715
```

Calculăm SS_A, df_A și MS_A:

```
SS_A = s*c*sum((m_i-m_T)^2)
cat("SS_A = ", SS_A,"\n")
```

```
## SS_A =  205.35
```

```
df_A = r-1
cat("df_A = ", df_A,"\n")
```

```
## df_A =  1
```

```
MS_A = SS_A/df_A
cat("MS_A = ", MS_A,"\n")
```

```
## MS_A =  205.35
```

Calculăm SS_B, df_B și MS_B:

```
SS_B = s*r*sum((m_j-m_T)^2)
cat("SS_B = ", SS_B,"\n")
```

```
## SS_B =  2426.434
```

```
df_B = c-1
cat("df_B = ", df_B,"\n")
```

```
## df_B =  2
```

```
MS_B = SS_B/df_B
cat("MS_B = ", SS_B,"\n")
```

```
## MS_B =  2426.434
```

Calculăm $SS_{A\times B}$, $df_{A\times B}$ și $MS_{A\times B}$:

```
v = m_ij - matrix(rep(m_i, c), ncol = c) - matrix(rep(m_j, r), nrow = r, byrow = T) + m_T
SS_AB = s*sum(v^2)
cat("SS_AB = ", SS_AB,"\n")
```

```
## SS_AB =  108.319
```

```
df_AB = (r-1)*(c-1)
cat("df_AB = ", df_AB,"\n")
```

```
## df_AB =  2
```

```
MS_AB = SS_AB/df_AB
cat("MS_AB = ", SS_AB,"\n")
```

```
## MS_AB =  108.319
```

Calculăm SS_T:

```
SS_T = SS_A + SS_B + SS_AB + SS_W
cat("SS_T = ", SS_T,"\n")
```

```
## SS_T =  3452.209
```

```
# verificam prin formula
sum((y-m_T)^2)
```

```
## [1] 3452.209
```

Calculăm tabelul ANOVA cu funcția `aov`:

```
ToothGrowth$dose = as.factor(ToothGrowth$dose)
ToothGrowth$supp = as.factor(ToothGrowth$supp)

model_anova_2w = aov(len~supp*dose, data = ToothGrowth)

summary(model_anova_2w)
```

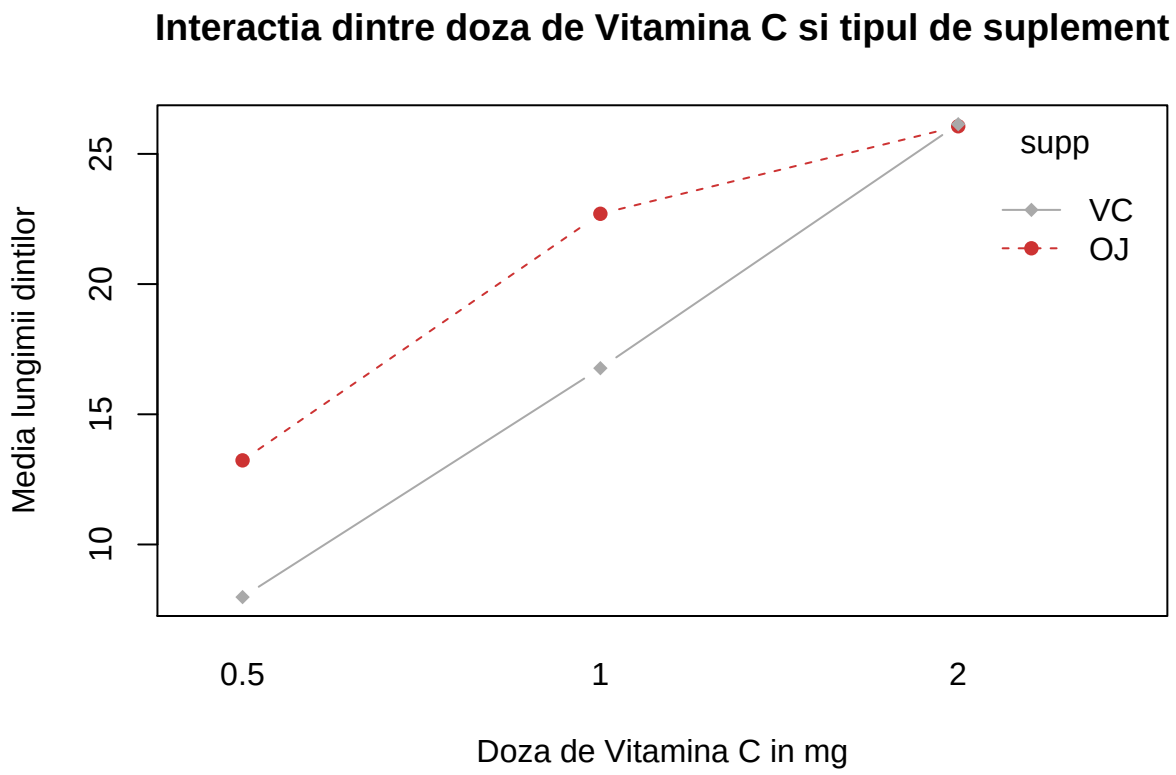
```
##             Df Sum Sq Mean Sq F value   Pr(>F)
```

```
## supp         1  205.4   205.4  15.572 0.000231 ***
## dose         2 2426.4  1213.2  92.000  < 2e-16 ***
## supp:dose    2  108.3    54.2   4.107 0.021860 *
## Residuals   54  712.1    13.2
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

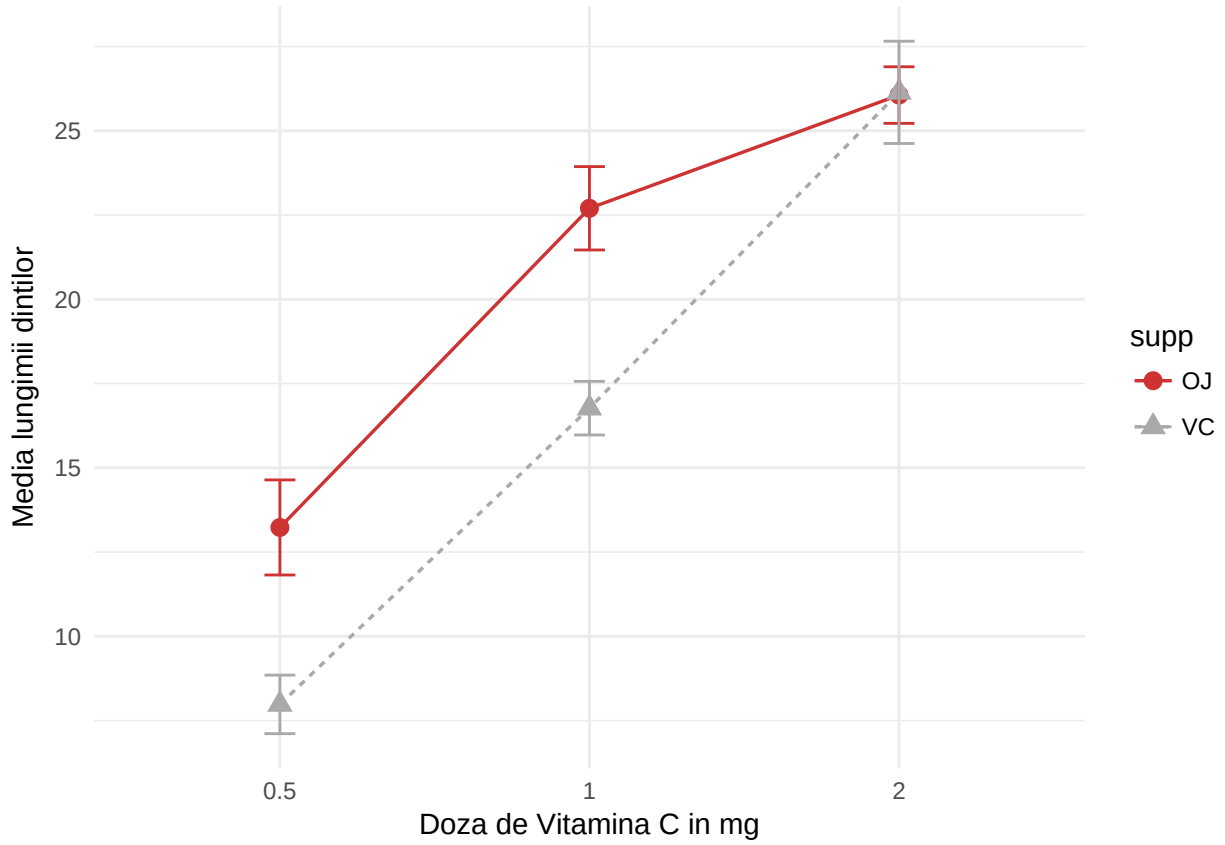
Tabelul ANOVA de mai sus ne arată că atât efectele principale (`supp` și `dose`) cât și interacția dintre cei doi factori (`supp:dose`) sunt semnificative.

Pentru a vedea interacția dintre cei doi factori putem să folosim funcția `interaction.plot`:

```
interaction.plot(dose, supp, len, type="b",
        col=c("brown3","darkgray"), pch=c(16, 18),
        main = "Interactia dintre doza de Vitamina C si tipul de suplement",
        xlab = "Doza de Vitamina C in mg",
        ylab = "Media lungimii dintilor")
```



Dacă vrem să includem și intervalele de încredere atunci avem:



Graficele ne arată că dinții de la porcușorii de guinea cresc cu doza de Vitamina C atât pentru sucul de portocale cât și pentru soluția de acid ascorbic. Pentru dozele de 0.5 și 1 mg, sucul de portocale produce în medie o creștere mai mare a dinților decât soluția de acid ascorbic. Pentru doza de 2 mg, ambele metode produc aceeași creștere.

2.1.1 Verificarea ipotezelor ANOVA

Vom începe prin a testa **condiția de normalitate** a observațiilor. Pentru aceasta vom folosi testul `Shapiro-Wilks` (funcția `shapiro.test`) și metoda grafică a *dreptei lui Henry* (Q-Q plot). Ca și în cazul ANOVA cu un factor, vom testa normalitatea datelor pentru toate datele și nu pentru fiecare eșantion în parte. În acest sens va trebui să calculăm reziduurile:

$$\hat{e}_{ijk} = y_{ijk} - \bar{y}_{ij\cdot}$$

lucru care poate fi realizat sau prin calcul direct:

```
res_model_direct = y-rep_ij
```

sau folosind funcția residuals:

```
res_model = residuals(model_anova_2w)
```

Aplicăm testul Shapiro-Wilks pentru reziduuri și obținem:

```
shapiro.test(res_model)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  res_model
## W = 0.98499, p-value = 0.6694
```

de unde concluzionăm că ipoteza de normalitate este satisfăcută. Aceeași concluzie o obținem și prin metoda grafică:

```
qqPlot(lm(len~supp*dose, data = ToothGrowth),
       simulate = TRUE,
       main = "Q-Q plot",
       xlab = "Cuantile teoretice",
       ylab = "Reziduurile standardizate")
```

Trebuie menționat că în cazul în care ipoteza de normalitate era respinsă atunci puteam folosi testul neparametric `Kruskal-Wallis` (funcția `kruskal.test`) ca alternativă la ANOVA.

Pentru a verifica **condiția de omogenitate** a datelor (homoscedasticitatea) pentru fiecare factor în parte folosim unul din testele următoare: *testul lui Bartlett* (funcția `bartlett.test`), *testul lui Fligner-Killeen* (funcția `fligner.test`), *testul lui Levene* (funcția `leveneTest` din pachetul `car`) sau *testul Brown-Forsythe* (funcția `leveneTest` din pachetul `car` sau funcția `hov()` din pachetul `HH`). Obținem:

- testul lui Bartlett

```
bartlett.test(len~supp, data = ToothGrowth)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  len by supp
## Bartlett's K-squared = 1.4217, df = 1, p-value = 0.2331
```

```
bartlett.test(len~dose, data = ToothGrowth)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  len by dose
## Bartlett's K-squared = 0.66547, df = 2, p-value = 0.717
```

- testul lui Fligner

```
fligner.test(len~supp, data = ToothGrowth)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  len by supp
## Fligner-Killeen:med chi-squared = 0.97034, df = 1, p-value =
## 0.3246
```

```
fligner.test(len~dose, data = ToothGrowth)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  len by dose
## Fligner-Killeen:med chi-squared = 1.3879, df = 2, p-value = 0.4996
```

- testul lui Levene (clasic)

```
leveneTest(len~supp, data = ToothGrowth, center = mean)
```

```
## Levene's Test for Homogeneity of Variance (center = mean)
##       Df F value Pr(>F)
## group  1  1.0973 0.2992
##       58
```

```
leveneTest(len~dose, data = ToothGrowth, center = mean)
```

```
## Levene's Test for Homogeneity of Variance (center = mean)
##       Df F value Pr(>F)
## group  2  0.7328  0.485
##       57
```

- testul lui Brown-Forsythe (similar cu testul lui Levene numai că folosește mediana în loc de medie și este mai robust)

```
leveneTest(len~supp, data = ToothGrowth)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##       Df F value Pr(>F)
## group  1  1.2136 0.2752
##       58
```

```r
leveneTest(len~dose, data = ToothGrowth)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##       Df F value Pr(>F)
## group  2  0.6457 0.5281
##       57
```

2.1.2 Comparări multiple

Am văzut din tabelul ANOVA cu doi factori că atât efectele principale (`supp` și `dose`) cât și interacția dintre cei doi factori (`supp:dose`) sunt semnificative. Pentru a vedea care interacțiune este semnificativă vom folosi metodologia testării multiple. Vom folosi *Testul lui Tukey HSD* (Honestly Significant Difference) deoarece suntem în situația unui plan de experiență echilibrat (în caz contrar am putea folosi *testul lui Scheffe*), test care permite compararea tuturor perechilor de diferențe dintre mediile grupurilor:

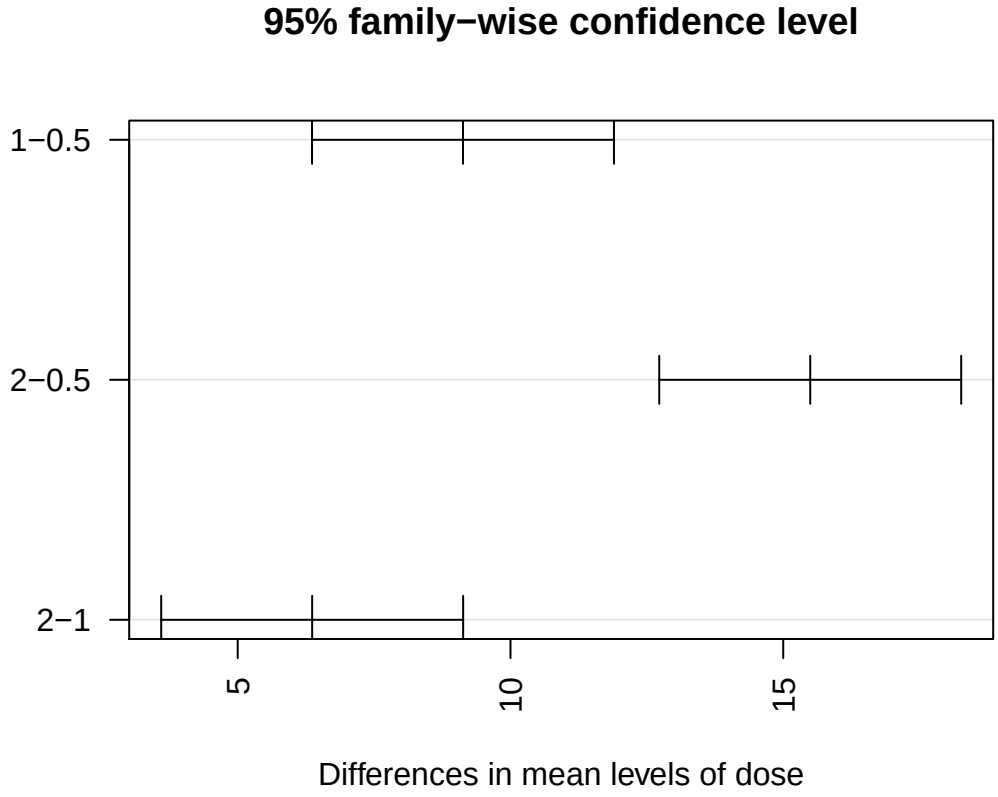
```r
TukeyHSD(model_anova_2w)
```

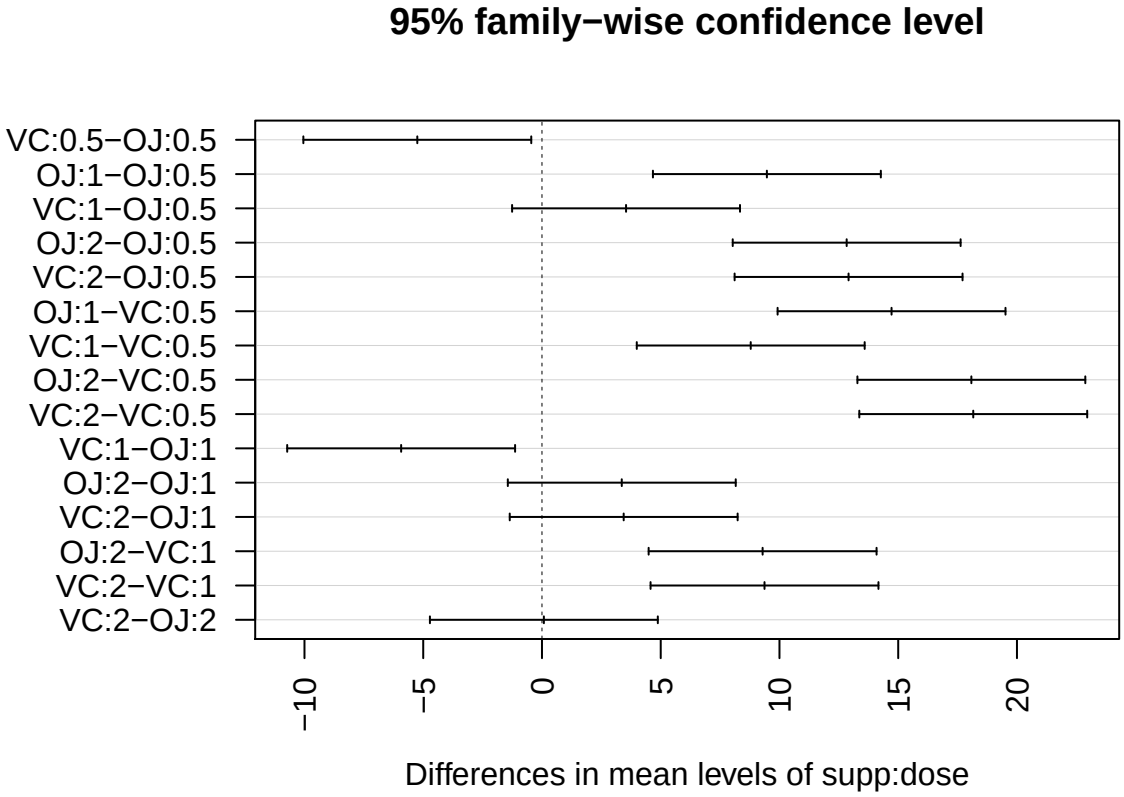
```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = len ~ supp * dose, data = ToothGrowth)
##
## $supp
##       diff       lwr       upr     p adj
## VC-OJ -3.7 -5.579828 -1.820172 0.0002312
##
## $dose
##         diff       lwr       upr   p adj
## 1-0.5  9.130  6.362488 11.897512 0.0e+00
## 2-0.5 15.495 12.727488 18.262512 0.0e+00
## 2-1    6.365  3.597488  9.132512 2.7e-06
##
## $`supp:dose`
##                 diff        lwr        upr     p adj
## VC:0.5-OJ:0.5 -5.25 -10.048124 -0.4518762 0.0242521
## OJ:1-OJ:0.5    9.47   4.671876 14.2681238 0.0000046
## VC:1-OJ:0.5    3.54  -1.258124  8.3381238 0.2640208
## OJ:2-OJ:0.5   12.83   8.031876 17.6281238 0.0000000
## VC:2-OJ:0.5   12.91   8.111876 17.7081238 0.0000000
## OJ:1-VC:0.5   14.72   9.921876 19.5181238 0.0000000
## VC:1-VC:0.5    8.79   3.991876 13.5881238 0.0000210
## OJ:2-VC:0.5   18.08  13.281876 22.8781238 0.0000000
## VC:2-VC:0.5   18.16  13.361876 22.9581238 0.0000000
## VC:1-OJ:1     -5.93 -10.728124 -1.1318762 0.0073930
## OJ:2-OJ:1      3.36  -1.438124  8.1581238 0.3187361
## VC:2-OJ:1      3.44  -1.358124  8.2381238 0.2936430
## OJ:2-VC:1      9.29   4.491876 14.0881238 0.0000069
## VC:2-VC:1      9.37   4.571876 14.1681238 0.0000058
## VC:2-OJ:2      0.08  -4.718124  4.8781238 1.0000000
```

Observăm, de exemplu, că diferența dintre mediile lungimii dinților pentru porcușorii de gunineea care au primit o doză de 1 mg de Vitamina C prin suplementul de suc de portocale și prin suplementul de soluție de acid ascorbic este semnificativă (VC:1-OJ:1 $p = 0.0073$) pe când diferența dintre mediile lungimii dinților

pentru porcușorii de gunineea care au primit o doză de 2 mg și una de 1 mg de Vitamina C prin suplementul de suc de portocale nu este semnificativă (OJ:2-OJ:1 $p = 0.318$).

Aceste diferențe se pot observa și grafic:


95% family−wise confidence level
1−0.5
2−0.5
2−1
5
10
15
Differences in mean levels of dose


95% family−wise confidence level
VC:0.5−OJ:0.5
OJ:1−OJ:0.5
VC:1−OJ:0.5
OJ:2−OJ:0.5
VC:2−OJ:0.5
OJ:1−VC:0.5
VC:1−VC:0.5
OJ:2−VC:0.5
VC:2−VC:0.5
VC:1−OJ:1
OJ:2−OJ:1
VC:2−OJ:1
OJ:2−VC:1
VC:2−VC:1
VC:2−OJ:2
−10
−5
0
5
10
15
20
Differences in mean levels of supp:dose

Curs Biostatistica 2017 - Laborator 7 & 8

Regresie

Contents

1 Regresie liniară simplă

1.1 Introducere

Regresia liniară simplă (sau *modelul liniar simplu*) este un instrument statistic utilizat pentru a descrie relația dintre două variabile aleatoare, X (variabilă *cauză*, *predictor* sau *covariabilă*) și Y (variabilă *răspuns* sau *efect*) și este definit prin

$$\mathbb{E}[Y|X = x] = \beta_0 + \beta_1 x$$

sau altfel spus

$$Y = \beta_0 + \beta_1 X + \varepsilon.$$

În relațiile de mai sus, β_0 și β_1 sunt cunoscute ca ordonata la origine (*intercept*) și respectiv panta (*slope*) dreptei de regresie.

Ipotezele modelului sunt:

i. **Linearitatea**: $\mathbb{E}[Y|X = x] = \beta_0 + \beta_1 x$
ii. **Homoscedasticitatea**: $\mathbb{V}\text{ar}(\varepsilon_i) = \sigma^2$, cu σ^2 constantă pentru $i = 1, \ldots, n$
iii. **Normalitatea**: $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$ pentru $i = 1, \ldots, n$
iv. **Independența erorilor**: $\varepsilon_1, \ldots, \varepsilon_n$ sunt independente (sau necorelate, $\mathbb{E}[\varepsilon_i \varepsilon_j] = 0$, $i \neq j$, deoarece sunt presupuse normale)

Altfel spus

$$Y|X = x \sim \mathcal{N}(\beta_0 + \beta_1 x, \sigma^2)$$

– Nicio ipoteză nu a fost făcută asupra repartiției lui X (poate fi sau deterministă asu aleatoare)

– Modelul de regresie presupune că Y **este continuă** datorită normalității erorilor. În orice caz, X **poate fi o variabilă discretă!**

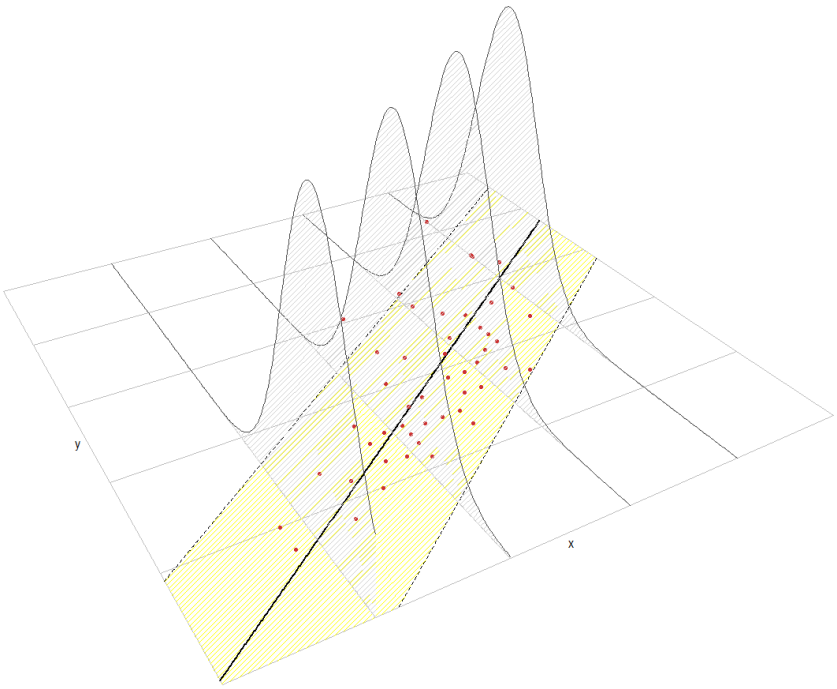


Figure 1: Regresia liniara simpla

Dat fiind un eșantion $(X_1, Y_1), \ldots, (X_n, Y_n)$ pentru variabilele X și Y putem estima coeficienții necunoscuți β_0 și β_1 minimizând *suma abaterilor pătratice reziduale* (*Residual Sum of Squares* - RSS)

$$\text{RSS}(\beta_0, \beta_1) = \sum_{i=1}^{n} (Y_i - \beta_0 - \beta_1 X_i)^2$$

ceea ce conduce la

$$\hat{\beta}_0 = \bar{Y} - \hat{\beta}_1 \bar{X}, \quad \hat{\beta}_1 = \frac{s_{xy}}{s_x^2} = \frac{\sum_{i=1}^{n} (X_i - \bar{X}) Y_i}{\sum_{i=1}^{n} (X_i - \bar{X})^2}$$

unde

- $\bar{X} = \frac{1}{n} \sum_{i=1}^{n} X_i$ este *media eșantionului*
- $s_x^2 = \frac{1}{n} \sum_{i=1}^{n} (X_i - \bar{X})^2$ este *varianța eșantionului*
- $s_{xy} = \frac{1}{n} \sum_{i=1}^{n} (X_i - \bar{X})(Y_i - \bar{Y})$ este *covarianța eșantionului*

Odată ce avem estimatorii $(\hat{\beta}_0, \hat{\beta}_1)$, putem defini:

- *valorile prognozate* (*fitted values*) $\hat{Y}_1, \ldots, \hat{Y}_n$ (valorile verticale pe dreapta de regresie), unde

$$\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 X_i, \quad i = 1, \ldots, n$$

- *reziduurile estimate* (*estimated residuals*) $\hat{\varepsilon}_1, \ldots, \hat{\varepsilon}_n$ (distanțele verticale dintre punctele actuale (X_i, Y_i) și cele prognozate $(X_i, \hat{Y}_i)$), unde

$$\hat{\varepsilon}_i = Y_i - \hat{Y}_i, \quad i = 1, \ldots, n$$

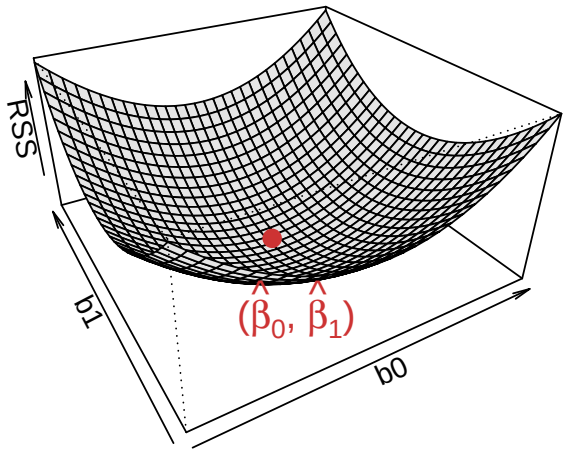


Figure 2: Graficul functiei RSS pentru modelul $y = -0.5 + 1.5x + e$.

Estimatorul pentru σ^2 este

$$\hat{\sigma}^2 = \frac{RSS(\hat{\beta}_0, \hat{\beta}_1)}{n-2} = \frac{\sum_{i=1}^{n} \hat{\varepsilon}_i^2}{n-2}.$$

1.2 Exemplul 1

> În acest exercițiu vrem să investigăm relația dintre consumul de clorură de sodiu (sarea de bucătărie) și tensiunea arterială la persoanele trecute de 65 de ani. Pentru aceasta vom folosi setul de date `saltBP` care conține informații despre tensiunea arterială a 25 de pacienți.

Începem prin a înregistra setul de date

```
saltBP = read.table("data/saltBP.txt", header = T)

plot(saltBP$salt, saltBP$BP,
     xlab = "sare",
     ylab = "tensiunea arteriala",
     col = "brown3",
     pch = 16,
     bty="n")
```

```
summary(saltBP)
```

```
##        BP              salt          saltLevel
##  Min.   :128.3   Min.   : 1.130   Min.   :0.0
##  1st Qu.:131.8   1st Qu.: 2.650   1st Qu.:0.0
##  Median :135.7   Median : 5.210   Median :0.0
```

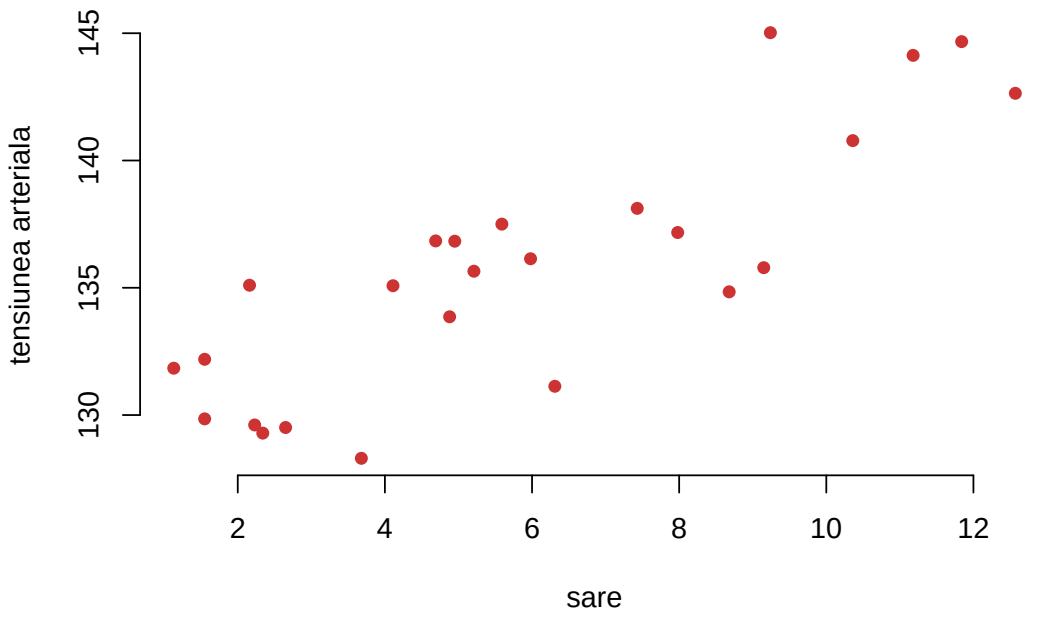


Figure 3: Diagrama de imprastiere

```
##  Mean   :135.7   Mean   : 5.898   Mean   :0.4
##  3rd Qu.:137.5   3rd Qu.: 8.680   3rd Qu.:1.0
##  Max.   :145.0   Max.   :12.570   Max.   :1.0
```

1.2.1 Estimarea parametrilor

Considerăm modelul de regresie $Y = \beta_0 + \beta_1 X + \varepsilon$ (unde $X =$`saltBP$salt` iar $Y =$`saltBP$BP`), $\varepsilon \sim \mathcal{N}(0, \sigma^2)$, a cărui parametrii sunt β_0, β_1 și σ^2.

- estimatorii parametrilor β_0 și β_1

```
# pentru b1

b1 = cov(saltBP$salt, saltBP$BP)/var(saltBP$salt)
cat("b1 = ", b1)
```

```
## b1 =  1.196894
```

```
# sau

sum((saltBP$salt-mean(saltBP$salt))*(saltBP$BP))/sum((saltBP$salt-mean(saltBP$salt))^2)
```

```
## [1] 1.196894
```

```
# pentru b0

b0 = mean(saltBP$BP) - b1*mean(saltBP$salt)
cat("b0 = ", b0)
```

```
## b0 =  128.6164
```

sau folosind funcția `lm`:

```
saltBP_model = lm(BP~salt, data = saltBP)
names(saltBP_model)
```

```
##  [1] "coefficients"  "residuals"     "effects"       "rank"
##  [5] "fitted.values" "assign"        "qr"            "df.residual"
##  [9] "xlevels"       "call"          "terms"         "model"
```

```
saltBP_model$coefficients
```

```
## (Intercept)        salt
##  128.616397    1.196894
```

Dreapta de regresie este:

```
plot(saltBP$salt, saltBP$BP,
     xlab = "nivelul de sare",
     ylab = "tensiunea arteriala",
     col = "brown3",
     pch = 16,
     bty="n",
     main = paste("y = ", format(b0, digits = 4), " + ", format(b1, digits = 4), " x"))

abline(a = b0, b = b1, col = "grey", lwd = 2)
points(mean(saltBP$salt), mean(saltBP$BP), pch = 16, col = "dark green", cex = 1.2)
text(mean(saltBP$salt), mean(saltBP$BP)-1.3, col = "dark green", cex = 1.2,
     labels = expression(paste("(", bar(x), ",", bar(y),")")))
```

- estimatorul lui σ ($\hat{\sigma}$)

```
n = length(saltBP$BP)
e_hat = saltBP$BP - (b0+b1*saltBP$salt)

rss = sum(e_hat^2)

sigma_hat = sqrt(rss/(n-2))
sigma_hat
```

```
## [1] 2.745374
```

sau cu ajutorul funcției `lm`

```
sqrt(deviance(saltBP_model)/df.residual(saltBP_model))
```

```
## [1] 2.745374
```

sau încă

```
saltBP_model_summary = summary(saltBP_model)
# names(saltBP_model_summary)
saltBP_model_summary$sigma
```

```
## [1] 2.745374
```

1.2.2 Intervale de încredere pentru parametrii

Repartițiile lui $\hat{\beta}_0$ și $\hat{\beta}_1$ sunt

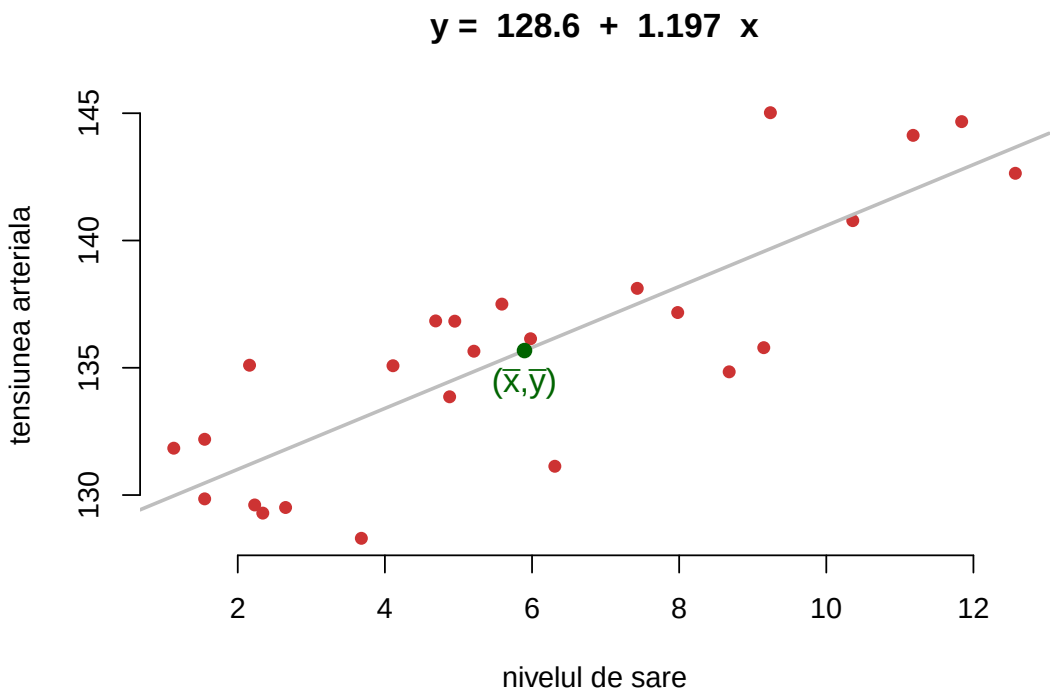


Figure 4: Dreapta de regresie

$$\hat{\beta}_0 \sim \mathcal{N}\left(\beta_0, \mathrm{SE}(\hat{\beta}_0)^2\right), \quad \hat{\beta}_1 \sim \mathcal{N}\left(\beta_1, \mathrm{SE}(\hat{\beta}_1)^2\right)$$

unde

$$\mathrm{SE}(\hat{\beta}_0)^2 = \frac{\sigma^2}{n}\left[1 + \frac{\bar{X}^2}{s_x^2}\right], \quad \mathrm{SE}(\hat{\beta}_1)^2 = \frac{\sigma^2}{n s_x^2}.$$

Folosind estimatorul $\hat{\sigma}^2$ pentru σ^2 obținem că

$$\frac{\hat{\beta}_0 - \beta_0}{\hat{\mathrm{SE}}(\hat{\beta}_0)} \sim t_{n-2}, \quad \frac{\hat{\beta}_1 - \beta_1}{\hat{\mathrm{SE}}(\hat{\beta}_1)} \sim t_{n-2}$$

unde

$$\hat{\mathrm{SE}}(\hat{\beta}_0)^2 = \frac{\hat{\sigma}^2}{n}\left[1 + \frac{\bar{X}^2}{s_x^2}\right], \quad \hat{\mathrm{SE}}(\hat{\beta}_1)^2 = \frac{\hat{\sigma}^2}{n s_x^2}$$

prin urmare, intervalele de încredere de nivel $1-\alpha$ pentru β_0 și β_1 sunt

$$IC = \left(\hat{\beta}_j \pm \hat{\mathrm{SE}}(\hat{\beta}_j) t_{n-2;\alpha/2}\right), \quad j = 0, 1.$$

```r
alpha = 0.05

# trebuie avut grija ca functia var si sd calculeaza impartind la (n-1) si nu la n !!!
se_b0 = sqrt(sigma_hat^2*(1/n+mean(saltBP$salt)^2/((n-1)*var(saltBP$salt))))
se_b1 = sqrt(sigma_hat^2/((n-1)*var(saltBP$salt)))

lw_b0 = b0 - qt(1-alpha/2, n-2)*se_b0
up_b0 = b0 + qt(1-alpha/2, n-2)*se_b0

cat("CI pentru b0 este (", lw_b0, ", ", up_b0, ")\n")
```

```
## CI pentru b0 este ( 126.337 ,  130.8958 )
```

```r
lw_b1 = b1 - qt(1-alpha/2, n-2)*se_b1
up_b1 = b1 + qt(1-alpha/2, n-2)*se_b1

cat("CI pentru b1 este (", lw_b1, ", ", up_b1, ")")
```

```
## CI pentru b1 este ( 0.8617951 ,  1.531993 )
```

Același rezultat se obține apelând funcția `confint` :

```r
confint(saltBP_model)
```

```
##                   2.5 %     97.5 %
## (Intercept) 126.3369606 130.895834
## salt          0.8617951   1.531993
```

Putem construi și o regiune de încredere pentru perechea (β_0, β_1):

```r
plot(ellipse(saltBP_model, c(1,2)), type = "l", col = "grey30",
     xlab = expression(beta[0]),
     ylab = expression(beta[1]),
     bty = "n")
points(coef(saltBP_model)[1], coef(saltBP_model)[2], pch = 18, col = "brown3")
abline(v = confint(saltBP_model)[1,], lty = 2)
abline(h = confint(saltBP_model)[2,], lty = 2)
```

1.2.3 ANOVA pentru regresie

Este predictorul X folositor în prezicerea răspunsului Y ? Vrem să testăm ipoteza nulă $H_0: \ \beta_1 = 0$.

Introducem următoarele *sume de abateri pătratice*:

- $SS_T = \sum_{i=1}^{n} \left(Y_i - \bar{Y}\right)^2$, **suma abaterilor pătratice totală** (variația totală a lui $Y_1, \ldots, Y_n$).
- $SS_{reg} = \sum_{i=1}^{n} \left(\hat{Y}_i - \bar{Y}\right)^2$, **suma abaterilor pătratice de regresie** (variabilitatea explicată de dreapta de regresie)
- $RSS = \sum_{i=1}^{n} \left(Y_i - \hat{Y}_i\right)^2$, **suma abaterilor pătratice reziduale**

Avem următoarea descompunere ANOVA

$$\underbrace{SS_T}_{\text{Variația lui } Y_i} = \underbrace{SS_{reg}}_{\text{Variația lui } \hat{Y}_i} + \underbrace{RSS}_{\text{Variația lui } \hat{\varepsilon}_i}$$

și tabelul ANOVA corespunzător

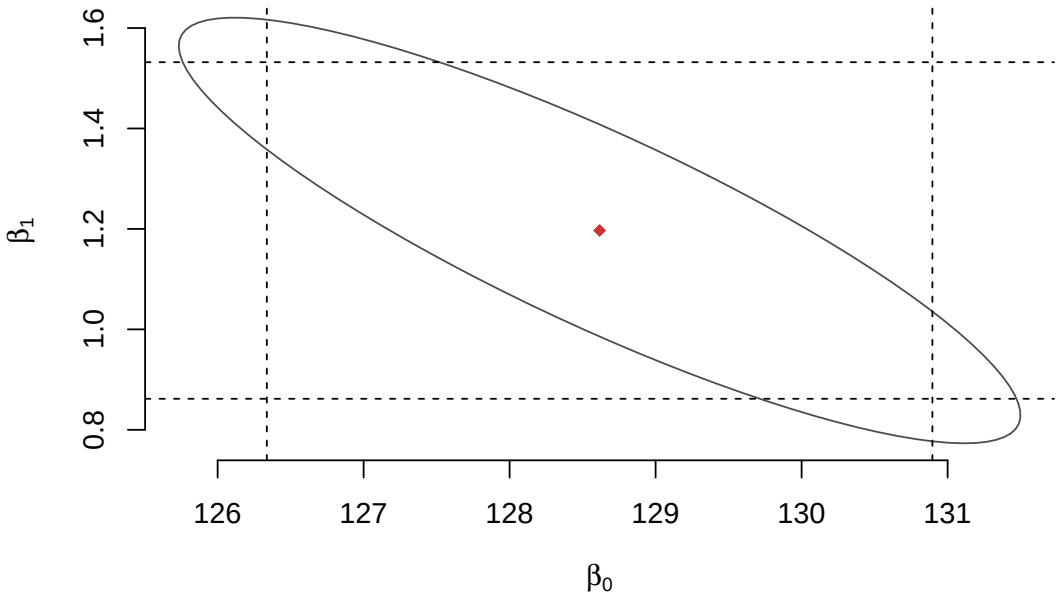


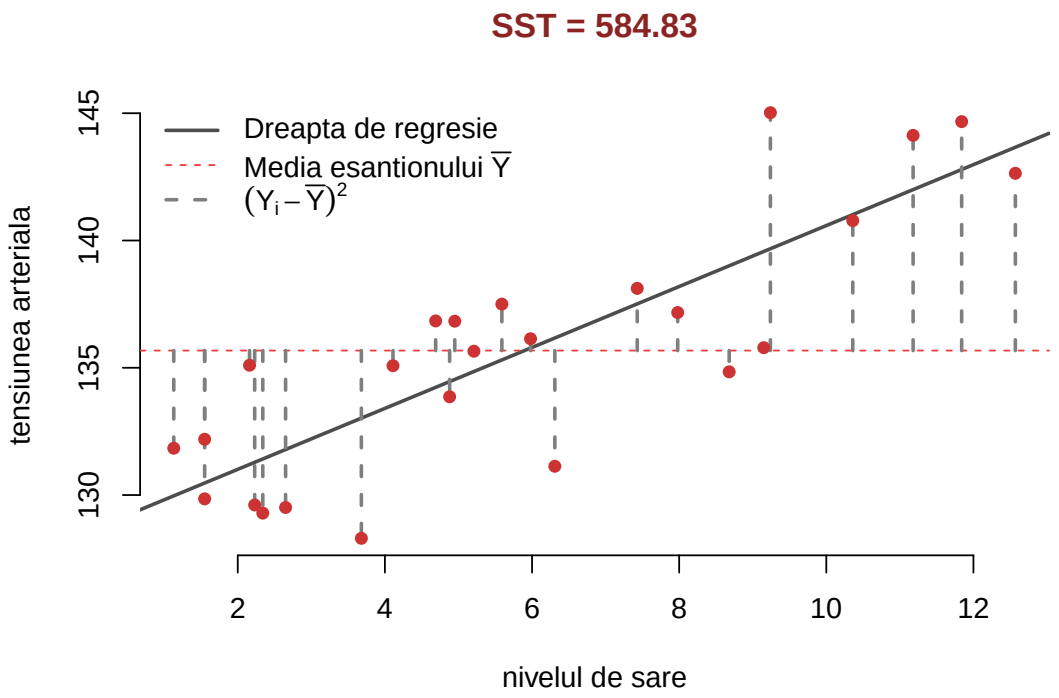
Figure 5: Regiune de incredere

	Df	SS	MS	F	p-value
Predictor	1	SS_{reg}	$\frac{SS_{reg}}{1}$	$\frac{SS_{reg}/1}{RSS/(n-2)}$	p
Residuuri	$n-2$	RSS	$\frac{RSS}{n-2}$		

Descompunerea ANOVA pentru problema noastră poate fi ilustrată astfel:

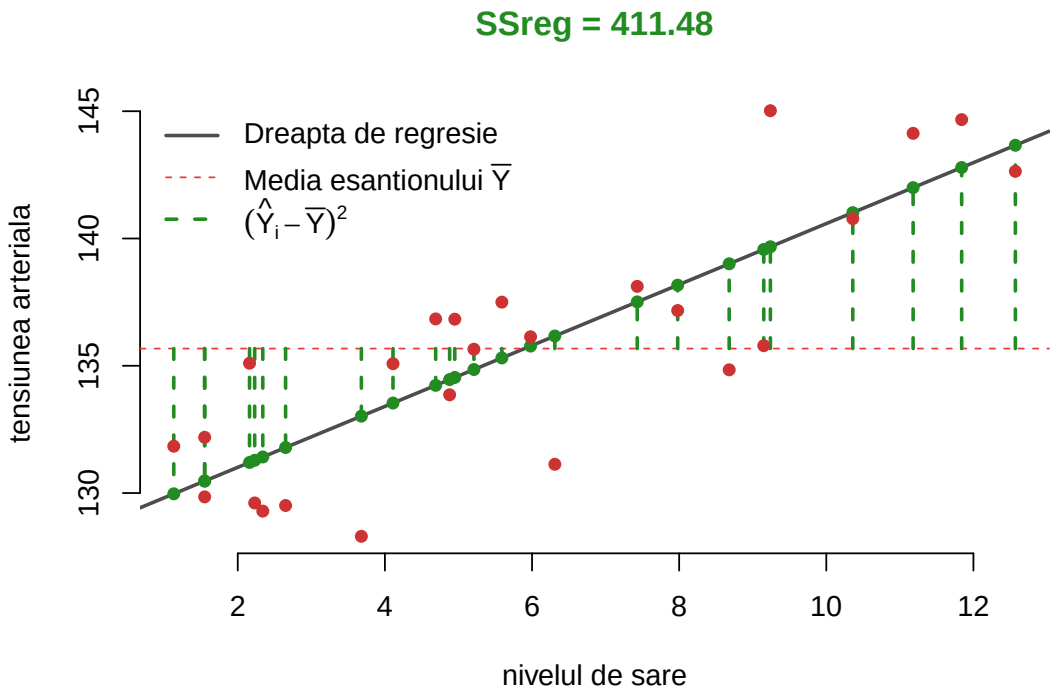
- *suma abaterilor pătratice totală*:

```
plot(saltBP$salt, saltBP$BP, pch = 16, type = "n",
     main = paste("SST =", round(sum((saltBP$BP - mean(saltBP$BP))^2), 2)),
     col.main = "brown4",
     xlab = "nivelul de sare",
     ylab = "tensiunea arteriala",
     bty = "n")

abline(saltBP_model$coefficients, col = "grey30", lwd = 2)
abline(h = mean(saltBP$BP), col = "brown2", lty = 2)

segments(x0 = saltBP$salt, y0 = mean(saltBP$BP), x1 = saltBP$salt, y1 = saltBP$BP,
         col = "grey50", lwd = 2, lty = 2)

legend("topleft", legend = expression("Dreapta de regresie", "Media esantionului " * bar(Y),
                                      (Y[i] - bar(Y))^2),
       lwd = c(2, 1, 2),
       col = c("grey30", "brown2", "grey50"),
       lty = c(1, 2, 2),
       bty = "n")

points(saltBP$salt, saltBP$BP, pch = 16, col = "brown3")
```



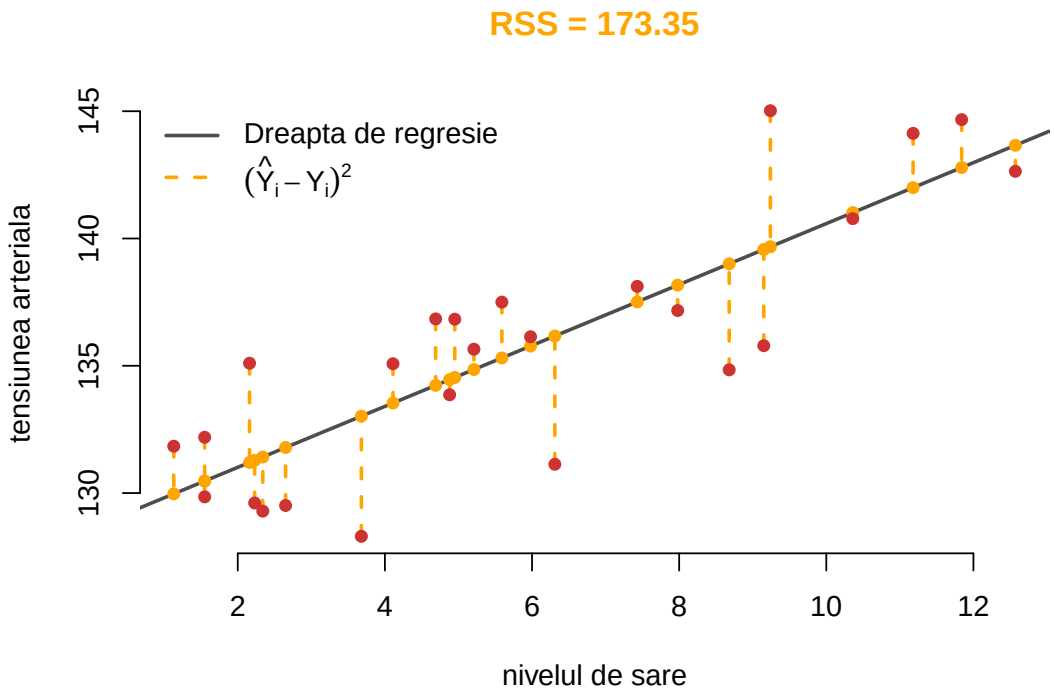
- *suma abaterilor pătratice de regresie*

```
plot(saltBP$salt, saltBP$BP, pch = 16, type = "n",
     main = paste("SSreg =", round(sum((saltBP_model$fitted.values - mean(saltBP$BP))^2), 2)),
     col.main = "forestgreen",
     xlab = "nivelul de sare",
     ylab = "tensiunea arteriala",
     bty = "n")

abline(saltBP_model$coefficients, col = "grey30", lwd = 2)
abline(h = mean(saltBP$BP), col = "brown2", lty = 2)

segments(x0 = saltBP$salt, y0 = mean(saltBP$BP), x1 = saltBP$salt, y1 = saltBP_model$fitted.values,
         col = "forestgreen", lwd = 2, lty = 2)

points(saltBP$salt, saltBP_model$fitted.values, pch = 16, col = "forestgreen")

legend("topleft", legend = expression("Dreapta de regresie", "Media esantionului " * bar(Y),
                                      (hat(Y)[i] - bar(Y))^2),
       lwd = c(2, 1, 2),
       col = c("grey30", "brown2", "forestgreen"),
       lty = c(1, 2, 2),
       bty = "n")

points(saltBP$salt, saltBP$BP, pch = 16, col = "brown3")
```



- *suma abaterilor pătratice reziduale*

```
plot(saltBP$salt, saltBP$BP, pch = 16, type = "n",
     main = paste("RSS =", round(sum((saltBP$BP - saltBP_model$fitted.values)^2), 2)),
     col.main = "orange",
     xlab = "nivelul de sare",
     ylab = "tensiunea arteriala",
     bty = "n")

abline(saltBP_model$coefficients, col = "grey30", lwd = 2)

segments(x0 = saltBP$salt, y0 = saltBP$BP, x1 = saltBP$salt, y1 = saltBP_model$fitted.values,
         col = "orange", lwd = 2, lty = 2)

points(saltBP$salt, saltBP_model$fitted.values, pch = 16, col = "orange")

legend("topleft", legend = expression("Dreapta de regresie", (hat(Y)[i] - Y[i])^2),
       lwd = c(2, 2),
       col = c("grey30", "orange"),
       lty = c(1, 2),
       bty = "n")

points(saltBP$salt, saltBP$BP, pch = 16, col = "brown3")
```



Tabelul ANOVA se obține prin

```
# tabel ANOVA
anova(saltBP_model)
```

```
## Analysis of Variance Table
##
## Response: BP
##           Df Sum Sq Mean Sq F value    Pr(>F)
## salt       1 411.48  411.48  54.594 1.631e-07 ***
## Residuals 23 173.35    7.54
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Definiția *coeficientului de determinare* R^2 este strâns legată de descompunerea ANOVA:

$$R^2 = \frac{SS_{reg}}{SS_T} = \frac{SS_T - RSS}{SS_T} = 1 - \frac{RSS}{SS_T}$$

R^2 măsoară **proporția din variația** variabilei răspuns Y **explicată** de variabila predictor X prin regresie. Proporția din variația totală a lui Y care nu este explicată este $1 - R^2 = \frac{RSS}{SS_T}$. Intuitiv, R^2 măsoară cât de bine modelul de regresie este în concordanță cu datele (cât de strâns este norul de puncte în jurul dreptei de regresie). Observăm că dacă datele concordă *perfect* cu modelul (adică $RSS = 0$) atunci $R^2 = 1$.

Putem vedea că $R^2 = r_{xy}^2$, unde r_{xy} este *coeficientul de corelație* empiric:

$$r_{xy} = \frac{s_{xy}}{s_x s_y} = \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^n (X_i - \bar{X})^2}\sqrt{\sum_{i=1}^n (Y_i - \bar{Y})^2}}$$

Mai mult se poate verifica și că $R^2 = r^2_{y\hat{y}}$, adică *coeficientul de determinare este egal cu pătratul coeficientului de corelație empirică dintre* $Y_1, \ldots, Y_n$ *și* $\hat{Y}_1, \ldots, \hat{Y}_n$.

Verificăm relația $R^2 = r^2_{xy} = r^2_{y\hat{y}}$ numeric:

```
yHat = saltBP_model$fitted.values

saltBP_model_summary$r.squared # R^2
```

```
## [1] 0.7035842
```

```
cor(saltBP$salt, saltBP$BP)^2 # corelatia^2 dintre x si y
```

```
## [1] 0.7035842
```

```
cor(saltBP$BP, yHat)^2 # corelatia^2 dintre y si yHat
```

```
## [1] 0.7035842
```

1.2.4 Inferență asupra parametrilor

Este predictorul X folositor în prezicerea răspunsului Y ? Vrem să testăm ipoteza nulă $H_0 : \beta_j = 0$ (pentru $j = 1$ spunem că predictorul `nivel de sare` nu are un efect *liniar* semnificativ asupra `tensiunii arteriale`). Pentru aceasta vom folosi statistica de test

$$t_j = \frac{\hat{\beta}_j}{\hat{SE}(\hat{\beta}_j)} \sim_{H_0} t_{n-2}.$$

Funcția `summary` ne întoarce p-valoarea corespunzătoare a acestor teste:

```
summary(saltBP_model)
```

```
##
## Call:
## lm(formula = BP ~ salt, data = saltBP)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.0388 -1.6755  0.3662  1.8824  5.3443
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  128.616      1.102 116.723  < 2e-16 ***
## salt           1.197      0.162   7.389 1.63e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.745 on 23 degrees of freedom
## Multiple R-squared:  0.7036, Adjusted R-squared:  0.6907
## F-statistic: 54.59 on 1 and 23 DF,  p-value: 1.631e-07
```

Observăm că ambele ipoteze sunt respinse în favoarea alternativelor bilaterale (la aceeași concluzie am ajuns și utitându-ne la intervalele de încredere - nu conțineau valoarea 0). Putem observa că t_1^2 este exact valoarea F statisticii, deci cele două abordări ne dau aceleași rezultate numerice.

1.2.5 Predicții

Pentru un nou set de predictori, x_0, răspunsul prognozat este $\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_0$ și vrem să investigăm incertitudinea din această predicție. Putem face distincția între două tipuri de predicție: predicție asupra răspunsului viitor mediu (inferență asupra mediei condiționate $\mathbb{E}[Y|X = x_0]$) sau predicție asupra observațiilor viitoare (inferență asupra răspunsului condiționat $Y|X = x_0$).

Un interval de încredere pentru răspunsul viitor mediu este:

$$\left(\hat{y} \pm t_{n-2:\alpha/2}\sqrt{\frac{\hat{\sigma}^2}{n}\left(1 + \frac{(x_0 - \bar{x})^2}{s_x^2}\right)}\right)$$

Un interval de încredere pentru valoarea prezisă (interval de predicție) este:

$$\left(\hat{y} \pm t_{n-2:\alpha/2}\sqrt{\hat{\sigma}^2 + \frac{\hat{\sigma}^2}{n}\left(1 + \frac{(x_0 - \bar{x})^2}{s_x^2}\right)}\right)$$

Pentru a găsi aceste intervale vom folosi funcția `predict`:

```
newData = data.frame(salt = 14)
newData2 = data.frame(salt = c(13, 14, 15))

# Predictie
predict(saltBP_model, newdata = newData)
```

```
##        1
## 145.3729
```

```
# Predictie pentru valoarea raspunsului mediu
predict(saltBP_model, newdata = newData, interval = "confidence")
```

```
##        fit      lwr     upr
## 1 145.3729 142.4298 148.316
```

```
predict(saltBP_model, newdata = newData2, interval = "confidence")
```

```
##        fit      lwr      upr
## 1 144.1760 141.5389 146.8132
## 2 145.3729 142.4298 148.3160
## 3 146.5698 143.3150 149.8246
```

```
# Predictie asupra observatiilor viitoare
predict(saltBP_model, newdata = newData, interval = "prediction")
```

```
##        fit      lwr      upr
## 1 145.3729 138.9764 151.7695
```

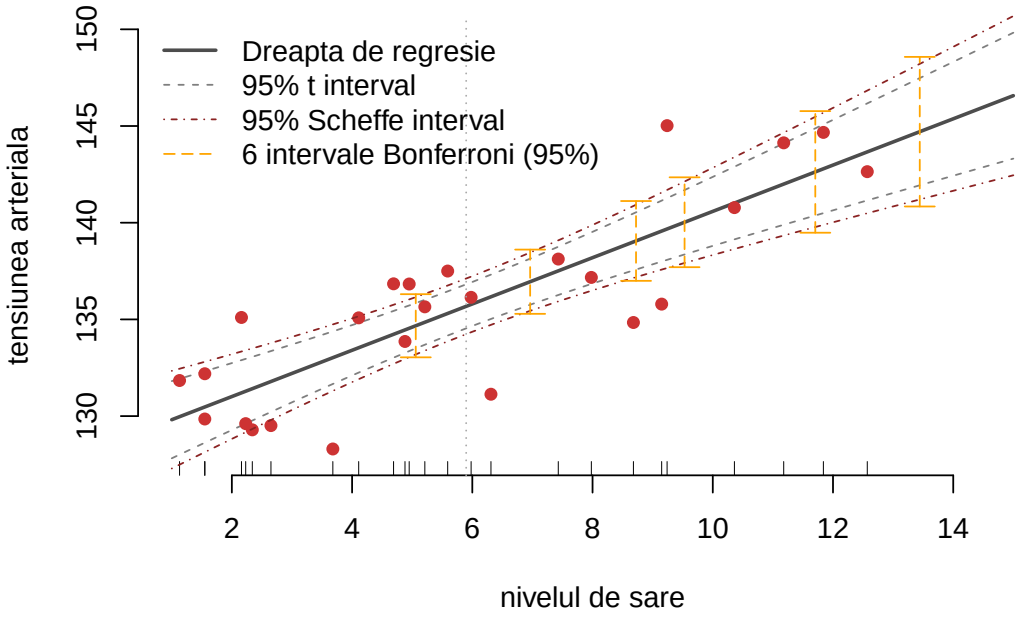
```
predict(saltBP_model, newdata = newData2, interval = "prediction")
```

```
##        fit      lwr      upr
## 1 144.1760 137.9144 150.4377
## 2 145.3729 138.9764 151.7695
## 3 146.5698 140.0240 153.1156
```

```
g = seq(1,15,0.5)

p = predict(saltBP_model, data.frame(salt = g), se = T, interval = "confidence")
```

```r
matplot(g, p$fit, type = "l", lty = c(1,2,2),
        lwd = c(2,1,1),
        col = c("grey30", "grey50", "grey50"),
        xlab = "nivelul de sare",
        ylab = "tensiunea arteriala",
        bty = "n")
rug(saltBP$salt)
points(saltBP$salt, saltBP$BP, col = "brown3", pch = 16)
abline(v = mean(saltBP$salt), lty = 3, col = "grey65")

# Scheffe's bounds
M = sqrt(2*qf(1-alpha, 2, n-2))

s_xx = (n-1)*var(saltBP$salt)
lw_scheffe = b0 + b1*g - M*sigma_hat*sqrt(1/n+(g-mean(saltBP$salt))^2/s_xx)
up_scheffe = b0 + b1*g + M*sigma_hat*sqrt(1/n+(g-mean(saltBP$salt))^2/s_xx)

lines(g, lw_scheffe, lty = 4, col = "brown4")
lines(g, up_scheffe, lty = 4, col = "brown4")

# Bonferroni bounds
# x0 = c(7, 8, 13, 14)
x0 = 1 + 14*runif(6)
m = length(x0)

t_bonf = qt(1-alpha/(2*m), n-2)

lw_bonf = b0 + b1*x0 - t_bonf*sigma_hat*sqrt(1/n+(x0-mean(saltBP$salt))^2/s_xx)
up_bonf = b0 + b1*x0 + t_bonf*sigma_hat*sqrt(1/n+(x0-mean(saltBP$salt))^2/s_xx)

segments(x0 = x0, y0 = lw_bonf, x1 = x0, y1 = up_bonf, col = "orange", lty = 5)
segments(x0 = x0-0.25, y0 = lw_bonf, x1 = x0+0.25, y1 = lw_bonf, col = "orange", lty = 1)
segments(x0 = x0-0.25, y0 = up_bonf, x1 = x0+0.25, y1 = up_bonf, col = "orange", lty = 1)

legend("topleft", legend = c("Dreapta de regresie", "95% t interval",
                                    "95% Scheffe interval", paste0(m, " intervale Bonferroni (95%)"))
       lwd = c(2, 1, 1, 1),
       col = c("grey30", "grey50", "brown4", "orange"),
       lty = c(1, 2, 4, 5),
       bty = "n")
```

\begin{figure}



{

}

\caption{Nivelul de sare prezis impreuna cu intervalul de incredere de nivel 95% pentru raspunsul mediu}
\end{figure}

1.2.6 Diagnostic

În această secțiune vom vedea dacă setul nostru de date verifică ipotezele modelului de regresie liniară.

- *Independența*

Ipoteza de independență a variabilei răspuns (prin urmare și a erorilor) reiese, de cele mai multe ori, din modalitatea în care s-a desfășurat experimentul.

- *Normalitatea*

Pentru a verifica dacă ipoteza de normalitate a erorilor este satisfăcută vom trasa dreapta lui Henry (sau Q-Q plot-ul):

```
library(car)
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:ellipse':
##
##     ellipse
```

```
qqPlot(saltBP_model, col = "brown3", col.lines = "grey50", pch = 16,
       simulate = TRUE,
       xlab = "Cuantile teoretice",
```

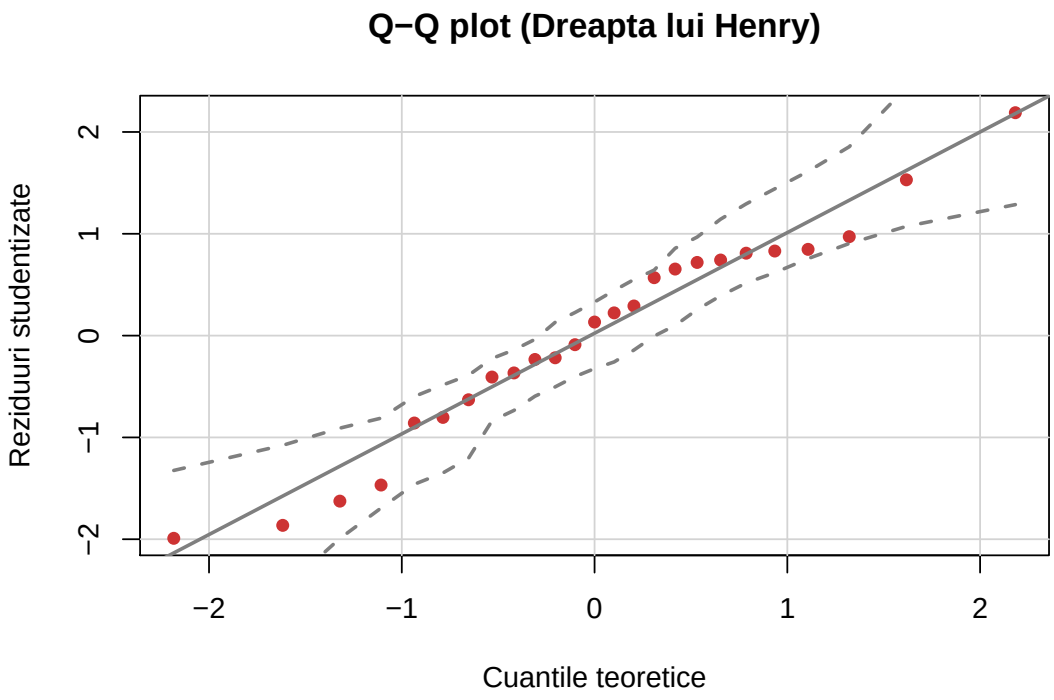


Figure 6: Q-Q plot

```
       ylab = "Reziduuri studentizate",
       main = "Q-Q plot (Dreapta lui Henry)")
```

Putem folosi și testul `Shapiro-Wilk`:

```
shapiro.test(residuals(saltBP_model))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(saltBP_model)
## W = 0.96871, p-value = 0.6125
```

```
residplot <- function(fit, nbreaks=15) {
    z <- rstudent(fit)
    hist(z, breaks=nbreaks, freq=FALSE,
    xlab="Studentized Residual",
    main="Distribution of Errors")
    rug(jitter(z), col="brown")
    curve(dnorm(x, mean=mean(z), sd=sd(z)),
    add=TRUE, col="blue", lwd=2)
    lines(density(z)$x, density(z)$y,
    col="red", lwd=2, lty=2)
    legend("topright",
    legend = c( "Normal Curve", "Kernel Density Curve"),
    lty=1:2, col=c("blue","red"), cex=.7)
}
```

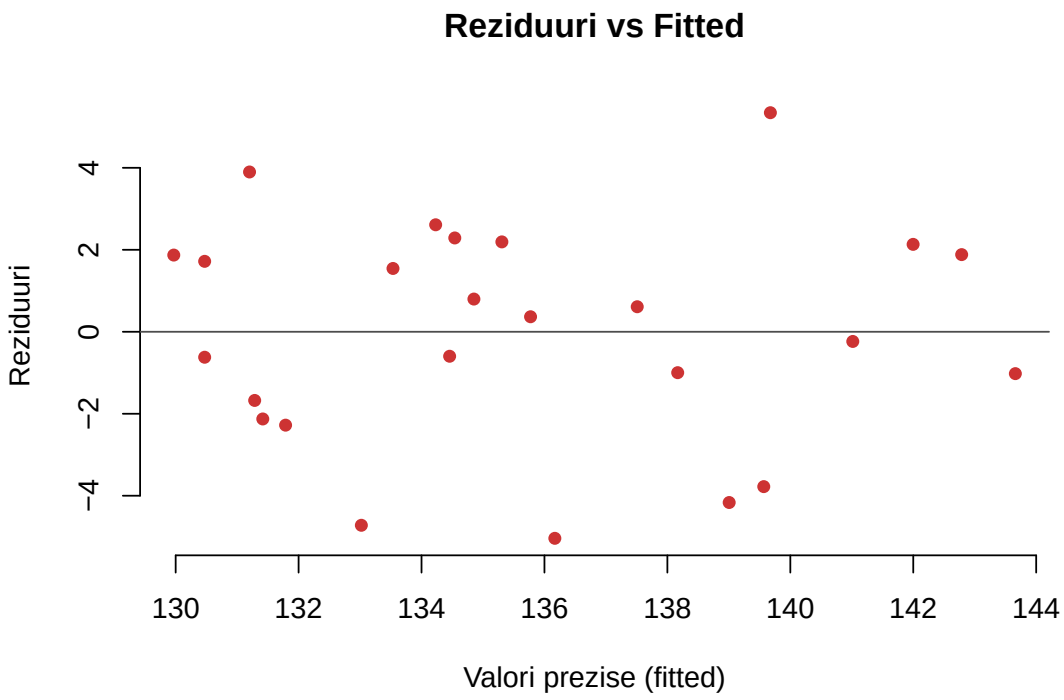


Figure 7: Reziduuri vs Valori prezise (Fitted)

```
residplot(saltBP_model)
```

- *Homoscedasticitatea*

Pentru a verifica proprietatea de homoscedasticitate a erorilor vom trasa un grafic al reziduurilor versus valorile prezise (fitted), i.e. $\hat{\varepsilon}$ vs $\hat{y}$. Dacă avem homoscedasticitate a erorilor atunci ar trebui să vedem o variație constantă pe verticală ($\hat{\varepsilon}$).

```
plot(residuals(saltBP_model)~fitted(saltBP_model),  col = "brown3", pch = 16,
     xlab = "Valori prezise (fitted)",
     ylab = "Reziduuri",
     main = "Reziduuri vs Fitted",
     bty = "n")

abline(h = 0, col = "grey30")
```

Tot în acest grafic putem observa dacă ipoteza de liniaritate este verificată (în caz de liniaritate între variabila răspuns și variabila cauză nu are trebui să vedem o relație sistematică între reziduuri și valorile prezise - ceea ce se și întâmplă în cazul nostru) ori dacă există o altă legătură structurală între variabila dependentă (răspuns) și cea independentă (predictor).

Curs Biostatistica 2017 - Laborator 9 & 10

Regresie

Contents

1 Regresie liniară multiplă

1.1 Introducere

Modelul de regresie liniară multiplă reprezintă o generalizare a modelului de regresie simplă. Dacă în regresia liniară simplă se folosea o singură variabilă predictor X ca să explice variabila răspuns Y, în modelul de regresie liniară multiplă se folosesc mai multe variabile predictor $X_1, \ldots, X_k$ pentru a explica răspunsul Y:

$$\mathbb{E}[Y|X_1 = x_1, \ldots, X_k = x_x] = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_k x_k$$

sau altfel scris

$$Y = \beta_0 + \beta_1 X_1 + \ldots + \beta_k X_k + \varepsilon.$$

Date fiind observațiile actuale, cu alte cuvinte dat fiind un eșantion $(X_{11}, \ldots, X_{1k}, Y_1), \ldots, (X_{n1}, \ldots, X_{nk}, Y_n)$ al lui $(X_1, \ldots, X_k, Y)$, unde X_{ij} reprezintă a i-a observație a predictorului X_j, modelul se poate scrie

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \ldots + \beta_k x_{ik} + \varepsilon_i, \quad i = 1, \ldots, n$$

a cărui formă compactă (matriceală) este

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

- $\mathbf{X}$ este *matricea de design*

$$\mathbf{X} = \begin{pmatrix} 1 & X_{11} & \cdots & X_{1k} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & X_{n1} & \cdots & X_{nk} \end{pmatrix}_{n \times (k+1)}$$

- $\mathbf{Y}$ este *vectorul răspuns*, $\boldsymbol{\beta}$ este *vectorul coeficienților* iar $\boldsymbol{\varepsilon}$ este *vectorul eroare*

$$\mathbf{Y} = \begin{pmatrix} Y_1 \\ \vdots \\ Y_n \end{pmatrix}_{n\times 1}, \quad \boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_k \end{pmatrix}_{(k+1)\times 1} \text{ și } \quad \boldsymbol{\varepsilon} = \begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{pmatrix}_{n\times 1}.$$

Să observăm că pentru $k = 1$ modelul se reduce la regresia liniară simplă. În acest caz:

$$\mathbf{X} = \begin{pmatrix} 1 & X_{11} \\ \vdots & \vdots \\ 1 & X_{n1} \end{pmatrix}_{n\times 2} \text{ și } \quad \beta = \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix}_{2\times 1}$$

Suma abaterilor pătratice reziduale pentru modelul de regresie liniară multiplă este

$$RSS(\boldsymbol{\beta}) = \sum_{i=1}^{n}(Y_i - \beta_0 - \beta_1 X_{i1} - \ldots - \beta_k X_{ik})^2 = (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})^T(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})$$

ceea ce conduce la *sistemul de ecuații normale*

$$\mathbf{X}^\intercal\mathbf{X}\hat{\boldsymbol{\beta}} = \mathbf{X}^\intercal\mathbf{Y}$$

a cărui soluție, dat fiind că $\mathbf{X}^\intercal\mathbf{X}$ este inversabilă, este

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{X}^\intercal\mathbf{Y}$$

Odată ce avem estimatorul $\hat{\boldsymbol{\beta}}$, putem defini:

- *valorile prognozate* (*fitted values*) $\hat{Y}_1, \ldots, \hat{Y}_n$ (valorile verticale pe hiperplanul de regresie), unde

$$\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 X_{i1} + \cdots + \hat{\beta}_k X_{ik}, \quad i = 1, \ldots, n$$

și sub formă matriceală

$$\hat{\mathbf{Y}} = \mathbf{X}\hat{\boldsymbol{\beta}} = \mathbf{X}(\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{X}^\intercal\mathbf{Y} = \mathbf{H}\mathbf{Y}$$

unde $\mathbf{H} = \mathbf{X}(\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{X}^\intercal$ se numește *matricea căciulă* (*hat matrix*) și reprezintă proiecția ortogonală a lui $\mathbf{Y}$ în spațiul generat de $\mathbf{X}$.

- *reziduurile estimate* (*estimated residuals*) $\hat{\varepsilon}_1, \ldots, \hat{\varepsilon}_n$, unde

$$\hat{\varepsilon}_i = Y_i - \hat{Y}_i, \quad i = 1, \ldots, n$$

și sub formă matriceală

$$\hat{\boldsymbol{\varepsilon}} = \boldsymbol{Y} - \hat{\boldsymbol{Y}} = (\boldsymbol{I} - \boldsymbol{H})\boldsymbol{Y}$$

Ipotezele modelului sunt:

i. **Linearitatea**: $\mathbb{E}[Y|X_1 = x_1, \ldots, X_k = x_k] = \beta_0 + \beta_1 x_1 + \ldots + \beta_k x_k$

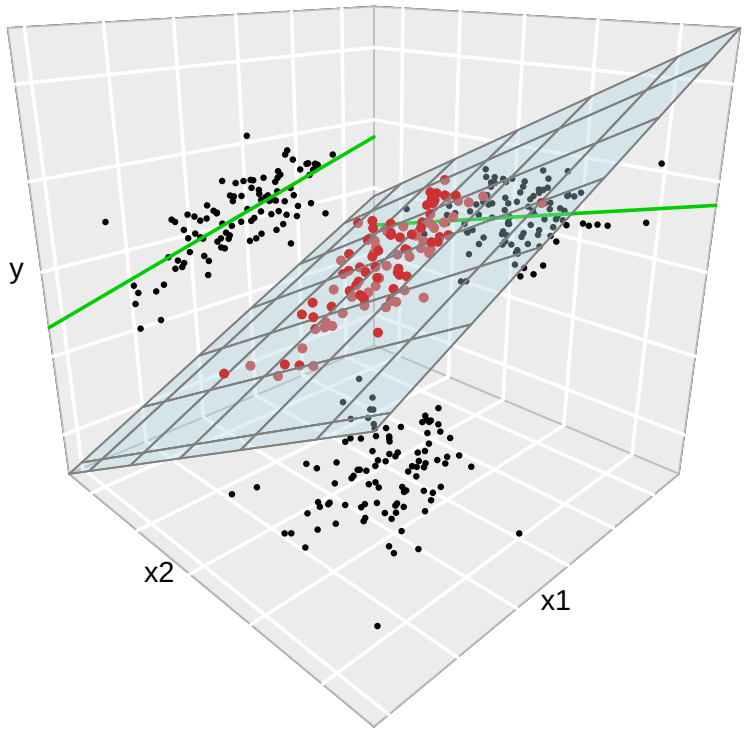


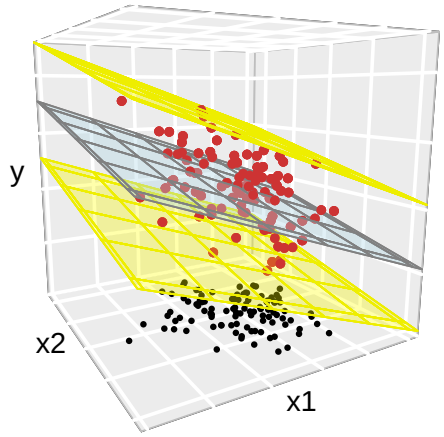
Figure 1: Planul de regresie (albastru) si relatia cu regresiile liniare simple (liniile verzi). Punctele rosii reprezinta un esantion pentru (X_1, X_2, Y) iar punctele negre sunt subesantioane pentru (X_1, X_2) (la baza), (X_1, Y) (stanga) si (X_2, Y) (dreapta).

ii. **Homoscedasticitatea**: $\mathbb{V}\text{ar}(\varepsilon_i) = \sigma^2$, cu σ^2 constantă pentru $i = 1, \ldots, n$
iii. **Normalitatea**: $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$ pentru $i = 1, \ldots, n$
iv. **Independența erorilor**: $\varepsilon_1, \ldots, \varepsilon_n$ sunt independente (sau necorelate, $\mathbb{E}[\varepsilon_i \varepsilon_j] = 0$, $i \neq j$, deoarece sunt presupuse normale)

Altfel spus

$$Y|(X_1 = x_1, \ldots, X_k = x_k) \sim \mathcal{N}(\beta_0 + \beta_1 x_1 + \ldots + \beta_k x_k, \sigma^2)$$

\begin{figure}



{

}

\caption{Planul de regresie. Spatiul dintre cele doua plane galbene arata unde se afla 95% din observatii (dupa modelul ales).} \end{figure}

Estimatorul pentru σ^2 este

$$\hat{\sigma}^2 = \frac{RSS(\hat{\beta}_0, \hat{\beta}_1, \ldots, \hat{\beta}_k))}{n - (k+1)} = \frac{\hat{\boldsymbol{\varepsilon}}^\intercal \hat{\boldsymbol{\varepsilon}}}{n - (k+1)} = \frac{\sum_{i=1}^n \hat{\varepsilon}_i^2}{n - (k+1)}.$$

1.2 Exemplul 1

Considerăm setul de date `galapagos` care conține informații despre numărul de specii de broaște țestoase din diferite insule din arhipelagul Galapagos (vezi articol). Setul conține date din 30 de insule despre numărul de specii de țestoase (`Species`), numărul de specii endemice (`Endemics`), suprafața insulei (`Area`), înălțimea maximă a insulei (‘`Elevation`), distanța la cea mai apropiată insulă (`Nearest`), distanța față de insula Snata Cruz (`Scruz`) și suprafața insulei adiacente (`Adjacent`). Vrem să investigăm relația liniară dintre numărul de specii și celelalte variabile.

Începem prin a citi datele

```
# gala = read.csv("data/galapagos.csv")

data("gala") # este nevoie de libraria faraway
head(gala)
```

```
##              Species Endemics  Area Elevation Nearest Scruz Adjacent
## Baltra            58       23 25.09       346     0.6   0.6     1.84
## Bartolome         31       21  1.24       109     0.6  26.3   572.33
## Caldwell           3        3  0.21       114     2.8  58.7     0.78
## Champion          25        9  0.10        46     1.9  47.4     0.18
## Coamano            2        1  0.05        77     1.9   1.9   903.82
## Daphne.Major      18       11  0.34       119     8.0   8.0     1.84
```

Considerăm modelul de regresie liniară multiplă cu 5 predictori:

```
gala_model = lm(Species ~ Area + Elevation + Nearest + Scruz + Adjacent, data=gala)

gala_model_summary = summary(gala_model)
gala_model_summary
```

```
##
## Call:
## lm(formula = Species ~ Area + Elevation + Nearest + Scruz + Adjacent,
##     data = gala)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -111.679  -34.898   -7.862   33.460  182.584
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.068221  19.154198   0.369 0.715351
## Area        -0.023938   0.022422  -1.068 0.296318
## Elevation    0.319465   0.053663   5.953 3.82e-06 ***
## Nearest      0.009144   1.054136   0.009 0.993151
## Scruz       -0.240524   0.215402  -1.117 0.275208
## Adjacent    -0.074805   0.017700  -4.226 0.000297 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 60.98 on 24 degrees of freedom
## Multiple R-squared:  0.7658, Adjusted R-squared:  0.7171
## F-statistic:  15.7 on 5 and 24 DF,  p-value: 6.838e-07
```

1.2.1 Estimarea parametrilor

Pentru început extragem matricea de design X

```
X = model.matrix( ~ Area + Elevation + Nearest + Scruz + Adjacent,
    data = gala)

head(X)
```

```
##              (Intercept)  Area Elevation Nearest Scruz Adjacent
```

```
## Baltra                 1 25.09       346      0.6   0.6     1.84
## Bartolome              1  1.24       109      0.6  26.3   572.33
## Caldwell               1  0.21       114      2.8  58.7     0.78
## Champion               1  0.10        46      1.9  47.4     0.18
## Coamano                1  0.05        77      1.9   1.9   903.82
## Daphne.Major           1  0.34       119      8.0   8.0     1.84
```

și răspunsul y

```
y = gala$Species
```

Vrem să găsim $\hat{\boldsymbol{\beta}} = (\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{X}^\intercal\mathbf{Y}$

```
# determinam (\mathbf{X}^\intercal\mathbf{X})^{-1}

xtxi = solve(t(X) %*% X) # t() - este transpusa
                         # %*% - produsul matriceal
                         # solve() - calculeaza pseudoinversa

bHat = xtxi %*% t(X) %*% y
bHat
```

```
##                      [,1]
## (Intercept)  7.068220709
## Area        -0.023938338
## Elevation    0.319464761
## Nearest      0.009143961
## Scruz       -0.240524230
## Adjacent    -0.074804832
```

```
# sau alternativ folosind ecuatiile normale
solve(crossprod(X,X), crossprod(X,y)) # crossprod calculeaza X^Ty
```

```
##                      [,1]
## (Intercept)  7.068220709
## Area        -0.023938338
## Elevation    0.319464761
## Nearest      0.009143961
## Scruz       -0.240524230
## Adjacent    -0.074804832
```

Estimatorul pentru σ^2 este dat de

```
sHat = sqrt(deviance(gala_model)/df.residual(gala_model))
sHat
```

```
## [1] 60.97519
```

```
# sau inca

gala_model_summary$sigma
```

```
## [1] 60.97519
```

Dacă vrem să determinăm erorile standard ale coeficienților, i.e. $\hat{\mathrm{SE}}(\hat{\beta}_i)$, să observăm pentru început că acestea sunt date de următoarea formulă

$$\hat{\mathrm{SE}}(\hat{\beta}_{i-1}) = \hat{\sigma}\sqrt{(\mathbf{X}^\intercal\mathbf{X})^{-1}_{ii}}$$

unde $(\mathbf{X}^\intercal\mathbf{X})^{-1}_{ii}$ reprezintă elementul i de pe diagonala matricii $(\mathbf{X}^\intercal\mathbf{X})^{-1}$.

```
seBHat = sHat*sqrt(diag(xtxi))
seBHat
```

```
## (Intercept)        Area   Elevation     Nearest       Scruz    Adjacent
## 19.15419782  0.02242235  0.05366280  1.05413595  0.21540225  0.01770019
```

```
# sau inca

gala_model_summary$coefficients[, 2]
```

```
## (Intercept)        Area   Elevation     Nearest       Scruz    Adjacent
## 19.15419782  0.02242235  0.05366280  1.05413595  0.21540225  0.01770019
```

1.2.2 Inferență asupra parametrilor

Având mai mulți predictori pentru o variabilă răspuns, ne întrebăm dacă avem nevoie de toți. Fie Θ spațiul parametrilor pentru un model mai mare și Θ_0 spațiul parametrilor pentru un model mai mic ($\Theta_0 \subset \Theta$). Dacă nu avem o diferență prea mare între concordanța celor două modele atunci îl preferăm pe cel mai simplu. Testul bazat pe raportul de verosimilități ($H_0 : \theta \in \Theta_0$ vs $H_1 : \theta \in \Theta$) conduce la respingerea ipotezei nule în cazul în care raportul

$$\frac{RSS_{\Theta_0} - RSS_\Theta}{RSS_\Theta}$$

este suficient de mare. Dacă spațiul parametrilor Θ are dimensiunea p (la noi $k+1$) iar spațiul parametrilor modelului redus Θ_0 are dimensiunea q atunci

$$F = \frac{\frac{RSS_{\Theta_0} - RSS_\Theta}{(p-q)}}{\frac{RSS_\Theta}{n-p}} = \frac{\frac{RSS_{\Theta_0} - RSS_\Theta}{(df_{\Theta_0} - df_\Theta)}}{\frac{RSS_\Theta}{df_\Theta}} \sim F_{p-q,n-p}.$$

unde $df_{\Theta_0} = n - q$ iar $df_\Theta = n - p$ (gradele de libertate sunt în general numărul de observații minus numărul de parametrii ai modelului).

a) Test asupra tuturor predictorilor

Să presupunem că vrem să testăm ipoteza nulă

$$H_0 : \beta_1 = \cdots = \beta_k = 0$$

cu alte cuvinte vrem să răspundem la întrebarea dacă vreuna din variabilele explicative este folositoare în prezicerea răspunsului. În această situație modelul (complet Θ) este $\boldsymbol{y} = \boldsymbol{X\beta} + \boldsymbol{\varepsilon}$ și are $k+1$ parametrii ($k+1$ coeficienți β_i) iar modelul redus (Θ_0) este $\boldsymbol{y} = \boldsymbol{\mu} + \boldsymbol{\varepsilon}$ și are 1 parametru (β_0). Prin urmare avem statistica F

$$F = \frac{\frac{RSS_{\Theta_0} - RSS_\Theta}{(k+1-1)}}{\frac{RSS_\Theta}{n-(k+1)}} = \frac{\frac{SS_T - RSS}{k}}{\frac{RSS}{n-(k+1)}} = \frac{\frac{SS_{reg}}{k}}{\frac{RSS}{n-(k+1)}} \sim F_{k,n-(k+1)}$$

unde RSS este suma abaterilor pătratice reziduale, $SS_T = (\boldsymbol{y} - \bar{\boldsymbol{y}})^\intercal(\boldsymbol{y} - \bar{\boldsymbol{y}})$ este suma abaterilor pătratice totale iar $SS_{reg} = SS_T - RSS$ este suma abaterilor de regresie, ceea ce conduce la tabelul ANOVA

	Df	SS	MS	F	p-value
Regresie	k	SS_{reg}	$\frac{SS_{reg}}{k}$	$F = \frac{SS_{reg}/k}{RSS/(n-(k+1))}$	p
Residuuri	$n-(k+1)$	RSS	$\frac{RSS}{n-(k+1)}$		
Total	$n-1$	SS_T			

Chiar dacă ipoteza nulă a fost respinsă asta nu înseamnă că modelul dat de alternativă este cel mai bun (nu știm dacă toți predictorii sunt necesari în model sau doar o parte dintre ei).

Pentru setul nostru de date să considerăm modelul nul (cel ce corespunde lui Θ_0)

```
gala_null_model = lm(Species ~ 1, data = gala)
```

Tabelul ANOVA este dat de

```
anova(gala_model, gala_null_model)
```

```
## Analysis of Variance Table
##
## Model 1: Species ~ Area + Elevation + Nearest + Scruz + Adjacent
## Model 2: Species ~ 1
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1     24  89231
## 2     29 381081 -5   -291850 15.699 6.838e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Observăm că ipoteza nulă este respinsă în acest caz în favoarea alternativei (p valoarea este aproximativ 6.8×10^{-7}).

Putem calcula această p-valoare și fără a apela la ajutorul funcției `anova`:

```
# pentru modelul redus
RSS0 = deviance(gala_null_model)
df0 = df.residual(gala_null_model)

# pentru modelul intreg
RSS = deviance(gala_model)
df = df.residual(gala_model)

# statistica F

Fstat = ((RSS0 - RSS)/(df0 - df))/(RSS/df)

1-pf(Fstat, df0-df, df)
```

```
## [1] 6.837893e-07
```

b) Test asupra unui predictor

Să presupunem acum că vrem să testăm dacă putem exclude din model un anumit predictor i (fixat). Prin urmare vrem să testăm ipoteza nulă

$$H_0 : \beta_i = 0$$

Considerăm modelul întreg Θ în care avem toți predictorii și modelul redus Θ_0 în care avem toți predictorii

mai puțin predictorul i (în cazul problemei noastre o să testăm să vedem dacă putem exclude sau nu variabila explicativă `Area`):

```
gala_Area_model = lm(Species ~ Elevation + Nearest + Scruz + Adjacent,
    data = gala)

anova(gala_model, gala_Area_model)
```

```
## Analysis of Variance Table
##
## Model 1: Species ~ Area + Elevation + Nearest + Scruz + Adjacent
## Model 2: Species ~ Elevation + Nearest + Scruz + Adjacent
##   Res.Df   RSS Df Sum of Sq      F Pr(>F)
## 1     24 89231
## 2     25 93469 -1   -4237.7 1.1398 0.2963
```

Observăm că nu putem respinge ipoteza nulă (p valoarea > 0.05).

O abordare alternativă constă în folosirea statisticii de test

$$t_i = \frac{\hat{\beta}_i}{\hat{\text{SE}}(\hat{\beta}_j)} \sim_{H_0} t_{n-k-1}$$

care verifică relația $t_i^2 = F$. Putem vedea statistica student în output-ul funcției `summary`:

```
gala_model_summary$coefficients
```

```
##                 Estimate  Std. Error      t value     Pr(>|t|)
## (Intercept)  7.068220709 19.15419782  0.369016796 7.153508e-01
## Area        -0.023938338  0.02242235 -1.067610554 2.963180e-01
## Elevation    0.319464761  0.05366280  5.953187968 3.823409e-06
## Nearest      0.009143961  1.05413595  0.008674366 9.931506e-01
## Scruz       -0.240524230  0.21540225 -1.116628222 2.752082e-01
## Adjacent    -0.074804832  0.01770019 -4.226216850 2.970655e-04
```

c) Test pentru o pereche de predictori

Să presupunem că vrem să testăm dacă suprafața insulei curente sau a insulei adiacente au vreo relație relativ la variabila răspuns. Prin urmare vrem să testăm ipoteza nulă (**să ținem cont că trebuie să specificăm care sunt toți predictorii !**)

$$H_0 : \beta_i = \beta_j = 0 \quad (\beta_{Area} = \beta_{Adjacent} = 0)$$

Putem testa această ipoteză folosind procedura descrisă anterior:

```
gala_Area_Adjacent_model = lm(Species ~ Elevation + Nearest + Scruz,
    data = gala)

anova(gala_Area_Adjacent_model, gala_model)
```

```
## Analysis of Variance Table
##
## Model 1: Species ~ Elevation + Nearest + Scruz
## Model 2: Species ~ Area + Elevation + Nearest + Scruz + Adjacent
##   Res.Df    RSS Df Sum of Sq      F  Pr(>F)
```

```
## 1     26 158292
## 2     24  89231  2     69060 9.2874 0.00103 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Observăm că ipoteza nulă este respinsă deoarece p-valoarea este mică (prin urmare excluderea celor doi predictori nu este justificată).

1.2.3 Intervale de încredere pentru parametrii

Cum repartiția lui $\hat{\boldsymbol{\beta}}$ este:

$$\hat{\boldsymbol{\beta}} \sim \mathcal{N}_{k+1}\left(\boldsymbol{\beta}, \sigma^2(\mathbf{X}^T\mathbf{X})^{-1}\right)$$

atunci estimatorul $\hat{\sigma}^2$ pentru σ^2 obținem că

$$\frac{\hat{\beta}_j - \beta_j}{\hat{\text{SE}}(\hat{\beta}_j)} \sim t_{n-(k+1)}$$

iar un interval de încredere de nivel $1-\alpha$ pentru parametrul β_j este

$$IC = \left(\hat{\beta}_j \pm \hat{\text{SE}}(\hat{\beta}_j) t_{n-2;\alpha/2}\right)$$

Putem construi intervale de încredere pentru parametrii folosind funcția `confint`:

```
confint(gala_model)
```

```
##                   2.5 %      97.5 %
## (Intercept) -32.4641006 46.60054205
## Area         -0.0702158  0.02233912
## Elevation     0.2087102  0.43021935
## Nearest      -2.1664857  2.18477363
## Scruz        -0.6850926  0.20404416
## Adjacent     -0.1113362 -0.03827344
```

Dacă vrem să construim o regiune de încredere pentru mai mult de un parametru atunci putem să folosim relația:

$$(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})^\intercal \boldsymbol{X}^\intercal \boldsymbol{X} (\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}) \leq (k+1)\hat{\boldsymbol{\sigma}}^2 F_{k+1,n-(k+1)}^{1-\alpha}$$

care reprezintă o regiune de încredere pentru $\boldsymbol{\beta}$.

De exemplu vrem să construim o regiune de încredere pentru perechea $(\beta_{Area}, \beta_{Adjacent})$:

```
plot(ellipse(gala_model, c(2,6)), type = "l", col = "grey30",
     xlab = "Area",
     ylab = "Adjacent",
     bty = "n")
points(0, 0, pch = 18, col = "grey50")
abline(v = confint(gala_model)[2,], lty = 2)
abline(h = confint(gala_model)[6,], lty = 2)
```

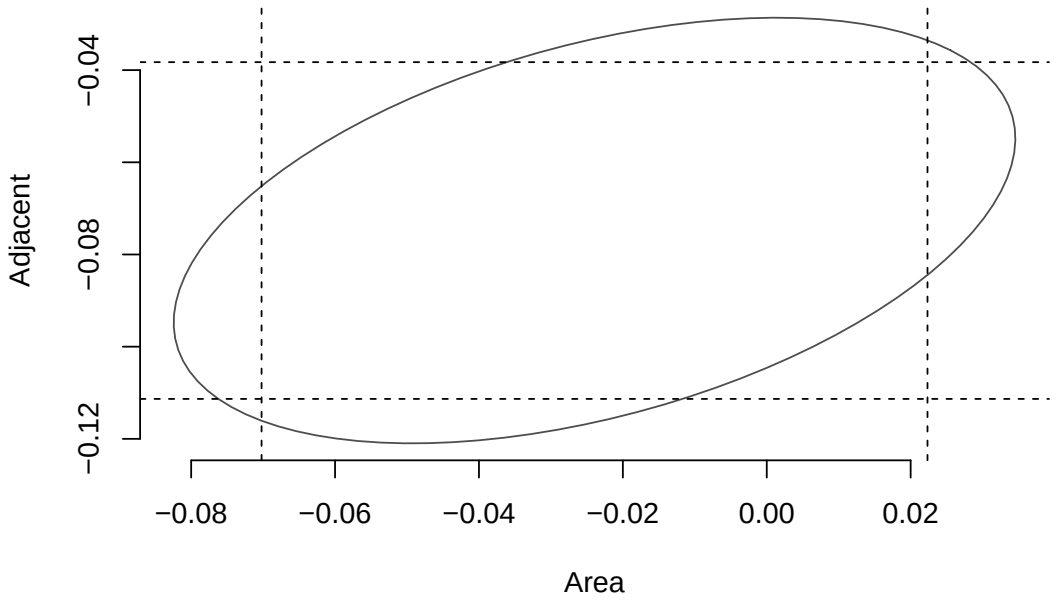


Figure 2: Regiune de incredere pentru Area si Adjacent

Cum punctul $(0, 0)$ nu aparține regiunii elipsoidale atunci putem respinge ipoteza nulă $H_0 : \beta_{Area} = \beta_{Adjacent} = 0$.