Lineaarsed mudelid
6. praktikum

Mudelite võrdlemine. F-test.

Arstitudengid alustavad sageli ravivõtete harjutamist esmalt mulaažide peal (enne katseloomade ja pärispatsientide juurde asumist). Alustame meiegi siis harjutamist ühe laest võetud andmestikuga – hoiame pärisandmed pärasteks, kui juba veidigi mõtteosavust on tekkinud. Oletame, et oleme mõõtnud inimeste pikkuseid kolmes maakonnas, kokku kuues vallas.

```
maakond=rep(c("Harjumaa","Tartumaa","Jõgevamaa"),c(5,4,4))
vald=rep(c("Viimsi","Kose","Kambja","Nõo","Põltsamaa","Mustvee"),
      c(3,2,2,2,2,2))
pikkus=c(210,190,185,180,176,176,180,156,166,167,156,155,171)

andmestik=data.frame(maakond, vald, pikkus)
andmestik
```

Hindame kaks mudelit (üks mudelitest on teise erijuht; üks on lihtsam ja teine mudel on „suurem“):

```
m1=lm(pikkus~1)
m2=lm(pikkus~factor(maakond))
```

Vaatame nende mudelite jääkide ruutude summasid:

```
sum(residuals(m1)**2)
sum(residuals(m2)**2)
```

$Y^T(I-P_{X1})Y$
$Y^T(I-P_{X2})Y$

Kas need kaks mudelit annavad sarnaseid prognoose? Kui prognoosides poleks märkimisväärset erinevust, võiksime ju kasutama jääda lihtsamat mudelit. Vaatame kahe mudeli prognooside summaarset ruuterinevust $Y^T(P_{X2} - P_{X1})Y$:

```
sum((predict(m2)-predict(m1))**2)
```

Märkus: Prognooside ruuterinevuste summa on sama, mis kahe erineva mudeli jääkide ruutude summade erinevus (lihtsam miinus keerulisem)

$$\begin{aligned} Y^T(P_{X2} - P_{X1})Y &= Y^T(-I + P_{X2} + I - P_{X1})Y \\ &= Y^T((I - P_{X1}) - (I - P_{X2}))Y \\ &= Y^T (I - P_{X1})\ Y - Y^T (I - P_{X2})Y \end{aligned}$$

Seega eelnenuga sama tulemuse oleksime saanud käsuga:

```
sum(residuals(m1)**2)-sum(residuals(m2)**2)
```

Arvutame järgnevalt kaks erinevat hinnangut jääkide dispersioonile. Üks neist (s1) on õige vaid siis, kui lihtsam mudel peab paika, teine on õige ka siis, kui vaid keerukam mudel on õige (s2):

```
s1=(sum(residuals(m1)**2)-sum(residuals(m2)**2))/(3-1)
s2=sum(residuals(m2)**2)/(13-3)

s1; s2
```

Nende kahe dispersioonihinnangu suhe ongi *F*-statistik, mille järgi otsustame mudelite käekäigu üle:

```
F=s1/s2
F
```

Kas meie *F* on liiga suur?
Nullhüpoteesi kehtides on *F*-statistiku jaotuseks (tsentraalne) *F*-jaotus. Vaatame, kui sageli (kui paljudes valimites) võiksime näha meie *F*-statistiku väärtusest väiksemaid *F*-statistiku väärtuseid:

```
pf(F, df1=3-1, df2=13-3)
```

Olulisustõenäosuseks on tõenäosus näha sedavõrd ekstreemset (sedavõrd suurt) või veel ekstreemsemat (veel suuremat) teststatistiku väärtust:

```
1 - pf(F, df1=3-1, df2=13-3)
```

Kontrolli oma tulemusi:

```
> anova(m1,m2)
Analysis of Variance Table

Model 1: pikkus ~ 1
Model 2: pikkus ~ factor(maakond)
  Res.Df    RSS Df Sum of Sq     F  Pr(>F)
1     12 2881.2
2     10 1242.5  2    1638.7 6.594 0.01492 *
```

Lihtsama mudeli jääkide ruutude summa $Y^T(I-P_{X1})Y$

Keerukama mudeli jääkide ruutude summa $Y^T(I-P_{X1})Y$

Erinevate dispersiooni-hinnangute suhe ehk *F*-statistik

Prognoosierinevuste ruutude summa (=jääkide ruutude summade erinevus)

Kui mitu hinnatavat parameetrit on keerukamas mudelis enam (p_2 - p_1)?

Ülesanne 1

Proovime, kas midagi jäi külge ka. Lisame järgmise mudeli, kus mängus ka vald:

```
m3=lm(pikkus~factor(maakond)+factor(vald))
```

Soovime võrrelda, kas võiksime piirduda vaid maakonda sisaldava mudeliga või peaksime eelistama mudelit, kus sees on ka vald.

Leia:

1. $Y^T(P_{X3} - P_{X2})Y$: .....

2. F-statistiku väärtus:

   F= ..... / ..... = ...

3. Mudelite m2 ja m3 võrdlemisel saadud p-väärtus:

   .....

Võrdle isearvutatud tulemusi käsu `anova(m2,m3)` tulemustega!

Käsud `anova` ja `drop1`.

Praktikas on sageli mugav teha kiiresti ja automatiseeritult läbi palju erinevaid mudelite omavahelisi võrdluseid. Käsk drop1 proovib mudelist kordamööda kõiki mõjusid eemaldada:

```
m3=lm(pikkus~factor(maakond)+factor(vald))
m4=lm(pikkus~factor(vald))
m5=lm(pikkus~factor(maakond))
m6=lm(pikkus~1)
```

```
> drop1(m3, test="F")
Single term deletions

Model:
pikkus ~ factor(maakond) + factor(vald)
               Df Sum of Sq    RSS    AIC F value Pr(>F)
<none>                      604.5 61.913
factor(maakond)  0      0.00  604.5 61.913
factor(vald)     3    638.05 1242.5 65.280  2.4628 0.1471
```

Kas mudel ilma maakonnata sobiks?
`anova(m3, m4)`

Kas mudel ilma vallata sobiks?
`anova(m3, m5)`

Ülesanne 2

Oskad sa seletada, miks käsu `anova(m3, m4)` väljund
On mõnevõrra ebaharilik (pole näiteks p-väärtust raporteritud)?
Vihje: mille poolest erinevad mudelid m3 ja m4?

Käsk anova proovib järgemööda lisada mudelisse mõjusid (selles järjekorras nagu nad on mudeli kirjelduses esitatud) kuni saavutatakse soovitud mudel.

NB! Teistmoodi! Mudelite m6 ja m5 võrdlust ei teha siin mitte käsuga `anova(m5, m6)` ehk ei arvutata teststatistikut *F* valemiga

```
sum((residuals(m5)-residuals(m6))**2)/2 /
    (sum(residuals(m5)**2)/10 )
```

Vaid kasutatakse järgmist arvutusvalemit:

```
sum((residuals(m5)-residuals(m6))**2)/2 /
   sum(residuals(m3)**2)/7
```

Kas oskad arvata, miks?

```
> anova(m3)
Analysis of Variance Table

Response: pikkus
                Df  Sum Sq Mean Sq F value  Pr(>F)
factor(maakond)  2 1638.68  819.34  9.4878 0.01016 *
factor(vald)     3  638.05  212.68  2.4628 0.14709
Residuals        7  604.50   86.36
```

anova(m5, m3)

Ülesanne 3

Vaata, mis juhtub, kui muuta mudelis seletavate tunnuste järjekorda, st kui kasutada mudeli

```
m3=lm( pikkus ~ factor(maakond) + factor(vald) )
```

asemel mudelit

```
m3a=lm( pikkus ~ factor(vald) + factor(maakond) )
```

Kas käskude `drop1(m3a, test="F")` või `anova(m3a)` väljundis midagi muutub? Miks?

Ülesanne 4

Ka summary-käsk raporteerib ühe *F*-testi tulemust. Millist kahte erinevat mudelit võrreldakse? Vaata lõigus **„Käsud `anova` ja `drop1`“** defineeritud mudeleid – kas leiad, milliseid neist on omavahel F-testi abil võrreldud saamaks summary-käsu poolt antava raporti lõpurida?

```
> summary(m3)

Call:
lm(formula = pikkus ~ factor(maakond) + factor(vald))


                         Estimate Std. Error t value Pr(>|t|)
(Intercept)               195.000      5.365  36.345  3.1e-09 ***
factor(maakond)Jõgevamaa  -33.500      8.483  -3.949  0.00554 **
factor(maakond)Tartumaa   -17.000      8.483  -2.004  0.08512 .
factor(vald)Kose          -17.000      8.483  -2.004  0.08512 .
factor(vald)Mustvee         1.500      9.293   0.161  0.87633
factor(vald)Nõo           -17.000      9.293  -1.829  0.11005
factor(vald)Põltsamaa          NA         NA      NA       NA
factor(vald)Viimsi             NA         NA      NA       NA
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 9.293 on 7 degrees of freedom
Multiple R-squared:  0.7902,    Adjusted R-squared:  0.6403
F-statistic: 5.273 on 5 and 7 DF,  p-value: 0.02515
```

anova(..... , ...)

Ülesanne 5

Vaatame järgmist näidet. Inimesi mõõdetakse kaks korda – üks tohter mõõdab inimese pikkuseid sentimeetrites, teine meetrites. Soovime leida mudelit, mis võimaldaks prognoosida inimese kaalu kasutades inimese pikkust.

Tekitame andmed (soovime ju kontrollida oma analüüsioskust ning avastada võimalikke probleeme enne pärisandmete kogumist):

```
# Genereerime andmed
set.seed(1)
n=100
pikkus=rnorm(n, mean=17, sd=1.5)
pikkusM=0.1*(pikkus+rnorm(n,sd=0.05))
pikkusCM=10*(pikkus+rnorm(n,sd=0.05))
kaal=10*pikkus-10+rnorm(n,5); rm(pikkus)

# Hindame mudeli
m=lm(kaal~pikkusM+pikkusCM)

# Võrdleme mudeleid
summary(m)
drop1(m, test="F")
anova(m)
```

Proovi vastata järgmistele küsimustele (maini kas vastuse leidsid summary, anova või drop1-käsu väljundist ja millisest kohast seal väljundis!):

1. Kas kaalu prognoosimisel on pikkusest abi (kas üleüldse on meil mudelis mõni kaalu prognoosimisel kasutamist vääriv tunnus)?
2. Teame inimese pikkust sentimeetrites. Kui me teaksime lisaks, kui pikk on inimene mõõdetuna meetrites, kas saaksime siis parema prognoosi tema kaalule?
3. Kas lisaks meetrites mõõdetud inimese pikkusele oleks hea teada tema pikkust mõõdetuna sentimeetrites?
4. Kas näeksime sarnast tulemust, kui mõõdaksime inimese pikkuse kõigest ühel korral ja arvutaksime sellest mõõtmisest tema pikkuse meetrites ja sentimeetrites?

Vastused:

1. Pikkusest on abi kaalu prognoosimisel: summary-käsu väljund, p-value: < 2.2e-16.

2. Täiendavalt teades inimese pikkust meetrites (lisaks pikkusele sentimeetrites) aitab kaalu täpsemalt prognoosida. Vaata drop1 käsk, rida pikkusM, p-value: 0.01211.

3. Täiendavalt teades inimese pikkust sentimeetrites aitab kaalu täpsemalt prognoosida. Vaata drop1 käsk, rida pikkusCM, p-value: 3.852e-05 või anova-käsk pikkusCM-rida.

4. Ei. Sellisel juhul oleks mudel, kus on sees inimese pikkus cm ja m sama mis mudel kus sees on vaid pikkus meetrites (samuti langeks ta kokku mudeliga kus sees on vaid pikkus sentimeetrites). Sama mudel siin tähenduses: mudeli veergudest moodustatud vektorruumid on samad ehk mudeli prognoosid/jäägid on samad.

Kuna kaalu prognoosimisel osutusid kasulikuks nii inimese pikkus sentimeetrites kui ka tema pikkus meetrites, siis otsustame lisada mudelisse ka inimese pikkuse kilomeetrites – äkki saame veel paremini kaalu prognoosiva mudeli?

Paraku pole meil takkajärgi kuskilt võtta inimeste mõõtmistulemusi kilomeetrites mõõdetuna. Arvutame need siis olemasolevate mõõtmistulemuste järgi:

```
pikkusK=(0.001*pikkusM+0.00001*pikkusCM)/2
```

Proovime hinnata nüüd ühe tõeliselt uhke ja hea mudeli inimese kaalu prognoosimiseks:

```
m_uhkeim=lm(kaal~pikkusM+pikkusCM+pikkusK)
anova(m_uhkeim)
drop1(m_uhkeim, test="F")
```

Käsu drop1 väljund näeb veidi imelik välja. Milles on asi?

Kui me aga prooviks alljärgnevat mudelit, siis kuidas nüüd tõlgendada drop1-käsu tulemust?

```
m_uhke=lm(kaal~pikkusM+pikkusK)
drop1(m_uhke, test="F")
```

Milliseid tunnuseid mainitutest – pikkusM, pikkusCM, pikkusK – kasutaksid lõppkokkuvõttes inimese kaalu prognoosimiseks? Milline võiks välja näha optimaalne kaalu prognoosiv mudel?

Mis oleks meie esialgsest tulemusest saanud, kui oleksime pikkust mõõtnud täpsemalt?

```
# Genereerime andmed
set.seed(1)
n=100
pikkus=rnorm(n, mean=17, sd=1.5)
pikkusM=0.1*(pikkus+rnorm(n,sd=0.01))
pikkusCM=10*(pikkus+rnorm(n,sd=0.01))
kaal=10*pikkus-10+rnorm(n,5); rm(pikkus)

# Hindame mudeli
m=lm(kaal~pikkusM+pikkusCM)

# Võrdleme mudeleid
drop1(m, test="F")
```

Miks nüüd pikkused (ei mõõdetuna sentimeetrites ega mõõdetuna meetrites) enam statistiliselt oluliseks ei osutu?

Näide

Mudeli otsimine tundmatu seose modelleerimiseks (NB! Hilisemates praktikumides/loengutes tutvume paremate strateegiatega selle ülesande lahendamiseks!!!)

```
# Algandmed

set.seed(1)
x=runif(100)*100;
y=15*log(x)+10*sin(x/18)+rnorm(10, sd=5)
plot(x,y)

# Algne mudel
m=lm(y~x+I(x**2)+I(x**3)+I(x**4)+I(x**5)+I(x**6)+I(x**7)+I(x**8)+I(x**9))
anova(m)

# Prognoosid
m2=lm(y~x+I(x**2)+I(x**3)+I(x**4))
xx=seq(0, 100, length=1000)
yy=predict(m2, data.frame(x=xx))
lines(xx,yy, col="red3", lwd=3)

# Soovi korral lisa joonisele tegelik seos:
tegelik=15*log(xx)+10*sin(xx/18)
lines(xx, tegelik, col="pink", lwd=2)
```

Arva, mis põhjusel on siin näites õppejõud kasutanud anova-käsku drop1-käsu asemel?
Muuseas – sarnaseid p-väärtuseid on võimalik summary-käsu abil saada siis, kui kasutad mudeli loomisel poly-käsku ilma raw=TRUE parameetrita:

```
m2=lm(y~poly(x, 9))
summary(m2)
```

Märka, et raw=TRUE variandi kasutamise korral saad märksa teistsugused stat. olulisused:

```
m2a=lm(y~poly(x, 9, raw=TRUE))
summary(m2a)
```

Ülesanne 6

Reaalsete andmete analüüsimisel armastavad inimesed sageli mudelit valida järgmisel viisil: otsitakse välja mõni statistiliselt oluline tunnus ja pistetakse see mudelisse. Seejärel vaadatakse, kas leitakse veel mõni statistiliselt oluline tunnus, mida võiks veel mudelisse lisada. Sellisel viisil jätkatakse, kuni enam ühtki statistiliselt olulist tunnust pole saadaval. F-testi teoreetilist tuletuskäiku vaadates peaksime märkama aga ühte väga suurt probleemi – teoreetik peaks taolise praktikute poolt armastatud lähenemisviisi täielikult maha laitma. Oskad ehk seletada, milles peitub probleem?

Märkus: On muidugi võimalik, et selline ebakorrektne F-testi kasutamine võib eksitada ning otsija juhatada väga vale mudeli juurde. Praktikas aga esineb suuri probleeme antud lähenemist kasutades suhteliselt harva – see on natuke nagu turvavööta autos sõitmine (enamasti jõuad ka turvavööta sõites elusana sihtpunkti).

Näide loengumaterjalis
Proovime järgnevalt läbi teha ka loengumaterjalides lk 47-48 esitatud näite.
Andmed (mõõdetud signaali tugevus üle taustafooni, ajahetked 1..7):

```
y=c(2,-10, 5, -4, 8, 89, 60)
```

Teame, et ajahetkedel 7 ja 8 edastati informatsiooni ja ajahetkedel 1,2,3 kindlasti ei edastatud. Küsimus on selles, kas ajavahetkedel 4 või 5 ka edastati midagi või mitte?

Lihtne mudel: signaali edastati vaid ajahetkedel 6,7:

```
X0 = cbind( t6=c(0,  0, 0,  0, 0,  1,  0),
            t7=c(0,  0, 0,  0, 0,  0,  1))
summary(lm(y~X0-1))
```

Keerukam mudel – signaali edastati ka ajahetkedel 4,5:

```
X1 = cbind( t4=c(0,  0, 0,  1, 0,  0,  0),
            t5=c(0,  0, 0,  0, 1,  0,  0),
            t6=c(0,  0, 0,  0, 0,  1,  0),
            t7=c(0,  0, 0,  0, 0,  0,  1))
summary(lm(y~X1-1))
```

Defineerime maatriksid P_{X0} ja P_{X1}:

```
PX0 = X0%*%solve(t(X0)%*%X0)%*%t(X0)
PX1 = X1%*%solve(t(X1)%*%X1)%*%t(X1)
```

Vaata järgmiseid vektoreid:

```
PX0%*%y
(PX1-PX0)%*%y
(diag(7)-PX1)%*%y
```

Kas mõistad, miks F-statistiku arvutamiseks vajalike hajuvuse hindamisel jagatakse vektorite $(\mathbf{P_{X1}} - \mathbf{P_{X0}})\mathbf{y}$ ja $(\mathbf{I} - \mathbf{P_{X1}})\mathbf{y}$ ruutpikkuseid vastavate vektorruumi dimensioonidega?