

Lineaarsed mudelid Ikka veel mitmesest testimisest ...

Loeme arvutisse tänase praktikumi andmestiku:

```
andmed=read.csv2(url("http://www.ms.ut.ee/mart/linmud2018/psych.csv"),
  header=TRUE)
head(andmed)

attach(andmed)
```

Andmestikus on järgmised tunnused:

Avatus – isiksuseomadus. Tsitaat webist: „**Avatud** inimesed on, nagu nimigi ütleb, avatud uutele ideedele ja kogemustele. Nad on väga uudishimulikud, loomingulised, laia silmaringiga ja originaalsete mõtetega. Neid iseloomustab teistest veidi erinev mõtlemine, autonoomia ja nad tahavad olla sõltumatud ning armastavad ka vaheldust.“ (www.mtk.ut.ee/doc/OKmka2004Aidlaseminar.doc)

Vanus, Sugu, Haridus – inimese vanus, sugu, haridus (alg/kesk/keskeri/kõrg).

Huvitume, kas eri haridust omandanud inimestel on avatus on erinev.

```
m1=lm(Avatus~Haridus)
drop1(m1, test="F")
```

Näeme, et tunnuse haridus väärtuste teadmisest on kasu inimese avatuse prognoosimisel. Aga milliste haridustasemete puhul saame ikka rääkida tõestatavalt erinevast avatusest?

Kasutame keskväärtuste võrdlemiseks Tukey-Kramer'i meetodit:

```
TukeyHSD(aov(m1), "Haridus")
```

või tulemused joonisena esitatult:

```
plot(TukeyHSD(aov(m1), "Haridus"))
```

NB! TukeyHSD-käsk eeldab, et faktortunnus on faktortunnuseks tehtud enne valemisse panemist. Kui oleksime hinnanud mudeli käsuga

```
m1=lm(Avatus~factor(Haridus))
```

siis poleks TukeyHSD-käsk meile soovitud vastust andnud!

Süüvime hetkeks TukeyHSD-käsu abil tehtud arvutuste detailidesse. Rehkendame saadud tulemused üle ühe võrdluse – näiteks alghariduse ja keskhariduse võrdluse jaoks.

Vaata hinnatud parameetrite järjekorda kas summary või coef-käsu abil:

```
summary(m1)
```

	Estimate	Std. Error	t value	Pr(> t)	
(Intercept)	86.075	2.941	29.267	< 2e-16	***
Hariduskesk	10.474	3.677	2.848	0.00470	**
Hariduskeskeri	9.799	3.430	2.857	0.00459	**
Hariduskõrg	17.728	3.633	4.879	1.75e-06	***

Paneme kirja lineaarkombinatsiooni, mis võrdleb alg-ja keskharidust,

$E(\text{Avatus}|\text{Keskharidus}) - E(\text{Avatus}|\text{Algharidus})$:

```
lambda1=c(0,1,0,0)
```

```
lambda1%%coef(m1)
```

Või täpsema ülevaate antud võrdluse jaoks saame estimable-käsuga (lisamoodulist gmodels):

```
library(gmodels)
lambda1_T_beeta=estimable(m1, lambda1)
lambda1_T_beeta
```

	Estimate	Std. Error	t value	DF	Pr(> t)
(0 1 0 0)	10.4743	3.677299	2.848367	294	0.004704426

Tukey-Krameri testi idee kohaselt võetakse vastu alternatiivne hüpotees siis, kui

$$\frac{\bar{Y}_i - \bar{Y}_j}{\sqrt{\hat{D}(\bar{Y}_i - \bar{Y}_j)}} \sqrt{2} > Q_{1-\alpha; k; df},$$

Kus $Q_{1-\alpha}$ on Tukey haardejaotuse $(1-\alpha)$ -kvantiil. Seega saame Tukey-Krameri testi teststatistiku leida käsuga:

```
lambda1_T_beeta[1,3]*sqrt(2)
```

ja vastava p-väärtuse käsuga:

```
1-ptukey(lambda1_T_beeta[1,3]*sqrt(2), 4, 294)
```

Samaaegsed 95%-usaldusintervallid aga saame leida käsuga:

```
lambda1_T_beeta[1,1]+c(-1,1)*qtukey(0.95, 4, 294)*
lambda1_T_beeta[1,2]/sqrt(2)
```

Võrdle saadud tulemusi TukeyHSD-käsu poolt saadud tulemustega!

Antud juhul on anmestik üsnagi mittetasakaaluline (mõne haridustasemega inimesi on palju, mõne haridustasemega – näiteks algharidusega – inimesi on aga vähe...):

```
table(Haridus)
```

Seega üritame vaadelda ka alternatiivset meetodit, mis töötaks sõltumata sellest, kui tasakaaluline on algne andmestik. Selleks teeme natuke eeltööd.

Esimene ülesanne: installeerige ja võtke kasutusse kolm erinevat lisamoodulit:

```
library(multcomp)
library(gmodels)
library(mvtnorm)
```

Teeme valmis kõik soovitud võrdlused:

Keskharidusega inimeste ja algharidusega inimeste keskmiste erinevus:

```
predict(m1, data.frame(Haridus="kesk")) -
  predict(m1, data.frame(Haridus="alg"))
l1=c(0,1,0,0)
estimable(m1, l1)
```

Kõigi teiste keskmiste omavahelised võrdlused:

```
# E keskeri - E alg:
l2=c(0,0,1,0)
estimable(m1, l2)

# E kõrg - E alg:
l3=c(0,0,0,1)
estimable(m1, l3)

# E keskeri - E kesk:
l4=c(0,-1,1,0)
estimable(m1, l4)

# E kõrg - E kesk:
l5=c(0,-1,0,1)
estimable(m1, l5)

# E kõrg - E keskeri:
l6=c(0,0,-1,1)
estimable(m1, l6)
```

Kõik need võrdlused on üksiktestid – iga üksiku testi puhul võime I liiki viga teha tõenäosusega 5% (kui kasutame olulisuse nivood 0,05). Mingit mitmese testimise korrektsiooni pole teostatud.

Kuidas öelda R-le, et kõigi testide peale kokku ei tohi I-liiki viga olla suurem kui 5%? Seda saab teha käsuga `glht` (lisamoodul `multcomp`):

```
vordlused=rbind(l1,l2,l3,l4,l5,l6)
rownames(vordlused)=c("kesk-alg","keskeri-alg",
  "kõrg-alg","keskeri-kesk",
  "kõrg-kesk","kõrg-keskeri")
mitmenev6rdlus=glht(m1, linfct=vordlused)

summary(mitmenev6rdlus)
confint(mitmenev6rdlus)
plot(mitmenev6rdlus)
```

Vahel tuleb tõepoolest kõik kontrollitavad võrdlused ise ükshaaval sisestada. Aga mõnikord saab soovitud võrdluseid ka automaatselt tekitada:

```
mitmenev6rdlus2=glht(m1, linfct=mcp(Haridus="Tukey"))
summary(mitmenev6rdlus2)
```

või

```
mitmenev6rdlus3=glht(m1, linfct=mcp(Haridus="Dunnett"))
summary(mitmenev6rdlus3)
```

Sageli kasutatakse ka Bonferroni või Bonferron-Holmi meetodit mitmeste võrdluste tegemiseks. Seega proovi ka järgmiseid käske:

```
estimable(m1, vordlused)
pvalues=estimable(m1, vordlused)[,5]
pvalues

p.adjust(pvalues, method="holm")
p.adjust(pvalues, method="bonferron")
```

Pane kirja ja võrdle:

Keskharidusega inimeste keskmine avatus vs algharidusega inimeste keskmine avatus

Korrigeerimata p-väärtus	Tukey-Krameri meetod p-väärtus	mitmemõõtmeline t-jaotus	
		kõik võrdlused p-väärtus	Võrdlus referentstasemega
.....			
Bonferroni meetodil korrigeeritud p-väärtus		Bonferron-Holmi meetodil korrigeeritud p-väärtus	
.....			

Ülesanne

Vaatasime enne vägagi lihtsat mudelit. Paneme oma võimed proovile veidi keerulisemas olukorras. Hindame näiteks järgmise mudeli:

```
m2=lm(Avatus~Vanus+Sugu+Haridus+Vanus*Haridus)
drop1(m2, test="F")
summary(m2)
```

Soovime teada, kas erineva haridusega inimeste keskmine avatus on erinev. Kas on tõestatavat erinevust inimeste avatuses 25 aasta vanuste eri haridusega inimeste vahel? Aga 70 aasta vanuste inimeste puhul?

Mis toimub glht-käsu sees ehk mitmese testimise seos mitmemõõtmelise t -jaotusega.

Pöördume tagasi lihtsama mudeli $m1$ juurde ja üritame iseseisvalt kontrollida glht-arvutusi. Kui oled vahepeal muutnud vektorite $l1$ - $l6$ väärtuseid, siis taasta $lk2$ antud programmi jooksumises vanad, mudeli $m1$ jaoks sobivad väärtused.

Testitavad hüpoteesid:

```
Lambda=rbind(l1,l2,l3,l4,l5,l6)
```

Testitavate hüpoteeside omavaheline kovariatsioonimaatriks:

```
# lin. komb. hinnangute dispersioonimaatriksi hinnang
# ehk  $\Lambda\beta$  hinnangu dispersioonimaatriksi hinnang:
sigma=Lambda*%vcov(m1)*%t(Lambda)
```

Mitmemõõtmelise t -jaotuse korral peavad dispersioonid olema kõik ühesugused (vähemalt osade käsitletute kohaselt), seega peame „korrigeerima“:

```
M=diag(1/sqrt(diag(sigma)))
sigma2=M*%sigma*%t(M)
sigma2
```

Esimese kontrollitava hüpoteesi korral milline oli t -väärtus:

```
estimable(m1, l1)
```

Näeme, et $t = 2.848367$. Milline on tõenäosus (juhul kui kõigi kontrollitavate hüpoteeside korral kehtib H_0), et kõigi 6 teststatistiku väärtused jäävad vahemikku $-2.84...2.84$?

```
pmvt(lower=rep(-2.848367,6), upper=rep(2.848367,6),
      df=294, sigma=sigma2)
```

Milline on tõenäosus, et mõni 6-st testist saab (nullhüpoteesi kehtides) tulemuseks t -statistiku, mis ei jää vahemikku $-2.84...2.84$?

```
1-pmvt(lower=rep(-2.848367,6), upper=rep(2.848367,6),
      df=294, sigma=sigma2)
```

Võrdle saadud tulemust glht-testi tulemusega!

```
summary(glht(m1, linfct=Lambda))
```

T -statistiku väärtuse (ebatäpse) ümberkirjutamise asemel võid t -statistiku väärtuse salvestada ja kasutada salvestatud väärtust::

```
tvaartus=estimable(m1, l1)[1,3]
1-pmvt(lower=rep(-tvaartus,6), upper=rep(tvaartus,6),
      df=294, sigma=sigma2)
```

Muuseas – samu käske uuesti käivitades võid saada veidi erineva tulemuse – milles asi?