

# Biomeetria

## 6. praktikum

Loeme sisse andmestiku fishcatch.dat:

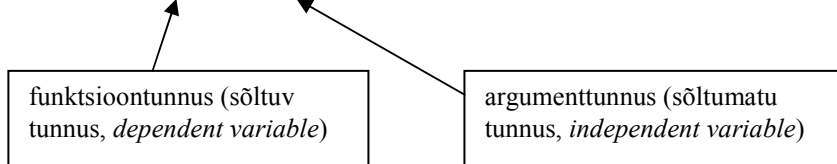
1. `andmed=read.table("http://www.ms.ut.ee/mart/biomeetria2009/fishcatch.dat", header=TRUE)`
2. Lühikeste tunnuste nimede kasutamiseks anna käsk  
`attach (andmed)`

## Regressioonanalüüs

Anna-Liisa tegi latikate elu kirjeldavat loodusfilmi. Peale veeluste kaadrite filmimist hakati tavainimese jaoks sobivaid selgitusi lisama. Telekanali esindaja, vana kalamees Kalamees soovis tungival, et ühe eriti uhke latika kaalu ka selgitavas tekstis mainitakse. Kuna filmitähest latikas oli juba ammu tont teab kuhu ujunud (või nahka pandud), tuli kala kaal kuidagi kaudsel viisil välja nuputada. Õnneks oli filmitud kaadrite pealt võimalik mõõta latika pikkust ja laiust. Anna-Liisa otsustaski kaalu prognoosida kas latika pikkuse või laiuse järgi (kumb iganes neist täpsema prognoosi annab), kasutades prognoosiva mudeli loomiseks Laenelmavesi järvest püütud latikate andmeid.

Mudeli loomine (latikate jaoks on tunnus *Species* väärtus 1):

```
> mudel=lm(Weight~Length3, data=andmed[Species==1,])
```



```
> mudel
```

Call:

```
lm(formula = Weight ~ Length3, data = andmed[Species == 1, ])
```

Coefficients:

```
(Intercept)      Length3  
-1194.40         47.37
```

Kaalu prognoosiv mudel on

$$\text{Kaal} = -1194,4 + 47,37 \cdot \text{Pikkus} + \epsilon$$

Seega 30 cm pikkuse latika kaaluks prognoosib mudel  $-1194,4 + 47,37 \cdot 30 = 226,7\text{g}$ , ehk teisisõnu – 30cm pikkuste latikate keskmine kaal on 30 cm.

```
> summary(mudel)
```

```
Call:
```

```
lm(formula = Weight ~ Length3, data = andmed[Species == 1, ])
```

```
Residuals:
```

Min	1Q	Median	3Q	Max
-101.671	-29.643	-8.777	28.855	176.486

```
Coefficients:
```

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	-1194.395	89.815	-13.30	8.29e-15 ***
Length3	47.369	2.328	20.34	< 2e-16 ***

```
---
```

```
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 56.45 on 33 degrees of freedom
```

```
Multiple R-Squared: 0.9261, Adjusted R-squared: 0.9239
```

```
F-statistic: 413.8 on 1 and 33 DF, p-value: < 2.2e-16
```

Determinatsioonikordaja  $R^2=0,9261$ .  
Mudel prognoosib kaalu küllaltki täpselt.

Milliste pikkuste jaoks prognoose soovime

Prognoosime erinevate pikkustega latikate kaale:

```
> prognoos=predict(mudel, data.frame(Length3=c(20,30,50)))
```

```
> prognoos
```

1	2	3
-247.0199	226.6678	1174.0432

20 cm pika latika kaaluks prognoositi -247 g (Näide sellest, et väljapoole olemasolevate andmete piire üldiselt prognoosida ei tohiks)

30 cm pika latika kaaluks prognoositi 226,7 g (Mis paistab juba pisut mõistlikum)

Saadud tulemused võib esitada graafiliselt. Kõigepealt joonistame latikate pikkuste ja kaalude hajuvusgraafiku:

```
plot(Length3[Species==1], Weight[Species==1],  
     main="Latika kaalu prognoosimine", xlab="Kala pikkus (cm)",  
     ylab="Kala kaal (g)")
```

Lisame saadud joonisele regressioonsirge. Seda on võimalik teha kahel moel. Lihtsaim viis (mida aga ei saa kasutada keerukamate mudelite puhul) oleks käsuga `abline(mudel)`. Teine võimalus (ja ehk veidi paremini üldistatav ka keerukamatele juhtudele) on kasutada `lines` ja `predict`-käsk:

```
lines(c(20,30,50), prognoos)
```

Saadud graafikule võime lisada ka usaldus- ja prognoosiintervalli. Selleks kasutame järgmiseid käsk:

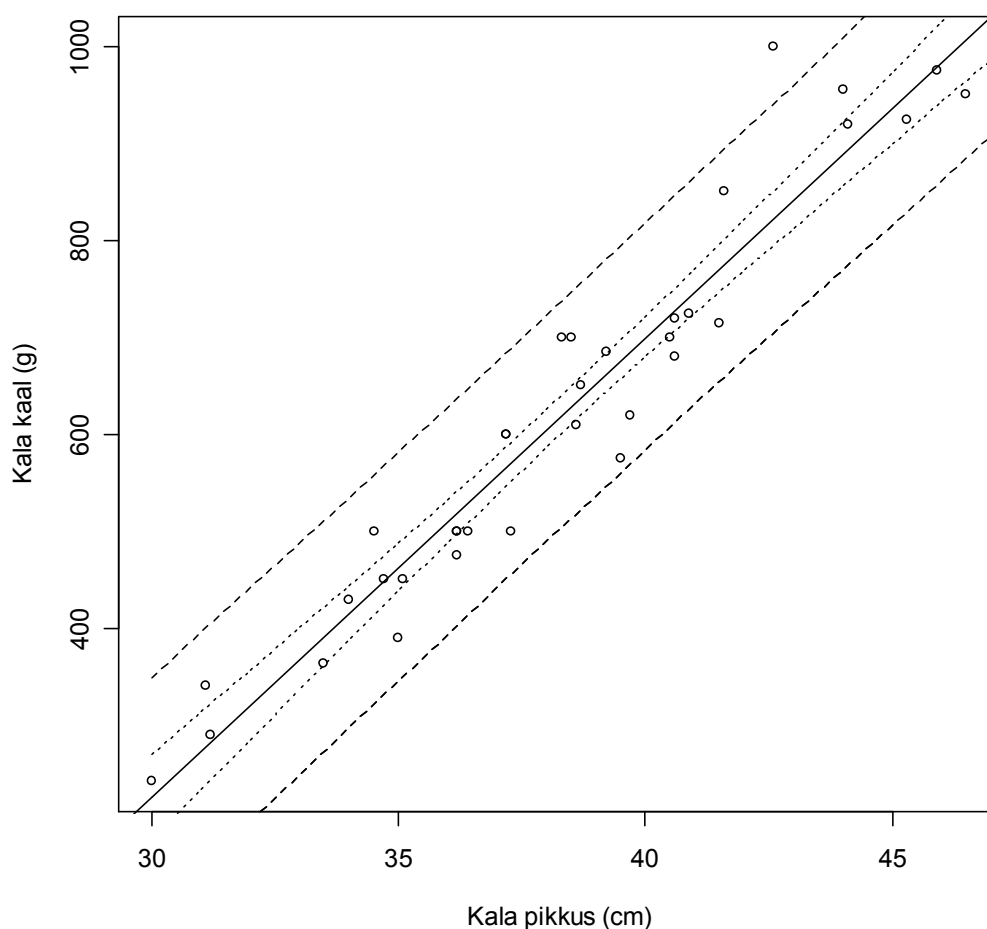
Prognosiintervalli lisamine (leiame prognoosid latikatele pikkusega 30cm, 31cm, ..., 50cm ja kanname leitud prognoosid ning prognoosiintervallid joonisele):

```
prognoos=predict(mudel, data.frame(Length3=c(30:50)),  
interval="prediction")  
lines(30:50,prognoos[,2],lty=2)  
lines(30:50,prognoos[,3],lty=2)
```

Usaldusintervalli lisamine:

```
prognoos=predict(mudel, data.frame(Length3=c(30:50)),  
interval="confidence")  
lines(30:50,prognoos[,2],lty=3)  
lines(30:50,prognoos[,3],lty=3)
```

### Latika kaalu prognoosimine



Konkreetsel latika kaal, keda filmisime (pikkus 44cm) on seega suure tõenäosusega vahemikus 770g...1009g, kaalu prognoos 889,8 e. ligikaudu 900g ehk ligikaudu kilo ehk ligikaudu ... :

```
> predict(mudel, data.frame(Length3=44), interval="prediction")  
      fit      lwr      upr  
[1,] 889.8305 770.3174 1009.344
```

## Eelduste kontroll

Paljude saadud tulemused on usaldusväärsed vaid siis, kui teatavad eeldused on täidetud. Põhilised eeldused, millele tähelepanu pöörata:

1. Kas sirge ikka sobib kaalu kasvamist seletama?
2. Kas mudeli jäägid on (ligikaudu) normaaljaotusega (kui ei, siis on valed leitud prognoosiintervallid, väikese valimi korral võivad kaheldavaks osutada ka hüpoteeside kontrolli osa ja usaldusintervallid)?
3. Kas jääkide hajuvus on ligikaudu konstantne (kui ei, siis prognoosiintervall pole õige ja mudeli parameetrite hindamisel pole olemasolevat informatsiooni mitte kõige paremini kasutatud – saaks täpsemalt...)?

Esimesele küsimusele saime vastuse juba joonistatud graafiku abil. Sageli kasutatakse ka mudeli jääkide ja prognoosiva tunnuse hajuvusgraafikut vaatamaks, kas seos tunnuste vahel on keerukam kui lineaarne (meetod väärib tundmaõppimist sest keerukamate mudelite korral osutub ta üsna kasulikuks):

```
plot (Length3[Species==1], resid(mudel))
```

Samalt graafikult võib kontrollida ka kolmanda eelduse paikapidavust. Mida teha, kui vaatlused ei paikne sirgel? Lihtsaid lahendusi on kaks:

1. teisenda kasutatud tunnuseid. Sageli võib aidata funktsioontunnuse (või nii sõltuva- kui sõltumatu tunnuse) logaritmine;
2. Teine võimalus on kaasata mudelisse argumenttunnuse kõrgemad astmed, st prognoosida  $y$ -tunnust kasutades polünoomi:  $y = c_0 + c_1 x + c_2 x^2 + c_3 x^3 + \dots + e$ . Kõrgemaid astmeid saame mudelisse kaasata R'is järgmisel moel:

```
mudel3=lm (Weight~Length3+I (Length3^2), data=andmed[Species==1,])
summary (mudel3)
predict (mudel3, data.frame (Length3=c (20, 30)))
```

või, saadud regressioonseose kujutamine joonisel:

```
x=seq (8, 50, 0.1)
plot (Length3[Species==1], Weight[Species==1])
y=predict (mudel3, data.frame (Length3=x))
lines (x, y)
```

NB! Lisa ise joonisele usaldusintervall regressioonkõverale ja 95%-tolerantsiintervall!

Antud juhul – kuna seos on tegelikult (peaaegu) lineaarne – ei muutu joonis pikkuse kõrgemate astmete lisamisel mudelisse kuigivõrd. Teistsuguste andmete puhul võib aga muutus osutada vägagi märkimisväärseks.

Jääkide normaaljaotust saab teadagi kontrollida tõenäosuspaberi (normaaljaotusgraafika) abil:

```
qqnorm (resid (mudel))
qqline (resid (mudel))
```

Tulemus on enam-vähem rahuldav (aga miks üks latikas nii eriline on? Ega tegemist pole sisestusveaga?). Kes ta üldse selline on? *Identify*-käsk võimaldab meil joonisel punkte hiirega klõpsida ja nende kohta informatsiooni saada (lõpetamiseks vajuta *Escape*'i):

```
identify (qqnorm (resid (mudel)))
andmed[Species==1,][30,]
```

### Parima prognoosiva tunnuse otsimisest:

Äkki saab aga kaalu kala kõrguse järgi paremini määrata?

```
> mudel2=lm(Weight~Height, data=andmed[Species==1,])
> summary(mudel2)
```

Call:

```
lm(formula = Weight ~ Height, data = andmed[Species == 1, ])
```

Residuals:

```
      Min       1Q   Median       3Q      Max
-362.72 -136.45   19.91  124.00  431.55
```

Coefficients:

```
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -1904.80     749.96  -2.540  0.01598 *
Height       63.94      18.96   3.373  0.00191 **
```

---

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Residual standard error: 179.1 on 33 degrees of freedom

Multiple R-Squared: 0.2563, Adjusted R-squared: 0.2338

F-statistic: 11.37 on 1 and 33 DF, p-value: 0.001915

Determinatsioonikordaja on märgatavalt väiksem,  $R^2=0,2363$ . Kala kõrgus on peaaegu kasutu kala kaalu prognoosimisel.

Näeme mudeli viletsust ka sellest, et tolerantsiintervall tuleb märksa laiem:

```
> predict(mudel2, data.frame(Height=40), interval="prediction")
      fit      lwr      upr
[1,] 652.725 282.6357 1022.814
```

Põhimõtteliselt on prognoosimiseks võimalik kasutada nii kala kõrgust kui pikkust:

```
mudel3=lm(Weight~Height+Length3, data=andmed[Species==1,])
summary(mudel3)
predict(mudel3, data.frame(Height=40, Length3=44),
interval="prediction")
```

ja prognoosiks sobivate tunnuste leidmisel võib olla abi järgmistest käskudest:

```
cor(andmed[Species==1,3:8])
plot(andmed[Species==1,3:8])
```

### Ülesanne

Proovi prognoosida ahvena kaalu kasutades ahvena pikkust (nähtud ahven oli 45cm pikk).

## Loe sisse andmestik lapsed2:

```
andmed = read.csv2(
  "http://www.ms.ut.ee/mart/biomeetria2009/lapsed2.csv",
  header=TRUE)
```

Vaata andmeid  
andmed[1:3,]

Antud andmestikus on järgmised tunnused:

vanus – lapse vanus mõõtmise tegemise hetkel (aastates)  
kaal – lapse kaal (kg)  
pikkus – lapse pikkus (cm).  
sugu – lapse sugu

NB! Millise käsu peaksid andma, et saaksid hiljem selle andmestiku tunnuseid (lihtsalt) kasutada?

Meid huvitab, kuidas lapse kasvades (vanuse suurenedes) muutub lapse kaal.

## Esialgne mudel

```
m1=lm(pikkus~vanus)
summary(m1)
```

Milline näeb välja hinnatud mudel? Pane see kirja!  
Kommenteeri tulemusi – kas vanuse abil saab pikkust prognoosida? Kui hästi?  
Mis on hinnatud mudeli puhul viltu?

Unustame hetkeks oma kahtlused ja prognoosime oma mudelit kasutades 1,2 aasta vanuse lapse pikkust. Leiame ka 95%-usaldusintervalli 1,2 aastaste laste keskmisele pikkusele ja 95%-prognoosiintervalli 1,2 aastase lapse pikkusele:

```
> predict(m1, data.frame(vanus=1.2))
[1] 78.55209
> predict(m1, data.frame(vanus=1.2), interval="confidence")
      fit      lwr      upr
[1,] 78.55209 78.50859 78.59559
> predict(m1, data.frame(vanus=1.2), interval="prediction")
      fit      lwr      upr
[1,] 78.55209 71.60648 85.4977
```

Joonistame vanuse-pikkuse hajuvusgraafiku ning lisame gaafikule meie poolt leitud regressioonisirge:

```
plot(vanus, pikkus, xlab="Vanus (aastates)", ylab="Pikkus (cm)")
x=seq(0,2.2,0.01)
y=predict(m1, data.frame(vanus=x))
lines(x,y, col="red", lwd=2)
```

Mida teevad ülaltoodud programmi kolm viimast rida? Mida loed välja joonistatud graafikult?  
Milliseid probleeme näed, kuidas võiks esinenud probleeme lahendada?

Vaatame ka teist regressioonanalüüsi eeldust – nõuet, et uuritav tunnus peab olema normaaljaotusega juhuslik suurus:

```
qqnorm(residuals(m1)); qqline(residuals(m1))
```

või vaatame lihtsalt jääkide histogrammi:

```
hist(residuals(m1))
```

Milline on otsus?