

Lineaarsed mudelid
Nihkega hinnangud

Käesolevas praktikumis vaatame paari erinevat võimalust leida nihkega hinnanguid lineaarse mudeli parameetritele ja prognoosime hinnatud mudelite abil uusi (tulevasi) vaatluseid. Millised meetodid viivad praegu täpsemate prognoosideni (NB! Teistsugustes oludes – teistsuguste tunnustevaheliste korrelatsioonide või vaatluste/tunnuste suhte puhul või lihtsalt teise algvalimi korral võib meetodite paremusjärjestus muutuda!)

Märgi üle uute vaatluste prognoosimisel tekkivad keskmised ruutvead (kasuta allpool toodud näiteprogrammi andmete genereerimiseks):

Meetod	MSE
Õige mudel	
Sammregressioon + AIC	
Kantregressioon	
Lasso	
Mudelite keskmistamine (Mallowsi' kaalud):	

Tulemuste kättesaamiseks uuri ja kasuta järgmist näiteprogrammi

```
# Algandmete tekitamine -----

# vaatluste arv
n=400
# tunnuste arv
tunnuseid=100

# Paneme paika regressioonmudeli tegelikud kordajad:
set.seed(1)
tegekordajad=rexp(tunnuseid)*(0.5)**(1:tunnuseid)
# maatriks mida kasutatakse x-tunnuste vaheliste korrelatsioonide loomiseks
seosX=matrix(rnorm(tunnuseid*tunnuseid, sd=4), ncol=tunnuseid)

# Funktsioon, mis tekitab soovitud arvu vaatluseid
vaatlused=function(n, kordajad, seosX){
  tunnuseid=length(kordajad)
  x0=matrix(rnorm(n*tunnuseid), ncol=tunnuseid)
  X=x0%*%t(seosX)
  y=5+X%*%tegekordajad+rnorm(n, sd=5)
  andmed=data.frame(y, X)
  return(andmed)
}
andmed=vaatlused(n, tegekordajad, seosX)
andmed_uus=vaatlused(10000, tegekordajad, seosX)
```

```
# Hindame andmestikku andmed kasutades meid huvitava mudeli ja vaatame,  
# kui hästi suudab hinnatud mudel prognoosida y-tunnuse väärtust uute andmete korral  
# ehk andmestikus andmed_uus.
```

```
# -----  
# Õige mudel  
# -----  
oigemudel=lm(y~. , data=andmed)  
prog1 = predict(oigemudel, newdata=andmed_uus)  
  
mean((andmed_uus$y-prog1)**2)
```

```
# -----  
# sammregressioon (AIC järgi mudeli valik)  
# -----  
m2 = step(oigemudel)  
prog2= predict(m2, newdata=andmed_uus)  
  
mean((andmed_uus$y-prog2)**2)
```

```
# -----  
# Mudelite keskmistamine - Mallowsi' kaalud  
# -----
```

```
# Peame esmalt paigaldama eksperimentaalse lisamooduli:  
install.packages("MAMI", repos=c("http://R-Forge.R-project.org",  
    "http://cran.at.r-project.org"), dependencies=TRUE)  
library(MAMI)
```

```
m3 = mma(andmed, formula=oigemudel, ycol=1, variance=c("BA","boot"), bsa=200)  
# Keskmise mudel:  
m3  
# Mudelite kaalud  
# Praegu vaadatakse mudeleid  $y \sim 1$ ;  $y \sim X_1$ ;  $y \sim X_1 + X_2$ ;  $y \sim X_1 + X_2 + X_3$ ; jne.  
# millise kaaluga millist mudelit arvestatakse?  
m3$averaging.weights  
sum(m3$averaging.weights)
```

```
# Leiame prognoosid "keskmise" mudeli järgi  
prog3=cbind(1,as.matrix(andmed_uus[, -1]))%*% (coef(m3)[1,])  
  
mean((andmed_uus$y-prog3)**2)
```

```

# -----
# kantregressioon (Ridge regression)
# -----
library(glmnet)

# Milline võiks olla optimaalne lambda väärtus?
lambdas=10^seq(-2, 4, length=1000)
m4 = cv.glmnet(as.matrix(andmed[, -1]), andmed$y, alpha = 0, lambda = lambdas)
opt_lambda = m4$lambda.min
opt_lambda

# Prognoosime kasutades optimaalset lambda väärtust:
prog4 <- predict(m4, s = opt_lambda, newx = as.matrix(andmed_uus[, -1]))

mean((andmed_uus$y-prog4)**2)

# -----
# Lasso
# -----
library(glmnet)

m5 = cv.glmnet(as.matrix(andmed[, -1]), andmed$y, alpha = 1, lambda = lambdas)
opt_lambda_lasso = m5$lambda.min
prog5 <- predict(m5, s = opt_lambda_lasso, newx = as.matrix(andmed_uus[, -1]))

mean((andmed_uus$y-prog5)**2)

```

Näide 2.

Soovime uurida ühe parameetri hinnangu nihet (andmestikus X1 ees olev kordaja) ja hinnangu täpsust (keskmine ruutviga). Vaata järgmist simulatsiooni (ja proovi aru saada, mis toimub). Seejärel täida simulatsioonile järgnev tabel.

Simulatsiooni kood:

```

# Selles osas proovime kantregressiooni funktsiooni lm.ridge (MASS) abil:
library(MASS)

# Mitu simulatsiooni teha
korduseid=100

# Mitu vaatlust, mitu tunnust simuleerida
n=100
tunnuseid=50

```

```

# Kuhu salvestame eri meetoditega leitud hinnangud
beta2_lm = rep(NA, korduseid)
beta2_ridge = rep(NA, korduseid)
beta2_lasso = rep(NA, korduseid)
beta2_ma = rep(NA, korduseid)

set.seed(1)

for (i in 1:korduseid){

  print(paste(i,"/",korduseid))

  # Vaatluste genereerimine
  X=matrix(rnorm(n*tunnuseid), ncol=tunnuseid)
  and=data.frame(X)
  and$y=4+2*and$X1+rnorm(n, sd=3)

  # Lineaarse mudeli hindamine
  mudel1=lm(y~., data=and)
  beta2_lm[i]=coef(mudel1)[2]

  # Kantregressioon:
  lambdas=10^seq(-2, 4, length=1000)
  m_test=lm.ridge(and$y~X, lambda=lambdas)
  k = lambda[m_test$GCV == min(m_test$GCV)]
  mudel2=lm.ridge(and$y~X, lambda=k)
  beta2_ridge[i]=coef(mudel2)[2]

  # Lasso
  mudel3 = cv.glmnet(as.matrix(and[,-(tunnuseid+1)]), and$y, alpha = 1,
                    lambda = lambdas)
  opt_lambda_lasso = mudel3$lambda.min
  beta2_lasso[i]=coef(mudel3, s=opt_lambda_lasso)[2]

  # Mudelite keskmistamine, Mallows'i kaalud
  # NB! Tunnuste järjekord on mma-käsu korral ülitähtis:
  # esmalt peavad olema kirjas oletuslikult suuremat mõju omavad tunnused!
  mudel4 = mma(and, formula=mudel1, ycol=(tunnuseid+1))
  beta2_ma[i]=coef(mudel4)[1,2]
}

```

Täida järgmine tabel (tunnuse X1 ees oleva kordaja hinnangu nihke ja keskmise ruutvea hinnangud):

Meetod	nihe	keskmine ruutviga
Lineaarne mudel		
Kantregressioon		
Lasso		
Mudelite keskmistamine		