Kuidas biomarkerite abil ennustada surma?

Risto Hinno

Friday, June 26, 2015

Sissejuhatus

Geenitehnoloogiat tudeeriv Elo "Elu" Eliksiir on kuulnud, et ka geneetika valdkonnas leidub edukaid ettevõtteid, nagu näiteks 23andMe, mis annab inimesele teada tema riski haigestuda erinevatesse haigustesse. Sestap plaanib ta teha idufirma, mis teeniks tulu inimese surma prognoosimisega. Täpsemalt, inimeselt võetaks vereproov, ning selle tulemuste põhjal öeldaks talle, milline on tema tõenäosus surra järgneva 5 aasta jooksul.

Et kirjutada taotlus rahastuse saamiseks, on tal esmalt vaja välja mõelda, millel see suremuse test üldse peaks põhinema. See tähendab, et millise ühendi sisaldust vereproovist oleks vaja mõõta? Ta leidis, et Geenivaramu teadlased eesotsas Krista Fischeriga (TÜ statistika vilistlane) on 2014. aastal avaldanud teadusartikli Biomarker Profiling by Nuclear Magnetic Resonance Spectroscopy for the Prediction of All-Cause Mortality: An Observational Study of 17,345 Persons.

Selle praktikumis uurimegi, kas ja kuidas saab biomarkerite abil ennustada surma. Meil on kasutada valim Geenivaramu andmestikust, mis sisaldab 5000 inimese verest mõõdetud 106 biomarkeri väärtuseid.

Aga mis üleüldse on biomarker?

A biomarker is a biological molecule found in blood, body fluids, or tissues that may signal an abnormal process, a condition, or a disease. The level of a particular biomarker may indicate a patient's risk of disease, or likely response to a treatment. For example, cholesterol levels are measured to assess the risk of heart disease.

Andmestik

Kasutame valimit Geenivaramu andmestikust, mida kasutati eelnevalt mainitud teadusartikli juures. Täpsemalt on andmestikus tunnused:

- sugu
- vanusgrupp
- s5 - indikaator, kas 5 aasta pärast oli surnud
- hyp - kas inimesel on hüpertooniatõbi ehk kõrgvererõhutõbi
- suits - kas on suitsetaja
- LDL_D - esimese biomarkeri väärtus
- L_HDL_FC - teise biomarkeri väärtus
- ...
- Cit - viimase biomarkeri väärtus

Laadi ÕISist alla andmestik *biomarkerid.csv* ja loe töökeskkonda.

Analüüsi lihtsuse huvides eemalda puuduvaid andmeid sisaldavad read. Abiks on funktsioon complete.cases.

```
biomarkerid=read.csv("./data/biomarkerid.csv")
biomarkerid=biomarkerid[complete.cases(biomarkerid),]
```

Kui hästi on kolesterooli abil võimalik ennustada surma?

Elo tutvus Geenivaramu andmestikuga, ent seal oli palju arusaamatute lühenditega biomarkereid. Samas on ta kuulnud, et kolesterool on üks nendest näitajatest, mille näit peab tingimata korras olema. Ehk saaks kolesterooli põhjal hästi prognoosida surma?

Eralda andmestikust alamandmestik, mis sisaldaks tunnuseid sugu, vanusgrupp, s5, hyp, suits ning järgmisi biomarkereid:

- Serum_C - üldkolesterool
- HDL_C - HDL kolesterool (“hea”)
- LDL_C - LDL kolesterool (“halb”)

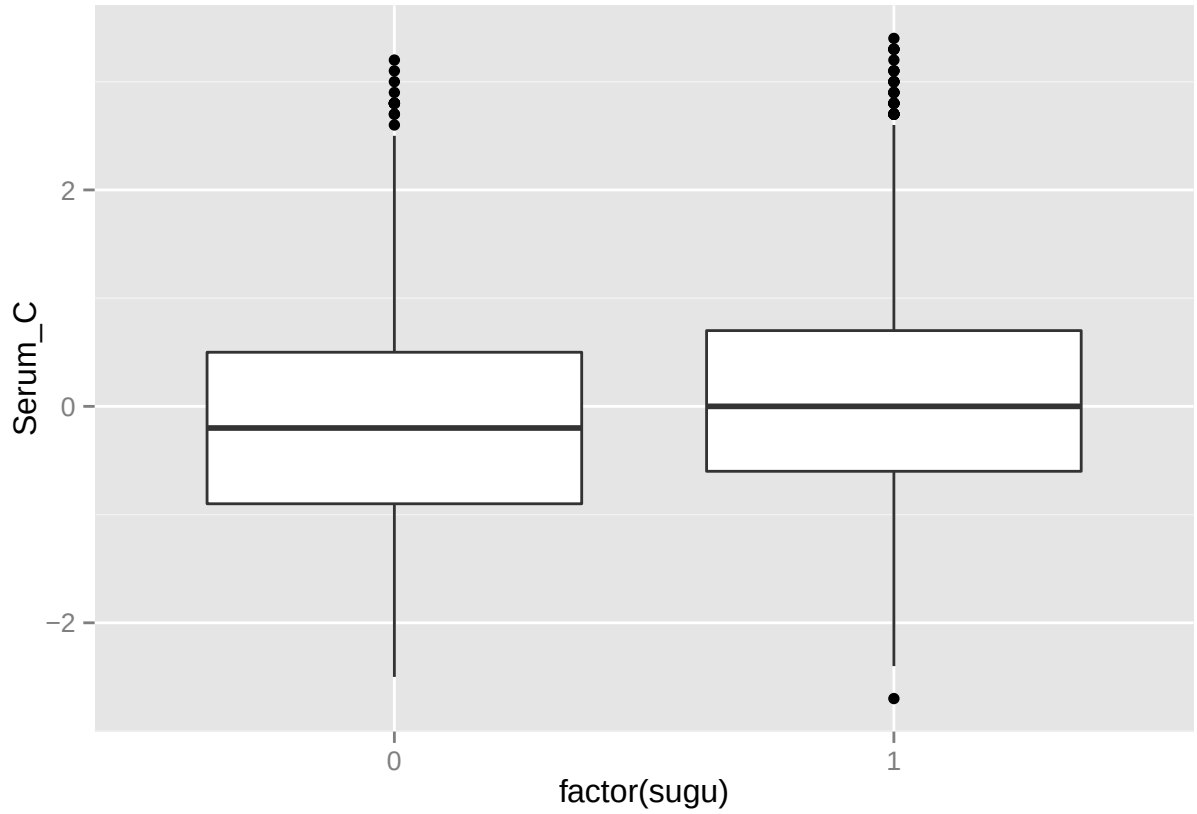
```
biom_subset=biomarkerid[, c("sugu", "vanusegrupp", "s5", "hyp", "suits",
                            "Serum_C", "HDL_C", "LDL_C")]
```

Ülesanne 1 (2 punkti) - kolesterool soo ja vanusegruppide lõikes

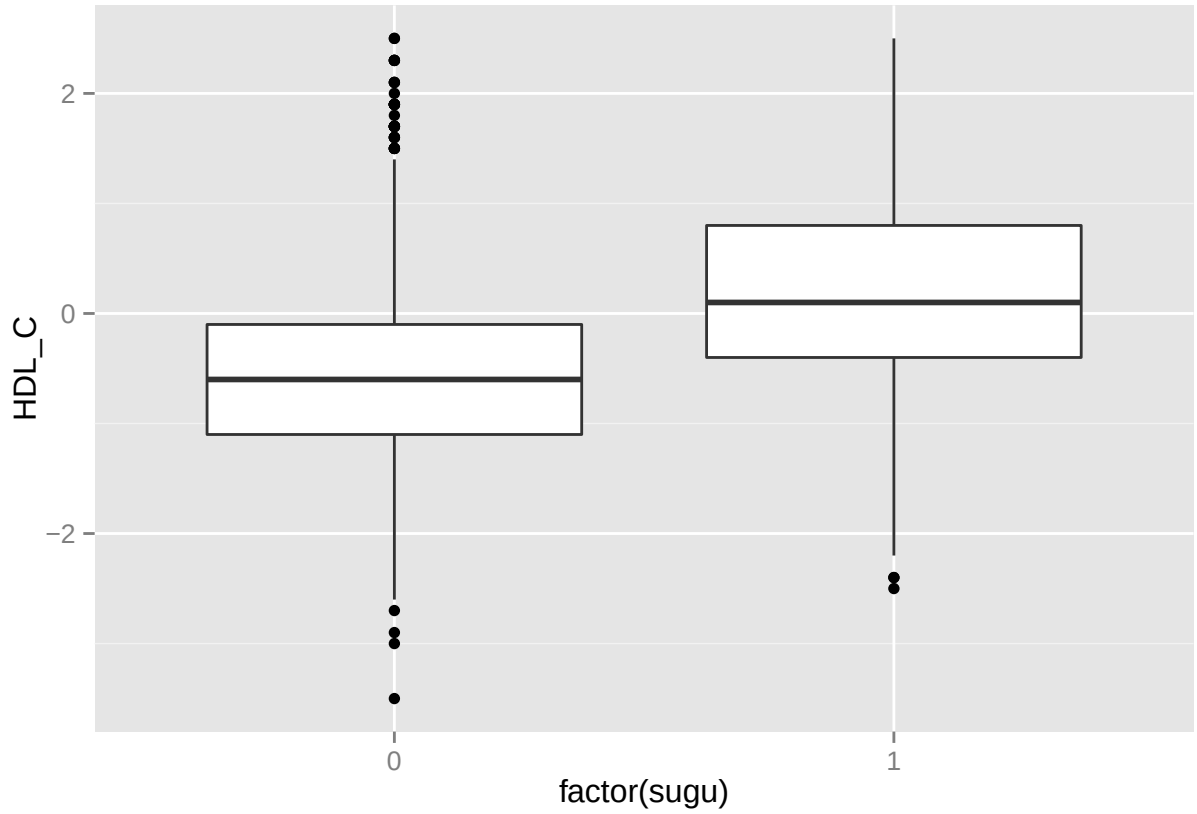
Tutvu andmestikuga ja selgita välja, kuidas on kodeeritud tunnus sugu (kas 0 tähistab meest või naist)?

Visualiseeri, kuidas nende 3 biomarkeri jaotused erinevad soo ja vanusegruppide lõikes.

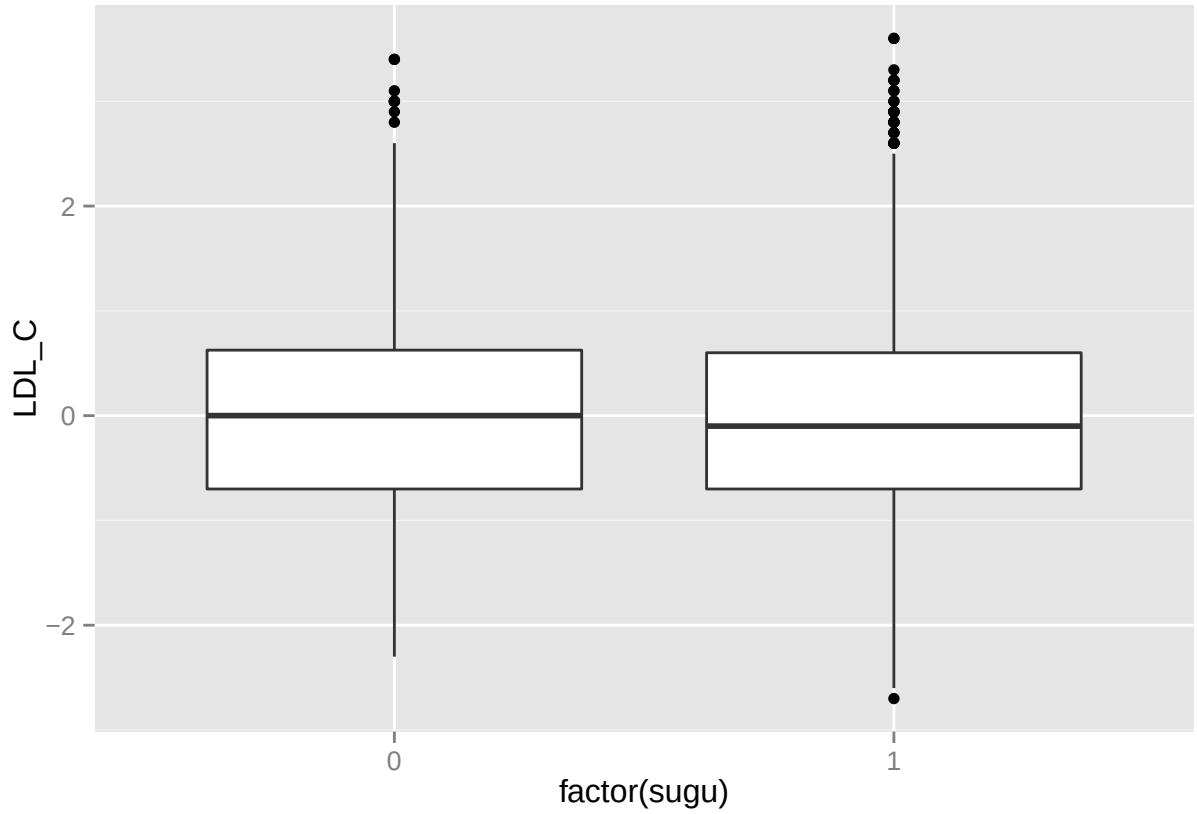
```
library(ggplot2)
ggplot(biom_subset, aes(x=factor(sugu), y=Serum_C))+
    geom_boxplot()
```



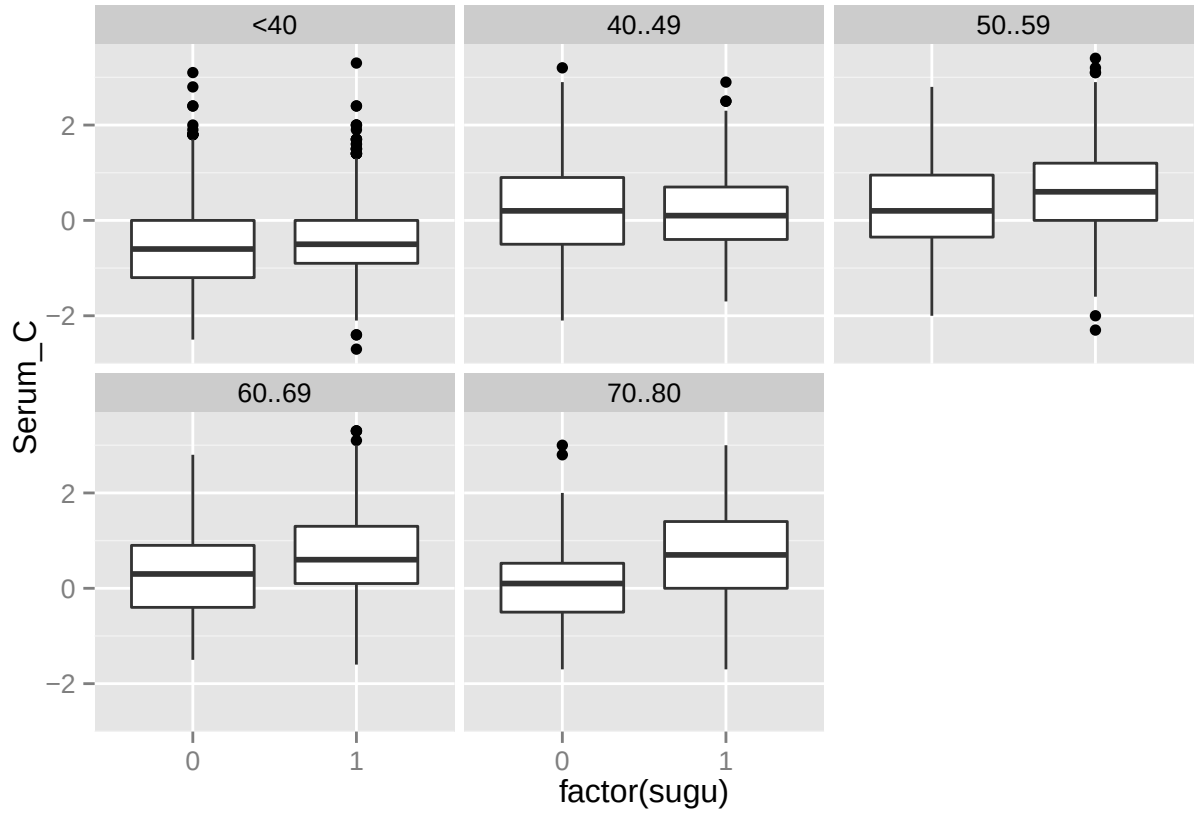

```
#HDL_C
ggplot(biom_subset, aes(x=factor(sugu), y=HDL_C))+
    geom_boxplot()
```



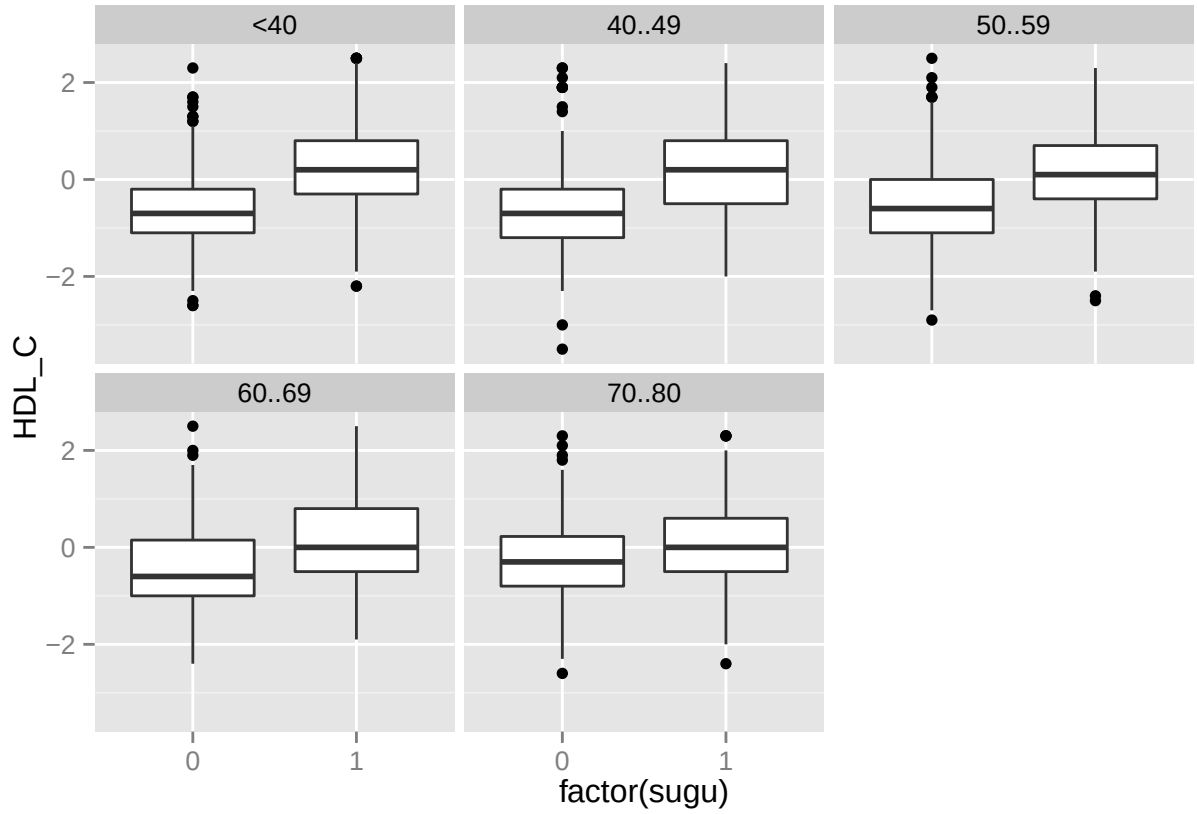

```
#LDL_C
ggplot(biom_subset, aes(x=factor(sugu), y=LDL_C))+
    geom_boxplot()
```



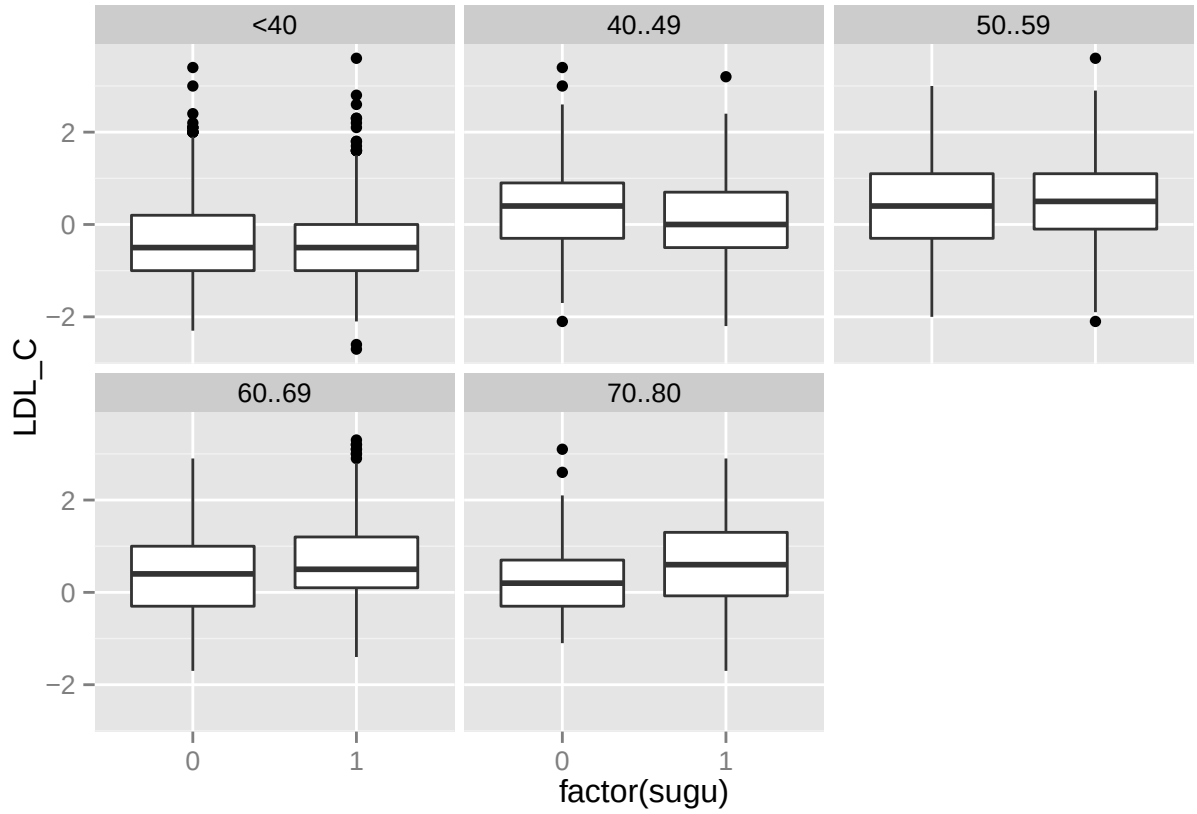

```
#vanusegrappide lõikes
ggplot(biom_subset, aes(x=factor(sugu), y=Serum_C))+
    geom_boxplot()+
    facet_wrap(~vanusegrupp)
```




```
#HDL_C
ggplot(biom_subset, aes(x=factor(sugu), y=HDL_C))+
    geom_boxplot()+
    facet_wrap(~vanusegrupp)
```




```
#LDL_C
ggplot(biom_subset, aes(x=factor(sugu), y=LDL_C))+
    geom_boxplot()+
    facet_wrap(~vanusegrupp)
```



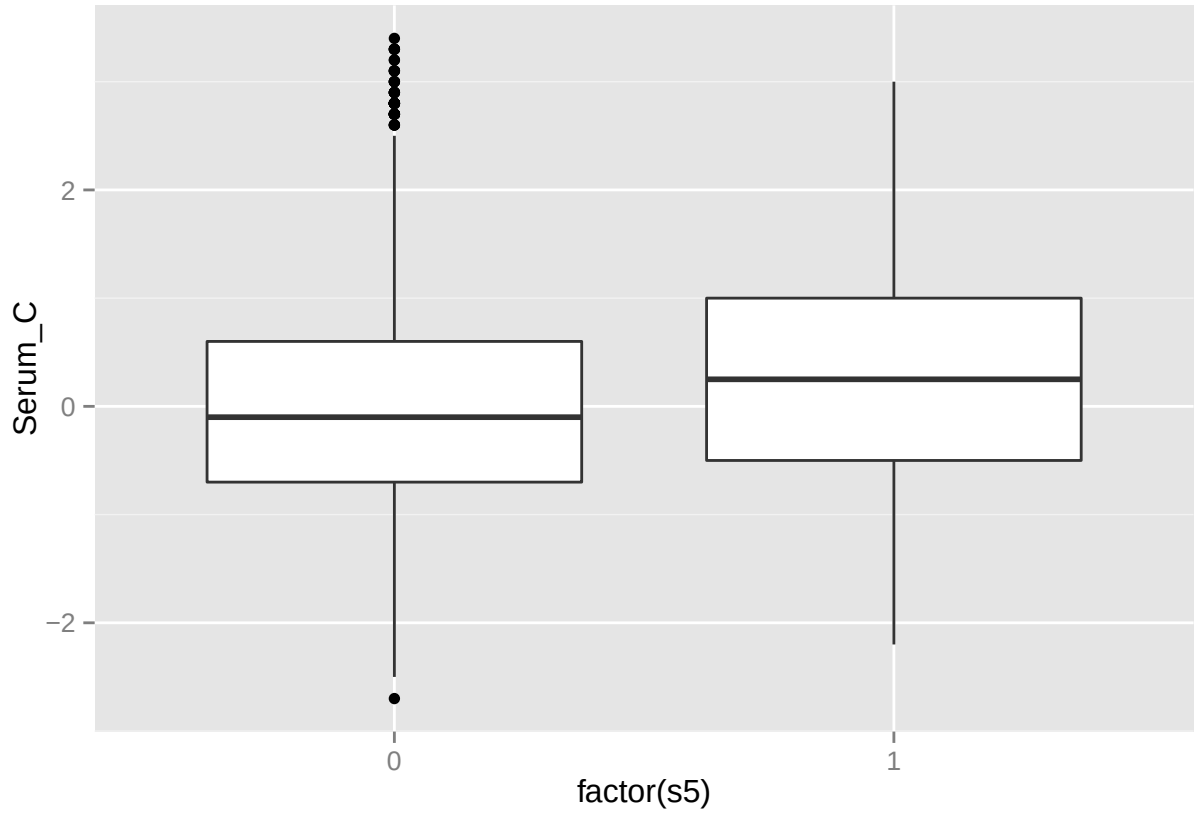
Tundub, et 1 on naine ja 0 mees, kuna naisi on artikli põhjal andmestikus rohkem.

Ülesanne 2 (4 punkti) - surma prognoosimine kolesterooli abil?

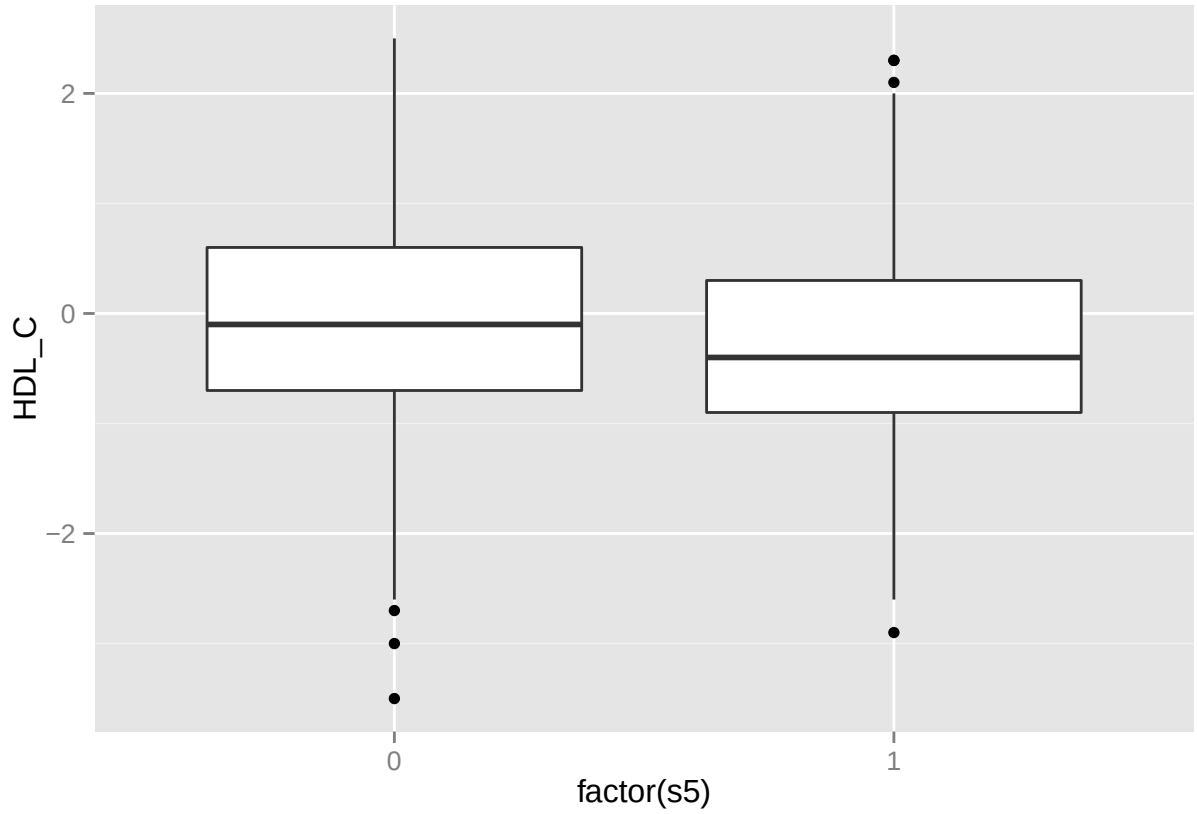
Uuri, kas kolesterool võimaldab prognoosida surma. Selleks tuleb andmetele sobitada mudel.

- Visualiseeri, kas kolesterooli (Serum_C, HDL_C, LDL_C) abil võiks saada prognoosida surma.

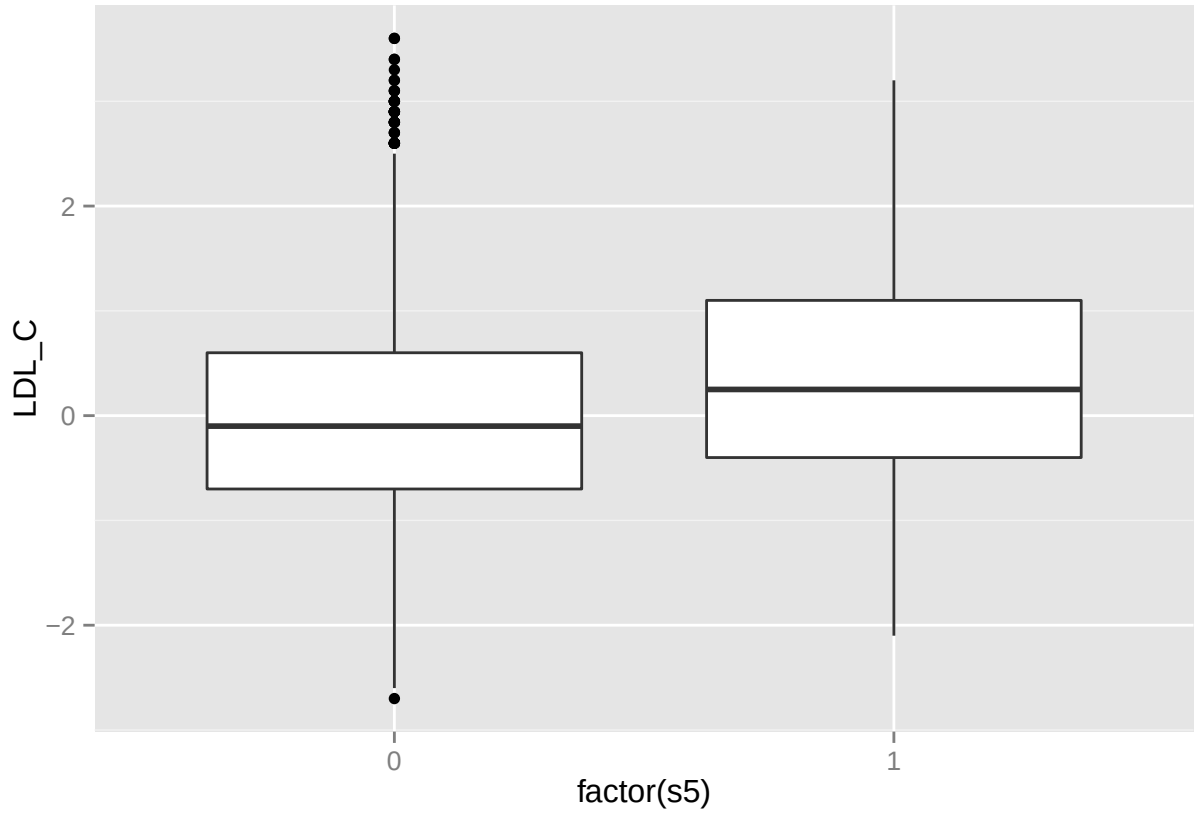
```
ggplot(biom_subset, aes(x=factor(s5), y=Serum_C))+
    geom_boxplot()
```




```
#HDL_C
ggplot(biom_subset, aes(x=factor(s5), y=HDL_C))+
    geom_boxplot()
```




```
#LDL_C
ggplot(biom_subset, aes(x=factor(s5), y=LDL_C))+
    geom_boxplot()
```



Mingi erinevus on, kuid väga kindlalt erinevust välja ei loe.

- Tundub, et joonisest ei piisa ning tuleb pöörduda statistiliste mudelite juurde. Kas kasutad lineaarset või logistilist regressiooni? Miks?
- Sobitasime mudeli glm(s5 ~ HDL_C, family=binomial, data=data) ning selgus, et HDL_C on oluline surma prognoosimisel. Seejärel aga sobitasime mudeli glm(s5 ~ HDL_C + sugu, family=binomial, data=data), siis miskipärast HDL_C enam ei ole oluline. Selgita, mis värk on. Kas siis kokkuvõttes on oluline või mitte?

```
mudel1=glm(s5 ~ HDL_C, family=binomial, data=biom_subset)
summary(mudel1)
```

```
##
## Call:
## glm(formula = s5 ~ HDL_C, family = binomial, data = biom_subset)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.4784  -0.3237  -0.2973  -0.2653   2.7625
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.12589    0.08092 -38.630  < 2e-16 ***
## HDL_C       -0.29026    0.08840  -3.283  0.00103 **
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1442.9  on 3991  degrees of freedom
## Residual deviance: 1431.8  on 3990  degrees of freedom
## AIC: 1435.8
##
## Number of Fisher Scoring iterations: 6
```

```
mudel2=glm(s5 ~ HDL_C + sugu, family=binomial, data=biom_subset)
summary(mudel2)
```

```
##
## Call:
## glm(formula = s5 ~ HDL_C + sugu, family = binomial, data = biom_subset)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.4428  -0.3728  -0.2523  -0.2396   2.7237
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.64215    0.11778 -22.433  < 2e-16 ***
## HDL_C       -0.10547    0.09512  -1.109    0.268
## sugu        -0.83123    0.16989  -4.893 9.95e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1442.9  on 3991  degrees of freedom
## Residual deviance: 1407.6  on 3989  degrees of freedom
## AIC: 1413.6
##
## Number of Fisher Scoring iterations: 6
```

Oleneb mudelist, mida kasutame. Ilma soota prognoosib päris hästi surma. Kuid kui korrigeerime seda sooga, siis enam ei selgita, sugu selgitab paremini varieeruvust.

- Sobita kolm mudelit, et uurida kolesterooli (tunnuste Serum_C, HDL_C ja LDL_C) seost surmaga. Muide, kas lisad mudelitesse ka tunnused sugu, vanusegrupp, suits ja hyp? Põhjenda oma otsust.

```
mudel3=glm(s5 ~ Serum_C+ HDL_C + LDL_C, family=binomial, data=biom_subset)
summary(mudel3)
```

```
##
## Call:
## glm(formula = s5 ~ Serum_C + HDL_C + LDL_C, family = binomial,
##     data = biom_subset)
##
## Deviance Residuals:
```

```
##     Min       1Q   Median       3Q      Max
## -0.5845  -0.3266  -0.2810  -0.2451   2.8080
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -3.1796     0.0846 -37.585  < 2e-16 ***
## Serum_C       0.1780     0.3349   0.531  0.59511
## HDL_C        -0.2864     0.1072  -2.672  0.00754 **
## LDL_C         0.1522     0.3381   0.450  0.65262
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1442.9  on 3991  degrees of freedom
## Residual deviance: 1413.4  on 3988  degrees of freedom
## AIC: 1421.4
##
## Number of Fisher Scoring iterations: 6
```

Ei lisanud, sest siis poleks ükski marker olnud statistiliselt oluline.

- Milline on tulemus, st kas siis mõni kolmest kolesterooli tunnusest on olulise mõjuga surma ennustamisel?

Jah, HDL_C on oluline.

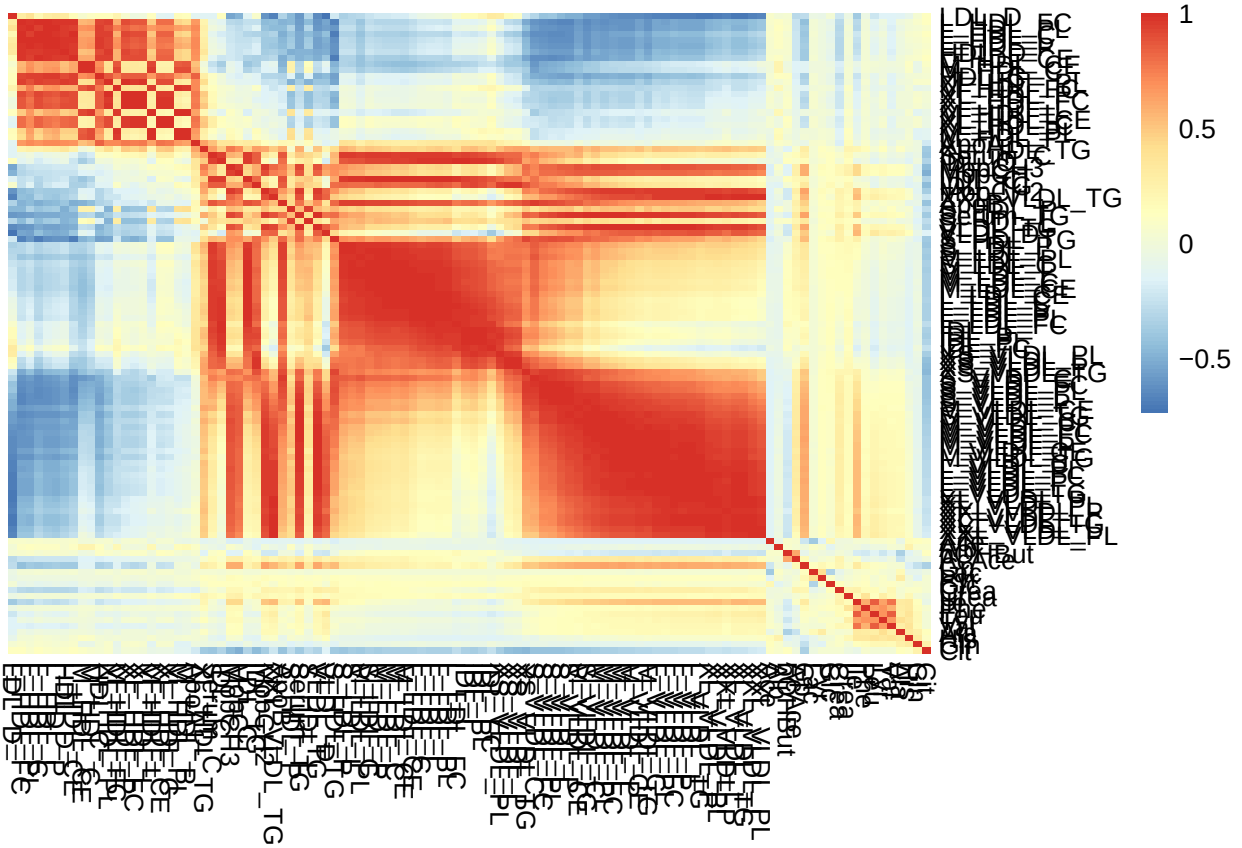
Kogu andmestikul põhinev analüüs

Eelnimetatud teadusartiklis vaadeldi kõiki 106 biomarkerit. Tegelenud esialgu kolme biomarkeriga, saime tuttavaks logistilise regressiooniga R-is ning julgeme nüüd asuda artiklis kirjeldatud analüüsi reprodutseerima. Kõigis järnevates ülesannetes kasutame kogu andmestikku (st kõiki 106 biomarkerit).

Ülesanne 3 (1 punkt) - korrelatsioonid biomarkerite vahel

Tee joonis, mis annaks hästi edasi, kas ja millised biomarkerid on omavahel korreleeritud. (Näpunäide: Arvuta korrelatsioonimaatriks käsuga cor ning visualiseeri seda.) Interpreteeri, milliseid mustreid ja seoseid näed?

```
korrel_maatriks=cor(biomarkerid[6:111])

library(pheatmap)
pheatmap(korrel_maatriks, cluster_rows=FALSE, cluster_cols=FALSE)
```



Ülesanne 4 (1 punkt) - Oluliste biomarkeri tuvastamine

Milline biomarker aitab kõige paremini ennustada surma kui võtame arvesse vanuse ja soo mõju?

Selleks sobita mudelid

- s5 ~ sugu + vanusegrupp + biomarker_1
- s5 ~ sugu + vanusegrupp + biomarker_2
- ...
- s5 ~ sugu + vanusegrupp + biomarker_106

ja iga biomarkeri korral eralda mudelist selle p-väärtus ja kordaja.

Kui sa ei soovi 106 korda glm mudelit käsitsi jooksutada ja manuaalselt p-väärtuseid välja noppida, siis automatiseeri see (näiteks for tsükli abil).

```
biomarkers = names(biomarkerid)[6:111]
formula0 = "s5 ~ sugu + vanusegrupp"

for (i in 1:length(biomarkers)) {
    formula = paste(formula0, biomarkers[i], sep=" + ")
    model = glm(formula, family=binomial, data=biomarkerid)
    summary_table = coef(summary(model))
```

```
    pvalue[i] = summary_table[nrow(summary_table), 4]
    estimate[i] = summary_table[nrow(summary_table), 1]
}
#vähem arve pärast komakohti
pvalue=round(pvalue, 4)
estimate=round(estimate, 4)
tulemid=data.frame(biomarkers, pvalue, estimate)
head(tulemid)
```

```
##   biomarkers pvalue estimate
## 1      LDL_D 0.0059   0.2846
## 2   L_HDL_FC 0.8867  -0.0138
## 3   L_HDL_PL 0.9308  -0.0086
## 4    L_HDL_C 0.4994  -0.0653
## 5    L_HDL_L 0.8082  -0.0239
## 6    L_HDL_P 0.9828  -0.0021
```

Ülesanne 5 (1 punkt)

Kirjuta eelnev kood funktsiooniks.

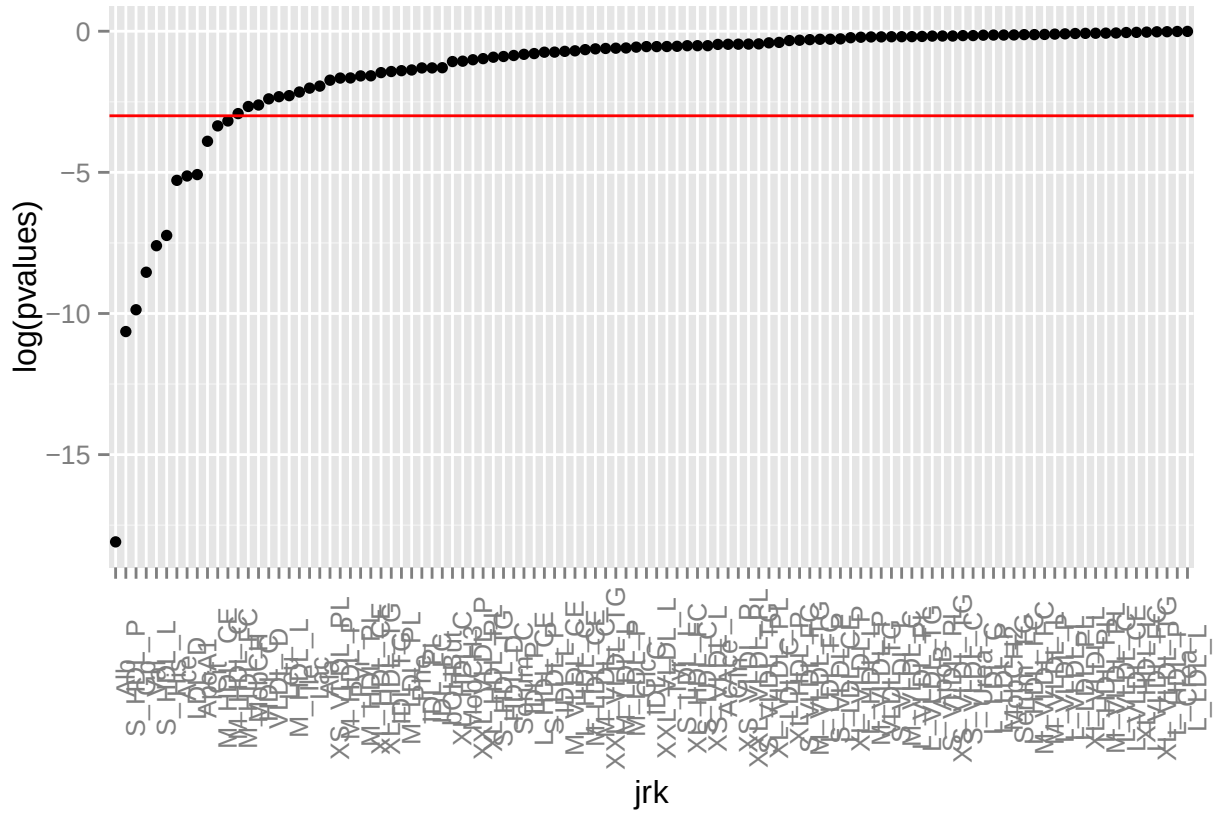
```
estimate_significance = function(formula0, biomarkers, data){
    coefs=list()
    pvalues=list()
    for (i in 1:length(biomarkers)) {
        formula = paste(formula0, biomarkers[i], sep=" + ")
        model = glm(formula, family=binomial, data=data)
        summary_table = coef(summary(model))
        pvalues[i] = summary_table[nrow(summary_table), 4]
        coefs[i] = summary_table[nrow(summary_table), 1]
    }
    #vähem arve pärast komakohti
    pvalues=round(as.numeric(pvalues), 10)
    coefs=round(as.numeric(coefs), 6)
    tulemid=data.frame(biomarkers, pvalues, coefs)

    return(tulemid)
}
#testin
head(estimate_significance(formula0="s5 ~ sugu + vanusegrupp",
                      biomarkers=c("LDL_D", "HDL_C"), data=biomarkerid))
```

```
##   biomarkers     pvalues     coefs
## 1      LDL_D 0.005926428  0.284601
## 2      HDL_C 0.073286899 -0.172940
```

Ülesanne 6 (1 punkt)

Visualiseeri saadud tulemust.

```
hinnangud=estimate_significance(formula0="s5 ~ sugu + vanusegrupp",
                      biomarkers=names(biomarkerid)[6:111], data=biomarkerid)
#Visualiseeri saadud tulemust.
library(ggplot2)
#leaime õige järjekorra
jrk=with(hinnangud, reorder(biomarkers, pvalues, mean))
#plotime
ggplot(hinnangud, aes(x=jrk, y=log(pvalues)))+
    geom_point()+
    geom_hline(yintercept=log(0.05), colour="red")+
    theme(axis.text.x=element_text(angle=90))
```



Ülesanne 7 (4 punkti + 1 boonuspunkt) - p-väärtuse piiri paikapanek

Nüüd saime kõigi biomarkerite jaoks teada p-väärtused. Jääb veel küsimus, millised neist peaksime liigitama olulisteks.

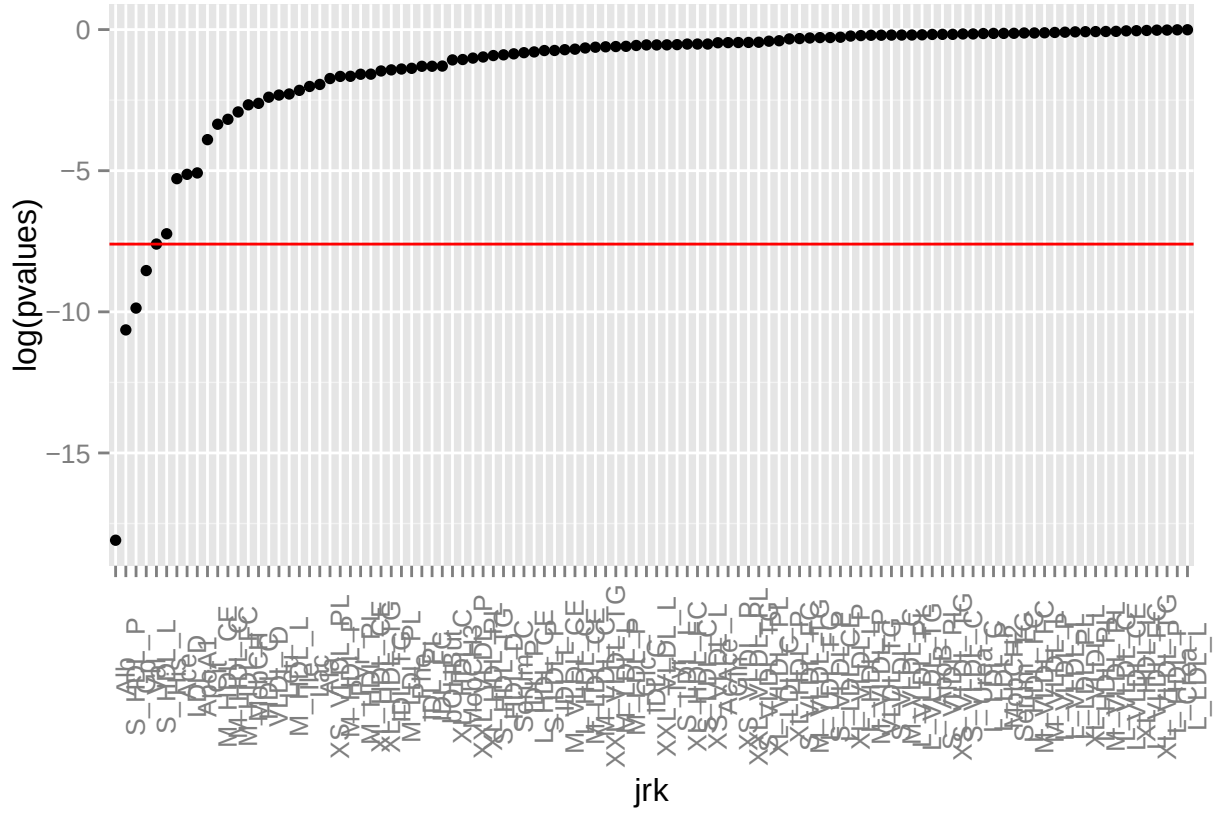
Kuna testisime kõigi 106 biomarkeri olulisust surma ennustamisel, puutume kokku mitmese testimise probleemiga. Vaata selle kohta koomiksit "Significant" ning uuri materjalist http://www.stat.berkeley.edu/~mgoldman/Section0402.pdf mitmese testimise ja Bonferroni korrektsiooni kohta.

- (1 punkt) Selgita, milles seisnes koomiksi idee. Idee seisneb selles, et mida rohkem teha statistilisi teste, siis tõenäosus, et saadakse valepositiivne, kasvab.
- (1 punkt) Artiklis kasutati olulisuse nivood $p < 0.0005$. Täpsemalt,

. . . significant at the Bonferroni-corrected threshold of p<0.0005, accounting for testing of 106 candidate biomarkers

- Selgita, miks kasutati sellist p-väärtuse piiri (aga mitte klassikalist p $<$ 0.05)? kasutati Bonferroni korrektsiooni (0.05/106 on ligikaudu 0.0005), et kontrollida valepositiivsete tulemuste hulka.

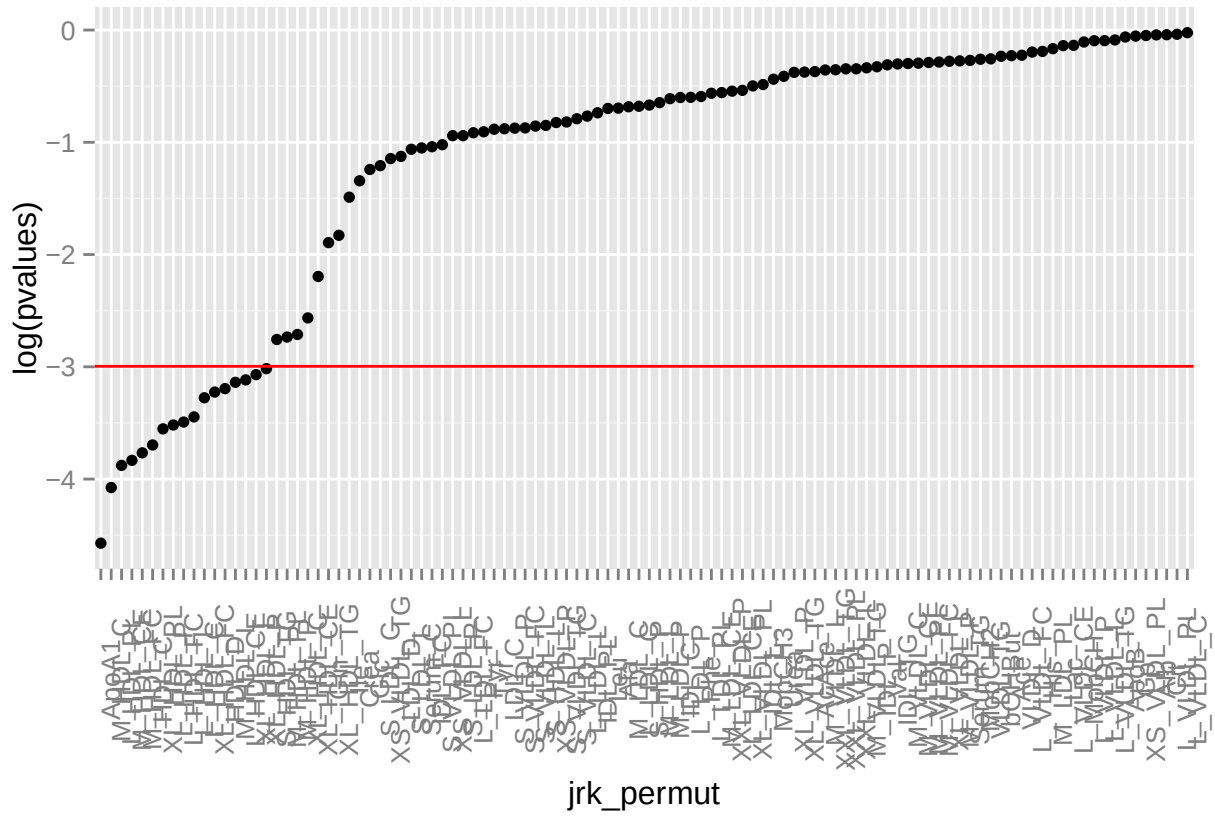
```
ggplot(hinnangud, aes(x=jrk, y=log(pvalues)))+
    geom_point()+
    geom_hline(yintercept=log(0.0005), colour="red")+
    theme(axis.text.x=element_text(angle=90))
```



- (2 punkti) Veendumaks, et p $<$ 0.05 kasutamisel võime tõepoolest saada liiga palju valepositiivseid tulemusi, tekita andmestik, kus puudub seos tunnuse s5 ja biomarkerite vahel. Selleks tekita uus tunnus, kus oleks s5 väärtuseid permuteeritud suvaliselt. Sobita nüüd mudelid, kus prognoosiksid permuteeritud s5 väärtuseid biomarkerite põhjal (selleks võid kasutada ülesandes 5 kirjutatud funktsiooni).

```
set.seed(100)
biomarkerid$s5_permut=sample(biomarkerid$s5, replace=T)
#names(biomarkerid)

hinnangud_permut=estimate_significance(formula0="s5_permut ~ sugu + vanusegrupp",
                                       biomarkers=names(biomarkerid)[6:111],
                                       data=biomarkerid)
#leiame faktorite õige järjekorra
jrk_permut=with(hinnangud_permut, reorder(biomarkers, pvalues, mean))
```

```
#plotime
ggplot(hinnangud_permut, aes(x=jrk_permut, y=log(pvalues)))+
    geom_point()+
    geom_hline(yintercept=log(0.05), colour="red")+
    theme(axis.text.x=element_text(angle=90))
```



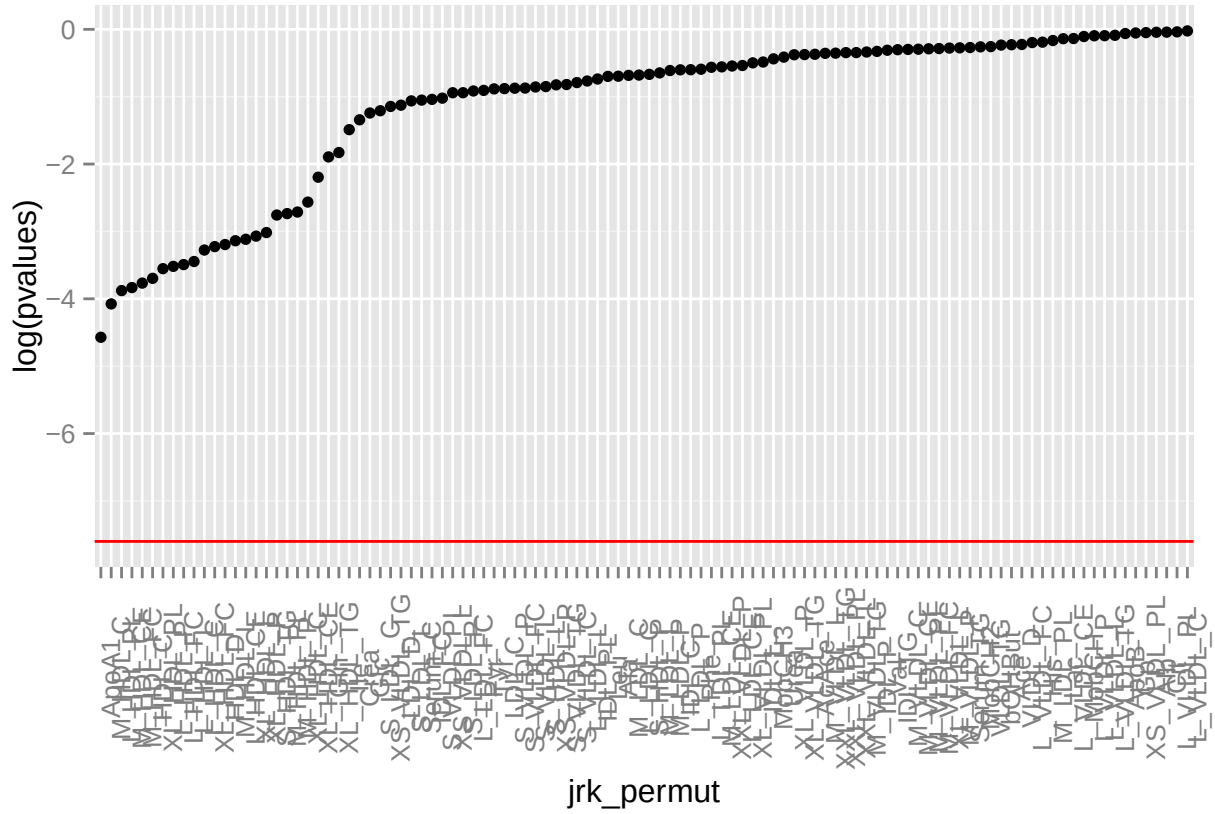
- Mitme biomarkeri p-väärtused tulid väiksemad kui 0.05?

```
nrow(subset(hinnangud_permut,pvalues<0.05))
```

```
## [1] 17
```

- Aga mitu tükki olid olulised Bonferroni korrektsiooni järgi?

```
ggplot(hinnangud_permut, aes(x=jrk_permut, y=log(pvalues)))+
    geom_point()+
    geom_hline(yintercept=log(0.0005), colour="red")+
    theme(axis.text.x=element_text(angle=90))
```

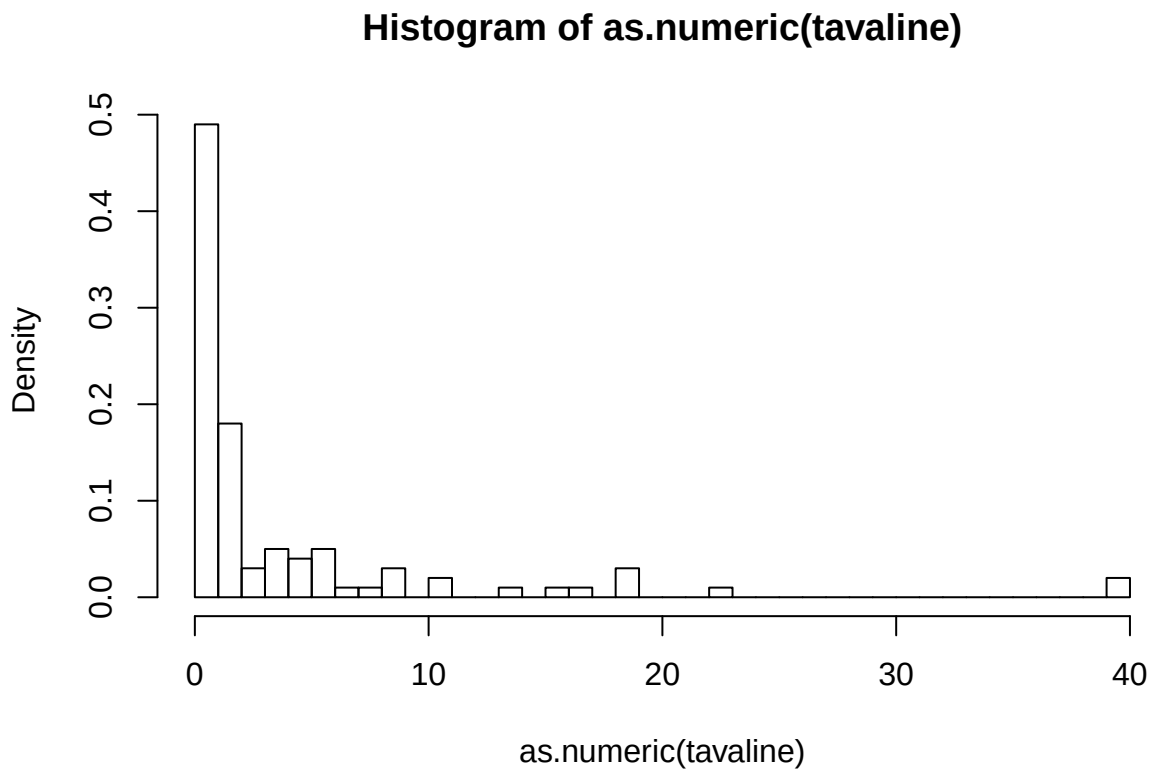



```
nrow(subset(hinnangud_permut,pvalues<0.0005)) #0
```

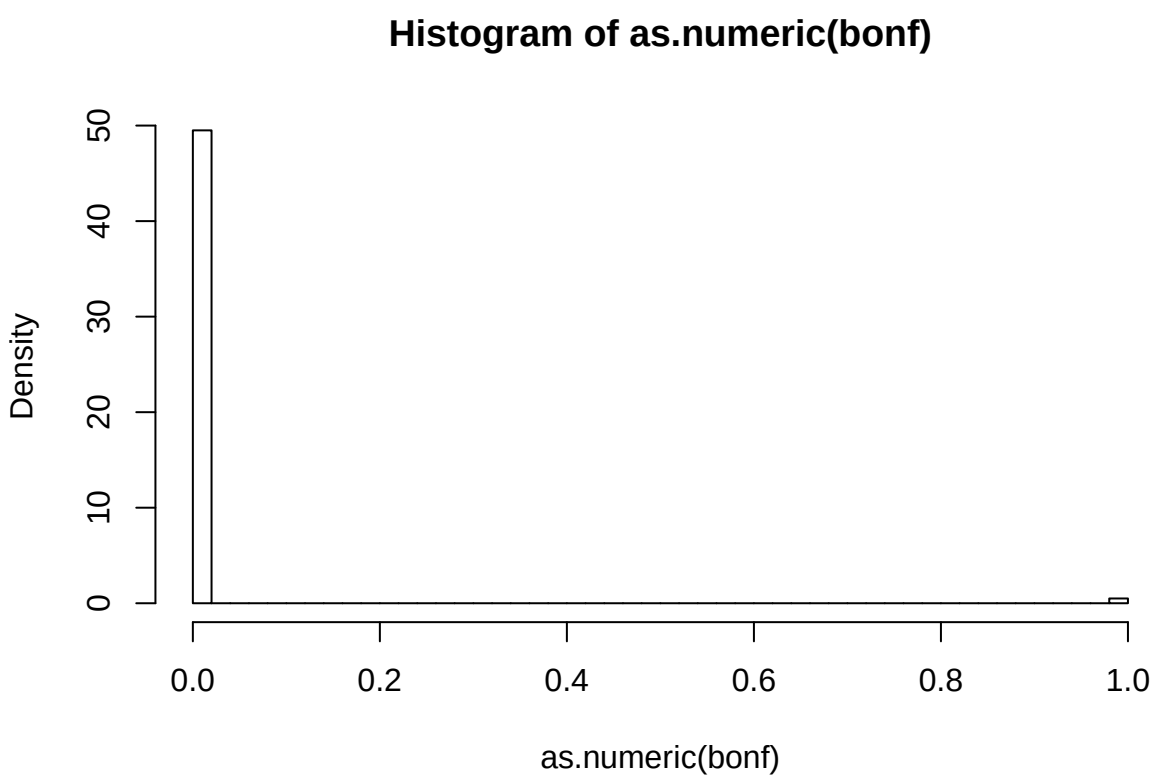
```
## [1] 0
```

- Mitut olulist p-väärtust oleksid oodanud kummalgi juhul? Selgita. Esimesel juhul 106 x 0,05 = 5.3, teisel juhul 106 x 0.0005 = 0.053
- (1 boonuspunkt) Korda permuteerimist 100 korral ning tee kokkuvõte tulemustest.

```
tavaline=list()
bonf=list()
#aeglane, 20min teeb!
for (i in 1:100) {

    biomarkerid$s5_permut=sample(biomarkerid$s5, replace=T)
    pvaartused=estimate_significance(formula0="s5_permut ~ sugu + vanusegrupp",
                                     biomarkers=names(biomarkerid)[6:111],
                                     data=biomarkerid)[,2]
    tavaline[i]=summary(pvaartused<0.05)[3]
    bonf[i]=summary(pvaartused<0.0005)[3]
}

hist(as.numeric(tavaline),  freq=FALSE, breaks=40)
```




```
hist(as.numeric(bonf), freq=FALSE, breaks=40)
```




```
simul=data.frame(as.numeric(tavaline),as.numeric(bonf))
summary(simul)
```

```
##  as.numeric.tavaline. as.numeric.bonf.
##  Min.   : 0.00        Min.   :0.00
##  1st Qu.: 1.00        1st Qu.:0.00
##  Median : 2.00        Median :0.00
##  Mean   : 4.12        Mean   :0.01
##  3rd Qu.: 4.25        3rd Qu.:0.00
##  Max.   :40.00        Max.   :1.00
```

Ülesanne 8 (1 punkt) - alternatiiv Bonferroni korrektsioonile

Ülesandes 3 nägime, et mitmed biomarkerid on omavahel tugevalt korreleeritud. Niisiis võib Bonferroni korrektsioon osutuda praegu liiga rangeks. Alternatiivselt võiksime leida, kui suur on meie andmestikus mittekorreleeritud tunnuste arv, ning teha nende arvu järgi Bonferroni korrektsiooni. Selleks, et leida andmestiku nn "efektiivne dimensionaalsus", kasuta PCA-d.

Juhised:

- Rakenda andmestikul PCA-d ning leia, mitu peakomponenti seletavad näiteks 99% variatsioonist.
- Leitud peakomponentide arv näitabki ligikaudu sõltumatute tunnuste arvu meie andmestikus. Tee Bonferroni korrektsioon selle arvu järgi. Millise p-väärtuse piiri saad?

```r
pca = prcomp(biomarkerid[, 6:111])
pca_sum=summary(pca)
pca_sum
```

```
## Importance of components:
##                           PC1    PC2    PC3     PC4     PC5    PC6     PC7
## Standard deviation     6.0485 4.1597 3.2737 2.17552 1.92864 1.5013 1.27880
## Proportion of Variance 0.4172 0.1973 0.1222 0.05397 0.04242 0.0257 0.01865
## Cumulative Proportion  0.4172 0.6145 0.7368 0.79073 0.83315 0.8589 0.87750
##                            PC8     PC9   PC10    PC11   PC12    PC13
## Standard deviation     1.15348 1.07317 1.0386 0.94700 0.9077 0.86415
## Proportion of Variance 0.01517 0.01313 0.0123 0.01023 0.0094 0.00852
## Cumulative Proportion  0.89268 0.90581 0.9181 0.92834 0.9377 0.94625
##                           PC14    PC15    PC16    PC17    PC18    PC19
## Standard deviation     0.7999 0.74099 0.71558 0.66602 0.62172 0.60976
## Proportion of Variance 0.0073 0.00626 0.00584 0.00506 0.00441 0.00424
## Cumulative Proportion  0.9536 0.95981 0.96565 0.97071 0.97512 0.97936
##                           PC20    PC21    PC22    PC23    PC24    PC25
## Standard deviation     0.54121 0.49441 0.42698 0.39458 0.37008 0.35348
## Proportion of Variance 0.00334 0.00279 0.00208 0.00178 0.00156 0.00142
## Cumulative Proportion  0.98270 0.98548 0.98756 0.98934 0.99090 0.99232
##                           PC26    PC27    PC28    PC29    PC30    PC31
## Standard deviation     0.33269 0.32290 0.27438 0.24944 0.24144 0.20580
## Proportion of Variance 0.00126 0.00119 0.00086 0.00071 0.00066 0.00048
## Cumulative Proportion  0.99359 0.99478 0.99563 0.99634 0.99701 0.99749
##                           PC32    PC33    PC34    PC35    PC36    PC37
## Standard deviation     0.17979 0.17099 0.14562 0.13023 0.11606 0.09649
## Proportion of Variance 0.00037 0.00033 0.00024 0.00019 0.00015 0.00011
## Cumulative Proportion  0.99786 0.99819 0.99844 0.99863 0.99878 0.99889
##                           PC38    PC39    PC40    PC41    PC42    PC43
## Standard deviation     0.08493 0.08409 0.07503 0.07068 0.06557 0.05505
## Proportion of Variance 0.00008 0.00008 0.00006 0.00006 0.00005 0.00003
## Cumulative Proportion  0.99897 0.99905 0.99912 0.99917 0.99922 0.99926
##                           PC44    PC45    PC46    PC47    PC48    PC49
## Standard deviation     0.05427 0.05304 0.04662 0.04512 0.04227 0.04051
## Proportion of Variance 0.00003 0.00003 0.00002 0.00002 0.00002 0.00002
## Cumulative Proportion  0.99929 0.99932 0.99935 0.99937 0.99939 0.99941
##                           PC50    PC51    PC52    PC53    PC54    PC55
## Standard deviation     0.03851 0.03827 0.03663 0.03590 0.03518 0.03506
## Proportion of Variance 0.00002 0.00002 0.00002 0.00001 0.00001 0.00001
## Cumulative Proportion  0.99943 0.99944 0.99946 0.99947 0.99949 0.99950
##                           PC56    PC57    PC58    PC59    PC60    PC61
## Standard deviation     0.03446 0.03365 0.03344 0.03288 0.03271 0.03229
## Proportion of Variance 0.00001 0.00001 0.00001 0.00001 0.00001 0.00001
## Cumulative Proportion  0.99951 0.99953 0.99954 0.99955 0.99956 0.99958
##                           PC62    PC63    PC64    PC65    PC66    PC67
## Standard deviation     0.03195 0.03157 0.03133 0.03114 0.03101 0.03095
## Proportion of Variance 0.00001 0.00001 0.00001 0.00001 0.00001 0.00001
## Cumulative Proportion  0.99959 0.99960 0.99961 0.99962 0.99963 0.99964
##                           PC68    PC69    PC70    PC71    PC72    PC73
## Standard deviation     0.03082 0.03080 0.03052 0.03038 0.03030 0.03026
## Proportion of Variance 0.00001 0.00001 0.00001 0.00001 0.00001 0.00001
## Cumulative Proportion  0.99965 0.99967 0.99968 0.99969 0.99970 0.99971
```

```
##                            PC74    PC75    PC76    PC77    PC78    PC79
## Standard deviation     0.03014 0.03009 0.02982 0.02963 0.02952 0.02950
## Proportion of Variance 0.00001 0.00001 0.00001 0.00001 0.00001 0.00001
## Cumulative Proportion  0.99972 0.99973 0.99974 0.99975 0.99976 0.99977
##                            PC80    PC81    PC82    PC83    PC84    PC85
## Standard deviation     0.02941 0.02914 0.02900 0.02889 0.02878 0.02868
## Proportion of Variance 0.00001 0.00001 0.00001 0.00001 0.00001 0.00001
## Cumulative Proportion  0.99978 0.99979 0.99980 0.99981 0.99982 0.99983
##                            PC86    PC87    PC88    PC89    PC90    PC91
## Standard deviation     0.02864 0.02850 0.02836 0.02824 0.02807 0.02796
## Proportion of Variance 0.00001 0.00001 0.00001 0.00001 0.00001 0.00001
## Cumulative Proportion  0.99984 0.99984 0.99985 0.99986 0.99987 0.99988
##                            PC92    PC93    PC94    PC95    PC96    PC97
## Standard deviation     0.02776 0.02770 0.02748 0.02734 0.02730 0.02716
## Proportion of Variance 0.00001 0.00001 0.00001 0.00001 0.00001 0.00001
## Cumulative Proportion  0.99989 0.99990 0.99991 0.99992 0.99992 0.99993
##                            PC98    PC99   PC100   PC101   PC102   PC103
## Standard deviation     0.02691 0.02686 0.02678 0.02654 0.02621 0.02533
## Proportion of Variance 0.00001 0.00001 0.00001 0.00001 0.00001 0.00001
## Cumulative Proportion  0.99994 0.99995 0.99996 0.99996 0.99997 0.99998
##                           PC104   PC105   PC106
## Standard deviation     0.02510 0.02446 0.02272
## Proportion of Variance 0.00001 0.00001 0.00001
## Cumulative Proportion  0.99999 0.99999 1.00000
```

Umbes 24 muutujat selgitab 99% variatsioonist ära. sellel juhul saan bonferroni korrektsiooni p-väärtuseks 0.05/24=0.00208.

Boonusülesanne 1 (2 punkti) - usaldusintervallid kordajate jaoks

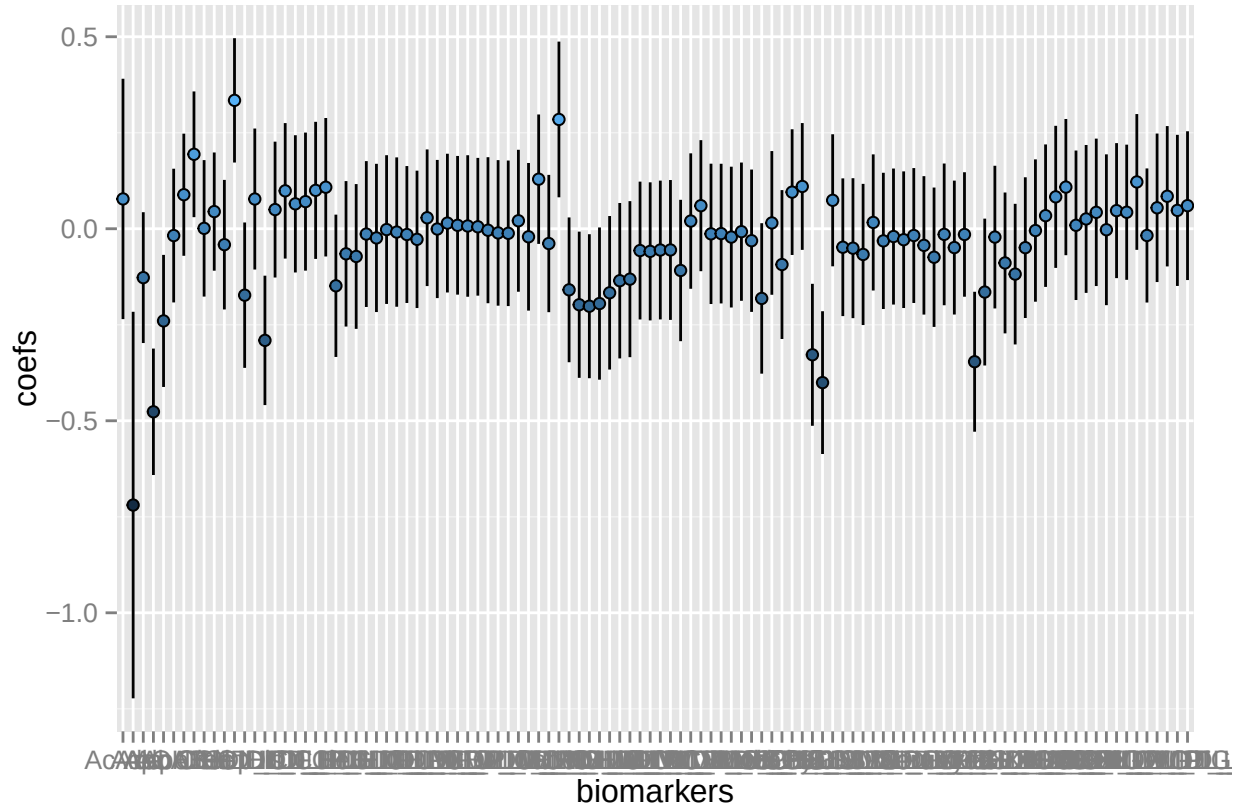
(1 boonuspunkt) Muuda funktsiooni estimate_significance selliselt, et iga biomarkeri kordajale arvutad ka 95% usaldusintervalli. Võid kasutada normaaljaotusel põhinevat lähendit ning arvutada selle kordaja_hinnang +- 1.96 * SE, kus SE on summary(model) väljundis toodud Std. Error. Funktsiooni tagastatavas andmetabelis peaksid nüüd olema ka veerud lower ja upper.

```
estimate_significance2 = function(formula0, biomarkers, data){
    coefs=list()
    pvalues=list()
    upper=list()
    lower=list()
    for (i in 1:length(biomarkers)) {
        formula = paste(formula0, biomarkers[i], sep=" + ")
        model = glm(formula, family=binomial, data=data)
        summary_table = coef(summary(model))
        pvalues[i] = summary_table[nrow(summary_table), 4]
        coefs[i] = summary_table[nrow(summary_table), 1]
        upper[i]=summary_table[nrow(summary_table), 1]+summary_table[nrow(summary_table), 2]*1.96
        lower[i]=summary_table[nrow(summary_table), 1]-summary_table[nrow(summary_table), 2]*1.96

    }
    pvalues=round(as.numeric(pvalues), 10)
    coefs=round(as.numeric(coefs), 6)
    upper=round(as.numeric(upper), 6)
```

```
    lower=round(as.numeric(lower), 6)
    tulemid=data.frame(biomarkers, pvalues, coefs, upper, lower)
    return(tulemid)
}

#test
tulem2=estimate_significance2(formula0="s5 ~ sugu + vanusegrupp",
                      biomarkers=names(biomarkerid)[6:111], data=biomarkerid)
```

(1 boonuspunkt) Visualiseeri tulemust (näiteks iga biomarkeri kohta näita kordaja hinnangut koos usaldusintervalliga).

```
ggplot(tulem2, aes(x=biomarkers, ymax=upper, ymin=lower, y=coefs))+
    geom_pointrange(aes(fill=coefs), shape=21)+
    guides(fill=FALSE)
```



Ülesanne 9 (1 punkt) - forward selection

Artiklis on kirjeldatud mudeli koostamist järgnevalt:

For biomarker discovery in the Estonian Biobank cohort, a multivariate model was derived in a forward stepwise fashion (Figure 2). First, the biomarker leading to the smallest p-value in the model adjusted for age and sex only was included as a predictor. Subsequently, the biomarker leading to the smallest p-value in the multivariate model adjusted for age, sex, and the first biomarker was included in the prediction model. The

process was repeated until no additional biomarkers were significant at the Bonferroni-corrected threshold of p<0.0005, accounting for testing of 106 candidate biomarkers.

Eelmistes ülesannetes leidsid kõige olulisema p-väärtusega biomarkeri. Jätka nüüd forward selection-iga:

- Lisa leitud biomarker mudelisse ning lähtu mudelist s5 ~ sugu + vanusegrupp + kõige_olulisem_biomarker
- Kasuta funktsiooni estimate_significance ning leia nüüd järgmine biomarker, mis mudelisse lisada.
- Jätka senikaua, kuni mudelisse lisatavad biomarkerite p-väärtused on väiksemad kui sinu määratud piir.

Artiklis saadi sellise protsessi tulemusena 4 olulist biomarkerit: Alb, VLDL_D, Gp, Cit. Kas said samasugused?

```
esimene=estimate_significance(formula0="s5 ~ sugu + vanusegrupp",
                        biomarkers=names(biomarkerid)[6:111], data=biomarkerid)
head(esimene[order(esimene$pvalues),])
```

```
##     biomarkers      pvalues     coefs
## 89         Alb 0.0000000139 -0.476595
## 33     S_HDL_P 0.0000239417 -0.400615
## 92          Gp 0.0000518255  0.334326
## 102        Val 0.0001957651 -0.346219
## 35     S_HDL_L 0.0005011386 -0.328187
## 104        His 0.0007210405 -0.290675
```

```
teine=estimate_significance(formula0="s5 ~ sugu + vanusegrupp + Alb",
                            biomarkers=names(biomarkerid)[6:111], data=biomarkerid)
head(teine[order(teine$pvalues),])
```

```
##     biomarkers      pvalues     coefs
## 89         Alb 0.0000000139 -0.476595
## 92          Gp 0.0000392554  0.337842
## 88         Ace 0.0016021510 -0.822712
## 102        Val 0.0029253764 -0.272515
## 33     S_HDL_P 0.0070881530 -0.267325
## 1        LDL_D 0.0224441839  0.231824
```

```
kolmas=estimate_significance(formula0="s5 ~ sugu + vanusegrupp + Alb + Gp",
                           biomarkers=names(biomarkerid)[6:111], data=biomarkerid)
head(kolmas[order(kolmas$pvalues),])
```

```
##     biomarkers      pvalues     coefs
## 89         Alb 0.0000392554  0.337842
## 92          Gp 0.0000392554  0.337842
## 106        Cit 0.0003184735  0.328433
## 33     S_HDL_P 0.0004225684 -0.356018
## 37      VLDL_D 0.0013495008 -0.334614
## 1        LDL_D 0.0047917531  0.272144
```

```
neljas=estimate_significance(formula0="s5 ~ sugu + vanusegrupp + Alb + Gp + Cit",
                             biomarkers=names(biomarkerid)[6:111], data=biomarkerid)
head(neljas[order(neljas$pvalues),])
```

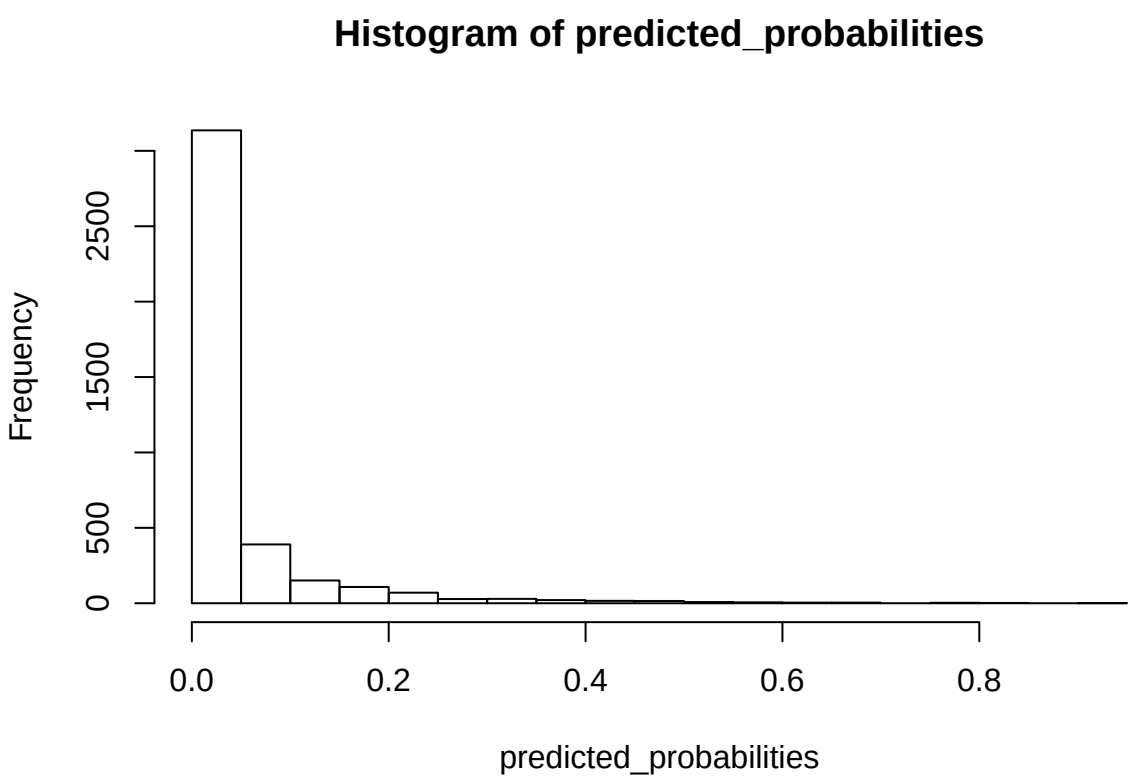
```
##     biomarkers      pvalues     coefs
## 89         Alb 0.0003184735  0.328433
## 92          Gp 0.0003184735  0.328433
## 106        Cit 0.0003184735  0.328433
## 37      VLDL_D 0.0022948808 -0.316779
## 33     S_HDL_P 0.0039993055 -0.296757
## 56      IDL_FC 0.0067108449  0.258449
```

Sain samad, peale VLDL_D (mul pole ka originaalandmed, need on nati permuteeritud).

Ülesanne 10 (1 punkt) - surma tõenäosuse prognoosimine

Eelmise ülesande tulemusena on sul nüüd olemas lõplik mudel, mis võtab arvesse kõik, mis on oluline surma tõenäosuse prognoosimiseks. Prognoosi iga andmestikus oleva inimese kohta tema tõenäosust surra 5 aasta jooksul ja visualiseeri tulemust (näiteks histogrammi abil)

```
model = glm(s5 ~ sugu + vanusegrupp + Alb + Gp + Cit,
            family=binomial, data=biomarkerid)
newdata = biomarkerid[,c("s5", "sugu", "vanusegrupp", "Alb", "Gp", "Cit")]
# On oluline, et newdata sisaldaks kindlasti kõik need veerud, mida on
#vaja prognoosimisel
predicted_probabilities = predict(model, newdata=newdata, type = "response")
hist(predicted_probabilities)
```



Boonusülesanne 2 (2 punkti) - prognooside täpsus

Eelmises ülesandes prognoosisid surma tõenäosust. Aga mida hakkab tavainimene peale tõenäosusega? Olgem ikka konkreetsed, kas siis sureb 5 aasta jooksul või mitte.

Selleks otsusta piir, millisest väiksemad tõenäosused klassifitseerid ei sure ja suuremad tõenäosused sureb. Kasutades seda piiri ning eelmises ülesandes kirjutatud funktsiooni, arvuta kõigi andmestikus olnud inimeste jaoks 5 aasta jooksul suremise prognoos (justkui meil poleks olnud teada tunnuse s5 väärtus).

Milline on sinu prognooside täpsus (st kui suur osa prognoosidest langes kokku tunnuse s5 väärtusega)?

Võrdlusmomendi saamiseks paku välja veel mingi teine, naiivne klassifitseerija (see võib põhineda ükskõik kui lihtsal reeglil). Milline on selle täpsus?

```
#teeme 0.5 peale
proov=ifelse(predicted_probabilities<0.5, 0, 1)
vordlus=data.frame(biomarkerid$s5, proov)
#kui palju ma puusse panin
summary(factor(vordlus$biomarkerid.s5-vordlus$proov)) #pole paha, 4.158% viga
```

```
##   -1    0    1
##    9 3826  157
```

```
#kui palju surnute osas puusse panin
surnud=subset(vordlus, biomarkerid.s5==1)
summary(factor(surnud$biomarkerid.s5-surnud$proov)) #väga perses mudel
```

```
##   0   1
##  19 157
```

```
#teeme confusionmatrixi
library(caret)
```

```
## Loading required package: lattice
```

```
confusionMatrix(vordlus$proov, biomarkerid$s5)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 3807  157
##          1    9   19
##
##                Accuracy : 0.9584
##                  95% CI : (0.9518, 0.9644)
##     No Information Rate : 0.9559
##     P-Value [Acc > NIR] : 0.2336
##
##                   Kappa : 0.1763
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.9976
##             Specificity : 0.1080
##          Pos Pred Value : 0.9604
##          Neg Pred Value : 0.6786
##              Prevalence : 0.9559
##          Detection Rate : 0.9537
##    Detection Prevalence : 0.9930
##       Balanced Accuracy : 0.5528
##
##        'Positive' Class : 0
##
```

```
#Võrdlusmomendi saamiseks paku välja veel mingi teine, naiivne
#klassifitseerija (see võib põhineda ükskõik kui lihtsal reeglil).
#Milline on selle täpsus?
#kõige lihtsam, kuna surijaid oli vähe, ennustan kõigile, et jäävad elama
naive=data.frame(rep.int(0, length(vordlus$proov)))
naive$s5=biomarkerid$s5
summary(factor(naive$s5-naive$rep.int.0..length.vordlus.proov..)) #4,4% viga
```

```
##    0    1
## 3816  176
```

Täpne mudel, kuid kuna surnuid on vähe saaks hea mudeli, kui ennustaks kõigile, et jäävad elama. Surma ette ennustamiseks kehv mudel (seal pani enamikuga puusse).