Tartu Ülikool
Loodus- ja täppisteaduste valdkond
Matemaatika ja statistika instituut

Kristi Ernits

Logistiline regressioon ja klassifitseerimispuu binaarse tunnuse modelleerimisel

Matemaatilise statistika eriala
Bakalaureusetöö (9 EAP)

Juhendaja Natalja Lepik

Tartu 2017

Logistiline regressioon ja klassifitseerimispuu binaarse tunnuse modelleerimisel

Bakalaureusetöö

Kristi Ernits

Lühikokkuvõte. Tihti on uurija huvitatud binaarse tunnuse seose modelleerimisest teiste tunnustega. Käesoleva bakalaureusetöö eesmärk on kirjeldada ja omavahel võrrelda logistilist regressiooni ning klassifitseerimispuu meetodit kaheväärtuselise uuritava tunnuse modelleerimisel. Töö teooriaosas kirjeldatakse lühidalt levinumaid meetodeid binaarse tunnuse modelleerimiseks, täpsem ülevaade antakse logistilisest regressioonimudelist ning klassifitseerimispuu meetodist. Praktilises osas rakendatakse nii logistilist regressiooni kui ka klassifitseerimispuu meetodit reaalsetel andmetel südame- ja veresoonkonna haiguste esinemise prognoosimiseks. Töö viimases osas viiakse läbi simuleerimisülesanne ning võrreldakse nimetatud kahte meetodit.

Märksõnad: üldistatud lineaarsed mudelid, puud (mat.), klassifitseerimine

CERCS teaduseriala: P160 Statistika, operatsioonanalüüs, programmeerimine, finants- ja kindlustusmatemaatika

Using logistic regression and classification tree method for modelling a binary variable

Bachelor's thesis

Kristi Ernits

Abstract. An investigator is often interested in modelling a binary dependent variable. The purpose of this bachelor's thesis is to describe and compare logistic regression and classification tree method for modelling a binary variable. In the theoretical part a brief overview of more common methods for modelling a binary variable is given, which is followed by a more detailed description of logistic regression and classification tree method. In the practical part both methods are used to estimate the occurrence of cardiovascular diseases. Finally, a simulation experiment is conducted to compare logistic regression and classification tree method.

Keywords: generalized linear models, trees (math.), classification

CERCS research specialisation: P160 Statistics, operation research, programming, actuarial mathematics

Sisukord

Sissejuhatus

Binaarse ehk kaheväärtuselise tunnuse seose modelleerimine teiste tunnustega pakub uurijale tihtipeale huvi. Näiteks soovitakse teada saada, millest sõltub inimese motivatsioon osaleda või mitte osaleda uuringus, mis iseloomustab allpool või ülalpool vaesuse piiri elavaid leibkondi või mis mõjutab inimesel teatud haiguse esinemist või mitteesinemist. Binaarsed tunnused tekivad kõikides küsitlustes, mis sisaldavad jah/ei vastustega küsimusi. Mõnikord luuakse neid ka arvtunnuste baasil, arvtunnuse väärtuste piirkonna kaheks osaks jagades.

Kaheväärtuselisele tunnusele mudeli leidmiseks eksisteerib mitu traditsioonilist statistilist meetodit, millest kõige tuntum on logistiline regressioon. Viimasel ajal on samuti palju kasutust leidnud uuemad ehk masinõppe meetodid, mille hulka kuulub klassifitseerimispuu meetod.

Käesoleva bakalaureusetöö eesmärk on kirjeldada ja omavahel võrrelda logistilist regressiooni ning klassifitseerimispuu meetodit juhul, kui uuritav tunnus on binaarne. Logistiline regressioon on parameetriline meetod kaheväärtuselise tunnuse seose modelleerimiseks seletavate tunnustega. Klassifitseerimispuu on mitteparameetriline meetod kvalitatiivse tunnuse, sealhulgas binaarse tunnuse, modelleerimiseks.

Bakalaureusetöö esimeses peatükis kirjeldatakse binaarset tunnust ning lühidalt levinumaid meetodeid selle seose modelleerimiseks teiste tunnustega. Teises ja kolmandas osas antakse pikem ülevaade vastavalt logistilise regressiooni ning klassifitseerimispuu kasutamisest kaheväärtuselisele tunnusele mudeli leidmisel. Samuti rakendatakse mõlemat meetodit reaalsetel andmetel südame- ja veresoonkonna haiguste esinemise prognoosimiseks. Viimases peatükis viiakse artikli (Phipps ja Toth, 2012) põhjal läbi simuleerimisülesanne ja võrreldakse eelpool nimetatud kahte meetodit.

Töö kirjutamiseks on kasutatud tekstitöötlusprogrammi LaTeX ja statistilise analüüsi läbiviimiseks rakendustarkvara R.

1 Binaarne tunnus

Olgu uuritaval tunnusel kaks võimalikku väärtust ehk vaadeldakse binaarset tunnust. Tavapäraselt kasutatakse sellise tunnuse väärtuste kodeerimiseks arve 0 ja 1 nii, et arv 1 tähistab huvipakkuva sündmuse toimumist ning arv 0 mittetoimumist (Agresti, 2002: 5).

Näiteks on binaarsed tunnused sugu (väärtustega 0, kui tegu on naisega, ja 1, kui tegu on mehega) ning haiguse A esinemine (väärtustega 0, kui inimesel ei esine haigus A, ja 1, kui inimesel esineb haigus A).

Kaheväärtuseline tunnus tekib andmestikus ka juhul kui uuritakse, kas objekti kohta on vaatluse all oleva tunnuse info olemas ehk enamasti, kas isik on uuringu küsimusele vastanud või mitte. Sel juhul võidakse defineerida vastamist tähistav tunnus väärtustega 0, kui isik ei vastanud küsimusele, ja 1, kui isik vastas küsimusele. Sarnaselt defineeritakse ka tunnus vastaja uuringus osalemise kohta. Viimase põhjal leitud vastamismäära peetakse andmete kvaliteedi juures oluliseks näitajaks (Phipps ja Toth, 2012).

1.1 Tähistused

Antud alapunkti kirjutamisel on kasutatud õpikut (James, Witten, Hastie ja Tibshirani, 2013).

Tähistagu n valimimahtu ning p seletavate tunnuste arvu. Olgu antud andmestik, mis sisaldab ühte uuritavat kaheväärtuselist tunnust y, mille seost teiste p tunnusega soovitakse leida. Uuritava binaarse tunnuse väärtus i-ndal ($i = 1, \ldots, n$) objektil olgu y_i ning vektor kõigi objektide uuritava tunnuse väärtustega

$$\boldsymbol{y} = \left(y_1, y_2, \ldots, y_n\right)^T.$$

Seletavaid ehk argumenttunnuseid võib panna kirja maatriksi $\boldsymbol{X}$ abil, mille i-nda rea j-nda veeru element on x_{ij} ehk

$$\boldsymbol{X} = \begin{pmatrix} x_{11} & x_{12} & \ldots & x_{1p} \\ x_{21} & x_{22} & \ldots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \ldots & x_{np} \end{pmatrix}.$$

Seega saab j-nda ($j = 1, \ldots, p$) seletava tunnuse kõiki mõõdetud väärtusi esitada vektorina

$$\boldsymbol{X_j} = \Big(x_{1j}, x_{2j}, \ldots, x_{nj}\Big)^T.$$

Binaarse tunnuse seose leidmisel teiste tunnustega pakub huvi nii mudeli prognoos y_i väärtusele kui ka prognoos vaadeldava sündmuse esinemise tõenäosusele $p_i = \mathrm{P}(Y_i = 1)$, kus

$$Y_i = \begin{cases} 0, & \text{tõenäosusega } 1 - p_i, \\ 1, & \text{tõenäosusega } p_i \end{cases}$$

on uuritava tunnuse realisatsioonile y_i vastav Bernoulli jaotusega $Be(p_i)$ juhuslik suurus, mille keskväärtus on $EY_i = p_i$. Tõenäosuseid p_i ($i = 1, \ldots, n$) saab kujutada vektorina

$$\boldsymbol{p} = \Big(p_1, p_2, \ldots, p_n\Big)^T.$$

1.2 Binaarse tunnuse seose modelleerimine

Kaheväärtuselisele tunnusele mudeli leidmiseks kasutatakse erinevaid meetodeid, mille hulgas on nii traditsioonilisi statistilisi kui ka uuemaid masinõppe meetodeid. Tuntumateks traditsioonilisteks meetoditeks on lineaarne regressioon, logistiline regressioon ning probit-regressioon. Uuemad meetodid on näiteks otsustuspuud, mis jagunevad regressiooni- ja klassifitseerimispuudeks, ning juhuslik mets.

1.2.1 Klassikalised meetodid

Antud alajaotuse kirjutamisel on kasutatud teost (Agresti, 2002), kui ei ole viidatud teisiti.

Üldistatud lineaarse mudeli korral on üheks eelduseks, et uuritava tunnuse jaotus pärineb eksponentsiaalsest jaotuste perest. See tähendab, et uuritava tunnuse jaotuse tihedus- või tõenäosusfunktsioon $p(\theta, y)$ on avaldatav kujul

$$p(\theta, y) = \exp\left(A(\theta) \cdot B(y) + C(\theta) + D(y)\right),$$

kus θ on jaotuse parameeter, y on funktsiooni argument ja A, B, C ja D on etteantud muutujate funktsioonid, ning funktsiooni $p(\theta, y)$ diferentseerimine parameetri θ järgi ja integreerimine (või summeerimine) argumendi y järgi on vahetatavad operatsioonid

(Parring, 1989: 53, 63). Juhusliku suuruse Y_i jaotus $Be(p_i)$ on pärit eksponentsiaalsest jaotuste perest. Üldistatud lineaarse mudeli korral kasutatakse ka funktsiooni uuritava tunnuse jaotuse keskväärtusest ehk seosefunktsiooni sidumaks omavahel nimetatud jaotuse keskväärtus ja argumenttunnused. Eeldatakse, et lineaarselt on seotud funktsioon keskväärtusest ning seletavad tunnused. Seega sõltub seosefunktsiooni kuju uuritava tunnuse jaotusest. Eelneva alapunkti põhjal ning tähistades seosefunktsiooni tähega g saadakse binaarse uuritava tunnuse korral üldistatud lineaarse mudeli kuju

$$\boldsymbol{g}(\boldsymbol{p}) = \beta_0 \mathbf{1} + \beta_1 \boldsymbol{X_1} + \ldots + \beta_p \boldsymbol{X_p} + \boldsymbol{\varepsilon}, \tag{1}$$

kus $\boldsymbol{g}(\boldsymbol{p}) = (g(p_1), g(p_2), \ldots, g(p_n))^T$, $\mathbf{1} = (1, \ldots, 1)^T$ on n-mõõtmeline vektor, hinnatavad parameetrid on $\beta_0, \beta_1, \ldots, \beta_p$ ja $\boldsymbol{\varepsilon} = (\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_n)^T$ on juhuslike vigade vektor (James jt, 2013: 16).

Lineaarne regressioon

Juhul kui seosefunktsioon on samasusteisendus ehk $g(p_i) = p_i$, nimetatakse üldistatud lineaarset mudelit lineaarseks regressioonimudeliks. Teades tundmatute parameetrite hinnanguid $\hat{\beta}_0, \hat{\beta}_1, \ldots, \hat{\beta}_p$, leitakse prognoosid tõenäosustele kasutades valemit

$$\hat{\boldsymbol{p}} = \hat{\beta}_0 \mathbf{1} + \hat{\beta}_1 \boldsymbol{X_1} + \ldots + \hat{\beta}_p \boldsymbol{X_p},$$

kus $\hat{\boldsymbol{p}} = (\hat{p}_1, \hat{p}_2, \ldots, \hat{p}_n)^T$. Tundmatute kordajate hinnangud leitakse vähimruutude meetodil (James jt, 2013: 72).

Lineaarse regressioonimudeli abil binaarse tunnuse prognoosimisel pole aga tagatud, et leitud hinnangud $\hat{p}_1, \hat{p}_2, \ldots, \hat{p}_n$ satuvad kõik lõiku $[0, 1]$. Seega tõlgendatakse neid jämedate hinnangutena tõenäosustele $p_1, p_2, \ldots, p_n$ (James jt, 2013: 130).

Vältimaks olukorda, et mõne tõenäosuse hinnang asub väljaspool lõiku $[0, 1]$ kasutatakse binaarsele tunnusele mudeli leidmisel lõigul $[0, 1]$ monotoonset diferentseeruvat seosefunktsiooni, mille muutumispiirkond on reaalarvude hulk $\mathbb{R}$. Nendele tingimustele vastavad funktsioonid on näiteks

- logit-funktsioon

$$\operatorname{logit}(p_i) = \ln\left(\frac{p_i}{1 - p_i}\right),$$

- probit-funktsioon

$$\text{probit}\left(p_i\right) = \Phi^{-1}\left(p_i\right),$$

kus Φ on standardse normaaljaotuse jaotusfunktsioon.

Logistiline regressioon

Üldistatud lineaarset mudelit, mille seosefunktsioon on logit-funktsioon, nimetatakse logistiliseks regressioonimudeliks. Sel juhul leitakse mudeli kordajate hinnanguid enamasti suurima tõepära meetodil. Prognoosid tõenäosustele $p_1, p_2, \ldots, p_n$ arvutatakse aga järgnevalt:

$$\hat{p}_i = \frac{\exp\left(\hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \ldots + \hat{\beta}_p x_{ip}\right)}{1 + \exp\left(\hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \ldots + \hat{\beta}_p x_{ip}\right)}. \tag{2}$$

Logistilist regressiooni on põhjalikumalt kirjeldatud peatükis 2.

Probit-regressioon

Kui mudelis (1) on seosefunktsioon probit-funktsioon, siis nimetatakse seda probit-regressiooni mudeliks. Mudeli tundmatuid parameetreid hinnatakse üldjuhul suurima tõepära meetodil. Huvipakkuva sündmuse tõenäosuste hinnangud leitakse valemi

$$\hat{p}_i = \Phi\left(\hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \ldots + \hat{\beta}_p x_{ip}\right)$$

abil.

1.2.2 Masinõppe meetodid

Vaadeldav alajaotus põhineb õpikul (James jt, 2013), kui pole märgitud teisiti.

Otsustuspuude meetodite korral jaotatakse objektid argumenttunnuste võimalike väärtuste põhjal erinevatesse piirkondadesse. Segmenteerimise kriteeriumite põhjal koostatakse otsustuspuu, kus objektid jagunevad lõikumatute piirkondade $R_1, R_2, \ldots, R_m$ vahel (vt joonis 3 lk 25).

Kvantitatiivse uuritava tunnuse korral kasutatakse regressioonipuu meetodit ning kvalitatiivse sõltuva tunnuse korral klassifitseerimispuu meetodit. Üldiselt võib binaarset tunnust y vaadelda nii arvulise kui ka mittearvulisena, kuid antud töös käsitletakse täpsemalt binaarset tunnust y kvalitatiivse tunnusena, millel on kaks võimalikku väärtust

ehk klassi. Põhjuseks on regressiooni- ja klassifitseerimispuu meetodite erinevus uuritavale tunnusele prognoosi leidmisel. Põhjalikum selgitus on toodud allpool.

Regressioonipuu

Kuulugu huvipakkuv objekt piirkonda R_k, kus $k \in \{1, \ldots, m\}$. Uuritava tunnuse väärtuse prognoos sellele objektile (tähistatakse $\hat{y}_{R_k}$) leitakse regressioonipuu korral piirkonda R_k kuuluvate objektide tunnuse y väärtuste aritmeetilise keskmisena:

$$\hat{y}_{R_k} = \frac{1}{|R_k|} \sum_{i \in R_k} y_i, \tag{3}$$

kus $|R_k|$ on piirkonda R_k kuuluvate objektide arv kasutatud andmestikus. Regressioonipuu korral jaotatakse kõik objektid lõikumatutesse piirkondadesse $R_1, R_2, \ldots, R_m$ nii, et jääkide ruutude summa (*residual sum of squares*)

$$RSS = \sum_{k=1}^{m} \sum_{i \in R_k} (y_i - \hat{y}_{R_k})^2$$

oleks minimaalne.

Saadud prognoosid (3) kuuluvad lõiku $[0, 1]$ ning binaarse tunnuse korral tekib küsimus, kuidas jaotada saadud hinnangud kahe klassi, 0 ja 1, vahel. Analoogiline probleem ilmneb huvipakkuva sündmuse toimumise tõenäosusele hinnangu leidmisel. Üldiselt ei rakendata regressioonipuud kaheväärtuselise uuritava tunnuse korral.

Klassifitseerimispuu

Klassifitseerimispuu korral kasutatakse y_i, $i \in R_k$, hinnanguna sõltuva tunnuse sagedaimat väärtust ehk moodi piirkonnas R_k. Kahe klassiga uuritava tunnuse korral on selleks

$$\hat{y}_{R_k} = \underset{i \in R_k}{\mathrm{Mod}}(y_i) = \begin{cases} 0, & \text{kui } \dfrac{1}{|R_k|} \displaystyle\sum_{i \in R_k} y_i < \dfrac{1}{2}, \\ 1, & \text{kui } \dfrac{1}{|R_k|} \displaystyle\sum_{i \in R_k} y_i > \dfrac{1}{2}. \end{cases} \tag{4}$$

Juhul, kui piirkonnas R_k on mõlema väärtusega vaatluseid võrdselt ehk $\dfrac{1}{|R_k|} \displaystyle\sum_{i \in R_k} y_i = \dfrac{1}{2}$, valitakse hinnang, 0 või 1, uuritavale tunnusele juhuslikult (Ripley, 2016). Huvipakkuva sündmuse toimumise tõenäosuse hinnang on ühtede osakaal antud piirkonnas R_k:

$$\hat{p}_{R_k} = \frac{1}{|R_k|} \sum_{i \in R_k} y_i. \tag{5}$$

Klassifitseerimispuu koostamisel leitakse piirkonnad $R_1, R_2, \ldots, R_m$ minimeerides näiteks klassifitseerimisviga, Gini indeksit või hälbimust. Põhjalikumalt on klassifitseerimispuu meetodit kirjeldatud peatükis 3.

Puudel põhinevad meetodid on kergesti interpreteeritavad ja nende põhjal saadud mudeleid on väga lihtne tõlgendada. Hinnangute täpsuse poolest jäävad otsustuspuudel põhinevad meetodid alla keerukamatele masinõppe meetoditele, näiteks juhusliku metsa meetodile.

Juhuslik mets

Juhusliku metsa meetodi korral konstrueeritakse B otsustuspuud (regressiooni- või klassifitseerimispuud). Iga puu koostamiseks võetakse p seletava tunnuse seast juhuslikult $s \approx \sqrt{p}$ tunnust, mille põhjal luuakse otsustuspuu. Leitud B otsustuspuu alusel jagunevad seletavate tunnuste väärtused a piirkonda R_l^{mets} $(l = 1, \ldots, a)$.

Uuritava tunnuse hinnang piirkonda R_l^{mets} kuuluvatele objektidele leitakse järgnevalt:

$$\hat{y}_{R_l^{\text{mets}}} = \frac{1}{B} \sum_{b=1}^{B} \hat{y}_{b, R_l^{\text{mets}}},$$

kus $\hat{y}_{b, R_l^{\text{mets}}}$ on b-nda otsustuspuuga saadud uuritava tunnuse prognoos piirkonda R_l^{mets} kuuluvatele objektidele. Rakendades otsustuspuudena klassifitseerimispuid saadakse leida sündmuse toimumise tõenäosuse hinnang piirkonda R_l^{mets} kuuluvatele objektidele:

$$\hat{p}_{R_l^{\text{mets}}} = \frac{1}{B} \sum_{b=1}^{B} \hat{p}_{b, R_l^{\text{mets}}},$$

kus $\hat{p}_{b, R_l^{\text{mets}}}$ on b-nda klassifitseerimispuuga leitud huvipakkuva sündmuse toimumise tõenäosuse hinnang piirkonda R_l^{mets} kuuluvatele objektidele.

1.2.3 Mudeli täpsuse mõõtmine

Antud alajaotuse kirjutamisel on kasutatud õpikut (James jt, 2013).

Modelleerimisel huvitutakse, et statistiline meetod oleks võimalikult täpne ehk mudeliga leitud hinnangud oleksid lähedased tunnuse tegelikele väärtustele. Meetodi täpsuse mõõtmiseks kasutatakse näiteks mudeli ruutkeskmist viga või klassifitseerimisviga.

Huvitudes kaheväärtuselise tunnuse modelleerimisel prognoosist uuritava tunnuse väärtusele y_i, leitakse mudeli täpsuse mõõtmiseks klassifitseerimisviga (*classification error rate*)

$$E = \frac{1}{n}\sum_{i=1}^{n} I(y_i \neq \hat{y}_i), \tag{6}$$

kus $I(y_i \neq \hat{y}_i) = 1$, kui $y_i \neq \hat{y}_i$ ehk vaatlus on valesti klassifitseeritud ja $I(y_i \neq \hat{y}_i) = 0$, kui $y_i = \hat{y}_i$. Kui vaadeldakse prognoosi huvipakkuva sündmuse toimumise tõenäosusele p_i, siis mõõdetakse mudeli täpsust ruutkeskmise vea (*mean squared error*)

$$MSE = \frac{1}{n}\sum_{i=1}^{n} (y_i - \hat{p}_i)^2 \tag{7}$$

abil. Nii klassifitseerimisvea kui ka ruutkeskmise vea väärtus on väike, kui mudeliga leitud hinnangud on lähedased tegelikele tunnuse väärtustele ning suur, kui mõne vaatluse korral erineb hinnang märgatavalt tegelikust väärtusest.

Näitajad (6) ja (7) arvutatakse mudeli koostamiseks kasutatud andmete põhjal. Üldiselt pole aga uurija huvitatud sellest, kui hästi töötab mudel juba kasutatud andmetel. Pigem soovitakse, et prognoosid oleksid võimalikult täpsed mudeli rakendamisel uutele andmetele, mida mudeli leidmisel ei kasutatud. Kui on antud m uut vaatlust, siis nende põhjal arvutatakse test-klassifitseerimisviga (*test error rate*)

$$E^t = \frac{1}{m}\sum_{i=1}^{m} I(y_i \neq \hat{y}_i), \tag{8}$$

ja test ruutkeskmine viga (*test MSE*)

$$MSE^t = \frac{1}{m}\sum_{i=1}^{m} (y_i - \hat{p}_i)^2, \tag{9}$$

kus $\hat{y}_i$ ja $\hat{p}_i$ on vastavalt prognoosid uue objekti uuritava tunnuse väärtusele ja huvipakkuva sündmuse esinemise tõenäosusele vanade andmete põhjal koostatud mudeliga. Mida väiksemad on nimetatud näitajad, seda täpsem on hinnangute leidmisel kasutatud mudel.

Uute andmete puudumisel saab mudeli täpsust hinnata ka teisiti, näiteks ristvalideerimise (*cross-validation*) abil. Kasutades k-kordset ristvalideerimist jagatakse mudeli leidmise jaoks kasutatud vaatlused juhuslikult k umbes sama suurusega gruppi. Praktikas jagatakse tihtipeale vaatlused $k = 5$ või $k = 10$ gruppi. Esmalt vaadeldakse esimest gruppi kui uute vaatluste hulka ning ülejäänud $k-1$ gruppi kuuluvate vaatluste põhjal

sobitatakse mudel. Seejärel arvutatakse esimese grupi vaatluste põhjal test-klassifitseerimisviga $\hat{E}_1^t$ ja test ruutkeskmine viga $\hat{MSE}_1^t$. Kirjeldatud protsessi korratakse k korda käsitledes igal korral uute vaatluste hulgana erinevat gruppi. Tulemusena saadakse k hinnangut test-klassifitseerimisveale, $\hat{E}_1^t, \hat{E}_2^t, \ldots, \hat{E}_k^t$, ja k hinnangut test ruutkeskmisele veale, $\hat{MSE}_1^t, \hat{MSE}_2^t, \ldots, \hat{MSE}_k^t$.

Ristvalideerimisel leitud vigade hinnangute põhjal saadakse test-klassifitseerimisviga k-kordsel ristvalideerimise meetodil:

$$E_{CV}^t(k) = \frac{1}{k}\sum_{i=1}^{k}\hat{E}_i^t \tag{10}$$

ja test ruutkeskmine viga k-kordsel ristvalideerimise meetodil:

$$MSE_{CV}^t(k) = \frac{1}{k}\sum_{i=1}^{k}\hat{MSE}_i^t. \tag{11}$$

Erinevaid mudeleid võrreldes eelistatakse mudelit, mille korral on viga k-kordsel ristvalideerimise meetodil väiksem.

2 Logistiline regressioon

Antud peatüki koostamisel on kasutatud teost (Agresti, 2002), kui pole viidatud teisiti.

Logistiline regressioon on üks parameetrilistest meetoditest binaarse tunnuse seose modelleerimisel teiste tunnustega. Alajaotuse 1.2.1 põhjal on

$$\text{logit}(p_i) = \beta_0 + \beta_1 x_{i1} + \ldots + \beta_p x_{ip} + \varepsilon_i \tag{12}$$

logistilise regressioonimudeli kuju i-nda objekti jaoks.

2.1 Mudeli kordajate hindamine

Mudeli kordajaid $\beta_0, \beta_1, \ldots, \beta_p$ hinnatakse logistilise regressiooni korral tavaliselt suurima tõepära meetodil. Maksimeeritav tõepärafunktsioon on

$$L(\boldsymbol{p}, \boldsymbol{y}) = \prod_{i=1}^{n} p(p_i, y_i) = \prod_{i=1}^{n} p_i^{y_i} (1 - p_i)^{1-y_i}, \tag{13}$$

kus $p(p_i, y_i)$, $i \in \{1, \ldots, n\}$, on juhuslikule suurusele $Y_i \sim Be(p_i)$ vastava tõenäosusfunktsooni väärtus kohal y_i ning

$$p_i = \frac{\exp(\beta_0 + \beta_1 x_{i1} + \ldots + \beta_p x_{ip})}{1 + \exp(\beta_0 + \beta_1 x_{i1} + \ldots + \beta_p x_{ip})}.$$

Lihtsuse mõttes vaadeldakse logaritmilist tõepärafunktsiooni

$$\ln L(\boldsymbol{p}, \boldsymbol{y}) = \ln \prod_{i=1}^{n} p_i^{y_i} (1 - p_i)^{1-y_i} = \sum_{i=1}^{n} \left(y_i \ln p_i + (1 - y_i) \ln(1 - p_i) \right), \tag{14}$$

mis saavutab maksimumi samas punktis kui tõepärafunktsioon. Funktsioone (13) ja (14) maksimeerivad kordajate väärtused $\hat{\beta}_0, \hat{\beta}_1, \ldots, \hat{\beta}_p$ ongi suurima tõepära hinnangud mudeli parameetritele.

Nimetatud hinnangute valemite analüütilist kuju pole võimalik leida, kuid üldjuhul on hinnangud leitavad numbriliselt kasutades iteratsioonimeetodeid. Täpsemalt on Newton–Raphsoni iteratsioonimeetodi ja Fisheri skoorimeetodi rakendamist üldistatud lineaarsete mudelite parameetrite hindamisel kirjeldatud teoses (Agresti, 2002). Kuna logistilise regressiooni korral annab iteratiivne kaalutud vähimruutude meetod ligikaudu samad mudeli kordajate hinnangud, mis Fisheri skoorimeetod, siis võidakse hinnangute

leidmiseks kasutada ka neist esimest. Mittekoonduva iteratsiooniprotsessi korral tekib probleeme parameetite hinnangute leidmisel, sellest on kirjutatud teoses (Agresti, 2002). Rakendustarkvara R klassis „glm" logistilist regressioonimudelit sobitades kasutatakse vaikimisi iteratiivset kaalutud vähimruutude meetodit (Davies, *s.a.*).

Prognoosid tõenäosustele $p_1, p_2, \ldots, p_n$ leitakse logistilise regressioonimudeli korral saadud tundmatute parameetrite hinnangute abil vastavalt valemile (2).

2.2 Mudeli headuse näitajad

Logistilise regressioonimudeli headust ning sobivust mõõdetakse erinevate näitajate, näiteks hälbimuse ja Aikaiki informatsioonikriteeriumi abil.

Hälbimus näitab erinevust sobitatud ja küllastunud mudeli logaritmiliste tõepärafunktsioonide vahel. Küllastunud mudel sobib täielikult mudeli koostamiseks kasutatud andmetega ehk selle parameetriteks on kõik vaatlused. Seega arvestades, et logistilise regressioonimudeli korral on uuritaval tunnusel kaks võimalikku väärtust, 0 ja 1, ning defineerides $0 \cdot \ln 0 = 0$, avaldub küllastunud mudeli logaritmiline tõepärafunktsioon kui

$$\ln L(\boldsymbol{y}, \boldsymbol{y}) = \sum_{i=1}^{n} (y_i \ln y_i + (1 - y_i) \ln(1 - y_i)) = 0.$$

Funktsiooni (14) kuju arvesse võttes saadakse, et hälbimus (*deviance*) avaldub kui

$$D = 2\left(\ln L(\boldsymbol{y}, \boldsymbol{y}) - \ln L(\hat{\boldsymbol{p}}, \boldsymbol{y})\right) = -2 \ln L(\hat{\boldsymbol{p}}, \boldsymbol{y}) = -2 \sum_{i=1}^{n} \left(y_i \ln \hat{p}_i + (1 - y_i) \ln(1 - \hat{p}_i)\right),$$

kus $\ln L(\hat{\boldsymbol{p}}, \boldsymbol{y})$ sobitatud mudeli logaritmiline tõepärafunktsioon. Eeldusel, et kehtib nullhüpotees ehk mudel (12) sobib andmetega, on hälbimus asümptootiliselt hii-ruut-jaotusega vabadusastmete arvuga $n - (p + 1)$.

Mida väiksem on hälbimus, seda paremini sobib leitud mudel andmetega. Rakendustarkvara R meetod „glm" väljastab lisaks hinnatud mudeli hälbimusele võrdluseks ainult vabaliiget sisaldava mudeli M_0 hälbimuse D_0 (Davies, *s.a.*). Soovitakse, et leitud mudeli hälbimus oleks väiksem ainult vabaliiget sisaldava mudeli hälbimusest.

Logaritmilise tõepärafunktsiooni väärtus on suurem keerukamate ehk rohkemate argumentidega mudelite korral, seega on ka keerukamate mudelite hälbimus väiksem.

Tihtipeale on tarvis leida aga võimalikult lihtne mudel, mis kirjeldab piisavalt suure osa andmetest. Selleks defineeritakse uus mudeli headuse näitaja, Akaiki informatsioonikriteerium, mis arvestab ka mudeli parameetrite arvu. Akaiki informatsioonikriteerium (*Akaike information criterion*) saadakse parandusliikme $2(p+1)$ lisamisel hälbimusele:

$$AIC = D + 2(p+1), \tag{15}$$

kus $p+1$ on mudeli parameetrite arv.

2.3 Mudeli olulisus

Hinnatud logistilise regressioonimudeli olulisuse ehk selle, kas mõni seletav tunnus mõjutab uuritava tunnuse väärtust, testimiseks kasutatakse enamasti Waldi statistikut. Kontrollitakse, kas mudeli (12) kordajad $\beta_0, \beta_1, \ldots, \beta_p$ on nullist erinevad või mitte ehk vaadeldav nullhüpotees on kujul

$$H_0 : \boldsymbol{\beta} = \mathbf{0}, \tag{16}$$

kus $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_p)^T$ ning $\mathbf{0}$ on $(p+1)$-mõõtmeline nullvektor. Antud nullhüpoteesi kehtides on Waldi statistik

$$W = \hat{\boldsymbol{\beta}}^T \left[\mathrm{cov}(\hat{\boldsymbol{\beta}})\right]^{-1} \hat{\boldsymbol{\beta}},$$

kus $\hat{\boldsymbol{\beta}} = (\hat{\beta}_0, \hat{\beta}_1, \ldots, \hat{\beta}_p)^T$ ja

$$\mathrm{cov}(\hat{\boldsymbol{\beta}}) = \begin{pmatrix} D\hat{\beta}_0 & \mathrm{cov}(\hat{\beta}_0, \hat{\beta}_1) & \ldots & \mathrm{cov}(\hat{\beta}_0, \hat{\beta}_p) \\ \mathrm{cov}(\hat{\beta}_1, \hat{\beta}_0) & D\hat{\beta}_1 & \ldots & \mathrm{cov}(\hat{\beta}_1, \hat{\beta}_p) \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{cov}(\hat{\beta}_p, \hat{\beta}_0) & \mathrm{cov}(\hat{\beta}_p, \hat{\beta}_1) & \ldots & D\hat{\beta}_p \end{pmatrix},$$

asümptootiliselt hii-ruut-jaotusega vabadusastmete arvuga $\mathrm{rank}\left(\mathrm{cov}(\hat{\boldsymbol{\beta}})\right)$.

Waldi statistiku asemel võidakse hüpoteesi (16) kontrollimiseks kasutada ka tõepärasuhte või skooristatistikut, mille kujud ja asümpotootilise hii-ruut-jaotuse vabadusastmete arvud on toodud teoses (Agresti, 2002).

Statistiliselt olulise mudeli korral huvitutakse täpsemalt, millised mudeli kordajad erinevad nullist ehk millised argumenttunnused on olulised. Iga parameetri β_j

$(j = 0, 1, \ldots, p)$ jaoks kontrollitakse hüpoteesi

$$H_0 : \beta_j = 0 \tag{17}$$

Waldi teststatistiku

$$z_j = \frac{\hat{\beta}_j}{\sqrt{\hat{D}(\hat{\beta}_j)}},$$

kus $\sqrt{\hat{D}(\hat{\beta}_j)}$ on hinnangu $\hat{\beta}_j$ standardviga, abil. Waldi teststatistik on nullhüpoteesi (17) kehtimisel asümptootiliselt standardse normaaljaotusega. Kuna mitteolulised tunnused ei kirjelda olulist osa uuritava tunnuse hajuvusest, jäetakse need lõplikust mudelist välja.

2.4 Tunnuste valimine mudelisse

Kui andmestikus on seletavate tunnuste arv, p suur, siis võib optimaalse logistilise regressioonimudeli, kus kõik tunnused on olulised, leidmine olla aeganõudev. Parima sõltumatute tunnuste kombinatsiooni valiku lihtsustamiseks kasutatakse näiteks parima (*best subset*), ettepoole (*forward stepwise*) või tahapoole (*backward stepwise*) valiku meetodit. Järgnevad valikumeetodite kirjeldused põhinevad õpikul (James jt, 2013).

Parima valiku meetodit kasutades hinnatakse iga $l \in \{1, \ldots, p\}$ korral C_p^l logistilist regressioonimudelit, milles on täpselt l seletavat tunnust. Hinnatud C_p^l mudeli seast valitakse välja parim ehk vähima ruutkeskmise veaga (7) mudel, mida tähistatakse M_l.

Ettepoole valiku meetodit rakendades hinnatakse iga $l \in \{1, \ldots, p\}$ korral $p - l + 1$ logistilist regressioonimudelit, milles igas on üks seletav tunnus rohkem kui mudelis M_{l-1}. Hinnatud $p - l + 1$ mudeli seast valitakse parim mudel, mida tähistatakse M_l.

Tahapoole valiku meetodi korral alustatakse logistilisest regressioonimudelist M_p, milles on p argumenttunnust. Iga $l \in \{p-1, p-2, \ldots, 1\}$ korral hinnatakse $l+1$ mudelit, milles igas on üks seletav tunnus vähem kui mudelis M_{l+1}. Hinnatud $l+1$ mudeli seast valitakse parim mudel M_l.

Kõigi kolme meetodi korral valitakse viimase sammuna leitud mudelite $M_0, M_1, \ldots, M_p$ seast välja mudel, mille test ruutkeskmine viga k-kordsel ristvalideerimise meetodil (11) või Aikaiki informatsioonikriteeriumi (15) väärtus on kõige väiksem.

2.5 Mudeli interpreteerimine

Hinnatud logistilise regressioonimudeli kuju i-nda, $i \in \{1, \ldots, n\}$, objekti jaoks on

$$\text{logit}(\hat{p}_i) = \ln \frac{\hat{p}_i}{1-\hat{p}_i} = \hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \ldots + \hat{\beta}_p x_{ip}.$$

Mudeli vabaliikme hinnangut $\hat{\beta}_0$ on võimalik interpreteerida juhul, kui null on kõigi argumenttunnuste võimalik väärtus ning vabaliikme hinnang on positiivne. Sel juhul

$$\ln \frac{\hat{p}_i}{1-\hat{p}_i} > 0 \quad \text{ehk} \quad \frac{\hat{p}_i}{1-\hat{p}_i} > 1 \quad \text{ehk} \quad \hat{p}_i > 1 - \hat{p}_i$$

ehk sündmuse toimumise tõenäosus on suurem kui 0,5 (Käärik, 2013: 111). Positiivne mudeli kordaja hinnang $\hat{\beta}_j$, $j \in \{1, \ldots, p\}$, näitab samasuunalist seost vastava argumenttunnuse ja uuritava tunnuse vahel. Negatiivne kordaja aga vastassuunalist seost.

Huvipakkuva sündmuse toimumise ja mittetoimumise tõenäosuste jagatist

$$\Pi_i = \frac{p_i}{1-p_i}$$

nimetatakse antud sündmuse šansiks. Mudel huvipakkuva sündmuse šansile i-nda objekti korral on

$$\hat{\Pi}_i = \frac{\hat{p}_i}{1-\hat{p}_i} = \exp\left(\hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \ldots + \hat{\beta}_p x_{ip}\right).$$

Tavapäraselt interpreteeritakse logistilises regressioonimudelis parameetri suurust šansside suhte Π_i/Π_k abil.

Näide 1. Olgu vaatluse all kaks objekti, mille j-nda tunnuse väärtused erinevad c ühiku võrra ehk

$$x_{ij} = x_{kj} + c$$

ning ülejäänud tunnuste väärtused on samad. Toimub c-ühikuline muutus j-nda tunnuse väärtuses, millega kaasneb šansside suhte

$$\frac{\hat{\Pi}_i}{\hat{\Pi}_k} = \frac{\exp\left(\hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \ldots + \hat{\beta}_j (x_{kj} + c) + \ldots + \hat{\beta}_p x_{ip}\right)}{\exp\left(\hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \ldots + \hat{\beta}_j x_{kj} + \ldots + \hat{\beta}_p x_{ip}\right)} =$$

$$= \frac{\exp\left(\hat{\beta}_j x_{kj} + \hat{\beta}_j c\right)}{\exp\left(\hat{\beta}_j x_{kj}\right)} = \exp\left(\hat{\beta}_j c\right)$$

kordne muutus.

Järgnevalt on toodud näide logistilise regressioonimudeli rakendamisest reaalsetel Eesti andmetel, mis pärinevad Euroopa Sotsiaaluuringust (European Social Survey, 2014). Kasutatavasse andmestikku pole kaasatud vastajaid, kellel mõne vaadeldava tunnuse väärtus puudus. Näite koostamisel kasutatud R kood ja väljundid on kirjas lisas 1.

Näide 2. Uuritakse, kuidas mõjutavad vanus, kehamassiindeks, sugu ja kõrge vererõhk indiviidil südame- ja veresoonkonna haiguste esinemist. Andmestik `ESS` sisaldab viit tunnust ja 2004 objekti. Uuritav binaarne tunnus `süda` on võimalike väärtustega 0, kui inimene pole põdenud viimase aasta jooksul südame- ja veresoonkonna haigusi, ja 1, kui inimene on antud haigusi põdenud. Seletavad arvulised tunnused `vanus` ja `KMI` on väärtuste piirkondadega vastavalt 15–99 aastat ja 15,6–49,5 kg/m^2. Binaarsed argumenttunnused on `sugu`, mis näitab, kas tegu on mehega (väärtus 0) või naisega (väärtus 1), ning `vererõhk`, mille väärtus on 0, kui inimesel pole viimase aasta jooksul olnud probleeme kõrge vererõhuga, ning 1, kui isikul on olnud probleeme kõrge vererõhuga.

Mudeli koostamiseks kasutatakse andmestikust `ESS` juhuslikult valitud 1002 objekti. Ülejäänud vaatluste põhjal hinnatakse mudeli täpsust test ruutkeskmise vea (9) abil.

Huvi pakub tõenäosus $p_i = \mathrm{P}(\texttt{süda}_i = 1)$, mille hindamiseks leitakse esmalt logistiline regressioonimudel kujul

$$\operatorname{logit}(p_i) = \beta_0 + \beta_1 \cdot \texttt{vanus}_i + \beta_2 \cdot \texttt{KMI}_i + \beta_3 \cdot \texttt{sugu}_i + \beta_4 \cdot \texttt{vererõhk}_i + \varepsilon_i.$$

Leitud mudelis pole tunnused `sugu` ja `KMI` olulised, mistõttu jäetakse need järjest mudelist välja. Viimaks hinnatakse järgnev logistiline regressioonimudel inimesel südame- ja veresoonkonna haiguste esinemisele:

$$\operatorname{logit}(p_i) = \beta_0 + \beta_1 \cdot \texttt{vanus}_i + \beta_2 \cdot \texttt{vererõhk}_i + \varepsilon_i. \tag{18}$$

Tabel 1: Mudeli (18) parameetrite hinnangud, Waldi teststatistikud ja nende olulisus

	Parameetri hinnang	**Waldi teststatistik**	**Olulisuse tõenäosus**
`(vabaliige)`	$-5{,}02$	$-12{,}58$	$< 0{,}001$
`vanus`	$0{,}05$	$7{,}93$	$< 0{,}001$
`vererõhk`	$0{,}97$	$4{,}77$	$< 0{,}001$

Mudelis (18) on mõlemad tunnused olulised ehk nii vanus kui ka kõrge vererõhk mõjutavad südame- ja veresoonkonna haiguste esinemise tõenäosust (vt tabel 1).

Seega on mudel südame- ja veresoonkonna haiguste esinemise šansi logaritmile

$$\text{logit}(\hat{p}_i) = -5{,}02 + 0{,}05 \cdot \texttt{vanus}_i + 0{,}97 \cdot \texttt{vererõhk}_i. \tag{19}$$

Antud mudeli hälbimus on $D = 675{,}86$, mis on väiksem kui ainult vabaliiget sisaldava mudeli hälbimus $D_0 = 821{,}36$. Aikaiki informatsioonikriteeriumi väärtus on $AIC = 681{,}86$, mis on väiksem kui esialgse viie parameetriga mudeli näitaja. Mudeli koostamisest kõrvale jäetud vaatluste põhjal arvutatud test ruutkeskmise vea väärtus on

$$MSE^t = 0{,}105.$$

Mudeli argumentide ees olevaid kordajaid tõlgendatakse eraldi. Kui kahe isiku, kelle tunnuse `vererõhk` väärtus on sama, vanusevahe on üks aasta, siis südame- ja veresoonkonna haiguste esinemise šanss on vanemal inimesel $\exp(0{,}05) = 1{,}05$ korda ehk 5% võrra suurem kui nooremal isikul. Samavanustest inimestest on isikul, kellel esines viimasel aastal probleeme kõrge vererõhuga, südame- ja veresoonkonna haiguste esinemise šanss $\exp(0{,}97) = 2{,}64$ korda suurem kui inimesel, kellel probleeme kõrge vererõhuga ei esinenud.

Südame- ja veresoonkonna haiguste esinemise tõenäosuse leidmiseks konkreetsele inimesele avaldatakse mudelist (19) hinnang tõenäosusele

$$\hat{p}_i = \frac{\exp(-5{,}02 + 0{,}05 \cdot \texttt{vanus}_i + 0{,}97 \cdot \texttt{vererõhk}_i)}{1 + \exp(-5{,}02 + 0{,}05 \cdot \texttt{vanus}_i + 0{,}97 \cdot \texttt{vererõhk}_i)}.$$

Seega on näiteks hinnanguliselt tõenäosus, et 55-aastasel kõrge vererõhuga inimesel esineb südame- ja veresoonkonna haigusi

$$\hat{p}_i = \frac{\exp(-5{,}02 + 0{,}05 \cdot 55 + 0{,}97 \cdot 1)}{1 + \exp(-5{,}02 + 0{,}05 \cdot 55 + 0{,}97 \cdot 1)} \approx 0{,}214.$$

Antud näites kirjeldatud probleem on lahendatud ka klassifitseerimispuu abil näites 3.

3 Klassifitseerimispuu

Antud peatükk põhineb õpikul (James jt, 2013), kui pole viidatud teisiti.

Klassifitseerimispuu on üks mitteparameetrilistest meetoditest kaheväärtuselise tunnuse seose modelleerimisel teiste tunnustega.

3.1 Puu koostamine

Klassifitseerimispuu koostamisel jagatakse objektid seletavate tunnuste väärtuste järgi lõikumatutesse piirkondadesse $R_1, R_2, \ldots, R_m$. Igas piirkonnas leitakse uuritavale binaarsele tunnusele ning vaadeldava sündmuse esinemise tõenäosusele hinnangud vastavalt valemite (4) ja (5) abil.

Kaheväärtuselise uuritava tunnuse korral klassifitseerimispuu koostamisel piirkondade $R_1, R_2, \ldots, R_m$ leidmisel vaadeldakse näiteks klassifitseerimisviga, Gini indeksit või hälbimust. Klassifitseerimisviga leitakse piirkonnas R_k, $k \in \{1, \ldots, m\}$, kui

$$E_{R_k} = 1 - \max\{\hat{p}_{R_k}, 1 - \hat{p}_{R_k}\} = \min\{\hat{p}_{R_k}, 1 - \hat{p}_{R_k}\}.$$

Kogu puu klassifitseerimisviga leitakse valemi (6) abil või kaalutud klassifitseerimisvigade keskmisena üle piirkondade R_k:

$$E = \frac{1}{n} \sum_{k=1}^{m} |R_k| E_{R_k} = \frac{1}{n} \sum_{k=1}^{m} |R_k| \min\{\hat{p}_{R_k}, 1 - \hat{p}_{R_k}\}. \tag{20}$$

Gini indeks mõõdab uuritava tunnuse varieeruvust üle selle võimalike väärtuste, 0 ja 1. Piirkonnas R_k arvutatakse Gini indeks järgmiselt:

$$G_{R_k} = 2\hat{p}_{R_k}(1 - \hat{p}_{R_k}).$$

Terve klassifitseerimispuu Gini indeks leitakse keskmisena üle piirkondade R_k kaalutud Gini indeksite:

$$G = \frac{1}{n} \sum_{k=1}^{m} |R_k| G_{R_k} = \frac{2}{n} \sum_{k=1}^{m} |R_k| \hat{p}_{R_k}(1 - \hat{p}_{R_k}). \tag{21}$$

Hälbimus leitakse piirkonnas R_k kui

$$D_{R_k} = -2\left(\hat{p}_{R_k} \ln \hat{p}_{R_k} + (1 - \hat{p}_{R_k}) \ln(1 - \hat{p}_{R_k})\right).$$

Kogu klassifitseerimispuu hälbimus saadakse piirkondade R_k kaalutud hälbimuste summana:

$$D = \sum_{k=1}^{m} |R_k| D_{R_k} = -2 \sum_{k=1}^{m} |R_k| \left(\hat{p}_{R_k} \ln \hat{p}_{R_k} + (1 - \hat{p}_{R_k}) \ln(1 - \hat{p}_{R_k})\right). \tag{22}$$

Rakendustarkvara R pakett „tree" väljastab jääkide keskmise hälbimuse (*residual mean deviance*) $RMD = D/(n-m)$, millest (Ripley, 2016)

$$D = (n-m)\, RMD.$$

Klassifitseerimispuu kasvatamiseks kasutatakse rekursiivset binaarset tükeldamist. See tähendab, et esialgu on kõik objektid ühes suures piirkonnas, mis esimese tükelduse tulemusena jaguneb kaheks alampiirkonnaks. Jagunemise tegemiseks valitakse üks seletav tunnus ning selle väärtus t nii, et nende abil leitud kaks objektide piirkonda (esimeses piirkonnas on need objektid, mille valitud tunnuse väärtus on väiksem kui t ning teises ülejäänud objektid) annaksid minimeeritava näitaja suurima vähenemise. Järgnevalt korratakse eelnevat protsessi eraldi mõlemas saadud alampiirkonnas. Tulemuseks on neli lõikumatut piirkonda. Tükeldamist jätkatakse lõpetamise tingimuseni, milleks võib olla näiteks väike objektide arv piirkonnas.

Tavapäraselt on puu kasvatamisel minimeeritav näitaja Gini indeks (21) või hälbimus (22), kuna need näitajad on klassifitseerimisveast (20) piirkonna puhtuse suhtes tundlikumad. Piirkonna puhtus tähendab, et võimalikult paljud uuritava tunnuse väärtused kuuluvad piirkonnas samasse klassi. Rakendustarkvara R paketis „tree" kasutatakse klassifitseerimispuu kasvatamisel minimeeritava näitajana vaikimisi hälbimust (Ripley, 2016).

3.2 Puu pügamine

Klassifitseerimispuu kasvatamise algoritmi puuduseks on puu liigne sobivus kasutatud andmetega. Kuna enamasti soovitakse puu abil prognoosida uuritava tunnuse väärtust nendele objektidele, mille kohta on teada vaid seletavate tunnuste väärtused, on lihtsam ja vähemate piirkondadega puu parem. Parim viis hea väiksema klassifitseerimispuu saamiseks on väga suure puu T_0 kasvatamine ning seejärel selle alampuuks T pügamine.

Klassifitseerimispuu pügamisel kasutatakse minimeerimisel näitajat (20), (21) või (22). Edasises tähistatakse valitud näitaja väärtust puu T korral $Q(T)$. Võimalikult täpse uuritava tunnuse hinnangu saavutamiseks kasutatakse pügamisel klassifitseerimisviga. Rakendustarkvara R paketis „tree" on klassifitseerimispuu pügamisel vaikimisi kasutusel hälbimus (Ripley, 2016).

Puu pügamisel on oluline leida, milline lõplike piirkondade arv on optimaalne. See tähendab, et valida tuleb puu, mis pole väga keeruline, kuid samas sobib hästi andmetega. Seega soovitakse, et puu test-klassifitseerimisviga (8) oleks minimaalne. Viimast hinnatakse ristvalideerimise abil. Samas on puu iga võimaliku alampuu test-klassifitseerimisvea k-kordsel ristvalideerimise meetodil (10) leidmine tülikas. Selle asemel kasutatakse nn *cost-complexity* pügamismeetodit. Sel juhul valitakse väikse arvu leitud alampuude seast alampuu, mille viga (10) on väikseim.

Olgu antud esialgse puu T_0 alampuu T. Defineeritakse nn *cost-complexity* kriteerium

$$C_\alpha(T) = Q(T) + \alpha \cdot m^T,$$

kus m^T on lõplike piirkondade arv puus T ning $\alpha \geq 0$ on häälestusparameeter (*tuning parameter*). Iga parameetri α väärtuse korral on võimalik näidata, et leidub üks ja ainult üks alampuu T_α, mis minimeerib kriteeriumi $C_\alpha(T)$ väärtust (vt Ripley, 1996).

Hastie, Tibshirani ja Friedman (2011) kohaselt toimib *cost-complexity* pügamismeetod järgnevalt. Alustatakse esialgsest puust T_0, igal sammul kustutatakse ära puu selle sõlme alampuu, mille korral on minimeeritava näitaja $Q(T)$ kasvamine vähim. Kui jõutakse ühest piirkonnast koosneva alampuuni, siis lõpetatakse. Tulemuseks on jada parameetri α väärtustele vastavatest alampuudest. Parima alampuu valimiseks leitakse, millise parameetri α väärtuse korral on test-klassifitseerimisviga k-kordsel ristvalideerimise meetodil vähim.

Klassifitseerimispuu pügamine on kokku võetud järgnevas algoritmis:

1. Etteantud vaatluste põhjal kasvatatakse rekursiivset binaarset tükeldamist kasutades suur klassifitseerimispuu.
2. Rakendades *cost-complexity* pügamismeetodit leitakse jada parameetri α väärtustele vastavatest alampuudest.

3. Kasutades k-kordset ristvalideerimist valitakse välja parim α. Selleks jagatakse andmestik k umbes võrdse suurusega gruppi ning iga $i \in \{1, \ldots, k\}$ korral:
   (a) Korratakse 1. ja 2. sammu vaatlustel, mis ei kuulu i-ndasse gruppi. Tulemuseks on jada parameetri α väärtustele vastavatest alampuudest.
   (b) Arvutatakse i-nda grupi andmete põhjal iga parameetri α korral test-klassifitseerimisviga.

   Leitakse $E^t_{CV}(k)$ iga parameetri α väärtuse korral. Viimaks valitakse välja see parameetri α väärtus, mille korral on antud näitaja vähim.
4. Töö tulemuseks on alampuu, mis vastab valitud parameetri α väärtusele.

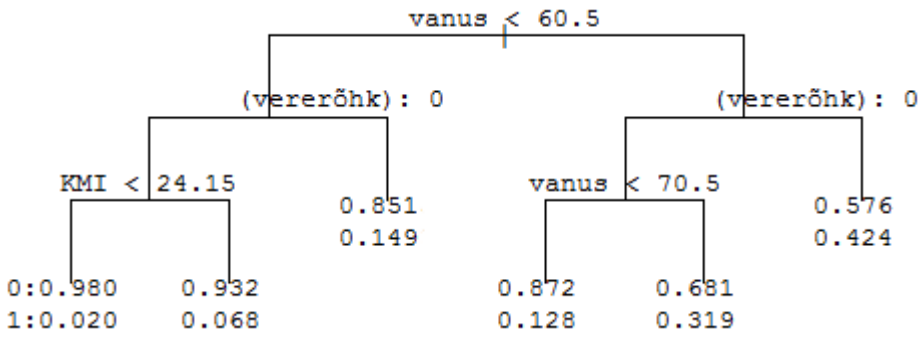
3.3 Puu interpreteerimine

Klassifitseerimispuud on lihtne interpreteerida, mistõttu on klassifitseerimispuu meetod atraktiivne ka praktikas. Konkreetsele objektile uuritava tunnuse prognoosi leidmiseks vaadeldakse selle objekti seletavate tunnuste väärtused. Nende abil leitakse, millisesse piirkonda R_k kuulub vaatluse all olev objekt. Selleks liigutakse alates klassifitseerimispuu tipust igal hargnemisel otsustuse põhjal lõpliku piirkonna R_k poole (vt joonis 1 lk 24). Hargnemisel oleva tingimuse tõesuse korral liigutakse vasakpoolsesse puu harru, vastasel juhul parempoolsesse harru. Uuritava tunnuse y prognoos on vaadeldavale objektile seega $\hat{y}_{R_k}$ ning huvipakkuva sündmuse toimumise tõenäosuse hinnang on $\hat{p}_{R_k}$.

Järgmises näites leitakse klassifitseerimispuu näites 2 vaadeldud ülesande lahendamiseks. Kasutatud R kood ja väljundid on toodud lisas 2.

Näide 3. Klassifitseerimispuu koostatakse sama 1002 objekti põhjal, mida kasutati näites 2 logistilise regressioonimudeli hindamisel. Mudelisse kaasatakse seletavad tunnused `vanus`, `KMI`, `sugu` ja `vererõhk`. Leitud klassifitseerimispuus toimuvad jagunemised inimese vanuse, kehamassiindeksi ja selle, kas isikul on olnud probleeme kõrge vererõhuga või mitte, järgi (vt joonis 1 lk 24).

Joonisel 1 on kujutatud igas lõplikus klassifitseerimispuu piirkonnas südame- ja veresoonkonna haiguste esinemise tõenäosus ning selle kohal antud haiguste mitte-diagnoosimise tõenäosus. Antud klassifitseerimispuu kohaselt on hinnanguliselt



Joonis 1: Klassifitseerimispuu südame- ja veresoonkonna haiguste esinemisele

tõenäosus, et 55-aastasel kõrge vererõhuga inimesel esineb südame- ja veresoonkonna haigusi

$$\hat{p}_i = 0{,}149.$$

Vaadeldavas klassifitseerimispuus on kuus lõplikku piirkonda ning puu pügamine pole võimalik, kuna ei leidu piisavalt heade omadustega alampuud. Puu klassifitseerimisviga on $E = 0{,}143$ ning hälbimus $D = 660{,}1$. Klassifitseerimispuu koostamisest kõrvale jäetud andmete põhjal arvutatud test ruutkeskmine viga on

$$MSE^t = 0{,}109.$$

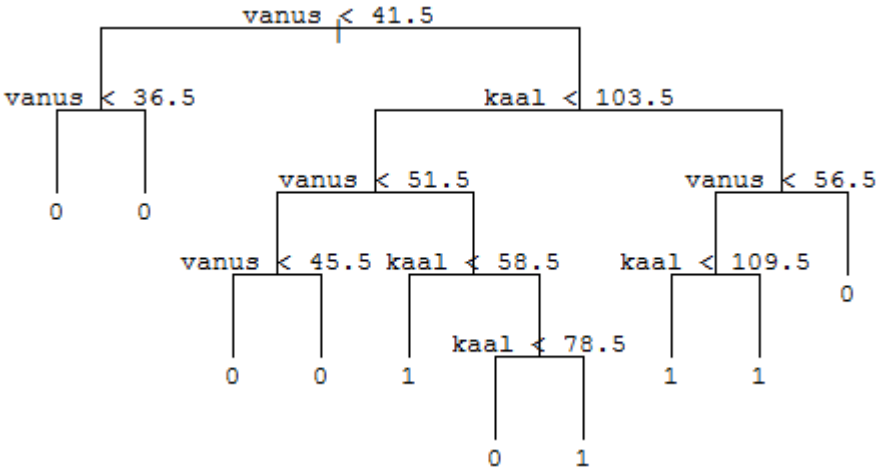
Viimase näitaja väärtus on vaid 0,004 võrra suurem kui sama probleemi lahendamiseks näites 2 leitud logistilise regressioonimudeli test ruutkeskmine viga.

Test ruutkeskmise vea väärtuse põhjal töötasid logistiline regressioon ning klassifitseerimispuu meetod südame- ja veresoonkonna haiguste esinemisele mudeli leidmisel umbes samaväärselt, kuid interpreteerida on lihtsam klassifitseerimispuud joonisel 1 kui hinnatud logistilist regressioonimudelit (19).

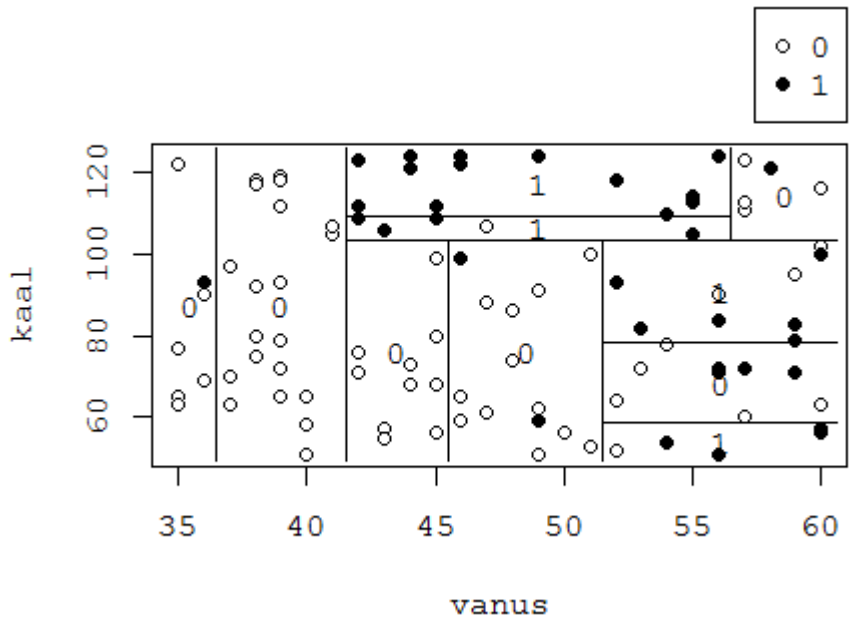
Järgnevalt on toodud näide klassifitseerimispuu koostamisest koos pügamisega. Kasutatav andmestik sisaldab kahte seletavat tunnust. Andmestiku genereerimise ning näites kasutatud R kood ja väljundid on kirjas lisas 3.

Näide 4. Huvitutakse, kuidas vanus ja kehakaal mõjutavad inimesel diabeedi esinemist. Genereeritud andmestik `veresuhkur` sisaldab kolme tunnust ning 100 objekti. Uuritav binaarne tunnus `diabeet` on võimalike väärtustega 0, kui inimesel pole diagnoositud suhkurtõbe, ja 1, kui inimesel on antud haigus diagnoositud. Seletavad arvulised

tunnused `vanus` ja `kaal` on väärtuste piirkondadega vastavalt 35–60 aastat ja 51–124 kilogrammi. Huvipakkuva tõenäosuse $p_i = \mathrm{P}(\texttt{diabeet}_i = 1)$ hindamiseks leitakse klassifitseerimispuu.



Joonis 2: Klassifitseerimispuu diabeedi diagnoosimisele

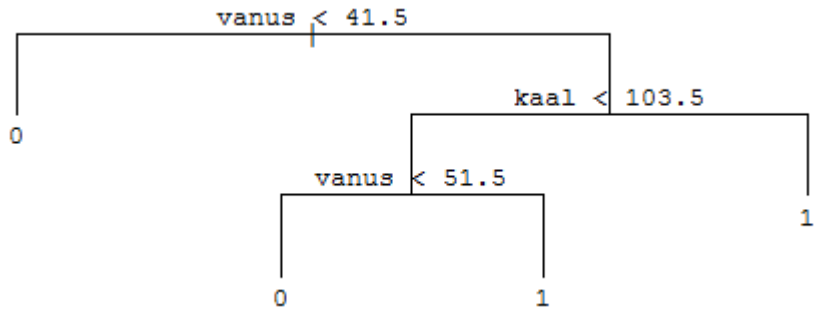


Joonis 3: Piirkondadeks jagunemine klassifitseerimispuus diabeedi diagnoosimisele

Saadud puus toimuvad objektide jagunemised nii inimese vanuse kui ka kehakaalu järgi. Tunnuste `vanus` ja `kaal` väärtuste põhjal moodustub 10 piirkonda (vt joonis 2 ja 3). Klassifitseerimisviga antud puus on $E = 0{,}13$ ning puu hälbimus on 57,15.

Antud klassifitseerimispuu on üsna haruline, mistõttu pügatakse seda väiksemaks. Väikeseim alampuu, nelja lõpliku piirkonnaga, saadakse alampuude seast, mille

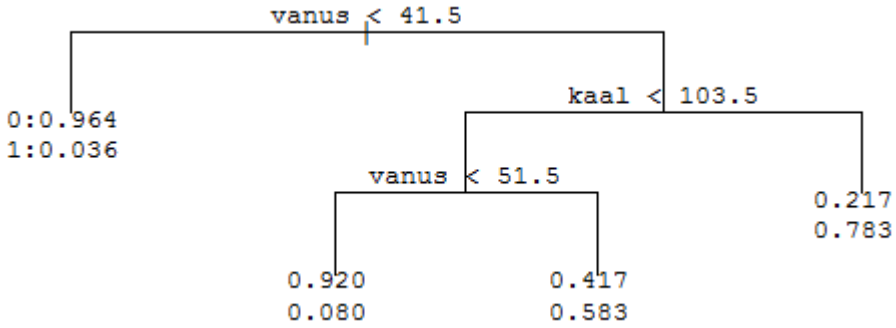
test-klassifitseerimisviga k-kordsel ristvalideerimise meetodil on vähim.



Joonis 4: Pügatud klassifitseerimispuu diabeedi diagnoosimisele

Joonisel 4 on kujutatud pügatud klassifitseerimispuu diabeedi diagnoosimisele. Jagunemised toimuvad taas nii tunnuse `vanus` kui ka `kaal` järgi. Pügatud puu klassifitseerimisviga on $E = 0{,}18$ ning antud klassifitseerimispuu hälbimus on $D = 79{,}25$.

Igas lõplikus piirkonnas on välja toodud hinnang isiku diabeedi põdemisele. Näiteks on inimestel, kes on nooremad kui 42 aastat, suurem tõenäosus suhkurtõbe mitte põdeda. Täpsemalt on hinnang diabeedi diagnoosimise tõenäosusele antud vanuses inimestel 0,036 (vt joonis 5). Joonisel 5 on kujutatud igas lõplikus klassifitseerimispuu piirkonnas diabeedi diagnoosimise prognoositud tõenäosus ning selle kohal suhkurtõve mitte-diagnoosimise tõenäosus.



Joonis 5: Pügatud klassifitseerimispuu tõenäosustega diabeedi diagnoosimisele

Konkreetsele isikule saab joonisel 5 kujutatud klassifitseerimispuu põhjal leida prognoosi suhkurtõve diagnoosimise tõenäosusele. Näiteks on hinnanguliselt tõenäosus, et 55-aastasel 100 kilogrammi kaaluval inimesel diagnoositakse suhkurtõbi

$$\hat{p}_i = 0{,}583.$$

4 Simuleerimisülesanne

Antud simuleerimisülesande aluseks on artikli (Phipps ja Toth, 2012) lisa, milles kirjeldatud simuleerimise mehhanisme on rakendatud andmestikule `ESS`, mida vaadeldi näidetes 2 ja 3. Nimetatud andmed pärinevad Euroopa Sotsiaaluuringust (European Social Survey, 2014). Simuleerimisülesande läbiviimisel kasutatud R kood ja väljundid on kirjas lisas 4.

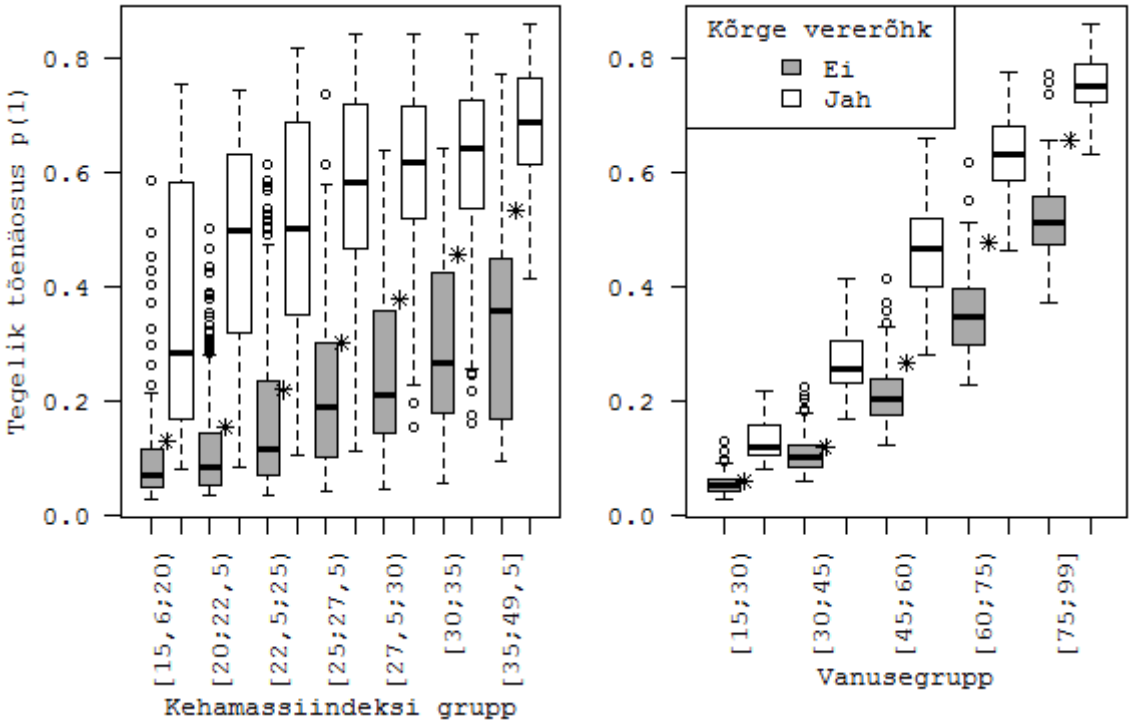
Antud osa eesmärk on omavahel võrrelda logistilist regressiooni ja klassifitseerimispuud nii meetodi täpsuse kui ka mudeli tõlgendamise keerukuse suhtes. Selleks genereeritakse esmalt iga isiku jaoks nn tegelik südame- ja veresoonkonna haiguste esinemise tõenäosus sõltuvalt tema vanusest, kehamassiindeksist ja sellest, kas indiviidil on kõrge vererõhk või mitte. Seejärel seatakse genereeritud tõenäosustele vastavusse binaarne tunnus ja leitakse vastav mudel nii logistilise regressiooni kui ka klassifitseerimispuu meetodi abil. Töös on tõenäosuste genereerimiseks kasutatud kahte erinevat eeskirja, mille korral meetodeid võrreldakse.

4.1 Binaarse tunnuse genereerimine

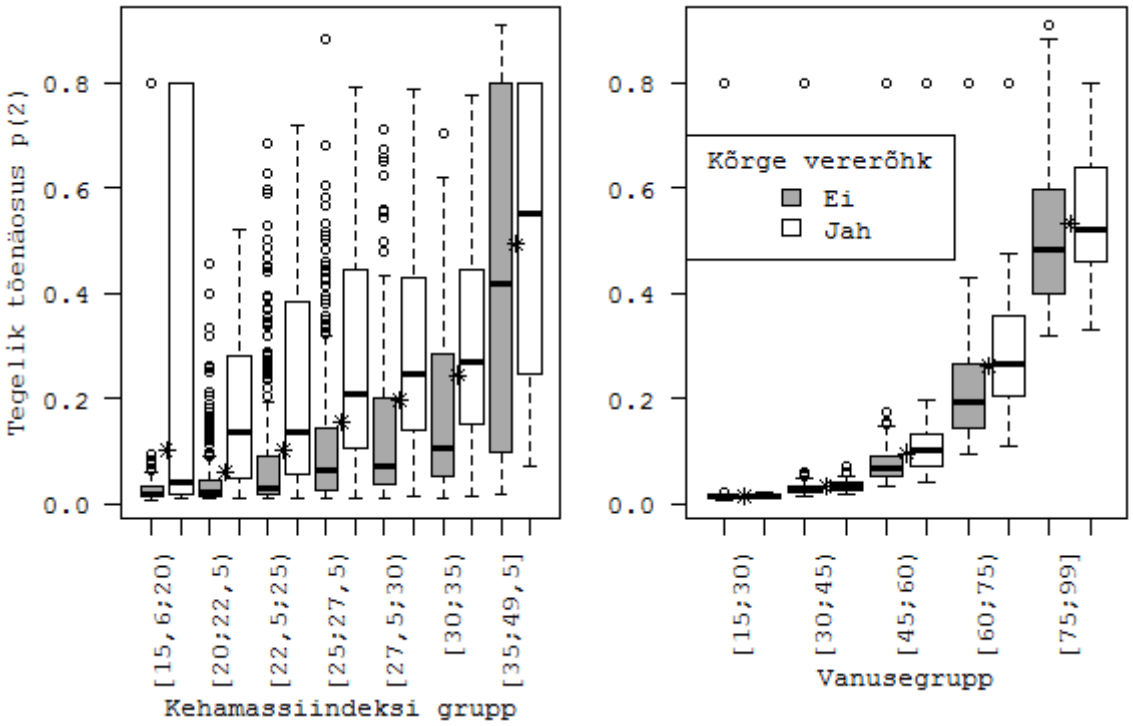
Esmalt eeldatakse, et tegelik südame- ja veresoonkonna haiguste esinemise šansi logaritm on seotud tunnustega `vanus`, `KMI` ja `vererõhk` lineaarselt ehk on täidetud logistilise regressioonimudeli eeldus. Täpsemalt eeldatakse, et südame- ja veresoonkonna haiguste esinemise tõenäosus avaldub kui

$$p_i^{(1)} = \frac{\exp\left(-5 + 0{,}05 \cdot \texttt{vanus}_i + 0{,}04 \cdot \texttt{KMI}_i + \texttt{vererõhk}_i\right)}{1 + \exp\left(-5 + 0{,}05 \cdot \texttt{vanus}_i + 0{,}04 \cdot \texttt{KMI}_i + \texttt{vererõhk}_i\right)}. \tag{23}$$

Genereeritud tõenäosuste jaotumine vastavalt kehamassiindeksi grupile, vanusgrupile ning kõrge vererõhu esinemisele on toodud joonisel 6 lk 28. Üldiselt on kõrgema kehamassiindeksiga inimestel südame- ja veresoonkonna haiguste esinemise tõenäosus suurem kui madalama kehamassiindeksiga inimestel. Samuti on nimetatud tõenäosus vanematel isikutel suurem kui noorematel. Südame- ja veresoonkonna haiguste esinemise tõenäosus on üldiselt suurem kõrge vererõhuga inimestel kui isikutel, kellel probleeme kõrge vererõhuga pole, vastavad keskmised tõenäosused on 0,574 ja 0,190.



Joonis 6: Tegelike tõenäosuste $p_i^{(1)}$ jaotumine kehamassiindeksi grupi, vanusegrupi ja kõrge vererõhu esinemise järgi (nn lineaarne seos)



Joonis 7: Tegelike tõenäosuste $p_i^{(2)}$ jaotumine kehamassiindeksi grupi, vanusegrupi ja kõrge vererõhu esinemise järgi (nn mittelineaarne seos)

Jooniselt 6 on näha ka, et keskmine südame- ja veresoonkonna haiguste esinemise tõenäosus, mis on joonisel tähistatud tärniga, suureneb nii kehamassiindeksi grupiti kui ka vanusgrupiti ligikaudu lineaarselt.

Teisel juhul eeldatakse, et südame- ja veresoonkonna haiguste esinemise tõenäosus on seotud tunnustega `vanus`, `KMI` ja `vererõhk` mittelineaarselt järgnevalt:

$$p_i^{(2)} = \begin{cases} 0{,}8, & \text{kui } \texttt{KMI}_i > 38 \text{ või } \texttt{vanus}_i > 60 \text{ ja } \texttt{KMI}_i < 20, \\ P, & \text{muidu,} \end{cases}$$

kus

$$P = \frac{\exp\left(-5 + 0{,}0005 \cdot \texttt{vanus}_i^2 + 0{,}0008 \cdot \texttt{vanus} \cdot \texttt{KMI}_i + 0{,}005 \cdot \texttt{vererõhk}_i \cdot \texttt{KMI}_i\right)}{1 + \exp\left(-5 + 0{,}0005 \cdot \texttt{vanus}_i^2 + 0{,}0008 \cdot \texttt{vanus} \cdot \texttt{KMI}_i + 0{,}005 \cdot \texttt{vererõhk}_i \cdot \texttt{KMI}_i\right)}.$$

Genereeritud tõenäosuste jaotumine vastavalt kehamassiindeksi grupile, vanusgrupile ning kõrge vererõhu esinemisele on näha joonisel 7. Ka sel juhul on kõrgema kehamassiindeksiga ja vanematel inimestel südame- ja veresoonkonna haiguste esinemise tõenäosus suurem. Samuti on südame- ja veresoonkonna haiguste esinemise tõenäosus suurem kõrge vererõhuga isikutel. Nende seas on vastav tõenäosus keskmiselt 0,303, kuid inimeste, kellel pole probleeme kõrge vererõhuga, seas 0,109.

Südame- ja veresoonkonna haiguste esinemise tõenäosus on nii tunnusega `KMI` kui ka tunnusega `vanus` seotud mittelineaarselt (vt joonis 7).

4.2 Simuleerimisülesande kirjeldus

Leitud tõenäosuste $p_i^{(1)}$ ja $p_i^{(2)}$ põhjal genereeritakse 100 korda uued tunnused `süda1` ja `süda2` nii, et $\texttt{süda1}_i$ on alati jaotusest $Be\left(p_i^{(1)}\right)$ pärit juhusliku suuruse realisatsioon ning $\texttt{süda2}_i$ on alati jaotusest $Be\left(p_i^{(2)}\right)$ pärit juhusliku suuruse realisatsioon. Igal korral hinnatakse mõlemale tekitatud tunnusele nii logistiline regressioonimudel (esimesel juhul oluliste tunnustega `vanus`, `KMI` ja `vererõhk` ning teisel juhul oluliste tunnustega `vanus` ja `KMI`) kui ka klassifitseerimispuu, kus on seletavad tunnused `vanus`, `KMI` ja `vererõhk`.

Meetodi täpsuse hindamiseks konkreetse seose korral arvutatakse selle meetodi ruutkeskmine viga Monte-Carlo meetodil

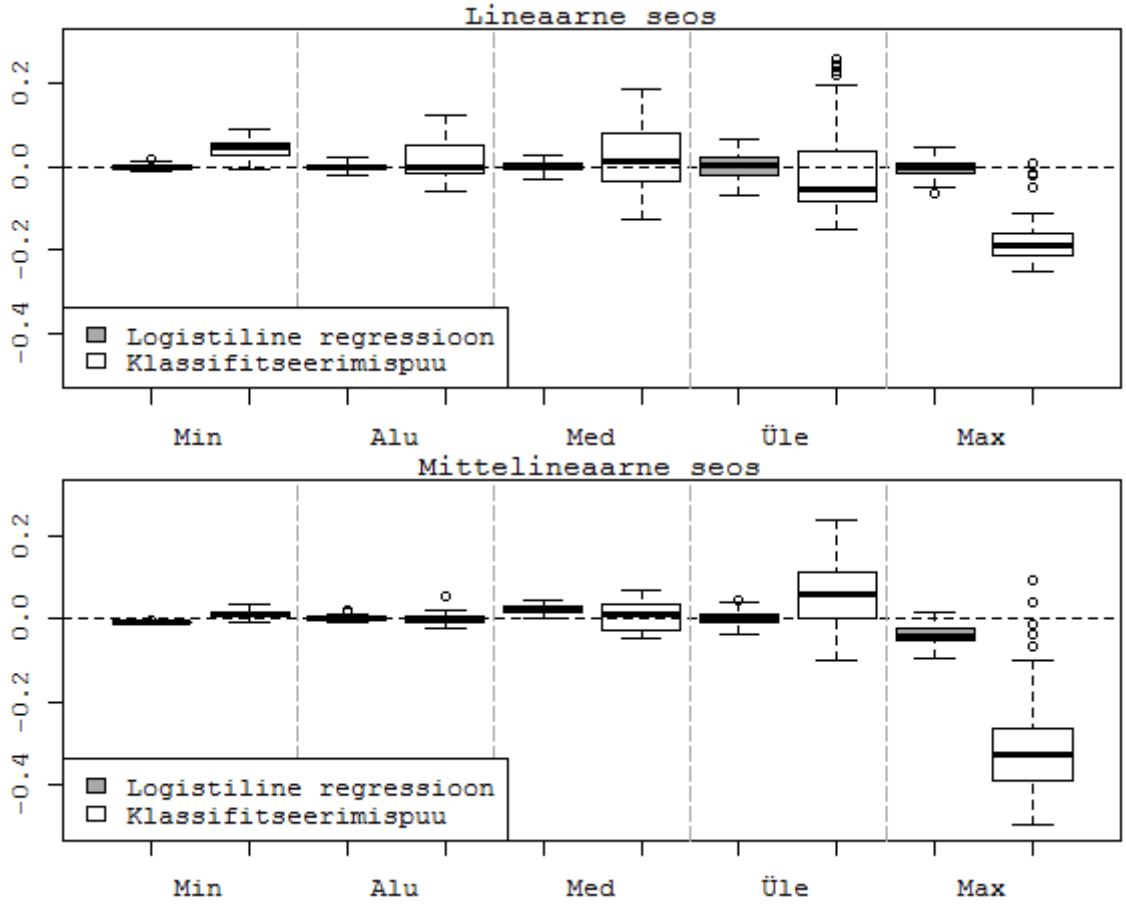
$$MSE_{MC} = \frac{1}{R}\sum_{j=1}^{R}\left[\frac{1}{n}\sum_{i=1}^{n}\left(p_i - (\hat{p}_i)_j\right)^2\right], \tag{24}$$

kus R on simuleerimissammude arv, n on vaatluste arv andmestikus, p_i on tegelik tõenäosus ning $(\hat{p}_i)_j$ on simuleerimissammul j saadud hinnang tõenäosusele p_i.

Lisaks võrreldakse tegelikke ja mudelite põhjal hinnatud tõenäosusi konkreetsete väärtuste korral. Nendeks on tegelike tõenäosuste miinimum, alumine kvartiil, mediaan, ülemine kvartiil ja maksimum. Võib juhtuda, et mediaanile lähedaste väärtuste korral töötab üks meetod tunduvalt paremini kui teine, kuid mediaanist kaugemal on olukord teistsugune.

4.3 Tulemused

Igal simuleerimissammul leitakse nii logistilise regressioonimudeli kui ka klassifitseerimispuu abil hinnatud tõenäosuste vahe tegelike südame- ja veresoonkonna haiguste esinemise tõenäosustega.



Joonis 8: Kahe meetodi abil hinnatud tõenäosuste ja tegelike südame- ja veresoonkonna haiguste esinemise tõenäosuste vahede karpdiagrammid

Tabel 2: Tegelike südame- ja veresoonkonna haiguste esinemise tõenäosuste kokkuvõte

Seos	Lineaarne	Mittelineaarne
Miinimum	0,028	0,009
Alumine kvartiil	0,094	0,023
Mediaan	0,211	0,067
Ülemine kvartiil	0,454	0,219
Maksimum	0,857	0,909

Joonisel 8 on kujutatud nimetatud vahede karpdiagrammid tegelike tõenäosuste miinimumide, alumiste kvartiilide, mediaanide, ülemiste kvartiilide ja maksimumide jaoks (vt tabel 2). Lisaks on joonisele kantud horisontaalne nulljoon, mis aitab võrrelda hinnangute nihete suurusi.

Üldiselt on näha, et logistilise regressioonimudeli abil saadud hinnangud varieeruvad vähem kui klassifitseerimispuuga leitud hinnangud. Lineaarse seose jaoks on logistilise regressioonimudeli abil saadud hinnangud ka väiksema nihkega kui klassifitseerimispuuga leitud hinnangud. Mittelineaarse seose korral on mõlema meetodiga leitud hinnangud ligikaudu sama nihkega ülemisest kvartiilist väiksemate väärtuste korral, kuid suuremate väärtuste korral on väiksema nihkega logistilise regressioonimudeli põhjal leitud hinnangud. Samuti on näha, et klassifitseerimispuu korral on mittelineaarse seose jaoks hinnangute varieeruvus suur mediaanist suuremate tõenäosuste hindamisel. Minimaalse väärtuse ja alumise kvartiili hindamisel mittelineaarse seose korral prognoosivad mõlemad meetodid tegelikke tõenäosusi sama hästi.

Tabelis 3 on toodud nii lineaarse kui ka mittelineaarse seose korral logistilise regressiooni ja klassifitseerimispuu meetodite vead (24). Lineaarse seose korral on antud viga klassifitseerimispuu meetodi korral märgatavalt suurem, kuid mittelineaarse seose korral pole meetodite vead oluliselt erinevad.

Tabel 3: Vead MSE_{MC}

Meetod \ **Seos**	Lineaarne	Mittelineaarne
Logistiline regressioon	0,00041	0,00659
Klassifitseerimispuu	0,00721	0,00662

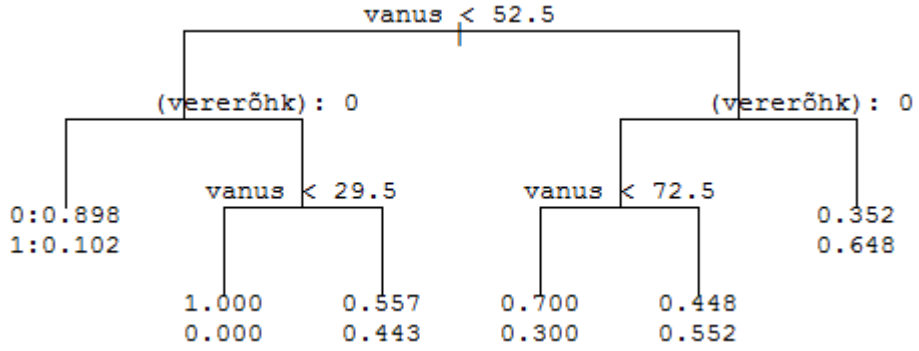
Joonise 8 ja tabelis 3 olevate vigade põhjal võib öelda, et logistiline regressioonimudel töötab südame- ja veresoonkonna haiguste esinemise tõenäosuse hindamisel paremini lineaarse tegeliku seose korral. Samas on klassifitseerimispuu meetod täpsem mittelineaarse tegeliku seose korral. Logistiline regressioonimudel on lineaarse seose korral märgatavalt parem kui klassifitseerimispuu, kuna esimese meetodi ruutkeskmine viga Monte-Carlo meetodil, MSE_{MC} on väiksem ning hinnangud väiksema nihkega. Mittelineaarse seose korral on nimetatud kaks meetodit samaväärsed, kuna nende vead MSE_{MC} erinevad vaid 0,00003 võrra ning saadud hinnangute nihked on ülemisest kvartiilist suuremate tõenäosuste hindamisel samas suurusjärgus.

Õpiku (James jt, 2013) põhjal võib klassifitseerimispuu meetod keerukamate mittelineaarsete seoste korral olla täpsem kui logistiline regressioonimudel. Klassifitseerimispuu on lihtsamini interpreteeritav kui logistiline regressioonimudel ning selle abil on hinnangute leidmine kergem kui logistilise regressioonimudeli põhjal. Seetõttu võib soovitada binaarse tunnuse modelleerimisel klassifitseerimispuu meetodit eriti mittelineaarse seose korral.

Näide 5. Esimesel simuleerimissammul tunnusele `süda1` hinnatud logistiline regressioonimudel on kujul

$$\operatorname{logit}\left(\hat{p}_i^{(1)}\right) = -5{,}26 + 0{,}05 \cdot \texttt{vanus}_i + 0{,}06 \cdot \texttt{KMI}_i + 0{,}99 \cdot \texttt{vererõhk}_i,$$

millest tõenäosuse $p_i^{(1)}$ hinnang on lähedane tegelikule tõenäosusele (23). Samale tunnusele hinnatud klassifitseerimispuu on toodud aga joonisel 9.



Joonis 9: Esimesel simuleerimissammul tunnusele `süda1` hinnatud klassifitseerimispuu

Südame- ja veresoonkonna haiguste esinemise tõenäosuse hinnangu leidmine 55-aastasele inimesele, kelle kehamassiindeks on 28,5 ja kellel pole kõrge vererõhuga

probleeme, on logistilise regressioonimudeli abil keerukam kui klassifitseerimispuu abil. Esimesel juhul peab välja arvutama avaldise

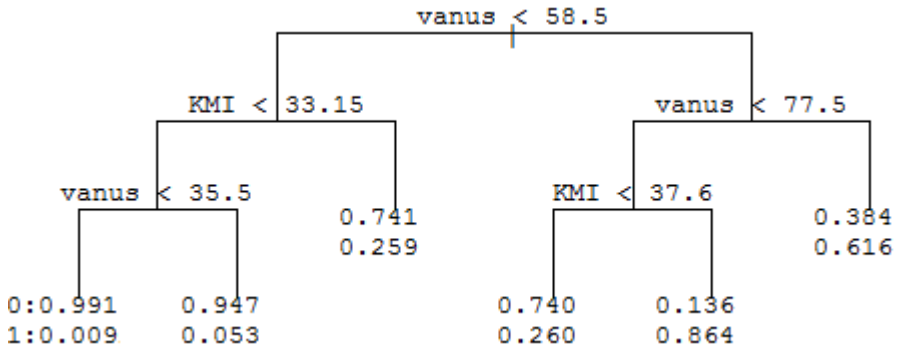
$$\hat{p}_i^{(1)} = \frac{\exp(-5{,}26 + 0{,}05 \cdot 55 + 0{,}06 \cdot 28{,}5 + 0{,}99 \cdot 0)}{1 + \exp(-5{,}26 + 0{,}05 \cdot 55 + 0{,}06 \cdot 28{,}5 + 0{,}99 \cdot 0)}$$

väärtuse, milleks on 0,310. Teisel juhul leitakse hinnang joonisel 9 oleva klassifitseerimispuus allapoole liikudes kuni jõutakse lõpliku piirkonnani. Sel juhul on hinnang soovitud tõenäosusele 0,300.

Vaadeldaval simuleerimissammul tunnusele `süda2` hinnatud logistilise regressioonimudeli kuju on järgnev:

$$\text{logit}\left(\hat{p}_i^{(2)}\right) = -8{,}36 + 0{,}08 \cdot \texttt{vanus}_i + 0{,}07 \cdot \texttt{KMI}_i.$$

Joonisel 10 on kujutatud tunnusele `süda2` hinnatud klassifitseerimispuud.



Joonis 10: Esimesel simuleerimissammul tunnusele `süda2` hinnatud klassifitseerimispuu

Eelnevalt vaadeldud isikule on südame- ja veresoonkonna haiguste esinemise tõenäosuse hinnang antud juhul logistilise regressioonimudeli põhjal

$$\hat{p}_i^{(2)} = \frac{\exp(-8{,}36 + 0{,}08 \cdot 55 + 0{,}07 \cdot 28{,}5)}{1 + \exp(-8{,}36 + 0{,}08 \cdot 55 + 0{,}07 \cdot 28{,}5)} \approx 0{,}123.$$

Samale tõenäosusele on joonisel 10 kujutatud klassifitseerimispuu põhjal hinnang 0,053.

Kokkuvõte

Antud bakalaureusetöö eesmärk oli kirjeldada ning omavahel võrrelda kahte binaarse tunnuse modelleerimiseks kasutatavat populaarsemat meetodit: logistilist regressiooni ja klassifitseerimispuu meetodit. Töös anti ka lühike ülevaade teistest levinumatest kaheväärtuselise tunnuse modelleerimisel kasutatavatest meetoditest.

Logistiline regressioon on parameetriline meetod, täpsemalt on tegu üldistatud lineaarse mudeli erijuhuga, kus seosefunktsioon on logit-funktsioon. Klassifitseerimispuu meetod on mitteparameetriline, tegu on otsustuspuu meetodiga kvalitatiivse uuritava tunnuse korral.

Nii logistilist regressioonimudelit kui ka klassifitseerimispuu meetodit rakendati südame- ja veresoonkonna haiguste esinemise prognoosimiseks Euroopa Sotsiaaluuringu andmetele (European Social Survey, 2014). Nimetatud probleemi lahendamisel töötasid mõlemad meetodid test ruutkeskmise vea väärtuste põhjal umbes sama hästi, kuid interpreteerida oli lihtsam leitud klassifitseerimispuud.

Logistilise regressioonimudeli ning klassifitseerimispuu meetodi võrdlemiseks läbi viidud simuleerimisülesandes genereeriti esmalt huvipakkuva sündmuse toimumise tõenäosus nii, et sündmuse toimumise šansi logaritm oli lineaarselt seotud mudeli argumenttunnustega. Teiseks genereeriti mittelineaarne seos. Lineaarse seose korral töötas vaadeldavatest meetoditest paremini logistiline regressioon, kuid teist liiki seose korral meetodid vea poolest märgatavalt ei erinenud. Seega võib soovitada mittelineaarse seose korral binaarse tunnuse modelleerimisel lihtsamini interpreteeritavat klassifitseerimispuu meetodit.

Kasutatud kirjandus

Agresti, A. (2002). *Categorical Data Analysis*. New Jersey: Wiley.

Davies, S. (*s.a.*). Class 'glm'. *Fitting Generalized Linear Models*. Kasutatud 26.02.2017 https://stat.ethz.ch/R-manual/R-patched/library/stats/html/glm.html

European Social Survey (2014). Kasutatud 08.04.2017 http://www.europeansocialsurvey.org

Hastie, T., Tibshirani, R., Friedman, J. (2011). *The Elements Of Statistical Learning: Data Mining, Inference, and Prediction*. New York: Springer.

James, G., Witten, D., Hastie, T., Tibshirani, R. (2013). *An Introduction to Statistical Learning: with Applications in R*. New York: Springer.

Käärik, E. (2013). *Andmeanalüüs II*. Loengukonspekt. Tartu Ülikool. Kasutatud 05.04.2017 http://dspace.ut.ee/bitstream/handle/10062/35401/AndmeanaluusII.pdf

Parring, A.-M. (1989). *Sissejuhatus matemaatilisse statistikasse*. Tartu: Tartu Ülikool.

Phipps, P., Toth, D. (2012). Analyzing establishment nonrepsonse using an interpretable regression tree model with linked administrative data. *The Annals of Applied Statistics*, **6**, 772—794.

Ripley, B. D. (1996). *Pattern Recognition and Neural Networks*. Cambridge: Cambridge University Press.

Ripley, B. D. (2016). Package 'tree'. *Classification and Regression Trees*. Kasutatud 17.03.2017 https://cran.r-project.org/web/packages/tree/tree.pdf

Lisad

Lisa 1. Näite 2 R kood ja väljundid

```
# Kasutatava andmestiku ESS loomine Euroopa Sotsiaaluuringu
# andmete põhjal
library(readxl)
andmed0=read_excel('<faili-asukoht>/<faili-nimi>.xlsx')

# Puuduvate andmetega vaatluste kustutamine
andmed0=subset(andmed0, ! YRBRN %in% c(7777, 8888, 9999))
andmed0=subset(andmed0, ! WEIGHT %in% c(777, 888, 999))
andmed0=subset(andmed0, ! HEIGHT %in% c(777, 888, 999))

# Kasutatava andmestiku ESS loomine
süda=andmed0$E28_1EE
ESS=data.frame(süda)
ESS$vanus=2014-andmed0$YRBRN
ESS$KMI=round(as.numeric(andmed0$WEIGHT)/
                ((as.numeric(andmed0$HEIGHT)/100)^2), 1)
ESS$sugu=ifelse(andmed0$GNDR==1, 0, 1)
ESS$vererõhk=andmed0$E28_2EE
attach(ESS)
summary(ESS)
```

```
      süda             vanus             KMI
 Min.   :0.0000   Min.   :15.00   Min.   :15.60
 1st Qu.:0.0000   1st Qu.:34.00   1st Qu.:22.70
 Median :0.0000   Median :51.00   Median :25.40
 Mean   :0.1477   Mean   :50.14   Mean   :26.02
 3rd Qu.:0.0000   3rd Qu.:65.00   3rd Qu.:28.70
 Max.   :1.0000   Max.   :99.00   Max.   :49.50
      sugu          vererõhk
 Min.   :0.0000   Min.   :0.000
 1st Qu.:0.0000   1st Qu.:0.000
 Median :1.0000   Median :0.000
 Mean   :0.5923   Mean   :0.257
 3rd Qu.:1.0000   3rd Qu.:1.000
 Max.   :1.0000   Max.   :1.000
```

```
# Andmestiku ESS jagamine mudeli koostamiseks kasutatavaks
# n-ö treeningandmestikuks ja uueks n-ö testandmestikuks
set.seed(12)
treening=sample(2004, 1002)
```

```
# Logistilise regressioonimudeli hindamine
log.mudel=glm(süda~vanus+KMI+as.factor(sugu)+as.factor(vererõhk),
              family=binomial(), data=ESS, subset=treening)
summary(log.mudel)
```

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.3656  -0.5470  -0.3334  -0.2143   2.9284

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)          -5.420902   0.645400  -8.399  < 2e-16 ***
vanus                 0.049529   0.006522   7.594 3.10e-14 ***
KMI                   0.015819   0.020778   0.761    0.446
as.factor(sugu)1      0.069805   0.208896   0.334    0.738
as.factor(vererõhk)1  0.934928   0.210206   4.448 8.68e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 821.36  on 1001  degrees of freedom
Residual deviance: 675.19  on  997  degrees of freedom
AIC: 685.19
```

```
# Tunnus sugu välja
log.mudel=glm(süda~vanus+KMI+as.factor(vererõhk),
              family=binomial(), data=ESS, subset=treening)
summary(log.mudel)
```

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.3593  -0.5488  -0.3329  -0.2144   2.9415

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)          -5.391380   0.640092  -8.423  < 2e-16 ***
vanus                 0.049844   0.006453   7.724 1.13e-14 ***
KMI                   0.015607   0.020792   0.751    0.453
as.factor(vererõhk)1  0.937959   0.209886   4.469 7.86e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 821.36  on 1001  degrees of freedom
Residual deviance: 675.30  on  998  degrees of freedom
AIC: 683.3
```

```
# Tunnus KMI välja
log.mudel=glm(süda~vanus+as.factor(vererõhk),
              family=binomial(), data=ESS, subset=treening)
summary(log.mudel)
```

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.3950  -0.5537  -0.3355  -0.2202   2.9076

Coefficients:
                     Estimate Std. Error z value Pr(>|z|)
(Intercept)          -5.02033    0.39900 -12.582  < 2e-16 ***
vanus                 0.05049    0.00637   7.926 2.27e-15 ***
as.factor(vererõhk)1  0.97437    0.20447   4.765 1.89e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 821.36  on 1001  degrees of freedom
Residual deviance: 675.86  on  999  degrees of freedom
AIC: 681.86
```

```
# Mudeli test ruutkeskmine viga
(tMSE=mean((süda-predict(log.mudel, ESS,
                         type="response"))[-treening]^2))
```

```
[1] 0.1045892
```

Lisa 2. Näite 3 R kood ja väljundid

Antud R kood on mõeldud kasutamiseks koos lisas 1 toodud R koodiga.

```
library(tree)
puu.mudel=tree(as.factor(süda)~vanus+KMI+as.factor(sugu)
               +as.factor(vererõhk), data=ESS, subset=treening)
summary(puu.mudel)
```

```
Classification tree:
Variables actually used in tree construction:
[1] "vanus"               "as.factor(vererõhk)" "KMI"
Number of terminal nodes:  6
Residual mean deviance:  0.6628 = 660.1 / 996
Misclassification error rate: 0.1427 = 143 / 1002
```

```
# Puu pügamise võimaluse kontrollimine
(puu.cv=cv.tree(puu.mudel, FUN=prune.misclass))
```

```
$size
[1] 6 1
$dev
[1] 144 144
$k
[1] -Inf    0
```

```
# Puu joonis tõenäosustega
par(family="mono")
par(mar=c(4, 1, 2, 1))
plot(puu.mudel, type="uniform");
text(puu.mudel, label="yprob", digits=3, pretty=0, cex=0.8)

# Puu test ruutkeskmine viga
(tMSE=mean((süda[-treening]
            -predict(puu.mudel,newdata=ESS[-treening,])[,2])^2))
```

```
[1] 0.1087237
```

Lisa 3. Näite 4 R kood ja väljundid

```
# Andmestiku 'veresuhkur' genereerimine
set.seed(2)
n=100
vanus=sample(35:60,n,replace=T)
kaal=sample(50:125,n,replace=T)
haigus=c(1:n)
for (i in 1:n){
  if (vanus[i]>=50){
    if (kaal[i]>=90){
      haigus[i]=rbinom(1,1,0.9) }
    haigus[i]=rbinom(1,1,vanus[i]/120) }
  else if (vanus[i]<=40){
    if (kaal[i]>=90){
      haigus[i]=rbinom(1,1,kaal[i]/150) }
    haigus[i]=rbinom(1,1,0.1) }
  else if (kaal[i]>=90){
    haigus[i]=rbinom(1,1,kaal[i]/125) }
  else {
    haigus[i]=rbinom(1,1,0.2) } }
diabeet=ifelse(haigus==1,"1","0")
veresuhkur=data.frame(diabeet,vanus,kaal)
summary(veresuhkur)
```

```
diabeet      vanus            kaal
 0:65    Min.   :35.00   Min.   : 51.00
 1:35    1st Qu.:40.00   1st Qu.: 65.00
         Median :46.00   Median : 81.00
         Mean   :47.33   Mean   : 85.98
         3rd Qu.:55.00   3rd Qu.:109.00
         Max.   :60.00   Max.   :124.00
```

```
# Klassifitseerimispuu koostamine
library(tree)
puu.mudel=tree(diabeet~vanus+kaal,data=veresuhkur)
summary(puu.mudel)
```

```
Classification tree:
Number of terminal nodes:  10
Residual mean deviance:  0.635 = 57.15 / 90
Misclassification error rate: 0.13 = 13 / 100
```

```
# Puu joonis klassidega
par(family="mono") # kirjastiili jaoks joonisel
plot(puu.mudel,type="uniform"); text(puu.mudel,pretty=0,cex=0.8)

# Puu piirkondade joonis
plot(vanus,kaal,col=1,bg=as.numeric(diabeet),pch=21)
partition.tree(puu.mudel,add=TRUE,cex=1)
plot.new() # legendi jaoks
legend("center",legend=unique(diabeet),
       col=1,bg=as.numeric(diabeet),pch=21)
```

```
# Puu pügamine
(puu.cv=cv.tree(puu.mudel, FUN=prune.misclass))
```

```
$size
[1] 10  7  5  4  3  1
$dev
[1] 28 28 28 28 32 36
$k
[1] -Inf  0.0  1.0  3.0  4.0  6.5
```

```
puu.pygatud=prune.misclass(puu.mudel, best=4)
summary(puu.pygatud)
```

```
Classification tree:
Number of terminal nodes:  4
Residual mean deviance:  0.8256 = 79.25 / 96
Misclassification error rate: 0.18 = 18 / 100
```

```
# Pügatud puu joonis klassidega
plot(puu.pygatud,type="uniform");
text(puu.pygatud,pretty=0,cex=0.8)

# Pügatud puu joonis tõenäosustega
plot(puu.pygatud,type="uniform");
text(puu.pygatud,label="yprob",digits=3,pretty=0,cex=0.8)
```

Lisa 4. Simuleerimisülesande R kood ja väljundid

```
library(tree)
set.seed(1)
attach(ESS) # Kasutatakse andmestiku 'ESS'
n=length(vanus)

# Genereeritakse lineaarne seos (seos 1)
p1=1/(1+exp(-(-5+0.05*vanus+0.04*KMI+vererõhk)))

# Genereeritakse mittelineaarne seos koosmõjudega (seos 2)
p2=1/(1+exp(-(
  -5+0.0005*vanus^2+0.0008*vanus*KMI+0.005*vererõhk*KMI)))
for (i in 1:n){
  if (vanus[i]>60){if (KMI[i]<20){p2[i]=0.8}}
  if (KMI[i]>38){p2[i]=0.8}
  }

# Kvantiilid ja neile vastavad vaatlused
(kv1=quantile(p1,type=3))
```

```
        0%         25%         50%         75%        100%
0.02814054  0.09363821  0.21081829  0.45412934  0.85717231
```

```
indeksid1=c(which(p1==kv1[1]),which(p1==kv1[2]),which(p1==kv1[3]),
            which(p1==kv1[4])[1],which(p1==kv1[5]))
(kv2=quantile(p2,type=3))
```

```
          0%          25%          50%          75%         100%
0.009238448  0.022903851  0.066732486  0.218525467  0.909181357
```

```
indeksid2=c(which(p2==kv2[1]),which(p2==kv2[2]),which(p2==kv2[3]),
            which(p2==kv2[4]),which(p2==kv2[5]))

# Tunnus KMI jagatakse gruppidesse
KMI_klass=ifelse(KMI<20,"[15,6;20)",
            ifelse(KMI<22.5,"[20;22,5)",
            ifelse(KMI<25,"[22,5;25)",
            ifelse(KMI<27.5,"[25;27,5)",
            ifelse(KMI<30,"[27,5;30)",
            ifelse(KMI<35,"[30,35)","[35;49,5]"))))))

# Tunnus vanus jagatakse gruppidesse
vanus_klass=ifelse(vanus<30,"[15;30)",
              ifelse(vanus<45,"[30;45)",
              ifelse(vanus<60,"[45;60)",
              ifelse(vanus<75,"[60;75)","[75;99]"))))

# Joonised tõenäosuste jaotumisest
# Lineaarne
par(family="mono")
op=par(mfrow=c(1,2),mar=c(7,4,1,0.1),font.main=1,cex=0.8)
boxplot(p1~vererõhk*KMI_klass,col=(c("darkgrey","white")),
        ylab="Tegelik tõenäosus p(1)",
        names=c("[15,6;20)","","[20;22,5)","","[22,5;25)",
                "","[25;27,5)","","[27,5;30)","","[30;35)",
```

```
                "","[35;49,5]",""),las=2)
points(c(1.5,3.5,5.5,7.5,9.5,11.5,13.5),
       t(aggregate(p1~KMI_klass,ESS,mean)[2]),pch=8)
title(sub="Kehamassiindeksi grupp",line=5.6)

boxplot(p1~vererõhk*vanus_klass,col=(c("darkgrey","white")),
        names=c("[15;30)","","[30;45)","","[45;60)","",
                "[60;75)","","[75;99]",""),
        las=2)
points(c(1.5,3.5,5.5,7.5,9.5),
       t(aggregate(p1~vanus_klass,ESS,mean)[2]),pch=8)
legend("topleft",title="Kõrge vererõhk", c("Ei","Jah"),
       fill=c("darkgrey","white"),cex=1)
title(sub="Vanusegrupp",line=4.6)
par(op)

# Mittelineaarne
op=par(mfrow=c(1,2),mar=c(7,4,1,0.1),font.main=1,cex=0.8)
boxplot(p2~vererõhk*KMI_klass,col=(c("darkgrey","white")),
        ylab="Tegelik tõenäosus p(2)",
        names=c("[15,6;20)","","[20;22,5)","","[22,5;25)",
                "","[25;27,5)","","[27,5;30)","","[30;35)",
                "","[35;49,5]",""),
        las=2)
points(c(1.5,3.5,5.5,7.5,9.5,11.5,13.5),
       t(aggregate(p2~KMI_klass,ESS,mean)[2]),pch=8)
title(sub="Kehamassiindeksi grupp",line=5.6)

boxplot(p2~vererõhk*vanus_klass,col=(c("darkgrey","white")),
        names=c("[15;30)","","[30;45)","","[45;60)","",
                "[60;75)","","[75;99]",""),
        las=2)
points(c(1.5,3.5,5.5,7.5,9.5),
       t(aggregate(p2~vanus_klass,ESS,mean)[2]),pch=8)
legend(x=0.1,y=0.7,title="Kõrge vererõhk", c("Ei","Jah"),
       fill=c("darkgrey","white"))
title(sub="Vanusegrupp",line=4.6)
par(op)

# Keskmised tõenäosused tunnuse vererõhk järgi
aggregate(p1~vererõhk,ESS,mean)
```

```
  vererõhk        p1
1        0 0.1899267
2        1 0.5743683
```

```
aggregate(p2~vererõhk,ESS,mean)
```

```
  vererõhk        p2
1        0 0.1091525
2        1 0.3030636
```

```
# Genereeritakse 100 korda uued tunnused süda1 ja süda2,
# millele hinnatakse nii logistiline regressioonimudel kui ka
# klassifitseerimispuu.
log1=matrix(nrow=5,ncol=100) # Kvartiilide vahede jaoks
ERlog1=c(1:100)
```

```
puu1=matrix(nrow=5,ncol=100)
ERpuu1=c(1:100)
log2=matrix(nrow=5,ncol=100)
ERlog2=c(1:100)
puu2=matrix(nrow=5,ncol=100)
ERpuu2=c(1:100)
for (i in 1:100){
  süda1=c(1:n)
  süda2=c(1:n)
  for (j in 1:n){
    süda1[j] = rbinom(1,1,p1[j])
    süda2[j] = rbinom(1,1,p2[j])
  }
  ESS$süda1=süda1
  ESS$süda2=süda2

  # Logistiline regressioonimudel tunnusele süda1
  log.mudel1=glm(süda1~vanus+KMI+as.factor(vererõhk),
                 family=binomial(), data=ESS)
  summary(log.mudel1)
```

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.9378  -0.7129  -0.4307   0.7946   2.6557

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)          -5.261718   0.360993 -14.576  < 2e-16 ***
vanus                 0.047211   0.003604  13.100  < 2e-16 ***
KMI                   0.057140   0.012346   4.628 3.69e-06 ***
as.factor(vererõhk)1  0.985921   0.125873   7.833 4.78e-15 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 2429.1  on 2003  degrees of freedom
Residual deviance: 1909.1  on 2000  degrees of freedom
AIC: 1917.1
```

```
  ERlog1[i]=mean((p1-predict(log.mudel1,ESS,type="response"))^2)
  log1[,i]=predict(log.mudel1,ESS[indeksid1,],type="response")
           -quantile(p1)

  # Klassifitseerimispuu tunnusele süda1
  puu.mudel1=tree(as.factor(süda1)~vanus+KMI
                  +as.factor(vererõhk),data=ESS)
  summary(puu.mudel1)
```

```
Classification tree:
Variables actually used in tree construction:
[1] "vanus"                "as.factor(vererõhk)"
Number of terminal nodes:  6
Residual mean deviance:  0.9695 = 1937 / 1998
Misclassification error rate: 0.227 = 455 / 2004
```

```
  par(mar=c(4,1,2,1))
  plot(puu.mudel1,type="uniform");
  text(puu.mudel1,label="yprob",
       digits=3,pretty=0,cex=0.8)
  ERpuu1[i]=mean((p1-predict(puu.mudel1)[,2])^2)
  puu1[,i]=(predict(puu.mudel1)[,2])[indeksid1]-quantile(p1)
  # Logistiline regressioonimudel tunnusele süda2
  log.mudel2=glm(süda2~vanus+KMI,
                 family=binomial(), data=ESS)
  summary(log.mudel2)
```

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.9688  -0.5394  -0.2597  -0.1228   3.1928

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -8.357986   0.518932 -16.106  < 2e-16 ***
vanus        0.082557   0.005269  15.668  < 2e-16 ***
KMI          0.066106   0.014263   4.635 3.57e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1750.1  on 2003  degrees of freedom
Residual deviance: 1293.5  on 2001  degrees of freedom
AIC: 1299.5
```

```
  ERlog2[i]=mean((p2-predict(log.mudel2,ESS,type="response"))^2)
  log2[,i]=predict(log.mudel2,ESS[indeksid2,],type="response")
           -quantile(p2)
  # Klassifitseerimispuu tunnusele süda2
  puu.mudel2=tree(as.factor(süda2)~vanus+KMI
                  +as.factor(vererõhk),data=ESS)
  summary(puu.mudel2)
```

```
Classification tree:
Variables actually used in tree construction:
[1] "vanus" "KMI"
Number of terminal nodes:  6
Residual mean deviance:  0.6317 = 1262 / 1998
Misclassification error rate: 0.1312 = 263 / 2004
```

```
  par(mar=c(4,1,2,1))
  plot(puu.mudel2,type="uniform");
  text(puu.mudel2,label="yprob",
       digits=3,pretty=0,cex=0.8)
  ERpuu2[i]=mean((p2-predict(puu.mudel2)[,2])^2)
  puu2[,i]=(predict(puu.mudel2)[,2])[indeksid2]-quantile(p2)
}

# Karpdiagrammide tegemise jaoks
koos1=cbind(t(log1)[,1],t(puu1)[,1],t(log1)[,2],t(puu1)[,2],
            t(log1)[,3],t(puu1)[,3],t(log1)[,4],t(puu1)[,4],
            t(log1)[,5],t(puu1)[,5])
koos2=cbind(t(log2)[,1],t(puu2)[,1],t(log2)[,2],t(puu2)[,2],
```

```
            t(log2)[,3],t(puu2)[,3],t(log2)[,4],t(puu2)[,4],
            t(log2)[,5],t(puu2)[,5])

# Karpdiagrammid
par(family="mono")
op=par(mfrow=c(2,1),mar=c(2,2,1,0.1),font.main=1,cex=0.8)
boxplot(koos1,ylim=c(-0.3,0.3),col=gray.colors(2),
        main="Lineaarne seos",
        names=c("Min"," ","Al."," ",
                "Med"," ","Ül."," ","Max"," "))
abline(h=0,lty=2)
legend("bottomleft",
       c("Logistiline regressioon","Klassifitseerimispuu"),
       fill=gray.colors(2))
boxplot(koos2,ylim=c(-0.3,0.3),col=gray.colors(2),
        main="Mittelineaarne seos",
        names=c("Min"," ","Al."," ",
                "Med"," ","Ül."," ","Max"," "))
abline(h=0,lty=2)
legend("bottomleft",
       c("Logistiline regressioon","Klassifitseerimispuu"),
       fill=gray.colors(2))
par(op)

# Arvutatakse ruutkeskmised vead Monte-Carlo meetodil
x=list(ERlog1,ERpuu1,ERlog2,ERpuu2)
sapply(x, mean)
```

```
[1] 0.0004071598 0.0072077329 0.0065907971 0.0066179706
```

Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks

Mina, Kristi Ernits,

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) enda loodud teose „Logistiline regressioon ja klassifitseerimispuu binaarse tunnuse modelleerimisel", mille juhendaja on Natalja Lepik,

    1.1. reprodutseerimiseks säilitamise ja üldsusele kättesaadavaks tegemise eesmärgil, sealhulgas digitaalarhiivi DSpace-is lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;

    1.2. üldsusele kättesaadavaks tegemiseks Tartu Ülikooli veebikeskkonna kaudu, sealhulgas digitaalarhiivi DSpace'i kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.

2. olen teadlik, et punktis 1 nimetatud õigused jäävad alles ka autorile.

3. kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest tulenevaid õigusi.

Tartus, 08.05.2017