TARTU ÜLIKOOL
MATEMAATIKA-INFORMAATIKATEADUSKOND
Matemaatilise statistika instituut
Finants- ja kindlustusmatemaatika eriala

Liina Muru

Kindlustuskahjude sageduse analüüs lokaalse regressiooni ja k-lähima naabri meetodil

Magistritöö (30 EAP)

Juhendaja: dotsent Meelis Käärik

Tartu 2015

Kindlustuskahjude sageduse analüüs lokaalse regressiooni ja k-lähima naabri meetodil

Kahjukindlustuses on üheks olulisemaks teemaks sobivate preemiate suuruste määramine. Sageli jagatakse selleks kindlustusvõtjad mingite tunnuste alusel erinevateks klassideks, et siis vastavas klassis hinnata kahjude suurust ja esinemise sagedust ning selle abil määrata preemiad. Klassidesse jagamise korral võib tekkida olukord, kus moodustatud klasside piiril asetsevate kindlustusvõtjate korral toob mõne vaadeldava tunnuse väike muutus kaasa sattumise teise klassi. See aga omakorda võib tuua kaasa preemia järsu muutumise ehk hinnašoki. Käesolevas töös uuritakse erinevaid meetodeid, et leida neist parim kindlustuskahjude esinemise sageduse võimalikult dünaamiliseks hindamiseks, mis vähendaks hinnašoki ohtu. Selleks kasutatakse lokaalset regressiooni, mille korral on piirkonnad määratud k-lähima naabri meetodit rakendades.
Märksõnad: regressioonanalüüs, kahjukindlustus, kindlustusmatemaatika

Claim frequency estimation using local regression and k-nearest neighbours method

Premium estimation is the main concept of non-life insurance. The most common approach is to divide the portfolio into subportfolis using some characteristic. We can estimate claim frequency and severity in each subportfolio to fix appropriate premium. In that case there is a possibility that small changes in client's data may result in large change in premiums – price shock – when client is situated on the border of a subportfolio. The main aim of this thesis is to analyse different methods to find the best and most dynamic method to estimate claim frequency and avoid the possibility of price shock in case of continuous variables. For that we use local regression with k-nearest neighbours method.
Keywords: regression analysis, non-life insurance, actuarial mathematics

Sisukord

Sissejuhatus

Kindlustuses on kõige olulisemaks teemaks õiglaste ning piisavate preemiate suuruste määramine, sest preemiad moodustavad kindlustusettevõtte sissetulekust suurima osa. Selleks, et määrata preemia suurust, peab hindama tekkida võivaid kahjusid – oluline on kahjude suurus ning nende esinemise sagedus. Esimeseks sammuks analüüsis on sageli kindlustusportfelli jagamine teatud kriteeriumite alusel alamportfellideks, siis saab vajalikke suurusi hinnata juba alamportfellis.

Klasside ehk alamportfellide moodustamise meetodeid on mitmeid ning sõltuvalt andmetest tuleb analüüsi käigus valida sobiv lähenemine. Üheks probleemiks portfelli klassifitseerimise juures on klasside piiridel paiknevad poliisid. Moodustades jäikade piiridega klassid võib mõne pideva tunnuse väikese muutuse korral kindlustusvõtja ühest poliisist teise liikuda, mis toob kaasa preemia järsu muutumise. Sellist olukorda nimetatakse hinnašokiks. See probleem tekib peamiselt pidevate tunnuste korral, milleks on kaskokindlustuses näiteks auto või omaniku vanus. Nominaalsete tunnuste korral, näiteks auto mark, mudel või see, kas varem on toimunud liiklusõnnetusi, on mõistetav preemia järsk muutumine. Lisaks muutuvad nominaalsed tunnused harvem, samas kui mitmed pidevad tunnused on ajas muutuvad.

Käesoleva töö eesmärgiks on uurida, kas dünaamilisemate klassipiiride kasutamine muudab kahjude hindamise täpsemaks, kui fikseeritud piiridega klasside moodustamine. Selleks kasutame iga poliisi analüüsimiseks lokaalset

regressiooni, arvestades tema ümbruseks k talle mingite tunnuste alusel lähimat poliisi. Sellist ümbruse defineerimise viisi nimetatakse k-lähima naabri meetodiks. Lisaks uurime erinevaid kauguse definitsioone, mida lisaks eukleidilisele kaugusele kasutada saab.

Magistritöö on jagatud viieks peatükiks. Töö esimeses osas kirjeldatakse kindlustuskahjude hindamise põhimõtteid, kindlustusportfelli jagamist alamportfellideks ja CART-meetodit. Teises peatükis tutvustatakse k-lähima naabri meetodit, mille abil määrata punkti ümbrust, ja selle erinevaid kasutusvõimalusi. Kolmandas peatükis defineeritakse erinevad kaugused, mida meetodi rakendamisel kasutada saab. Lisaks tutvustatakse lühidalt nende omadusi. Neljandas peatükis rakendatakse lokaalset regressiooni, et leida võrrandid, mille abil hinnata kindlustuskahjude esinemise sagedust. Viimases peatükis rakendatakse tutvustatud meetodeid reaalsetele Eesti kaskokindlustuse andmetele. Hindamiseks kasutatakse leitud lokaalse regressiooni mudelit, kus punkti ümbrus on leitud k-lähima naabri meetodil.

Töö on koostatud kasutades tekstitöötlusprogrammi LaTeX. Analüüsiks ja jooniste koostamiseks on kasutatud statistikaprogrammi R.

Autor tänab juhendaja dotsent Meelis Käärikut konsultatsioonide ja sisukate märkuste eest.

Peatükk 1

Kindlustuskahjude analüüs

Käesoleva töö aluseks on võetud 2012. aastal ilmunud Meelis Kääriku ja Ants Kaasiku artikkel [5] kahjude hindamisest CART-meetodil (*Classification and Regression trees*). Eesmärgiks on uurida meetodeid, mis on samadel eeldustel klassipiiride määramisel dünaamilisemad kui artiklis käsitletud.

1.1 Kollektiivmudel

Kollektiivmudeli ideeks on kindlustusportfell jagada alamportfellideks teatud tunnuste alusel. Nii saame hindamiseks väiksemad ja teatud tunnuste poolest sarnasemad klassid. Igas klassis saab seejärel leida hinnanguid nii kogu kahju suurusele kui kahjude esinemise sagedusele selles klassis. Antud töös keskendutakse kahjude esinemise sageduse hindamisele.

Defineerime kollektiivmudeli kogu kahju S kui juhusliku summa

$$S = \sum_{j=1}^{N} Z_j,$$

kus juhuslik suurus N on kahjude arv vaadeldavas perioodis ja Z_j nende suurus. [5]

Kollektiivmudeli korral tehakse eeldus, et kahjude arv N on sõltumatu üksikkahjude suurusest Z_j ja fikseeritud $N = n$ korral on kahjude suurused $Z_1, \ldots, Z_n$ sõltumatud sama jaotusega juhuslikud suurused.

1.2 CART-meetod

CART-meetodi ehk klassifitseerimise ja regressioonipuude meetodi peamine idee seisneb selles, et antakse ette vaadeldavatel tunnustel põhinevate lihtsate reeglite kogumik, mille alusel klassidesse jagamine toimub. Klassifitseerimine toimub sammhaaval ja igal sammul jagatakse antud klassi kuuluvad poliisid valitud tunnuse põhjal kahte klassi ehk lehte. Igal sammul lisandub puule üks leht. Järgmisel sammul jagatakse iga saadud klass omakorda kaheks. Nii võib klassifitseerimist jätkata, kuni igas klassis on ainult üks poliis, kuid enamasti saadakse piisavalt hea jaotus juba varem. Igal sammul tuleb valida klass, mida jagama hakatakse ning tunnus, mille alusel seda tehakse.

Valiku tegemisel on eesmärgiks minimiseerida puu hälvet

$$D(T) = \sum_{i=1}^{n} (\lambda_{[i]} t_i - n_i \log(\lambda_{[i]} t_i)),$$

kus T on meie vaadeldav mudel (puu), n_i on kahjude arv poliisis i, t_i on kindlustusperiood ja $\lambda_{[i]}$ on keskmine kahjude arv ühes ajaühikus klassis, kuhu kuulub poliis i. [5] Mudeli hälve võimaldab omavahel võrrelda erinevaid mudeleid. Iga järgmine klassifitseerimissamm peaks kirjeldatud hälvet vähendama fikseeritud väärtuse võrra, et saadud puu oleks parem kui eelmine. Vastasel juhul pole selle jaotuse tegemine enam kasulik ja klassifitseerimine lõpetatakse.

Defineerime uue suuruse

$$D_\alpha(T) = D(T) + \alpha|T|,$$

kus $|T|$ on lehtede arv puus T ja $\alpha \geq 0$ on fikseeritud parameeter, mis väljendab ühe lehe lisamise "hinda". Selline suurus arvestab lisaks hälbele ka

lehtede arvuga puus ja sõltub α väärtusest. Nii saame võrrandi, mida valitud α korral minimiseerides leiame parima puu kõikide maksimaalse puu T_∞ alampuude seast. Defineerides $\alpha = 0$ saame olukorra, kus lehtede arv puus pole oluline ning parimaks puuks on maksimaalne puu ehk selline, kus igas lehes on üks poliis. Optimaalseks puuks on see puu, mille korral D_α on minimaalne.

Sellise klassifitseerimise teel jagame kõik poliisid klassidesse ning leiame igas klassis keskmise kahjude esinemise sageduse. Poliisis i esinenud kahjude arvu n_i saab defineerida kui summa kahjude arvust poliisis i ajahetkel j

$$n_i = \sum_{j=1}^{t_i} n_{ij}.$$

Seega kahjude esinemise sagedus on igas klassis määratud kui

$$\lambda_{[i_1]} = \lambda_{[i_1]} = \ldots = \lambda_{[i_1]} = \frac{\sum_{j=1}^{n} n_{i_j}}{\sum_{j=1}^{n} t_{i_j}},$$

kus vaadeldav klass koosneb poliisidest $i_1, i_2, \ldots, i_n$.

CART-meetod fikseerib kindlad piirid, mille alusel jaotus klassidesse tehakse. Pidevate tunnuste korral võib sellise algoritmi korral tekkida probleeme klasside piiridel asuvate poliiside paigutamisel. Nende jaoks on väikese parameetrite muudatuse korral võimalik olukord, kus muutuse tulemusena langevad vaadeldud poliisid teise klassi. Kahjukindlustuses võib selline olukord tekkida näiteks vanuse alusel klassipiiride loomisel, kus väike muutus kasutaja andmetes toob kaasa väga suure muutuse kindlustusmakses, kui klient asub mõne vanuseklassi piiril ning selle ületab. Sellist olukorda nimetatakse hinnašokiks ning suurimaks probleemiks ongi see just pidevate tunnuste korral. Selle probleemi üheks lahenduseks oleks leida dünaamilisem klassideks jagamise meetod.

Peatükk 2

K-lähima naabri meetod

Antud uurimustöö raames vaatleme ühte lihtsamini rakendatavat klassideks jagamise meetodit, milleks on k-lähima naabri meetod. Erinevalt parameetrite kaudu üheselt fikseeritud klassidest võimaldab see meetod väärtusi klasterdada dünaamilisemalt. Nii nagu kõikide klassifitseerimisalgoritmide puhul, on ka selle meetodi eesmärk grupeerida vaadeldavad punktid nii, et ühte gruppi kuuluvad punktid on teatud tunnuste mõttes lähemal üksteisele kui teistesse gruppidesse kuuluvatele punktidele.

K-lähima naabri meetodi rakendamine on küllaltki lihtne, sest ainsateks eeldusteks on, et k oleks fikseeritud positiivne täisarv, olemas oleks treeningandmed ehk teadaolevad väärtused ning määratud peab olema meetrika, mida soovime kasutada. Meetodi eeliseks on selle kerge kasutamine ka suure valimi korral.

Seda mitteparameetrilist klassifitseerimise meetodit tutvustati esmakordselt 1951. aasta Fix'i ja Hodges'i artiklis [2] ning sellest ajast alates on seda edasi arendatud ning kasutusele võetud erinevates valdkondades, millede hulka ka kindlustusportfelli klasterdamine kuulub. Mainitud artiklis oli eesmärgiks teades mingit juhusliku suuruse Z realisatsiooni z määrata, kumb kahest ette antud jaotuseset sobib juhuslikule suurusele Z.

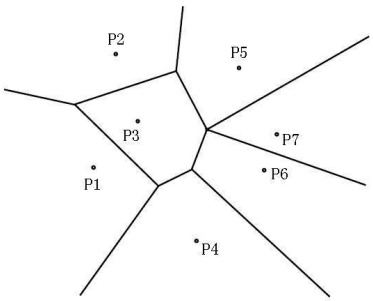
2.1 Regressioon ja klassifitseerimine

K-lähima naabri meetodi kasutamise saab vastavalt eesmärgile jagada kaheks: regressioon- või klassifitseerimisülesandeks.

2.1.1 Klassifitseerimisülesanne

Klassifitseerimise korral on meetodi väljundiks klass, millesse vaadeldav punkt kuulub, arvestades tema lähimaid naabreid, ehk punkt määratakse klassi, mille elemente on tema ümbruses kõige rohkem. Vaadeldav ümbrus sisaldab k talle lähimat punkti eelnevalt defineeritud meetrika mõttes. Näiteks võib ümbruse defineerimiseks leida punkti kauguse kõigist punktidest, saadud tulemused sorteerida kasvavas järjekorras ning võtta seejärel k esimest punkti.

Erijuhuks on olukord, kus $k = 1$ ning sel juhul määratakse punkt samasse klassi, kus on tema lähim naaber.



Joonis 2.1: Juhul, kui $k = 1$ jagavad teadaolevad punktid ruumi üheselt klassideks. [9]

Sel moel tasapinna jaotamist nimetatakse Voronoi diagrammiks. [9] Nii on teadaolevate punktide abil võimalik valimiruum vastavalt defineeritud kaugusele jagada üheselt klassideks nagu on näha joonisel 2.1.

Klassifitseerimiseks on erinevaid võimalusi ka siis, kui naabrite hulk on kindlaks määratud. Viise, kuidas iga punkt otsuse tegemisse panustab on erinevaid. Üheks on nn enamushääletus – klassi määramiseks loetakse ümbruses kokku punktide hulk iga klassi korral, kusjuures iga punkt panustab võrdselt. Punkt määratakse klassi, mille esindajaid vaadeldavas ümbruses kõige rohkem on. Teisel juhul on võimalik häälte andmist kaaluda. Kõige rohkem kasutatakse kaaluks punktidevahelise kauguse d pöördväärtust.

2.1.2 Regressioonülesanne

K-lähima naabri meetodi kasutamisel regressiooni korral on meetodi väljundiks otsitava tunnuse y hinnatud väärtus, mis leitakse sisendiks oleva punkti k lähima naabri vastavate tunnuste keskmise abil. Tunnuse hindamiseks on antud valem

$$\hat{y} = \frac{1}{k} \sum_{x_i \in N_k(x)} y_i,$$

kus $N_k(x)$ on punkti x selline naabrus, mis on defineeritud k talle lähima punkti $x_i = (x_{i1}, x_{i2}, \ldots, x_{im})$ kaudu [4]. Suurused x_{ij} tähistavad regressori ehk sisendtunnuse j väärtust i-ndal vaatlusel.

Klassifitseerimist võib pidada k-lähima naabri regressioonülesandeks, kus tunnus on nominaalne. Ka regressiooni korral on võimalik kõiki naabrusesse sattunud punkte arvestada võrdselt või lisada neile kaalud, et iga punkt panustaks proportsionaalselt.

Meetodi plussiks on kerge rakendatavus ka suurte andmehulkade korral. Puudusteks on suur tundlikkus ebaoluliste vaatluste osas, sest kõik punktid panustavad meetodisse võrdselt ja nn dimensionaalsuse needus: kui vaadeldavate tunnuste hulk kasvab, siis on punktid teineteisest kaugemal, sest iga tunnus panustab. See muudab uute punktide klassifitseerimise keerulisemaks. Lisaks on meetodi rakendamiseks vaja teada treeningandmeid ehk siis selleks, et ühte punkti hinnata peab eksisteerima teatud hulk punkte, mille tunnuste

väärtused on teada.

2.2 Optimaalse k valik

Meetodit rakendades on üks esimesi küsimusi, kuidas valida selline k, et tulemused oleks võimalikult täpsed. Üldiselt öeldes klassifitseerib suurem k andmeid paremini, sest võtab arvesse rohkem punkte ning vähendab sellega müra andmetes, tegu on silumisparameetriga. Samas suureneb k suurendamisel ka arvutuse keerukus ja lisaks tuleks tähele panna, et kui $k \to n$, kus n on valimi maht, ei toimu enam klassifitseerimist ja hindamisel arvestatakse kogu valimiga.

Mõned kindlamad soovitused on k valikuks diskreetsel juhul antud, näiteks binaarsete ehk kaheklassiliste ülesannete puhul tuleks k valida paaritu, sest see väldib viiki jäämise võimalust tehes otsust kahe klassi vahel. Regressiooni korral on kõige lihtsam k-d suurendada teatava sammuga seni, kuni järgmine samm ei anna enam nähtavalt paremaid tulemusi mudeli headuse parandamiseks. Näiteks alustada $k = 50$ ning igal sammul seda kahekordistada [7].

Käesoleva töö raames analüüsime k-lähima naabri meetodil kahjukindlustuse kindlustuspoliise, mis on erineva kestusega. See tähendab, et arvestame iga vaatluse korral lisaks ka sellega, kui pikk kindlustusperiood vaatlusele vastab. Sel juhul ei kasuta me naabruse suuruse määramisel ainult tema ümbrusesse sattunud poliiside arvu, vaid ka vastavate poliiside kindlustuspäevade arvu.

2.3 Puuduvate väärtustega tegelemine

Puuduvad väärtused on praktikas väga sagedasti esinev probleem ning nagu teisi meetodeid mõjutab see ka k-lähima naabri meetodit - punktidevahelist kaugust d pole võimalik leida, kui mõni vaadeldavatest väärtustest puudub.

Kõige lihtsam lahendus on puuduvate väärtustega punktid kõrvale jätta. Samas võib väikese valimi mahu ja paljude puuduvate väärtuste korral see keeruline olla. Üheks lahenduseks on asendamise meetod, kus punkti x_i puuduv väärtus tunnuse j korral x_{ij} asendatakse sama tunnuse keskmise väärtusega $\bar{x_j}$. Teine võimalus on kauguse asendamine keskmise kaugusega, mis on arvutatud vaadeldava tunnuse teadaolevate väärtuste pealt. Juhul, kui punktide x_i ja $x_{i'}$ j-nda tunnuse vahelist kaugust $d_j(x_{ij}, x_{i'j})$ pole võimalik leida, kuna x_{ij} või $x_{i'j}$ on puuduv väärtus asendame selle keskmise kaugusega vaadeldud tunnuse j korral

$$\bar{d_j} = \frac{1}{n^2} \sum_{i=1}^{n} \sum_{i'=1}^{n} d_j(x_{ij}, x_{i'j}),$$

kus n on teadaolevate vaatluste arv. [7]

Peatükk 3

Kauguse defineerimine meetodi rakendamiseks

Selleks, et rakendada lähinaabrite meetodit on vaja defineerida konkreetne kaugus, mida me ümbruste leidmiseks kasutame. Kõige sagedamini kasutatakse eukleidilist kaugust, kuid täpsemate tulemuste saamiseks võib kaaluda ka teisi definitsioone. Parimaks objektidevahelise sarnasuse mõõduks võivad erinevate andmete korral osutuda erinevalt defineeritud kaugused.
Kõik defineeritud kaugused vastavad järgmistele aksioomidele:

$$\begin{aligned} d(x,y) &= 0 \Leftrightarrow x = y \quad (samasus), \\ d(x,y) &= d(y,x) \quad (sümmeetria), \\ d(x,y) &\leq d(x,z) + d(z,y) \quad \forall z \quad (kolmnurga\ võrratus). \end{aligned}$$

Nende omaduste kehtivusega arvestame edaspidi iga kauguse defineerimisel.

3.1 Eukleidiline kaugus

Punktide x ja y vaheline eukleidiline kaugus avaldub kujul

$$d(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}.$$

Eukleidilise kauguse kasutamisel k-lähima naabri meetodi ümbruse määramiseks ei võeta arvesse kasutatavate tunnuste vahelisi seoseid ja see võib osutuda antud definitsiooni puuduseks. Eukleidilise kauguse kasutamise plussiks on kindlasti tema lihtne rakendatavus, kuna pole vaja teha lisasamme enne punkti kauguse leidmist, teades vaatlusandmeid.

3.2 Minkowski kaugus

Eukleidiline kaugus on erijuht Minkowski kaugusest juhul kui $q = 2$. Üldine valem avaldub kujul

$$d_{Minkowski}(x,y) = (\sum_{i=1}^{n} |x_i - y_i|^q)^{\frac{1}{q}}.$$

Erijuhul $q = 1$ on tegemist Manhattani kaugusega

$$d_{Manhattan}(x,y) = \sum_{i=1}^{n} |x_i - y_i|.$$

Suurema q-väärtusega erijuhte kasutatakse klassifitseerimises harva, sest q väärtuse kasvades antakse suurem kaal neile tunnustele, mille poolest punktid erinevad kõige rohkem. Minkowski kauguse puhul on tegu üldistatud meetrikaga, mis vastab aksioomidele ning samamoodi nagu eelnevalt vaadeldud erijuht $q = 2$ korral, ei võta see ka teiste q väärtuste korral arvesse vaatlusandmete vahelisi seoseid. [1]

3.3 Mahalanobise kaugus

Statistikas on kasutusel kaugus punkti x ja jaotuse F vahel, mida nimetatakse Mahalanobise kauguseks ning mis on defineeritud kui

$$D_M(x) = \sqrt{(x-\mu)^T C^{-1}(x-\mu)}, \tag{3.1}$$

kus $x = (x_1, x_2, x_3, \ldots, x_n)^T$ on vaatlus ja $\mu = (\mu_1, \mu_2, \mu_3, \ldots, \mu_n)^T$ on jaotuse F keskväärtus ning C selle jaotuse kovariatsioonimaatriks. [6]

Mahalanobise kaugust valemiga (3.1) võib vaadelda kui kaugust, mis mõõdab mitme standardhälbe kaugusel on vaadeldav punkt jaotuse keskväärtusest. Mida lähemal on punkt keskväärtusele, seda väiksem on kaugus. Selleks, et Mahalanobise kaugust kasutada klassifitseerimises, on kõigepealt vaja hinnata kõikide klasside kovariatsioonimaatrikseid teadaolevate vaatluste abil. Siis on võimalik testandmete korral arvutada kaugus kõikidest klassidest ning vaatlus määrata vastavalt algoritmile klassi, millest kaugus on minimaalne.

Mahalanobise kaugust on võimalik defineerida ka kui erisuse mõõtu kahe samast jaotusest juhusliku suuruse $X = (X_1, \ldots, X_n)$ ja $Y = (Y_1, \ldots, Y_n)$ vahel

$$d_M(x, y) = \sqrt{(x-y)^T C^{-1}(x-y)},$$

kus $x = (x_1, \ldots, x_n)^T$ on realisatsioon X-st ja $y = (y_1, \ldots, y_n)^T$ on realisatsioon Y-st. [3]

Sel kujul definitsiooni on võimalik kasutada ka lähinaabrite meetodi korral. Lisaks näeme siit, et kui kovariatsioonimaatriks on ühikmaatriks, siis

taandub Mahalanobise kaugus eukleidiliseks kauguseks

$$\begin{aligned} d_M(x,y) &= \sqrt{(x-y)^T I^{-1}(x-y)} = \sqrt{(x-y)^T(x-y)} = \\ &= \sqrt{(x_1-y_1, \quad \ldots, \quad x_n-y_n) \begin{pmatrix} x_1-y_1 \\ \ldots \\ x_n-y_n \end{pmatrix}} = \\ &= \sqrt{(x_1-y_1)(x_1-y_1)+\ldots+(x_n-y_n)(x_n-y_n)} = \\ &= \sqrt{\sum_{i=1}^{n}(x_i-y_i)^2} = d(x,y). \end{aligned}$$

Mahalanobise kaugust võib seega võtta kui eukleidilise kauguse edasiarendust, mis võtab arvesse ka tunnustevahelist kovariatsiooni.

Kui kovariatsioonimaatriks on diagonaalmaatriks, siis saame erijuhu, mida nimetatakse normeeritud eukleidiliseks kauguseks:

$$\begin{aligned} d_M(x,y) &= \sqrt{(x-y)^T C^{-1}(x-y)} = \\ &= \sqrt{(x_1-y_1, \quad \ldots, \quad x_n-y_n) \begin{pmatrix} c_{11} & 0 & \ldots & 0 \\ 0 & c_{22} & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & c_{nn} \end{pmatrix}^{-1} \begin{pmatrix} x_1-y_1 \\ \ldots \\ x_n-y_n \end{pmatrix}} = \\ &= \sqrt{(x_1-y_1, \quad \ldots, \quad x_n-y_n) \begin{pmatrix} \frac{1}{c_{11}}(x_1-y_1) \\ \ldots \\ \frac{1}{c_{nn}}(x_n-y_n) \end{pmatrix}} = \\ &= \sqrt{\sum_{i=1}^{n}\frac{(x_i-y_i)^2}{c_{ii}}} = \sqrt{\sum_{i=1}^{n}\frac{(x_i-y_i)^2}{\sigma_i^2}}. \end{aligned}$$

ja kuna x ja y on realisatsioonid samast jaotusest, siis kovariatsioonimaatriksi elementideks peadiagonaalil on vektorite elementide vahelised dispersioonid $c_{ii} = \sigma_i^2$.

3.4 Hamming'i kaugus

Eelnevalt kirjeldatud kaugused sobivad arvuliste tunnuste hindamiseks, kuid analüüsis tuleb sageli ette ka nominaaltunnuseid, mille korral kirjeldatud kauguste kasutamine pole võimalik. Lihtsaim nominaaltunnuste vahelise kauguse definitsioon on Hamming'i kaugus, mis on 0 kui tunnused on võrdsed ja 1 muudel juhtudel:

$$d_{Hamming}(x,y) = \begin{cases} 0, & kui \quad x = y, \\ 1, & mujal \end{cases} .$$

Selliste tunnuste hindamise korral, millest osa on nominaalsed ja osa mitte, peab kauguse definitsioon olema paindlik ning heaks lahenduseks oleks erinevate kauguste definitsioonide ühendamine vastavalt tunnuse tüübile

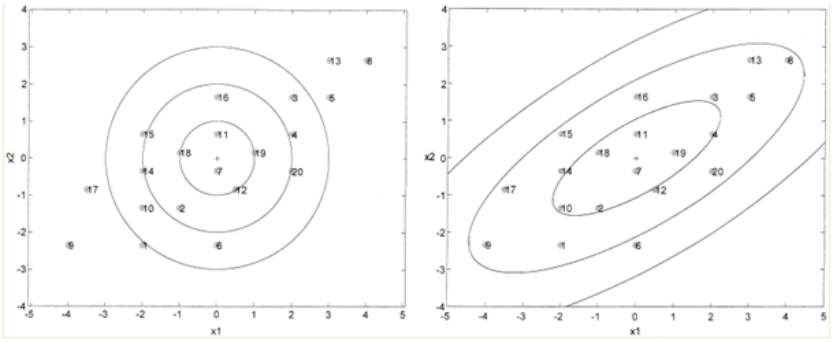
$$d(x,y) = \sum_{j=1}^{n} d_j(a_j, b_j),$$

kus $d_j(a_j, b_j)$ on Hamming'i kaugus, kui tegu on nominaaltunnusega, ja näiteks eukleidiline või Mahalanobise kaugus, kui tegu on arvtunnustega [8]. Käesolevas töös me nominaalsete tunnustega ei tegele, sest nende korral ei ole probleemiks kindlate piiridega klassifitseerimine, mida antud töös lahendada üritame.

3.5 Optimaalse kauguse valik

Kõiki tutvustatud kaugusi saab kasutada k-lähima naabri meetodis naabruste määramiseks. Minkowski kauguse ja selle erijuhtude leidmine on lihtsam, kuna ei pea tegema eraldi samme arvutamaks kovariatsioonide maatriksit. Samas võtab Mahalanobise kaugus arvesse ka vaadeldavate tunnuste vahelisi seoseid. Seetõttu tulebki sõltuvalt üldkogumist ja teadaolevast infost langetada otsus, kas vaadeldavate andmete vahelised seosed on piisavalt olulised, et neid arvesse võtta algoritmi keerukust tõstes või on olulisem meetodi kiirus.

Klassid, mis erinevate kauguste defineerimise kaudu moodustuvad, erinevad küllaltki suurel määral ning punkti kuulumine teatud klassi sõltub väga tugevalt definitsioonist. Seda on näha joonisel 3.1, kus Mahalanobise kaugus võtab arvesse punktide jaotust ning eukleidiline kaugus seda ei tee. Seega võivad punktid, mis ühe kauguse korral langevad samasse klassi, teise kauguse korral seda mitte teha.



Joonis 3.1: Erinevus eukleidilise(a) ja Mahalanobise(b) kauguse abil defineeritud klasside kujus ja punktide klassidesse paigutuses. [6]

Peatükk 4

Lokaalne regressioon kahjude esinemise sageduse analüüsiks

Selleks, et kindlustusjuhtumite esinemise sagedust teadaolevate tunnuste abil prognoosida, tuleb lahendada regressioonülesanne. Lokaalne regressioon on regressiooni vorm, kus prognoosi leidmiseks kasutatakse ainult vaadeldava punkti (poliisi) teatud ümbrusesse jäävaid punkte. Erinevalt tavalisest regressioonist, mis arvestab kogu valimiga, lihtsustab lokaalselt leitav mudel sobitamist ning erinevate ümbruste eripärade arvesse võtmist. Lokaalse regressiooni jaoks vajalike punkti ümbruste leidmiseks kasutame k-lähima naabri meetodit, mis on kirjeldatud teises peatükis.

Kindlustusportfelli jagamisel alamportfellideks eeldatakse sageli, et sellesse kuuluvate ja analüüsis kasutatavate kindlustuspoliiside kestused on võrdsed. Reaalsete andmete korral on sellise eelduse tegemine enamasti võimatu ja täpsema mudeli saamiseks tuleks arvesse võtta ka iga poliisi kestust. Käesolevas töös arvestame prognoosimisel ka iga mudeli kestusega ja seega on oluline kasutatavaid parameetreid defineerides aega arvesse võtta.

Tähistame

- t_i – poliisi i kestus päevades (kindlustusperiood),

- n_{ij} – kahjude arv poliisis i ajaühikus j,
- n_i – poliisi i kahjude arv, $n_i = \sum_{j=1}^{t_i} n_{ij}$,
- λ_i – keskmine kahjude arv ühes ajaühikus klassis, mis sisaldab poliisi i,
- N_{ij} – kahjude esinemise sagedus poliisis i ajahetkel j,
- N_i – kahjude esinemise sagedus kogu kindlustusperioodis t_i.

Juhusliku suuruse N_{ij} jaotuse valimiseks on kolm klassikalist võimalust [5]

- binoomjaotus $N_{ij} \sim B(n, p)$,
- negatiivne binoomjaotus $N_{ij} \sim NBin(n, p)$,
- Poissoni jaotus $N_{ij} \sim Po(\lambda)$.

Sel juhul kehtivad järgmised seosed kahjude esinemise sageduse kohta ajahetkes ja kogu kindlustusperioodi kahjude esinemise sageduse vahel

- kui $N_{ij} \sim Po(\lambda)$, siis $N_i \sim Po(\lambda t_i)$,
- kui $N_{ij} \sim NBin(n, p)$, siis $N_i \sim NBin(nt_i, p)$,
- kui $N_{ij} \sim B(n, p)$, siis $N_i \sim B(nt_i, p)$.

Käesolevas töös eeldame, et kahjude esinemise sagedus N_i on Poissoni jaotusega tõenäosusfunktsiooniga

$$\frac{(\lambda_i t_i)^{n_i}}{n_i!} e^{-(\lambda_i t_i)}.$$

Teades seda, leiame parameetri λ suurima tõepära hinnangu

$$\hat{\lambda} = \frac{\sum_{i=1}^{n} n_i}{\sum_{i=1}^{n} t_i}.$$

Lokaalse regressiooni rakendamiseks tähistagu $J(x)$ nende poliiside indeksite hulka, mille korral regressorite väärtused on vaadeldava väärtuse x naabruses,

a ja b regressiooni parameetreid ja x_i regressori väärtust poliisis i. Ühe regressoriga lokaalse regressiooni jaoks leiame nüüd Poissoni jaotusega mudeli tõepärafunktsiooni:

$$L_x(a,b) = \prod_{i \in J(x)} \frac{((a+bx_i)t_i)^{n_i}}{n_i!} e^{-(a+bx_i)t_i}.$$

Selle abil leitud log-tõepära avaldub

$$l_x(a,b) = \ln(L_x(a,b)) = \sum_{i \in J(x)} n_i \ln((a+bx_i)t_i) - \sum_{i \in J(x)} n_i! - \sum_{i \in J(x)} (a+bx_i)t_i.$$

Suurima tõepära hinnangu leidmiseks leiame nüüd tuletised mõlema regressiooni parameetri järgi:

$$\begin{aligned}
\frac{\partial l_x(a,b)}{\partial a} &= \frac{\partial}{\partial a}\Big(\sum_{i \in J(x)} n_i \ln(at_i + bx_i t_i) - \sum_{i \in J(x)} n_i! - \sum_{i \in J(x)} (at_i + bx_i t_i)\Big) \\
&= \sum_{i \in J(x)} \frac{n_i}{(a+bx_i)} - \sum_{i \in J(x)} t_i, \\
\frac{\partial l_x(a,b)}{\partial b} &= \sum_{i \in J(x)} \frac{n_i x_i}{(a+bx_i)} - \sum_{i \in J(x)} t_i x_i.
\end{aligned}$$

Nende tuletiste võrdsustamisel 0-ga saame võrrandid suurima tõepära hinnangute leidmiseks

$$\begin{cases} \sum_{i \in J(x)} \frac{n_i}{(a+bx_i)} = \sum_{i \in J(x)} t_i, \\ \sum_{i \in J(x)} \frac{n_i x_i}{(a+bx_i)} = \sum_{i \in J(x)} t_i x_i. \end{cases} \tag{4.1}$$

Ühe regressori korral võib regressoriks võtta näiteks omaniku või auto vanuse.

Kahe regressori korral olgu $J(x_1, x_2)$ nende poliiside indeksite hulk, mille regressorite väärtused langevad (x_1, x_2) ümbrusesse, a, b_1, b_2 regressioonipa-rameetrid ning $x_{1,i}$ ja $x_{2,i}$ regressori väärtused poliisi i jaoks. Siis on tõepärafunktsioon Poissoni jaotusega mudeli jaoks vastavalt

$$L_x(a,b_1,b_2) = \prod_{i \in J(x)} \frac{((a+b_1x_{1,i}+b_2x_{2,i})t_i)^{n_i}}{n_i!} e^{-(a+b_1x_{1,i}+b_2x_{2,i})t_i}.$$

Selle abil leitud log-tõepära avaldub

$$l_x(a,b_1,b_2) = \sum_{i\in J(x)} n_i \ln((a+b_1x_{1,i}+b_2x_{2,i})t_i) - \\ - \sum_{i\in J(x)} n_i! - \sum_{i\in J(x)} (a+b_1x_{1,i}+b_2x_{2,i})t_i.$$

Suurima tõepära hinnangu leidmiseks leiame nüüd tuletised iga parameetri järgi:

$$\begin{aligned}
\frac{\partial l_x(a,b_1,b_2)}{\partial a} &= \sum_{i\in J(x)} \frac{n_i}{(a+b_1x_{1,i}+b_2x_{2,i})} - \sum_{i\in J(x)} t_i, \\
\frac{\partial l_x(a,b_1,b_2)}{\partial b_1} &= \sum_{i\in J(x)} \frac{n_ix_{1,i}}{(a+b_1x_{1,i}+b_2x_{2,i})} - \sum_{i\in J(x)} t_ix_{1,i}, \\
\frac{\partial l_x(a,b_1,b_2)}{\partial b_2} &= \sum_{i\in J(x)} \frac{n_ix_{2,i}}{(a+b_1x_{1,i}+b_2x_{2,i})} - \sum_{i\in J(x)} t_ix_{2,i}.
\end{aligned}$$

Nende abil leiame võrrandid suurima tõepära hinnangute leidmiseks

$$\begin{cases}
\sum_{i\in J(x)} \frac{n_i}{(a+b_1x_{1,i}+b_2x_{2,i})} = \sum_{i\in J(x)} t_i, \\
\sum_{i\in J(x)} \frac{n_ix_{1,i}}{(a+b_1x_{1,i}+b_2x_{2,i})} = \sum_{i\in J(x)} t_ix_{1,i}, \\
\sum_{i\in J(x)} \frac{n_ix_{2,i}}{(a+b_1x_{1,i}+b_2x_{2,i})} = \sum_{i\in J(x)} t_ix_{2,i}.
\end{cases} \tag{4.2}$$

Kahe regressori korral võib korraga arvesse võtta nii omaniku kui auto vanust.

Samal viisil on võimalik jätkata hinnangute leidmist. Leiame vajalikud võrrandid m regressori korral. Olgu $J(x_1,\ldots,x_m)$ nende poliiside indeksite hulk, mille regressorite väärtused langevad $(x_1,\ldots,x_m)$ ümbrusesse, $a,b_1,b_2,\ldots,b_m$ regressiooniparameetrid ning $x_{1,i},\ldots,x_{m,i}$ regressori väärtused poliisi i jaoks. Siis on tõepärafunktsioon Poissoni jaotusega mudeli jaoks vastavalt

$$L_x(a,b_1,b_2,\ldots,b_m) = \prod_{i\in J(x_1,\ldots,x_m)} \frac{((a+\sum_{q=1}^m b_qx_{q,i})t_i)^{n_i}}{n_i!} e^{-(a+\sum_{q=1}^m b_qx_{q,i})t_i}.$$

Selle abil leiame log-tõepära funktsiooni

$$l_x(a, b_1, b_2, \ldots, b_m) = \sum_{i \in J(x_1,\ldots,x_m)} n_i \ln((a + \sum_{q=1}^{m} b_q x_{q,i})t_i) -$$
$$- \sum_{i \in J(x_1,\ldots,x_m)} n_i! - \sum_{i \in J(x_1,\ldots,x_m)} (a + \sum_{q=1}^{m} b_q x_{q,i})t_i.$$

Suurima tõepära hinnangute leidmiseks leiame nüüd tuletised kõigi parameetrite järgi:

$$\begin{aligned}
\frac{\partial l_x(a, b_1, b_2, \ldots, b_m)}{\partial a} &= \sum_{i \in J(x_1,\ldots,x_m)} \frac{n_i}{(a + \sum_{q=1}^{m} b_q x_{q,i})} - \sum_{i \in J(x_1,\ldots,x_m)} t_i, \\
\frac{\partial l_x(a, b_1, b_2, \ldots, b_m)}{\partial b_1} &= \sum_{i \in J(x_1,\ldots,x_m)} \frac{n_i x_{1,i}}{(a + \sum_{q=1}^{m} b_q x_{q,i})} - \sum_{i \in J(x_1,\ldots,x_m)} t_i x_{1,i}, \\
&\vdots \\
\frac{\partial l_x(a, b_1, b_2, \ldots, b_m)}{\partial b_m} &= \sum_{i \in J(x_1,\ldots,x_m)} \frac{n_i x_{m,i}}{(a + \sum_{q=1}^{m} b_q x_{q,i})} - \sum_{i \in J(x_1,\ldots,x_m)} t_i x_{m,i}.
\end{aligned}$$

Nende abil saame võrrandisüsteemi suurima tõepära hinnangute leidmiseks

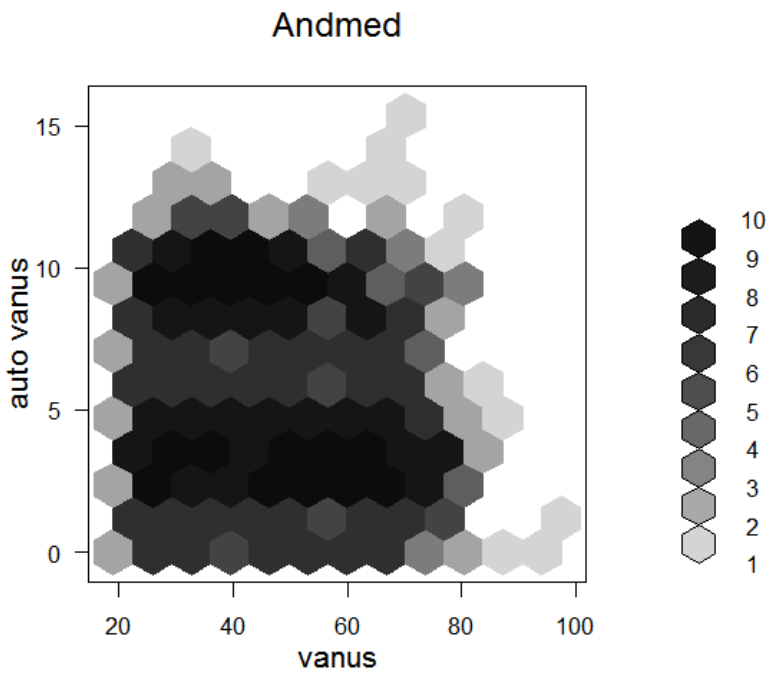
$$\begin{cases}
\sum_{i \in J(x_1,\ldots,x_m)} \frac{n_i}{(a + \sum_{q=1}^{m} b_q x_{q,i})} = \sum_{i \in J(x_1,\ldots,x_m)} t_i, \\
\sum_{i \in J(x_1,\ldots,x_m)} \frac{n_i x_{1,i}}{(a + \sum_{q=1}^{m} b_q x_{q,i})} = \sum_{i \in J(x_1,\ldots,x_m)} t_i x_{1,i}, \\
\quad \vdots \\
\sum_{i \in J(x_1,\ldots,x_m)} \frac{n_i x_{m,i}}{(a + \sum_{q=1}^{m} b_q x_{q,i})} = \sum_{i \in J(x_1,\ldots,x_m)} t_i x_{m,i}.
\end{cases}$$

Peatükk 5

Meetodi rakendamine reaalsetele andmetele

5.1 Andmete kirjeldus

Eelnevates peatükkides kirjeldatud meetodite rakendamiseks kasutati ühe Eesti kindlustusfirma kaskokindlustuse andmeid. Vaadeldud oli erinevaid poliise, mille algus- ja lõppkuupäevad jäid 7 aasta vahemikku 2007-2014 aastal. Neis poliisides on andmed erinevate riskide kohta alates klaasikahjudest kuni vargusohuni. Poliiside kestvused on erinevad ning lisaks oli kindlustusvõtjal olemas võimalus poliis varem lõpetada. Iga poliisi kohta on kehtivuse lõppkuupäevale lisaks teada reaalne lõpetamise kuupäev. Auto kohta olid teada mitmed olulised tunnused - vanus, mark, mudel, tüüp, esialgne väärtus, hetkeväärtus, valmistamise aasta jms. Ka olid olemas andmed omaniku kohta - vanus, sünniaasta, sugu ja eelnevalt esinenud kahjujuhtumite arv. Poliisi sõlmijate vanused jäid vahemikku 19-94 ja autode vanused 0-15 aastat.



5.2 Mudelite headuse mõõt

Selleks, et erinevate k väärtuste ja kauguse definitsioonide kombinatsioonidega loodud mudelite tulemusi võrrelda tuleb anda teatud headuse mõõt. Sageli kasutatakse selleks standardviga või AIC-kriteeriumit. Käesolevas töös kasutatakse järgnevalt defineeritud vea mõõtu

$$e = \frac{1}{365} \sum_{i=1}^{n} t_i (n_i - \hat{n}_i)^2,$$

kus

- n – poliiside hulk testandmetes,
- t_i – poliisi i kestvus päevades,
- n_i – tegelik kahjude hulk poliisis i,

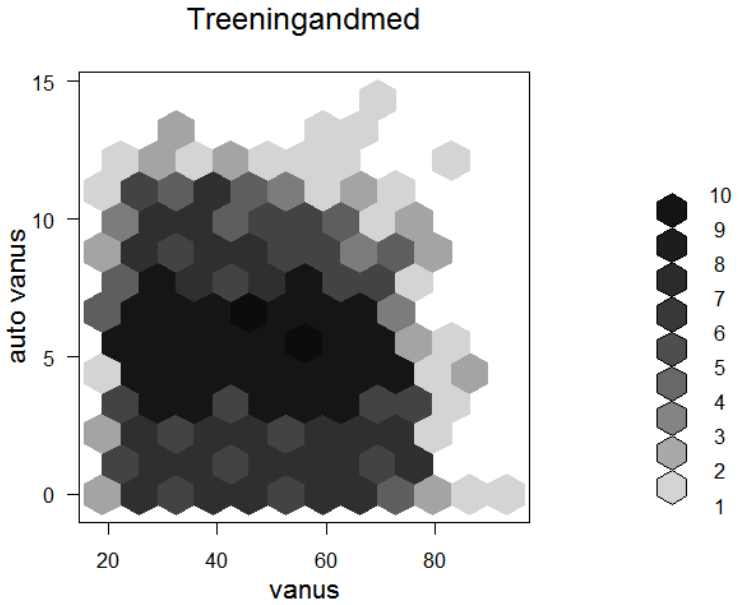
- $\hat{n}_i$ – prognoositud kahjude hulk poliisis i. [5]

Sellise vea mõõdu kasutamine võtab arvesse, et poliisi kestvuse kohta pole tehtud ühtegi eeldust ning testandmetes on erinevate poliisiide kestvused erinevad.

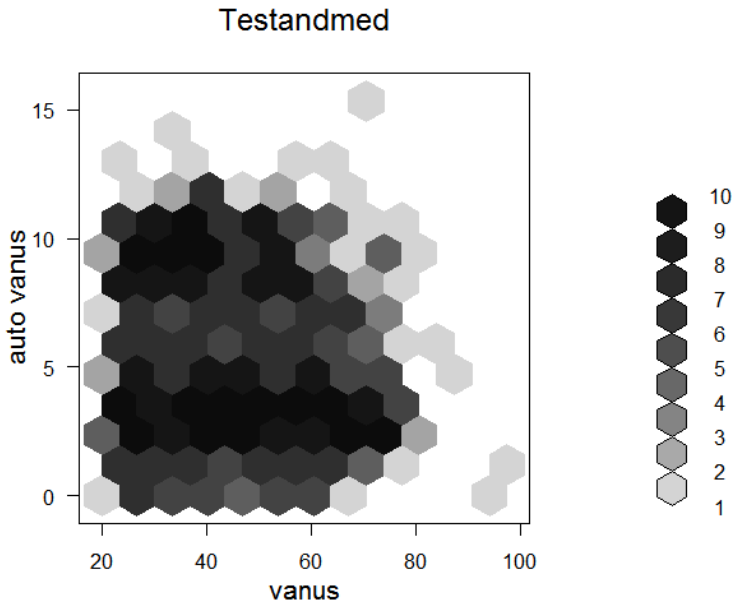
Eesmärgiks oleks kindlaks teha, millise kauguse ja k suuruse korral saame parimad tulemused. Lisaks toome sisse juhu, kus $k \to n$ ehk nn globaalse regressiooni, et hinnata, kas lokaalne regressioon selles olukorras annab paremad tulemused või pole lokaalsuse kasutamisel mõju mudeli headusele ning selle sammu võiks üldse kõrvale jätta. Lisaks võrdleme saadud tulemusi CART-meetodil leitud veaga, et näha, kas dünaamilisemad klassipiirid aitavad saada täpsema hinnangu või mitte.

5.3 Ülesande püstitus

Andmestiku jagame kahte ossa: testandmed ja treeningandmed.



Treeningandmeid kasutatakse mudeli koostamiseks ning testandmeid mudeli headuse testimiseks. Treeningandmeteks olid poliisid, mille alguskuupäev oli ajavahemikus 2007. aasta jaanuarist kuni 2009. aasta juunini. Testandmeteks olid need poliisid, mille alguskuupäevad jäid 2009. aasta juunist 2010. aasta juunini. Sellise jaotuse kasuks räägib asjaolu, et ka reaalselt soovitakse kindlustuses ajaliselt vanemate andmete abil prognoosida uuemaid. Treeningandmetes oli vaatluseid 15 745 ning testandmetes 9542. Nendest oli vastavalt 7569 ja 5029 vaatlust sellised, mille kohta oli teada nii auto kui inimese vanus ning sõiduki hetkeväärtus.



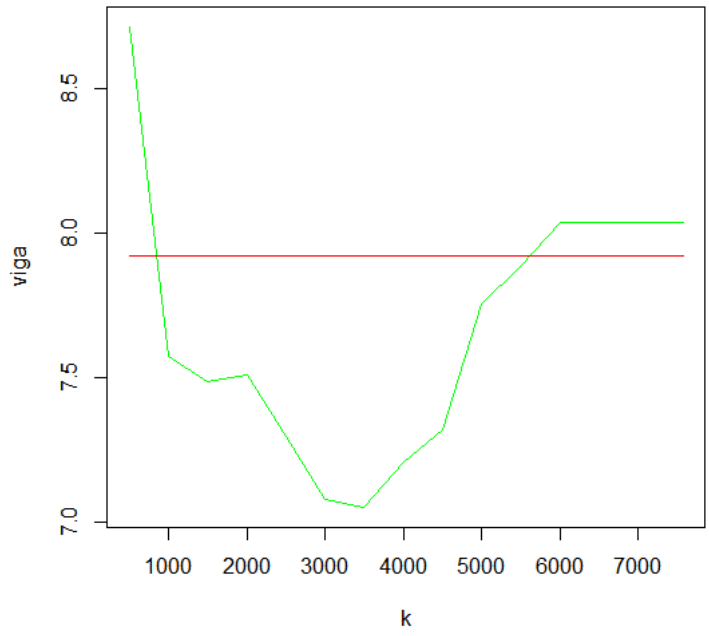
Analüüsi käigus proovitakse mitmeid erinevaid kombinatsioone kauguse valikust ja k suurusest k-lähima naabri meetodi rakendamisel. Lisaks tehakse ühe ja kahe regressoriga lokaalse regressiooni mudelid. K valimisel kasutatakse esialgu $k = 500$ ning suurendatakse seda iga sammuga 500 võrra. Leidsime iga k väärtuse jaoks vea suuruse nii eukleidilise kui Mahalanobise kaugusega. Mahalanobise kaugusest kasutati lihtsustatud versiooni, kus kovariatsioonimaatriks arvutati üks kord kogu andmestiku pealt, selle asemel, et seda tsük-

lisse lisada. Selline samm vähendab tunduvalt arvutusele kuluva aja mahtu. Puuduvate väärtusega poliisid jäetakse kõrvale, kuna valimi maht on ka peale nende eemaldamist piisavalt suur.

5.4 Tulemused

5.4.1 Ühe regressoriga lokaalne regressioon

Esmalt kasutame hindamiseks lokaalset regressiooni ühe regressoriga, mille korral leiame hinnangud vastavalt valemile (4.1). Regressoriks võtame omaniku vanuse.



Joonis 5.1: Vea e suurused, kui regressoriks on omaniku vanus.

Punasega on joonisele kantud CART-meetodi viga väärtusega 7.92. Rohelise joonega on toodud eukleidilise kaugusega leitud mudeli vead. Ühemõõtme-

lisel juhul on Mahalanobise kauguse arvutamisel kovariatsioonimaatriksiks vanuse dispersioon ja sisuliselt on tegu skaleeritud eukleidilise kaugusega. Tulemused tulevad mõlema kauguse korral samad, sest ühemõõtmelisel juhul ei muuda skaleerimine punktide valikut.

k	Eukleidiline kaugus
500	8.715839
1000	7.576187
1500	7.483801
2000	7.509538
2500	7.295515
3000	7.076690
3500	**7.049603**
4000	7.205904
4500	7.318590
5000	7.756306
5500	7.886840
6000	8.038482
6500	8.038482
7000	8.038482

Tabel 5.1: Vea e väärtused, kui regressoriks on omaniku vanus.

Näeme, et alates $k = 6000$ jõuame olukorrani, kus $k \to n$ ja saame globaalse regressiooni, mille viga on 8.04.

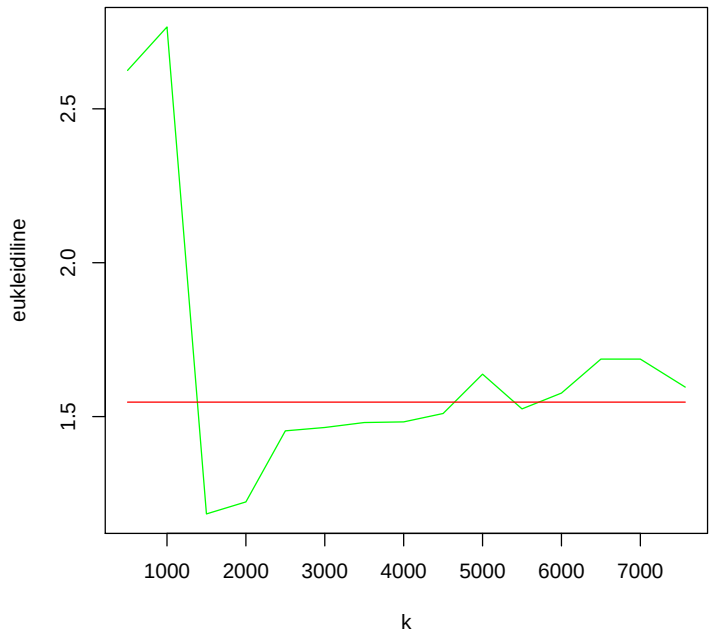
Võrreldes saadud tulemusi CART-meetodil leitud veaga 7.92, näeme, et lokaalne regressioon annab peaagu kõikide k väärtuste korral parema tulemuse. Globaalne regressioon ei anna CART-meetodist paremat tulemust. Parima tulemuse saime eukleidilise kaugusega kui $k = 3500$, kus $e = 7.05$.

Järgmisena kasutame regressorina auto vanust. Tulemusi näeme tabelist 5.2.

k	Eukleidiline kaugus
500	2.624786
1000	2.766021
1500	1.183770
2000	**1.222773**
2500	1.453705
3000	1.464608
3500	1.480690
4000	1.482796
4500	1.510074
5000	1.637857
5500	1.525042
6000	1.576388
6500	1.686811
7000	1.686811

Tabel 5.2: Vea e väärtused, kui regressoriks on auto vanus.

Võrreldes saadud tulemusi CART-meetodil leitud veaga 1.55 näeme, et lokaalne regressioon annab parema tulemuse $k = 1500$ kuni $k = 4500$ ja $k = 5500$ korral. Näeme, et alates $k = 6500$ jõuame olukorrani, kus $k \to n$ ja saame globaalse regressiooni, mille viga on 1.69. Globaalne regressioon ei anna CART-meetodist paremat tulemust. Parima tulemuse saime eukleidilise kaugusega kui $k = 2000$, kus $e = 1.22$.



Joonis 5.2: Vea e suurused, kui regressoriks on omaniku vanus.

Joonisele 5.2 on punasega toodud CART-meetodi viga väärtusega 1.55 ja rohelisega on toodud eukleidilise kaugusega leitud mudeli vead.

Dünaamilisemate klassipiiride eelis CART-meetodi ees sõltus regressori valikust, kuid ühemõõtmelisel juhul annab lokaalne regressioon enamasti parema tulemuse. Lisaks seisneb dünaamiliste klassipiiride eelis ka selles, et vähendatakse hinnašoki tekkimise võimalust ja preemiate muutumine on sujuvam.

	e	Regressor
CART-meetod	7.92	omaniku vanus
Lokaalne regressioon $k = 1500$	7.48	
Lokaalne regressioon $k = 2500$	7.30	
Lokaalne regressioon $k = 3500$	**7.05**	
Globaalne regressioon	8.04	
CART-meetod	1.55	auto vanus
Lokaalne regressioon $k = 1000$	2.77	
Lokaalne regressioon $k = 2000$	**1.22**	
Lokaalne regressioon $k = 3000$	1.46	
Globaalne regressioon	1.69	

Tabel 5.3: Erinevate meetodite vead ühe sisendtunnuse korral.

5.4.2 Kahe regressoriga lokaalne regressioon

Kahe regressoriga lokaalse regressiooni korral kasutame hindamiseks eelmises peatükis leitud võrrandeid (4.2). Võtame regressoriteks omaniku ja auto vanused. Tabelis 5.4 näeme saadud tulemusi.

Mahalanobise kauguse korral on kasutatud kovariatsioonimaatriksit, kus age tähistab omaniku vanust ja $v.age$ auto vanust.

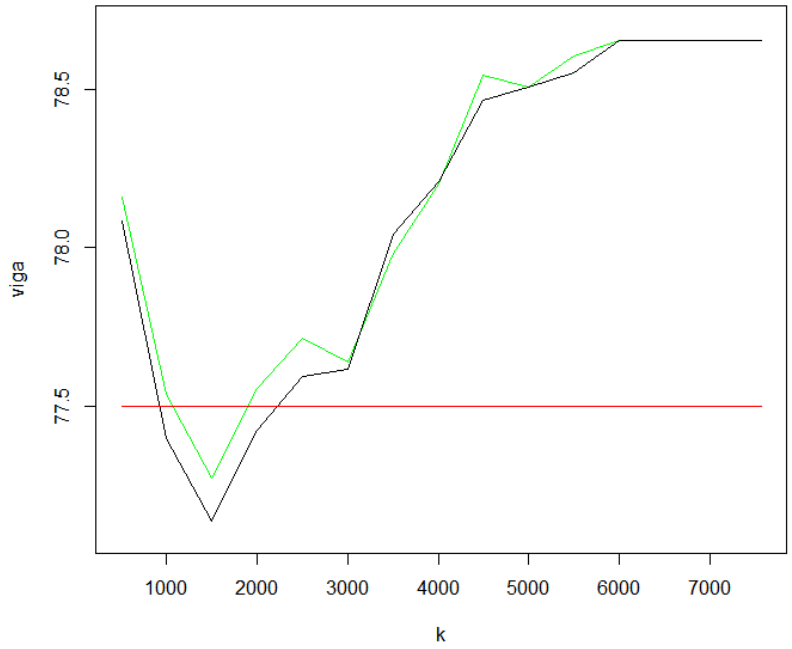
$$C = cov(age, v.age) = \begin{pmatrix} 151.62321 & -3.980510 \\ -3.98051 & 7.568443 \end{pmatrix}$$

Kahemõõtmelisel juhul on näha erinevust eukleidilise ja Mahalanobise kauguse vahel. Näeme, et Mahalanobise kaugus annab suurema osa k väärtuste korral parema tulemuse, vaid $k = 3500$ ja $k = 4000$ korral jääb ta eukleidilisele alla. Alates $k = 6000$ jõutakse olukorrani, kus $k \to n$ ja saame globaalse regressiooni, mille korral on veaks 78.65. Võrdluseks võtame CART-meetodil

k	Eukleidiline kaugus	Mahalanobise kaugus
500	78.161971	78.084263
1000	77.536584	77.396354
1500	**77.271121**	**77.136557**
2000	77.557201	77.422963
2500	77.713830	77.593561
3000	77.640166	77.616465
3500	77.983807	78.044727
4000	78.200291	78.211248
4500	78.546014	78.464993
5000	78.507941	78.506341
5500	78.606995	78.551386
6000	78.654487	78.654487
6500	78.654487	78.654487
7000	78.654487	78.654487

Tabel 5.4: Vea e väärtused kahe regressoriga lokaalse regressiooni korral.

saadud tulemuse, mille viga on 77.5. Eukleidilise kauguse korral saadakse sellest parem tulemus vaid $k = 1500$ korral, mis on ka eukleidilise kauguse parim tulemus 77.27. Mahalanobise kaugusega saadakse CART-meetodist parem tulemus alates $k = 1000$ kuni $k = 2000$. Ülejäänud k-väärtuste korral on mõlema kaugusega leitud lokaalse regressiooni hinnangute vead suuremad kui CART-meetodil leitud viga. Ka globaalse regressiooni viga 78.65 on suurem kui CART-meetodil leitud. Parima tulemuse kahe regressoriga lokaalse regressiooni korral saime Mahalanobise kaugusega $k = 1500$, kus $e = 77.14$.



Joonis 5.3: Vea e suurused kahe regressoriga lokaalse regressiooni korral.

Punasega on joonisele kantud CART-meetodi viga väärtusega 77.5. Rohelise joonega on toodud eukleidilise kaugusega leitud mudeli vead ja musta joonega Mahalanobise kaugusega leitud mudeli vead. On näha, et kahemõõtmelisel juhul annab Mahalanobise kaugus väiksema vea, sest arvesse võetakse tunnustevahelist kovariatsiooni, mis annab eelise eukleidilise kauguse ees. Vigade graafikud on sarnase kujuga, sest korrelatsioonimaatriksist

$$R = cor(age, v.age) = \begin{pmatrix} 1 & -0.117504 \\ -0.117504 & 1 \end{pmatrix},$$

kus age on omaniku vanus ja $v.age$ auto vanus, näeme, et tunnuste vahel on negatiivne ning üpris väike korrelatsioon. Seetõttu annavad Mahalanobise ja eukleidiline kaugus sarnased tulemused.

Kahemõõtmelisel juhul on lokaalse regressiooniga võimalik saavutada eelis

CART-meetodi ees, kuid tähelepanu tuleb pöörata k valikule, et saavutada väiksem viga. Samas võimaldavad dünaamilised klassipiirid, mis määratakse lokaalse regressiooniga, vähendada hinnašoki tekkimise võimalust, mis on kindlustuspoliisi sõlmija jaoks kindlasti eelistatud.

Eukleidiline kaugus	e
Lokaalne regressioon $k = 1500$	77.27
Lokaalne regressioon $k = 2500$	77.71
Lokaalne regressioon $k = 3500$	77.98
Mahalanobise kaugus	
Lokaalne regressioon $k = 1500$	**77.14**
Lokaalne regressioon $k = 2500$	77.59
Lokaalne regressioon $k = 3500$	78.04
Globaalne regressioon	78.65
CART-meetod	77.5

Tabel 5.5: Erinevate meetodite vead kahe sisendtunnuse korral.

Kokkuvõte

Töö eesmärgiks oli uurida, kas lokaalse regressiooni ja dünaamiliste klassipiiride kasutamine annab täpsemaid tulemusi kindlustuskahjude sageduse hindamisel, kui CART-meetod. Selleks tutvustasime töö esimeses osas k-lähima naabri meetodit. Tegu on mitteparameetrilise meetodiga, mida saab kasutada nii klassifitseerimiseks kui regressiooniks. Antud töös kasutasime seda koos lokaalse regressiooniga. Sageli on meetodi puhul elementaarseks valikuks eukleidiline kaugus, kuid antud töös uurisime ka Mahalanobise kauguse kasutamise võimalusi ning omadusi.

Lisaks erinevatele kauguse defineerimise võimalustele on lokaalse regressiooni korral mitu võimalust regressorite valikuks. Nii leidsimegi kahjude esinemise sageduse hinnangute valemid nii ühe, kahe kui ka m regressoriga juhul. Eesti kaskokindlustuse andmetel proovisime ühe ja kahe regressoriga hindamist, kus esimesel juhul oli regressoriks omaniku või auto vanus ning kahemõõtmelisel juhul võtsime arvesse mõlemad. Ühemõõtmelisel juhul on eukleidilise ja Mahalanobise kaugusega leitud tulemused võrdsed, sest kovariatsioonimaatriks ühemõõtmelisel juhul on vaadeldavate punktide jaotuse dispersioon. Kahemõõtmelisel juhul olid tulemused erinevad ning suurema osa k väärtuste korral saime Mahalanobise kaugusega paremad tulemused.

Ühemõõtmelisel juhul on lokaalse regressiooniga leitud mudeli vea suurus suurema osa k väärtuste korral väiksem kui CART-meetodil leitud viga, kuid kahemõõtmelisel juhul on vaid paari k väärtuse korral võimalik saada pa-

rem viga. Seega vaadeldud andmete korral on ühemõõtmelisel juhul lokaalse regressiooniga hindamisel ehk dünaamiliste klassipiiride kasutamisel nähtav eelis CART-meetodi ees, kuid kahemõõtmelisel juhul annab see paremad tulemused vaid väikses k-väärtuste piirkonnas. Samas vähendavad dünaamilised piirid hinnašoki esinemise võimalust ning kindlustusmaksete muutus on sujuvam.

Antud teemal on võimalus uurimist jätkata suurema arvu regressorite hulgaga mudelite analüüsimisel, mille kohta antud töös küll leiti hindamiseks vajalikud võrrandid kuid praktiliste andmete peal analüüsini ei jõutud.

Kirjandus

Cunningham, P; Delany, S. J (2007) „k-Nearest Neighbour Classifiers", *Technical Report UCD-CSI-2007-4*

Fix, E; Hodges, J.L (1989) „Discriminatory Analysis. Nonparametric Discrimination: Consistency Properties“, *International Statistical Reviw*, 57 (3), 238-247

Gnanadesikan, R; Kettenring, J.R (1972) „Robust Estimates, Residuals, and Outlier Detection with Multiresponse Data", *Biometrics*, 28 (1), 81-124

Hastie, T; Tibshirani, R; Friedman, J (2008) „The elements of statistical learning: data mining, inference and prediction", *Springer*, 14-18

Käärik, M; Kaasik, A (2012) „On premium estimation using the C&RT/Poisson model and its extensions", *Lithuanian Journal of Statistics*, 51 (1), 36-50

Maesschalck, R. De; Jouan-Rimbaud, D; Massart, D. L (2000) "Tutorial - The Mahalanobis distance", *Chemometrics and Intelligent Laboratory Systems*, 50, 1-18

Pärna, K; Kangro, R; Kaasik, A; Möls, M (2012) „K-Nearest Neighbors as Pricing Tool in Insurance: a Comparative Study", *Multivariate Statistics: Theory and Applications*, 130-131

Wilson, D.R; Martinez, T.R (1997) „Improved Heterogeneous Distance Functions", *Journal of Artificial intelligence Research*, 6, 1-34

Xiaoyu, S; Jingke, X; Zhichao, Y; Huanliang, S (2014) „RkNN Query Algorithm Based on K-order Voronoi Diagram", *International Journal of Control and Automation*, 7 (9), 11-26

Lisa: Kasutatud R-i kood

```
# ühe regressoriga lokaalne regressioon


# age - omaniku vanus
# v_age - sõiduki vanus
# claimfreq - kahjusagedus
# days - kindlustuspäevade arv
# a, b1, b2 - regressiooniparameetrid

#Võrrandid toodud juhul, kui regressoriks omaniku vanus
#Auto vanusele üleminekuks asendada väärtused
#age väärtusega v_age
fnL_1 = function (a,b1,age,claimfreq,days) {-
sum(claimfreq*log(a+b1*age))+sum(days*(a+b1*age))
}

#võrrandid STH leidmiseks
fnLgrad_1 = function(a,b1,age,claimfreq,days){
c(-sum(claimfreq/(a+b1*age))+sum(days),
-sum(claimfreq*age/(a+b1*age))+sum(days*age))
}

#vastavad võrrandid konkreetsete andmete korral
```

```
fnL_kon_1 = function(x){
a=x[1];
b1=x[2];
fnL_1(a=x[1],b1=x[2],age=age_sample,
claimfreq=claimfreq_sample,days=days_sample)
}

fnLgrad_kon_1 = function(x){
a=x[1];
b1=x[2];
fnLgrad_1(a=x[1],b1=x[2],age=age_sample,
claimfreq=claimfreq_sample,days=days_sample)
}

############################################################
# andmestiku sisselugemine ja mõned abiteisendused

...

# treeningandmed

data = sqldf("select inimvanus as age,
    sum(kahjudearv) as freq,
    sum(poliis_kehtinud) as days,
    sum(kahjudearv)/sum(poliis_kehtinud) as lambda_d,
    365*sum(kahjudearv)/sum(poliis_kehtinud) as lambda_y
  from treeningpoliisid
  where inimvanus != 'NA'
  group by inimvanus")

#auto vanus regressoriks
```

```
#data = sqldf("select autovanus as v_age,
#    sum(kahjudearv) as freq,
#    sum(poliis_kehtinud) as days,
#    sum(kahjudearv)/sum(poliis_kehtinud) as lambda_d,
#    365*sum(kahjudearv)/sum(poliis_kehtinud) as lambda_y
#  from treeningpoliisid
#  where autovanus != 'NA'
#  group by autovanus")


################################################
#keskväärtused ja dispersioonid
age_mean = sum(data$age*data$days)/
sum(data$days); age_mean
age_var = 1/(sum(data$days)-1)*
sum(data$days*(data$age-age_mean)^2); age_var
#v_age_mean = sum(data$v_age*data$days)/sum(data$days);
#v_age_mean
#v_age_var = 1/(sum(data$days)-1)*sum(data$days*
#(data$v_age-v_age_mean)^2);v_age_var
################################################

# testandmed

test_data = sqldf("select inimvanus as age,
    sum(kahjudearv) as freq,
    sum(poliis_kehtinud) as days,
    sum(kahjudearv)/sum(poliis_kehtinud) as lambda_d,
    365*sum(kahjudearv)/sum(poliis_kehtinud) as lambda_y
  from testpoliisid
  where inimvanus != 'NA'
```

```
  group by inimvanus")

#auto vanus regressoriks
#test_data = sqldf("select autovanus as v_age,
#    sum(kahjudearv) as freq,
#    sum(poliis_kehtinud) as days,
#    sum(kahjudearv)/sum(poliis_kehtinud) as lambda_d,
#    365*sum(kahjudearv)/sum(poliis_kehtinud) as lambda_y
#  from testpoliisid
#  where autovanus != 'NA'
#  group by autovanus")

##############################################################################
data2age=data.frame(age=c(min(data$age):max(data$age)))
data2 = sqldf("select a.age, c.freq, c.days, c.lambda_y
           from data2age as a
           left join data as c on a.age=c.age")

#auto vanus regressoriks
#data2age=data.frame(age=c(min(data$v_age):max(data$v_age)))
#data2 = sqldf("select a.v_age, c.freq, c.days, c.lambda_y
#           from data2v_age as a
#           left join data as c on a.v_age=c.v_age")

#erinevad k-väärtused, suurendatakse sammuga 500
k=c(500,1000,1500,2000,2500,3000,3500,4000
,4500,5000,5500,6000,6500,7000)
nimekiri=rep(NA,14)

for (m in (1:length(k))){
## naabruse leidmise funktsioon
```

```
datawindow =function(age,i1){
window = data2[round(abs(data2$age-age))<=i1 &
 !is.na(data2$days),] #eukleidiline
window
}

#########################################################################

for (j in (1:length(data2$age))) {
  age=data2$age[j]
  i = 0.01
  neighb_size = sum(datawindow(age,i)$days)
  while ((neighb_size < 365*k[m]) &
  ((age-i>min(data2$age))|(age+i<max(data2$age)))) {
    i = i + 0.1
    neighb_size = sum(datawindow(age,i)$days)
  }
  age_sample = datawindow(age,i)$age
  claimfreq_sample = datawindow(age,i)$freq
  days_sample = datawindow(age,i)$days/365

  fnL2_1 = function(x){
         a=x[1];
         b1=x[2];
         fnL_1(a=x[1],b1=x[2],age=age_sample
         ,claimfreq=claimfreq_sample,days=days_sample)
   }

   fnLgrad2_1 = function(x){
    a=x[1];
```

```
    b1=x[2];
    fnLgrad_1(a=x[1],b1=x[2],age=age_sample
    ,claimfreq=claimfreq_sample,days=days_sample)
   }

   params=optim(par=c(1,0),fn=fnL2_1,
   gr=fnLgrad2_1,method="BFGS")
   data2$a2[j] = round(params$par[1],digits=8)
   data2$b12[j] = round(params$par[2],digits=8)
   data2$neighb2[j] = neighb_size
   if (params$convergence != 0) { cat("j=",j,",
   age=",age,", neighb=",neighb_size," NOT CONVERGED")}
}

data2$lambda_new2 = round(data2$a2+data2$b12*data2$age
, digits=8)

test_data2 = sqldf("select a.age, a.freq, a.days,
    b.lambda_y, b.lambda_new2,
    a.lambda_y as lambda_actual
  from test_data as a left join
  data2 as b on a.age = b.age")
#####################################################################

e2_semipar2 = sum(test_data2$days*
(test_data2$lambda_actual-test_data2$lambda_new2)
^2/365,na.rm=TRUE);e2_semipar2 #leiame mudeli vea

nimekiri[m]=e2_semipar2
print(paste0("Tulemus on: ", e2_semipar2))
}
```

```
#####################################################################
#####################################################################
#####################################################################
#Kahe regressoriga lokaalne regressioon

fnL_2 = function (a,b1,b2,age,v_age,claimfreq,days) {
-sum(claimfreq*log(a+b1*age+b2*v_age))+
sum(days*(a+b1*age+b2*v_age))
}

#Võrrandid STH leidmiseks
fnLgrad_2 = function(a,b1,b2,age,v_age,claimfreq,days){
c(-sum(claimfreq/(a+b1*age+b2*v_age))+sum(days),
-sum(claimfreq*age/(a+b1*age+b2*v_age))+sum(days*age),
-sum(claimfreq*v_age/(a+b1*age+b2*v_age))+sum(days*v_age))
}

#Treeningandmed

data = sqldf("select inimvanus as age,
     autovanus as v_age,
     sum(kahjudearv) as freq,
     sum(poliis_kehtinud) as days,
     sum(kahjudearv)/sum(poliis_kehtinud)
     as lambda_d,
     365*sum(kahjudearv)/sum(poliis_kehtinud)
     as lambda_y
   from treeningpoliisid
   where inimvanus != 'NA' and autovanus != 'NA'
   group by inimvanus,autovanus")
```

```
#keskväärtused ja dispersioonid
age_mean = sum(data$age*data$days)/sum(data$days);
 age_mean
age_var = 1/(sum(data$days)-1)*sum(data$days*
(data$age-age_mean)^2); age_var
v_age_mean = sum(data$v_age*data$days)/sum(data$days);
v_age_mean
v_age_var = 1/(sum(data$days)-1)*sum(data$days*
(data$v_age-v_age_mean)^2);v_age_var
cov_2=cov.wt(data.frame(data$age,data$v_age),
data$days/sum(data$days),cor=TRUE)$cov
##############################################################################

#Testandmed

test_data = sqldf("select inimvanus as age,
    autovanus as v_age,
    sum(kahjudearv) as freq,
    sum(poliis_kehtinud) as days,
    sum(kahjudearv)/sum(poliis_kehtinud) as lambda_d,
    365*sum(kahjudearv)/sum(poliis_kehtinud) as lambda_y
  from testpoliisid
  where inimvanus != 'NA' and autovanus !='NA'
  group by inimvanus,autovanus")

##############################################################################

data2age = data.frame(age=c(min(data$age):max(data$age)))
data2vage = data.frame(v_age=c(min(data$v_age):max(data$v_age)))
```

```
data2 = sqldf("select a.age, b.v_age, c.freq, c.days, c.lambda_y
  from data2age as a
  left join data2vage as b
  left join data as c on a.age=c.age and b.v_age = c.v_age")

#erinevad k-väärtused, suurendatakse sammuga 500
k=c(500,1000,1500,2000,2500,3000,3500,4000,4500
,5000,5500,6000,6500,7000)
mahalanobis=rep(NA,14)

for (m in (1:length(k))){

## naabruse leidmise funktsioon (datawindow)

datawindow =function(age,v_age,i1){
#dispersioonidega korrigeeritud eukleidiline
#data2$kaugus=(data2$age-age)^2/(age_var*(i1)^2)+
(data2$v_age-v_age)^2/(v_age_var*(i1)^2)
#window=data2[data2$kaugus<1 & !is.na(data2$days),]

#mahalanobis
data2$kaugus=sqrt(mahalanobis(cbind(data2$age,
data2$v_age),cbind(age,v_age),cov_2))
window=data2[data2$kaugus<i1 & !is.na(data2$days),]

window
}


#####################################################################
for (j in (1:length(data2$age))) {
```

```
age=data2$age[j]
v_age=data2$v_age[j]
i = 0.01
neighb_size = sum(datawindow(age,v_age,i)$days)
while ((neighb_size < 365*k[m]) &
((age-i>min(data2$age))|(age+i<max(data2$age)))) {
  i = i + 0.1
  neighb_size = sum(datawindow(age,v_age,i)$days)
}
age_sample = datawindow(age,v_age,i)$age
v_age_sample = datawindow(age,v_age,i)$v_age
claimfreq_sample = datawindow(age,v_age,i)$freq
days_sample = datawindow(age,v_age,i)$days/365

 fnL2= function(x){
   a=x[1];
   b1=x[2];
   b2=x[3];
   fnL_2(a=x[1],b1=x[2],b2=x[3],age=age_sample
   ,v_age=v_age_sample,claimfreq=claimfreq_sample
   ,days=days_sample)
 }

 fnLgrad2 = function(x){
   a=x[1];
   b1=x[2];
   b2=x[3];
   fnLgrad_2(a=x[1],b1=x[2],b2=x[3],age=age_sample
   ,v_age=v_age_sample,claimfreq=claimfreq_sample
   ,days=days_sample)
 }
```

```
    params=optim(par=c(1,0,0),fn=fnL2,gr=fnLgrad2,method="BFGS")
    data2$a2[j] = round(params$par[1],digits=8)
    data2$b12[j] = round(params$par[2],digits=8)
    data2$b22[j] = round(params$par[3],digits=8)
    data2$neighb2[j] = neighb_size
    if (params$convergence != 0) { cat("j=",j,", age=",age,",
    v_age=",v_age,", neighb=",neighb_size," NOT CONVERGED")}
}

data2$lambda_new2 = round(data2$a2+data2$b12*data2$age+
data2$b22*data2$v_age, digits=8)

test_data2 = sqldf("select a.age, a.v_age, a.freq, a.days,
    b.lambda_y, b.lambda_new2, a.lambda_y as lambda_actual
  from test_data as a left join data2 as b
  on a.age = b.age and a.v_age = b.v_age")
##################################################################

#leiame mudeli vea
e2_semipar2 = sum(test_data2$days*(test_data2$lambda_actual-
test_data2$lambda_new2)^2/365,na.rm=TRUE);e2_semipar2
mahalanobis[m]=e2_semipar2
print(paste0("Tulemus on: ", e2_semipar2))
}
```

Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks

Mina, Liina Muru (sünnikuupäev: 23.02.1991)

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) enda loodud teose

   „Kindlustuskahjude sageduse analüüs lokaalse regressiooni ja k-lähima naabri meetodil“,

   mille juhendaja on Meelis Käärik,

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

   (b) 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, 13.05.2015