Tartu Ülikool
Loodus- ja täppisteaduste valdkond
Matemaatika ja statistika instituut

Triin Ree

Krediidibüroosse eraisikute kohta tehtud päringute informatsiooni kasutamine panga krediidiriski mudelis

Matemaatilise statistika erialal

Magistritöö (30 EAP)

Juhendajad: Raul Kangro (PhD)
Karl Märka (Creditinfo)

Tartu 2019

Krediidibüroosse eraisikute kohta tehtud päringute informatsiooni kasutamine panga krediidiriski mudelis

Magistritöö

Triin Ree

Lühikokkuvõte. Käesoleva magistritöö eesmärgiks on Creditinfo andmebaasis olevate eraisikute kohta tehtud maksehäirete päringute informatsiooni põhjal k-keskmiste klasterdamise abil leida inimeste finantskäitumise mustreid. Saadud klastrite tulemused kaasatakse krediidiriski mudeli loomisesse, et uurida, kas maksevõimelisuse tõenäosuse hindamisel kasutades taotlusele eelneva aasta jooksul tehtud päringute infot parandab logistilisel regressiooni mudelil põhineva krediidiriski mudeli klassifitseerimise täpsust.

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

Võtmesõnad: klasteranalüüs, kõrgdimensionaalsed andmed, krediidirisk, krediidiinfo, logistilise regressioon

Using inquiry information made about private individuals to the credit bureau in the bank's credit risk model

Master's thesis

Triin Ree

Abstract. The aim of this master's thesis is to find patterns of people's financial behavior based on the data of private persons payment default inqueries in the database of Creditinfo by using k-means clustering. The results obtained in the cluster analysis are used in the logistic regression analysis to develop a credit risk model to assess borrower's creditworthiness. Therefore, the effect of using inquiries information in the credit risk model on its accuracy is studied.

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

Keywords: cluster analysis, high-dimensional data, credit risk, credit information, logistic regression

Sisukord

1. Sissejuhatus

Käesolev magistritöö on valminud koostöös LHV Panga ja Creditinfo Eestiga ning selle eesmärgiks on välja töötada ja valideerida Eesti kontekstis unikaalne andmeallikas eraisikute krediidiriski hindamiseks finantsasutustele, kindlustusandjatele, telekommunikatsiooniettevõtetele ja teistele eraisikutele krediidi alusel kaupa või teenuseid pakkuvatele ettevõtetele.

Creditinfo (endine Krediidiinfo) on Eesti suurim krediidibüroo, mis haldab krediidiasutuste ametlikku maksehäireregistrit, kuhu salvestatakse nii ettevõtete kui eraisikute maksehäired. Maksehäire on sealjuures defineeritud kui vähemalt 30€ suurune makseviivitus, mis ületab maksetähtaega 45 päeva või enam. Maksehäireregistril on nii lepingulised liikmed, kes edastavad sinna kõigi oma maksehäirete info, kui ka tarbijad, kes võivad ka maksehäireid edastada, kuid kelle peamine huvi on teha päringuid oma klientide kohta. Maksehäireregister täidab kahte peamist eesmärki: see võimaldab laenuandjatel järgida seadusest tulenevaid vastutustundliku laenamise põhimõtteid ning distsiplineerib eraisikuid ja ettevõtteid oma finantskohustusi täitma.

Maksehäireregister sisaldab 2019. aasta seisuga ca 420 000 avaldatavat eraisiku maksehäiret ehk kuni 5 aastat tagasi tekkinud maksehäiret, mis on tasutud või kuni 13 aastat tagasi tekkinud maksehäiret, mis on tasumata. Sealjuures 9% täisealisest Eesti elanikkonnast esineb üks või enam maksehäiret, mis on hetkel tasumata. See tähendab, et juba võlakeerisesse sattunud inimest on laenuandjatel kerge tuvastada, kuna nendel on registris juba mitu maksehäiret üleval. Krediidiskoorimises nimetatakse seda ka negatiivseks informatsiooniks. Positiivne informatsioon – laenusoovija aktiivsete ja minevikus tasutud kohustuste info – ei ole Eestis üheski keskses registris talletatud ning laenuandjal on selline info ebatäielikul kujul ainult enda korduvklientide kohta.

Positiivse info puudumine ei võimalda laenuandjatel tuvastada laenutaotlejaid, kellel esineb riskikäitumine – varasemate kohustuste finantseerimine uute laenudega. Selliste klientide puhul on aja küsimus millal laenuintresside tasumine ei ole enam jõukohane ning iga krediidiasutuse huvi on kaitsta ennast selliste väga kõrge riskiga laenude väljastamise eest. Magistritöös kirjeldatud lähenemine põhineb hüpoteesil, et positiivset infot on mingil määral võimalik asendada surrogaatinfoga ning parandada seeläbi krediidiriski mudeli täpsust.

Vastavalt andmekaitseseaduse nõuetele on Creditinfo kohustatud salvestama iga eraisiku kohta tehtud maksehäireregistri päringu kellaaja, kuupäeva ning päringu teinud juriidilise või eraisiku andmed. Autori lähenemine antud töös põhineb eeldusel, et suurema osa päringute taga on krediidiandja õigustatud huvi eraisikule toote või teenuse pakkumiseks või täpsemalt – eraisiku soov omandada krediidi alusel mingit toodet või teenust. Kuna Creditinfol puudub igasugune info päringu tulemi kohta saab tehtud päringuid käsitleda kui positiivse info surrogaatinfot.

Edasine töö lähtub eeldusest, et sellise info kasutamine võimaldab luua olulisel määral täpsema mudeli kui vaid krediidiasutuse siseinfol ja negatiivse maksehäire infol põhinev skooringumudel. Töö eesmärkideks seati:

1. Näidata, et laenutaotlusele eelneva aasta jooksul eraisiku kohta tehtud päringute põhjal on võimalik ennustada taotlusele järgneva aasta maksekäitumist

2. Grupeerida eraisikud minevikus tehtud päringute mustri alusel ning vastavad grupid ära kirjeldada ja visualiseerida lähtuvalt päringute tegemise perioodile järgneva aasta maksekäitumisest
3. Disainida toorandmetest muutujad, mis kirjeldaksid võimalikult hästi päringute ajalugu ning implementeerida need LHV krediidiriski mudelis

Töö käigus rakendati erinevaid statistilisi meetodeid päringute tegemise ajaloost muutujate loomiseks. Lisaks päringute tulemi puudumisele pidi lähtuma ka teisest olulisest kitsendusest – maksehäireregistri leping ei luba Creditinfol avaldada päringuid teinud ettevõtete registrikoode. Töö läbiviimiseks tehti autorile kättesaadavaks päringu teinud ettevõtte anonümiseeritud tunnus, aastakäibe suurus, töötajate arv ning EMTAK klassifikaatori põhine tegevusala.

Töö on jaotatus neljaks: teooria, analüüs, tulemused ja järeldused. Esimeses peatükis on kirjeldatud klasteranalüüsi ja peakomponentide meetodeid, mille abil otsitakse Creditinfo eraisikute andmetele tuginedes finantskäitumise mustreid. Samuti on antud peatükis käsitletud logistilist regressiooni mudelit, millel põhineb skooringumudel hindamaks LHV panga klientide maksejõuetuse tõenäosust järgneva aasta jooksul peale taotluse tegemist. Teises peatükis on kirjeldatud analüüsiks kasutatud Creditinfo ettevõtete ja eraisikute andmestikke ning LHV krediiditaotlejate andmestikku. Täpsemalt kirjeldatakse analüüsiprotsessi ja implementeerimist. Kolmandas peatükis keskendutakse Creditinfo eraisikute kohta loodud muutujate põhjal saadud klasteranalüüsi tulemuste kirjeldamisele. Lisaks katsetatakse alternatiivina esialgsetest muutujatest leitud peakomponentidel põhinevat klasterdamist. Klasteranalüüsi käigus leitud klastrid kaasatakse LHV skoorigumudelisse, et hinnata erinevatel meetoditel saadud muutujate panust mudeli täpsuse parandamisel.

2. Teooria

2.1 Klasteranalüüs

Klasteranalüüsi kirjeldus põhineb raamatul [1].

Klasteranalüüsi ehk andmete segmenteerimist kasutatakse objektide jaotamisel rühmadesse ehk klastritesse omavahelise sarnasuse alusel. Objektid on klasterdatud ehk teisisõnu rühmitatud järgmiste põhimõtete alusel: klastrisisese sarnasuse maksimeerimine ja klastrite vahelise sarnasuse minimeerimine. Eesmärgiks on moodustada klastrid nii, et objektid klastri siseselt oleksid võimalikult sarnased üksteisele, aga objektide sarnasus klastrite vahel oleks võimalikult väike.

Klasterdamine on üks juhendamata õppe (*unsupervised learning*) meetoditest, mille eesmärgiks on leida andmetest struktuuri.

Erinevalt klassikalistest klassifikatsiooniprobleemidest, kus iga vaatlus kuulub teadaolevasse rühma ja eesmärgiks on ennustada, millisesse rühma kuulub uus vaatlus, püüab klasteranalüüs leida rühmade arvu ja nende koosseisu.

Klasterdamismeetodid võib jaotada tulemuse struktuuri järgi kaheks:

- hierarhilised meetodid (*hierarchial method*);
- eraldusmeetodid (*partitional method*).

Hierarhiliste meetodite korral on tulemiks üksteises sisalduvad klasterdused. Hierarhilised meetodid jagunevad omakorda jagavateks (*divisive*) ja ühendavateks (*agglomerative*). Jagava lähenemise korral alustatakse ühest suurest klastrist, kuhu kõik objektid kuuluvad ja igal järgneval sammul jagatakse klaster väiksemateks klastriteks. Ühendava klasterdamise korral moodustab iga objekt eraldi klastri ja igal järgneval sammul ühendatakse objektid või grupid, mis on üksteisele lähedal. Hierarhiline klasterdamine sobib kasutamiseks juhul, kui andmepunktide hulk on väike.

Eraldusmeetodite puhul jagatakse objektid etteantud arvuks k grupiks nii, et gruppide arv ei ületaks objektide arvu ja iga grupp sisaldaks vähemalt ühte objekti, kusjuures iga objekt kuulub vähemalt ühte gruppi.

K-keskmiste klasterdamine on üks populaarsemaid eraldusmeetodil põhinevaid algoritme, jaotamaks andmed klastritesse. Antud meetodi eeliseks on see, et ta on efektiivne suurte andmemahtude korral. [2]

2.2 K-keskmiste meetod

K-keskmiste meetodi kirjeldus põhineb raamatul [3].

K-keskmiste klasterdamine on meetod, mis jaotab vaadeldavad objektid K lõikumatuks klastriks, kus esmalt tuleb määrata klastrite arv K.

Olgu C_1,..., C_K indeksite hulgad, mis tähistavad vastavasse klastrisse kuulumist. Hulgad C_1,..., C_K täidavad järgmisi tingimusi:

1) Iga objekt kuulub vähemalt ühte K-st klastrist : $C_1 \cup C_2 \cup \ldots \cup C_K = \{1, \ldots, n\}$.

2) Klastrid on lõikumatud, ükski objekt ei kuulu rohkem kui ühte klastrisse: $C_k \cap C_{k'} = \emptyset, kui\ k \neq k'$.

K-keskmiste klasterdamise idee seisneb selles, et klastrisisene hajuvus oleks väike. Klastrisisene hajuvus klastri C_k korral on $W(C_k)$, mis näitab kui palju objektid klastri siseselt erinevad üksteisest. Seega soovime lahendada järgmist probleemi:

$$\min_{C_1,\dots,C_k} \sum_{k=1}^{K} W(C_k).$$

Ehk soovime eraldada objektid K-sse klastrisse nii, et kogu klastrisisene hajuvus summeerituna üle kõigi klastrite K oleks minimaalne.

Et antud optimeerimisülesannet lahendada, tuleb defineerida klastrisisene hajuvus. Selleks on palju erinevaid võimalusi, üheks levinumaks on eukleidilise kauguse (*Euclidean distance*) ruut. Olgu meil andmehulk X, mis koosneb n objektist ja neil on mõõdetud p tunnust. Seega, saame defineerida

$$W(C_k) = \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2,$$

kus $|C_k|$ tähistab objektide arvu k-ndas klastris.

Seega võime öelda, et klastrisisene hajuvus k-nda klastri korral on võrdne k-ndas klastris olevate objektide paariviisiliste eukleidiliste kauguste ruutude summa ja k-ndas klastris olevate objektide arvu jagatisega. Kombineerides eelpool toodud valemeid, saame defineerida k-keskmiste klasterdamise järgmiselt

$$\min_{C_1,\dots,C_k} \sum_{k=1}^{K} \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2.$$

Vastava optimeerimisülesande lahendamiseks kasutatakse erinevaid algoritme. Järgmistes alapeatükkides käsitleme nii k-keskmiste kui ka k-keskmiste++ algoritmi.

2.2.1 K-keskmiste algoritm

Järgnev alapeatükk põhineb raamatul [4], kui ei ole viidatud teisiti.

K-keskmiste algoritm on kirjeldatav järgmiste etappidena:

1) Esiteks määratakse loodavate klastrite arv K.
2) Seejärel valitakse juhuslikult andmestikust k objekti ja määratakse need esmasteks klastrite keskpunktideks.
3) Iga objekt määratakse klastrisse, mille keskpunkt on talle lähim, vastavalt valitud kaugusmõõdule.
4) Arvutatakse klastrite tsentroidide väärtused ümber. K-nda klastri tsentroid on p-mõõtmeline vektor iga tunnuse keskmisest.
5) Korratakse eelnevaid samme 3 ja 4 seni kuni klastritesse jaotamine enam ei muutu.

Kuna k-keskmiste algoritm leiab lokaalse miinimumi, siis sõltub tulemus paljustki esialgu juhuslikult määratud klastritest. Seepärast on oluline rakendada algoritmi mitmeid kordi erineva esmase jaotamisega klastritesse. [3]

2.2.2 K-keskmiste++ algoritm

K-keskmiste++ algoritmi idee on välja pakkunud D.Arthur ja S.Vassilivitskii ja antud algoritmi kirjeldus põhineb artiklil [5]. Võrreldes eelnevalt kirjeldatud k-keskmiste algoritmiga, mille idee autoriks on Lloyd ja mida siiani väga laialdaselt kasutatakse, on katsed näidanud, et k-keskmiste++ algoritmi kasutamine parandab nii kiirust kui ka täpsust k-keskmiste klasterdamisel. Antud meetodi puhul pakutakse välja konkreetne viis, kuidas valitakse keskpunktid k-keskmiste algoritmi jaoks. Täpsemalt, tähistagu $D(x)$ lühimat kaugust objekti ja lähima keskpunkti vahel, mis on varasemalt juba valitud. Seega saame kirjeldada k-keskmiste++ algoritmi järgnevalt:

1) Esiteks määratakse loodavate klastrite arv K.

2a) Määratakse üks keskpunkt $\boldsymbol{a_1}$, mis on valitud ühtlaselt juhuslikult andmehulgast X.

2b) Määratakse uus keskpunkt $\boldsymbol{a_i}$, vastav punkt on valitud X tõenäosusega $\frac{D(x)^2}{\sum_{x \in X} D(x)^2}$.

2c) Korratakse sammu 2b seni kuni kõik K keskpunkti on koos.

3) Edasi jätkatakse nagu tavalise k-keskmiste algoritmi puhul, sammudega 3-5.

2.2.3 Kaugusmõõt

Vaatluste klassifitseerimisel gruppidesse on tarvilik leida meetod , mille alusel arvutatakse iga vaatluspaari vaheline kaugus või teisisõnu sarnasus. Kaugusmõõdu valik on oluline samm klasterdamisel, kuna see mõjutab klastrite kuju. Vastav mõõt defineerib selle, kuidas sarnasus kahe elemendi (x, y) vahel on arvutatud. Kõige sagedamine kasutatav kaugusmõõt on eukleidiline kaugus, mis on defineeritud järgnevalt:

$$d_{euc}(\boldsymbol{x}, \boldsymbol{y}) = \sqrt{\sum_{i=1}^{p} (x_i - y_i)^2},$$

kus **x** ja **y** on p-mõõtmelised vektorid. [4]

Koosinuse sarnasuse (*Cosine similarity*) on kahe vektori vahelise sarnasuse näitaja, mõõtes nende vahelist nurka. Koosinuse sarnasuse mõõdu korral maksimeeritakse klastrisisest sarnasust. Antud kaugusmõõdu väärtuste vahemik on -1 ja 1 vahel, kus 0 tähendab, et kaks vektorit on ortogonaalsed ja 1 tähendab, et kaks vektorid osutavad samas suunas ning -1 tähendab seda, et kaks vektorit on diametraalselt vastupidised. Olgu meil kaks vektorit **x** ja **y**, nende omavaheline sarnasus on defineeritud järgmiselt:

$$\cos(\boldsymbol{x}, \boldsymbol{y}) = \frac{\boldsymbol{x} \cdot \boldsymbol{y}}{\|\boldsymbol{x}\| \|\boldsymbol{y}\|},$$

kus · tähistab vektorite **x** ja **y** skalaarkorrutist ja $\|\boldsymbol{x}\|$ning $\|\boldsymbol{y}\|$ eukleidilist normi. $\|\boldsymbol{x}\|$ on vektori $\boldsymbol{x} = (x_1, x_2, \dots, x_p)$ eukleidiline norm, mis on defineeritud kujul $\sqrt{x_1^2 + x_2^2 + \cdots + x_p^2}$. Sarnaselt on defineeritud ka ka vektori **y** eukleidiline norm $\|\boldsymbol{y}\|$. Kontseptuaalselt, eukleidiline norm tähistab vektori pikkust. [1]

Kui eukleidilise kauguse korral mõõdetakse objektide vahelist kaugust, siis koosinuse sarnasuse korral vaadeldakse hoopis vektorite vahelist nurka, et näidata, kui sarnased on objektid omavahel. Eukleidiline kaugusmõõt ei tööta hästi andmete korral, kus on palju 0 väärtuseid. Näiteks sagedus vektorid sisaldavad tavaliselt palju 0 väärtuseid. Üldjuhul kasutatakse koosinuse sarnasust kaugusmõõduna juhtudel kui vektorite suurus ei mängi niivõrd suurt rolli ja andmed sisaldavad palju nulle. Näiteks kui analüüsitakse tekstiandmeid, mis on kirjeldatud sõnade sagedustega tekstis. Käesoleva töö korral tundub olevat sobilikum kasutada kaugusmõõduna koosinuse sarnasust, kuna kasutatava andmestiku puhul on tegemist päringute sageduste andmetega ning andmetes on palju 0 väärtuseid . [1]

2.2.4 Normaliseerimine

Andmete normaliseerimise eesmärgiks on anda tunnustele võrdne kaal. Normaliseerimine on eriti kasulik klassifitseerimise meetodite korral nagu närvivõrkude või lähima naabri meetod ja klasteranalüüs. Kaugusmõõdul põhinevate meetodite puhul aitab normaliseerimine vältida olukorda, kus laia vahemikuga tunnus (näiteks sissetulek) omab rohkem kaalu, kui väikese vahemikuga tunnus (näiteks binaarne tunnus). Andmete normaliseerimiseks on mitmeid meetodeid. [1]

Käesolevas töös on kasutatud standardiseerimise teisendust, mille korral objekti tunnuse väärtusest lahutatakse tunnuse keskväärtus ning tulemus jagatakse tunnuse standardhälbega, vastav teisendus on kirjeldatav valemiga

$$z_{ij} = \frac{x_{ij} - mean(\boldsymbol{x_j})}{sd(\boldsymbol{x_j})},$$

kus z_{ij} tähistab vaatluse x_{ij} standardiseeritud väärtust ning $mean(\boldsymbol{x_j})$ ja $sd(\boldsymbol{x_j})$ on vastavalt tunnuse j keskväärtus ja standardhälve. [1]

2.3 Klastrite arvu valik

Üheks olulisimaks aspektiks k-keskmiste meetodi korral on klastrite arvu valik. Kirjanduses on toodud mitmeid indekseid ja meetodeid määramaks optimaalset klastrite arvu, kuid pole ühte kindlat õiget meetodit. Kõige tuntumaks meetodiks on küünarnuki meetod (*elbow method*), kus klastrite arv määratakse visuaalselt. Otsitakse sellist punkti, kus vastava klastrite arvu k korral väheneb klastrisiseste hajuvuste summa märgatavalt ja edasi toimub ühtlane vähenemine. Antud meetod siiski reaalelu andmetel sageli ei tööta ja klastrite arvu määramine võib olla ebatäpne, sest üldjuhul pole graafikul leitav märgatav punkt hajuvuse vähenemisest.[1]

Selleks, et leida optimaalset klastrite arvu antud töös, vaadeldi kahte erinevat indeksit: silueti (*Silhouette*) ja Davies- Bouldini indekseid.

2.3.1 Silueti meetod

Silueti meetodi kirjeldus põhineb artiklil [6], kui ei ole viidatud teisiti.

Silueti meetodit kasutatakse valideerimaks klastrite paikapidavust ja samal ajal ka sobiva arvu klastrite leidmiseks. Antud meetodit tutvustas Rousseeuw 1987.aastal. Silueti laius (*Silhouette witdh*) põhineb iga objekti $\boldsymbol{x_i}$ silueti väärtusel, mis mõõdab, kui hästi objekt $\boldsymbol{x_i}$ sobib määratud klastrisse, võrreldes klastrisisest ühtekuuluvust klastri eraldatusega. Silueti laius on defineeritud järgnevalt:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}},$$

kus $a(i)$ on keskmine erinevus $\boldsymbol{x_i},\ i \in C_k$ kõikide teiste $\boldsymbol{x_j},\ j \in C_k$ vahel,

$b(i)$ on minimaalne erinevus üle kõigi keskmiste erinevuste $\boldsymbol{x_j}, j \in C_l, l \neq k$ ja $\boldsymbol{x_i}, i \in C_k$ vahel.

Seetõttu, objekti $\boldsymbol{x_i}$ korral

$$-1 \leq s(i) \leq 1.$$

Kui $s(i)$ on lähedal nullile, siis objekti $\boldsymbol{x_i}$ võib määrata teise klastrisse, ilma klastrite lähedust ja eraldatavust halvemaks muutmata. Negatiivne $s(i)$ tähendab kehva klasterdamist, samas, mida lähemal on $s(i)$ 1-le, seda parem on klastritesse määramine olnud. Klastrite kehtivust saame valideerida silueti indeksiga, mis on määratud

$$Sil = \frac{1}{n}\sum_{i=1}^{n} s(i).$$

Kirjanduses pole võimalik välja tuua ühtegi klastrite määramise indeksit, millel oleks selge eelis teiste ees. Silueti indeks on paljudes katsetes toiminud hästi [7]. Lisaks antud meetod töötab mis tahes kaugusmõõtude korral.

Kuna silueti indeks põhineb paarikaupa kaugusmaatriksil üle kõigi andemete, on see üheks suureks väljakutseks antud meetodi puhul. Sellest vaatenurgast, tuleks silueti meetodit lihtsustada, et ta oleks k-keskmiste klasterdamisel suurte andmehulkade korral tõhus. [8]

Lihtsustatud silueti meetod (*Simplified Silhouette method*) on arvutuslikult lihtsustatud versioon silueti meetodist. Uurimustööd näitavad, et lihtsustatud silueti ja originaalse silueti meetodi tulemused on sarnane, aga esimese eeliseks on see, et ta on arvutustes tunduvalt kiirem. Lihtsustatud silueti meetodi korral on andmepunkti kaugus klastrist esitatud kaugusena klastri tsentroidist mitte keskmise kaugusena kõigi andmepunktide vahel antud klastris nagu see on silueti meetodi korral. [8]

2.3.2 Davies-Bouldini meetod

Davies-Bouldini meetodil põhinev indeks võeti kasutusele 1979.aastal. Davies-Bouldini indeks põhineb ideel, et hea eraldatuse korral peaks klastrisisene homogeensus ja kompaktsus samas ka klastrite vaheline eraldatus olema kõrge. [9]

Davies-Bouldini indeks on arvutatav järgmiselt:

$$DB = \frac{1}{n}\sum_{i=1,i\neq j}^{n} \max\left(\frac{\sigma_i + \sigma_j}{d(\boldsymbol{c_i}, \boldsymbol{c_j})}\right),$$

kus n tähistab klastrite arvu, σ_i on keskmine klastrisisene hajuvus kõigi objektide, kes kuuluvad klastrisse C_i ja klastri keskpunkti $\boldsymbol{c_i}$ vahel, σ_j on keskmine klastrisisene hajuvus kõigi objektide, kes kuuluvad klastrisse C_j ja klastri keskpunkti $\boldsymbol{c_j}$ vahel ning $d(\mathbf{c_i}, \mathbf{c_j})$ on kaugus klastrite keskpunktide $\boldsymbol{c_i}$ ja $\boldsymbol{c_j}$ vahel. Kuna optimaalsed klastrid peaksid olema kompaktsed ja olema üksteisest võimalikult erinevad, siis Davies-Bouldini indeksi väärtust peaks olema minimaalne. [10]

2.4 Peakomponentide analüüs

Peakomponentide analüüsi kirjeldus põhineb loengukonspektil [11].

Sageli kirjeldab meil vaatlusobjekte palju tunnuseid. Kõik nad on olulised kirjeldamaks objekti, aga nende rohkus muudab analüüsi ja tulemuste interpreteerimise raskeks. Seega oleks eesmärgiks vähendada andmete dimensionaalsust informatsiooni kokkusurumisega. Antud ülesande täitmiseks sobib hästi peakomponentide meetod.

Peakomponentide meetodi idee on kombineerida esialgsed lähtetunnused väiksemaks arvuks uuteks tunnusteks, mis on esialgsete tunnuste lineaarkombinatsioonid. Olgu meil n vaatlusobjektil mõõdetud p tunnust ja X on juhuslik vektor p tunnusega ning X' tähistab transponeeritud X. Seega $X = [X_1, X_2, \dots, X_p]'$. Esimeseks sammuks on leida lähtetunnustest X lineaarkombinatsioon $\alpha_1' X$ selliselt, et dispersioon oleks maksimaalne võimalik. Seega esimene peakomponent on lineaarkombinatsioon

$$P_1 = \alpha_1' X = \alpha_{11}X_1 + \alpha_{12}X_2 + \cdots + \alpha_{1p}X_p = \sum_{j=1}^{p} \alpha_{1j}X_j,$$

kus α_1 on ühikvektor p konstandist $\alpha_{11}, \alpha_{12}, \dots, \alpha_{1p}$ ja disperisoon oleks maksimaalne.

Järgmisena otsitakse teise peakomponendi lineaarkombinatsiooni $P_2 = \alpha_2' X$, mis oleks mittekorreleeritud esimese peakomonendiga $P_1 = \alpha_1' X$ ja peakomponendil P_2 oleks suuruselt järgmine dispersioon jne.

2.5 Logistiline regressioon

Järgnev alampeatükk põhineb loengukonspektil [12].

Logistilist regressiooni kasutatakse juhtudel, kui uuritav tunnus on binaarne. Näiteks, kas inimene haigestub või mitte, kas klient jääb võlgu või mitte. Eelpool kirjeldatud uuritavate tunnuste väärtused on tavaliselt kodeeritud väärtusteks 0 või 1, kus huvipakkuva sündmuse esinemist tähistab 1 ja mitteesinemist 0.

Antud töös uurime kahte klassi kuulumist, mille korral uuritav tunnus on Bernoulli jaotusega $Y \sim B(1, \pi)$, kus π on meid huvitava sündmuse tõenäosus. Seega huvitab meid seos uuritava tunnuse esinemise tõenäosuse π ja mõõdetud seletavate tunnuste vahel.

Logistilise regressiooni mudeli kuju on järgmine

$$ln\frac{\pi}{1-\pi} = \beta_0 + \beta_1 x_1 + \ldots + \beta_k x_k ,$$

kus

$\pi = \boldsymbol{P}(Y = 1)$ on sündmuse esinemise tõenäosus,

$1 - \pi = \boldsymbol{P}(Y = 0)$ on sündmuse mitteesinemise tõenäosus,

$\beta_0, \beta_1, \ldots , \beta_k$ on mudeli tundmatud parameetrid ehk argumenttunnused,

$x_0, x_1, \ldots , x_k$ on seletavad tunnused.

Üldjuhul hinnatakse logistilise regressiooni tundmatud parameetrid β_i suurima tõepära meetodil. Suurima tõepära hinnangu korral leitakse selline parameetri θ väärtus, mille korral tõepärafunktsioon $L(x, \theta)$ saavutab maksimumi.

Logit seosest saame avaldada sündmuse esinemise tõenäosuse

$$\pi = \frac{e^z}{1 + e^z},$$

kus $z = \beta_0 + \beta_1 x_1 + \ldots + \beta_k x_k$.

2.5.1 Mudeli headuse näitajad

Järgnev alapeatükk põhineb raamatul [13].

Üheks oluliseks aspektiks mudeli sobilikkuse juures on selle täpsuse hindamine. Logistilise regressiooni mudeli headust mõõdetakse erinevate näitajate abil. Antud töös kasutatakse regressiooni mudeli täpsuse hindamiseks ROC-kõvera (*receiver operating characteristic curve*) alust pindala ehk AUC (*area under the curve*) näitajat ja tõeselt positiivsete määra konkreetse lävendi korral, antud töös on selleks lävendiks valitud 0.05.

ROC-kõvera graafiliseks illustreerimiseks kasutatakse spetsiifilisust ja tundlikkust.

Spetsiifilisus (*specifity*) näitab, kui suure osa uuritava sündmuse mittetoimumisest ennustab mudel õigesti ehk tõeselt negatiivsete määra (*true negative rate*).

Tundlikkus (*sensitivity*) näitab, kui suure osa uuritava sündmuse toimumisest ennustab mudel õigesti ehk tõeselt positiivsete määra (*true positive rate*).

ROC-kõrvera graafiliseks illustreerimiseks kantakse abtsissteljele valepositiivsete määr (1-spetsiifilisus) ja ordinaatteljele tõeselt positiivsete määr (tundlikkus).

ROC- kõvera alune pindala (AUC) näitaja jääb alati 0 ja 1 vahele, mida ligilähedasem on vastav väärtus 1-le, seda parema klassifitseerimisvõimega on antud mudel. Logistilise regressiooni mudeli abil saadakse tõenäosused, mis klassifitseeritakse vastavalt etteantud lävendile (*cut-off*). Antud töös soovime mudeli alusel prognoosida maksjõuetuse tõenäosust, seega tõenäosused, mis on suuremad lävendist klassifitseeritakse 1, ehk klient on maksjõuetu ja vastupidisel juhul, kui tõenäosus on alla lävendi klassifitseeritakse 0, ehk maksejõuline klient.

Järgnevalt on toodud ROC-kõvera aluse pindala kokkuleppelised piirid mudeli headuse iseloomustamiseks:

- $AUC = 0.5$ eristusvõime puudub;
- $0.5 < AUC < 0.7$ kehv eristusvõime;
- $0.7 \leq AUC < 0.8$ aktsepteeritav eristusvõime;
- $0.8 \leq AUC < 0.9$ väga hea eristusvõime;
- $AUC \geq 0.9$ suurepärane eristusvõime.

3. Analüüs

3.1 Andmestike kirjeldus

3.1.1 Creditinfo andmestik

Creditinfo tegeleb Eesti ettevõtete ja eraisikute majandus-ja finantsandmete kogumise ning neile lisaväärtuse loomisega. Nende eesmärk on aidata klientidel teha tarku äriotsuseid, pakkudes selleks nutikaid ja kvaliteetseid andmevahetuse, andmeanalüütika ning tarkvaralahendusi. [14]

Käesolevas töös analüüsitakse Creditinfo klientideks olevate ning vaatlusaluse perioodi jooksul vähemalt ühe päringu teinud ettevõtete (edaspidi päringutegijate) ja täisealiste eraisikute, kelle kohta on vaatlusalusel perioodil päringuid tehtud (edaspidi päringusubjektide) andmeid, et luua neile lisaväärtust. Creditinfo andmebaasis on üle 400 000 ettevõtte ja ligikaudu 800 000 eraisiku andmed, sh. maksehäirete ajalugu.

Päringutegijate andmestik koosneb 1290 ettevõttest, kes vahemikus 30.09.2016 kuni 30.09.2017 tegid vähemalt ühe päringu eraisiku kohta. Ettevõtete kohta on teada järgnevad andmed: aastakäibe suurus, töötajate arv ning EMTAK klassifikaatori põhine tegevusala. Ettevõtete poolt tehtud päringute arv on ärisaladus ning seda ei ole tehtud töö autorile kättesaadavaks. Lisaks on iga ettevõte kirjeldatud ainult anonümiseeritud registrikoodiga, et ei oleks võimalik luua subjektiivsetel alustel gruppe.

Päringusubjektide andmestik koosneb 786 285 eraisikust, kelle kohta on teada kõik päringu teinud ettevõtted ja päringute kuupäevad perioodil 30.09.2016 kuni 30.09.2017.

Eraisikute andmestik on JSON (*JavaScript Object Notation*) formaadis, mis koosneb nimi/väärtus paaride kollektsioonidest. [15]

Joonisel 1 on antud ülevaade eraisikute toorandmete kohta, mida kasutati andmete eeltöötlemisel, et luua uusi tunnuseid.

```
{
'Isikukood1':
    {
        {
            'Ettevõte11': [[('2016-12-16 14:44:40', 3)]],
            'Ettevõte7': [[('2017-07-04 14:01:17', 3)]],
            'Ettevõte56': [[('2016-10-12 11:13:55', 3),('2016-11-13 11:47:50', 3),
            ('2017-01-24 17:45:32', 3)]],
            'Ettevõte47': [[('2017-06-01 12:09:54', 3),('2017-06-05 09:45:09', 9),
            ('2017-06-28 10:59:15', 9),('2017-06-28 10:59:15', 9)]],
            'Ettevõte1410': [[('2017-07-06 16:54:24', 3), ('2017-07-11 12:16:19', 3)]],
            'Ettevõte300': [[('2017-06-08 14:59:37', 3)]],
            'Ettevõte9104': [[('2017-08-24 09:30:23', 3),('2017-08-24 09:30:27', 3)]],
            'Ettevõte79': [[('2017-08-24 09:30:23', 3)]]
        }
    }
}
```

Joonis 1 Ülevaade Creditinfo eraisikute andmestikust.

3.1.2 LHV andmestik

LHV Pank (edaspidi LHV) on 1999.aastal loodud Eesti kapitalil põhinev pank, pakkudes hoiuseid, laene, arveldusteenuseid, väärtpaberivahendust, pensionifonde, liisingut ja varahaldusteenust nii eraisikutele kui ettevõtetele. [16]

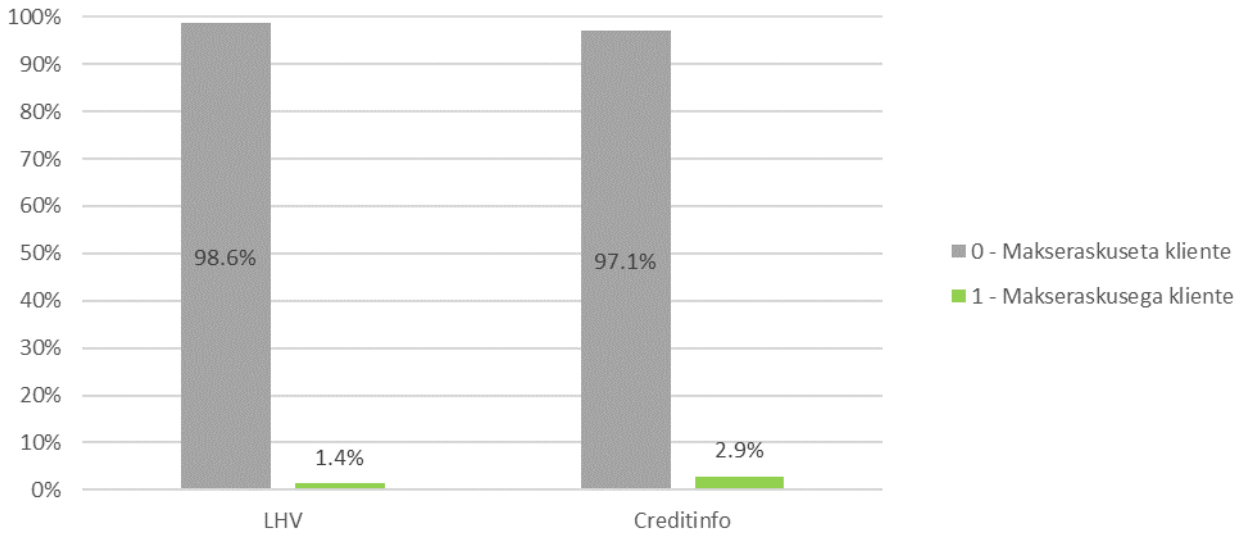
LHV laenusaajate andmestik koosneb 70 758 eraisikust, kes perioodil 01.01.2016 kuni 31.12.2017 tegid laenutaotluse ja said positiivse otsuse laenu saamiseks. Iga inimese kohta on kogutud järgmised andmed taotluse esitamise hetkel:

- Sotsiaal-demograafilised andmed: sugu, sünniaeg, kodakondsus, elamisviis, eluaseme tüüp, elukoht, haridustase jms;
- Finantsandmed: sissetulek, kohustused jms;
- Maksekäitumise andmed: maksehäired, maksuvõlad jms, mis põhinevad nii krediidibüroode andmetel, kui LHV enda ajaloolistel andmetel kliendi kohta;
- Laenuspetsiifilised andmed: laenu suurus, periood jms;

Lisaks on iga laenusaaja kohta teada, kas lepingu sõlmimisele järgneva aasta jooksul on klient muutunud maksejõuetuks või mitte. Seega uuritava tunnuse väärtus 1 tähistab maksejõuetut ja 0 maksejõulist klienti. Laenusaajate seast 69 787 klienti on maksevõimelised ja 971 makseraskustega.

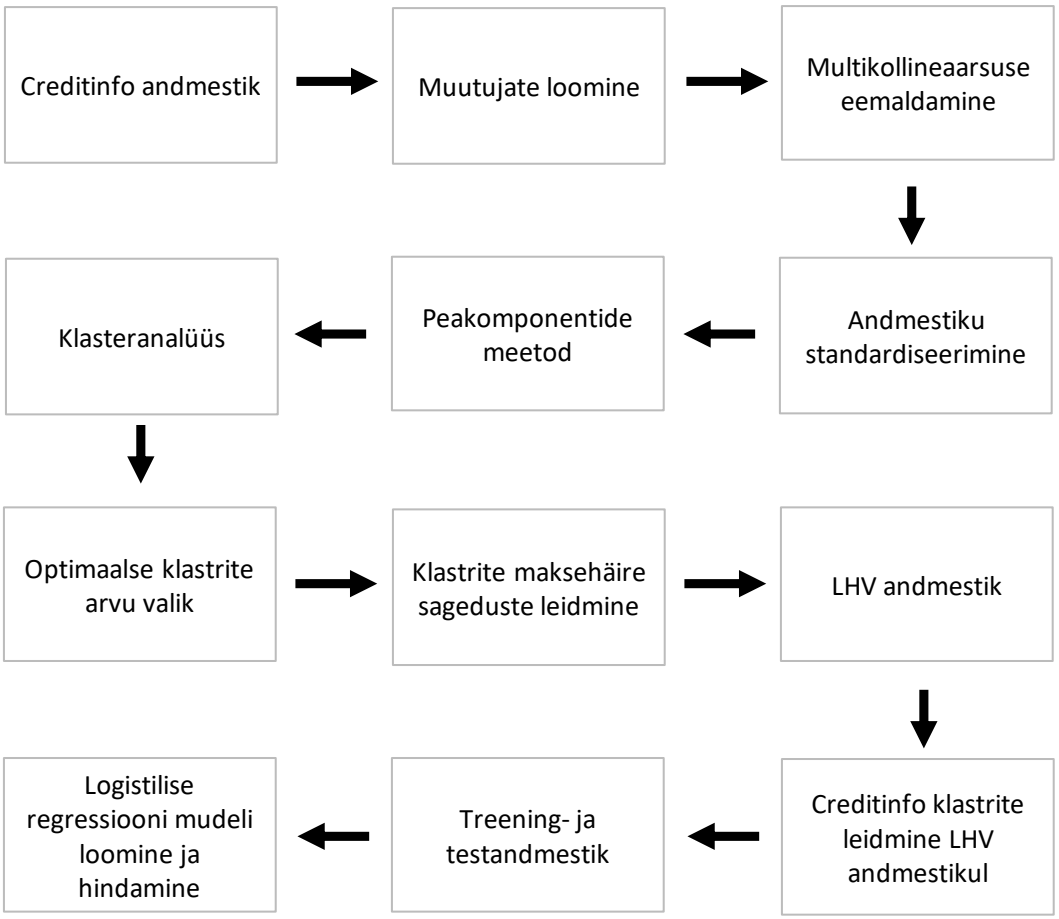
Täiendavalt on igale laenutaotlejale leitud Creditinfo andmebaasist laenutaotlusele eelneva aasta jooksul ettevõtete poolt tehtud päringute informatsioon, mida kasutatakse hiljem isikute klastritesse määramiseks.

Järgneval joonisel on toodud võrdluseks uuritava tunnuse- makseraskusega klientide arvu jaotus LHV ja Creditinfo andmestikul. Makseraskusega kliendiks loetakse LHV kontekstis üle 90 päevase võlgnevusega klienti. Creditinfo korral on makseraskusega klient defineeritud kui vähemalt 30€ suurune makseviivitus, mis ületab maksetähtaega 45 päeva või enam. Mõlema andmestiku korral on makseraskusega klientide osakaal kõikidest klientidest väga väike, jäädes paari protsendi juurde.



Joonis 2 Makseraskusega klientide protsentuaalne jaotus LHV ja Creditinfo andmestiku korral.

3.2 Tööprotsess



Joonis 3 Magistritöö tööprotsess.

3.3 Muutujate loomine

Muutuja (*feature*) kirjeldab toorandmeid numbrilisel kujul. Esialgses andmestikus võib oluline informatsioon esineda nii tekstilises, numbrilisel, audiovisuaalsel või mõnel muul serialiseeritud kujul. Muutujate loomine (*feature engineering*) on meetod, mille käigus luuakse toorandmetest muutujaid ja teisendatakse need formaatidesse, mis on sobilikud masinaõppe mudelitele. See on oluline etapp kogu masinaõppe protsessis, kuna õigesti valitud muutujad võivad lihtsustada modelleerimist ja seeläbi võimaldab saada parema täpsusega tulemusi. [17]

Eraisikute maksehäirete kohta on palju erinevaid ettevõtteid teinud väga palju päringuid. Et eraisikute andmestikust luua mõistlikke ja statistiliste mudelite jaoks sobilikke muutujaid on vajalik ettevõtete tasemeid mõistlikul viisil vähendada.

Esimese sammuna klassifitseeriti Creditinfo ettevõtete andmestik EMTAK koodi ja käibe alusel. Antud sammu eesmärk on vähendada tunnuste arvu ja grupeerida sarnased ettevõtted üheks grupiks. Saadud klassifikatsioonide alusel luuakse eraisikute andmestikus uued

muutujad. Sellel sammul on ka oluline äriloogikast lähtuv eesmärk: kuna ettevõtteid võib jooksvalt tekkida ja kaduda, oleksid puhtalt ettevõttepõhised muutujad ajas väga ebastabiilsed.

Enamik ettevõtteid on grupeeritud EMTAK klassifikaatori 1.taseme ehk tähtkoodi alusel. Tegevusvaldkondade korral, mis hõlmavad endas finantsteenuste osutamist on jaotus tehtud madalamate tasemete (2.taseme, 3.taseme, 4.taseme ja 5.taseme) lõikes, vastavalt sellel kui oluliseks töö autor pidas antud tegevusala ettevõtte päringute tegemist, näitamaks isiku käitumisharjumusi ja riskantsust.

Tegevusalad, mille päringute informatsioon ei pruugiks anda väärtuslikku infot isiku käitumise kohta, on kokku grupeeritud üheks klastriks. Näiteks ettevõtete grupp 1 koosneb kolmest suurest tegevusalast – põllumajandus, mäetööstus, töötlev tööstus (vt. Tabel 1).

Teisalt näiteks kuulub EMTAK klassifikaatori tegevusala kunst, meelelahutus ja vaba aeg alla detailsema liigituse alusel hasartmängude tegevusala. Võiks eeldada, et isik on riskantsem juhul, kui tema kohta on teinud päringu ettevõte, kelle tegevusalaks on hasartmängud, kui mõni teine ettevõte, kes ei kuulu küll hasartmängude alla, aga üldisemas tegevusalade kategoorias on samuti kunst, meelelahutus ja vaba aeg rühmas. Antud kaalutulustel on hasartmängu tegevusala eraldi grupina kirjeldatud.

Eelpool kirjeldatud printsiipide alusel on määratud 27 ettevõtete gruppi. Tabelis 1 on toodud ettevõtete grupp ja vastava grupi kirjeldus.

Tabel 1 Ettevõtete grupp ja kirjeldus.

Ettevõtete grupp	Kirjeldus
1	Põllumajandus, mäetööstus, töötlev tööstus
2	Elektrienergia
3	Veevarustus, ehitus
4	Mootorsõidukite müük
5	Hulgikaubandus, v.a mootorsõidukid
6	Jaekaubandus, v.a mootorsõidukid
7	Parklate tegevus
8	Veondus, laondus
9	Teenindavad tegevused ja teadmata tegevusalad
10	Kutse-, teadus- ja tehnikaalane tegevus
11	Hasartmängud
12	Elektroonilise side teenus
13	Muu info- ja side teenused
14	Kinnisvaraalane tegevus
15	Avalik haldus ja riigikaitse
16	Inkassoteenus ja krediidiinfo
17	Mootorsõidukite rentimine
18	Masinate, seadmete kasutusrent
19	Muu haldus-ja abitegevused
20	Suuremad pangad
21	Väiksemad pangad
22	Valdusfirmade - ja maaklertegevus
23	Kapitalirent (liising)
24	Muu laenuandmine, v.a pandimajad
25	Kahjukindlustus, elukindlustus
26	Kindlustuse abitegevusalad
27	Pandimajad

Teise etapina jaotati kõigepealt Creditinfo eraisikute andmestikus olevad ettevõtted eelpool toodud gruppidesse. Seejärel loodi kolm muutujate plokki iga ettevõtete grupi kohta :

- ettevõtete arv;
- päringute arv;
- ajapõhised näitajad sh:
    - keskmine päevade arv päringute vahel;
    - maksimaalne päevade arv päringute vahel;
    - päevade arv, mis on möödas varaseimast päringust;
    - minimaalne päevade arv päringute vahel;
    - päevade arv, mis on möödas hiliseimast päringust.

Esimene muutujate plokk sisaldab endas 27 tunnust. Iga tunnus näitab, mitu unikaalset ettevõtet on aastasel perioodil vastavas ettevõtte grupis päringu teinud antud isiku kohta. Kuna ettevõtete gruppe oli kokku 27, siis iga grupi kohta on leitud antud väärtus. Kui mõnes ettevõtte grupis polnud antud perioodil ühtegi päringut, siis vastava tunnuse väärtus määrati võrdseks 0-ga.

Teise muutujate ploki iga tunnus näitab, mitu päringut on tehtud aastasel perioodil vastavas ettevõtte grupis. Kui kaks ettevõtet, kes kuulusid samasse ettevõtte gruppi olid teinud samal päeval päringu, siis loeti päringute koguarvuks 2. Kui aga sama ettevõtte korral oli ühel päeval olnud isiku kohta rohkem kui üks päring, siis loeti päringute koguarvuks 1. Pigem on siin tegu kas tehnilise probleemiga, et andmed on salvestunud andmebaasi mitmekordselt või et andmed ei jõudnud koheselt päringutegijani ja päring on uuesti edastatud.

Kolmandas plokis on 5 erineva arvutusloogikaga ajapõhist tunnust, mis on arvutatud igale ettevõtte grupile. Ajaraam, milles päringuid vaadatakse on 30.09.2016 kuni 30.09.2017, siis varaseim päring saab olla 30.09.2016 kuupäeva seisuga ja hiliseim päring 30.09.2017. Esimene muutuja on päevade arv, mis on möödas varaseima päringu tegemise hetkest kuni tänase päringu tegemise hetkeni (siin ja edasi mõeldakse tänase päringu tegemise hetkena 30.09.2017 kuupäeva). Kuna päringuid vaadatakse vaid aastases ajaaknas, siis maksimaalne päevade arv, mis on möödas varaseimast päringust, saab olla 365 päeva.

Teine muutuja on päevade arv, mis on möödas viimasest päringust kuni tänase päringu hetkeni. Kolmas ajapõhine muutuja on minimaalne päevade arv erinevate päringute vahel antud ettevõtte grupis. Neljas muutuja on maksimaalne päevade arv erinevate päringute vahel antud ettevõtte grupis. Viies muutuja on keskmine päevade arv päringute vahel. Kui isiku kohta pole ühtegi päringut toimunud viimase aasta jooksul vaadelduna uue päringu tegemise hetkest, siis loetakse kõigi viie ajapõhise tunnuse väärtus võrdseks 365 päevaga. Kuna ajapõhiste tunnuste puhul peaks 0 tähendama seda, et päring eraisiku kohta on toimunud alles hiljuti, siis 365 viitaks sellele, et viimasest päringust on möödas vähemalt aasta, seega kuna antud juhul isiku kohta pole viimase aasta jooksul ühtegi päringut tehtud, siis näitaks väärtus 365, et viimasest päringust on kaua aega möödas.

Võiks eeldada, et mida väiksem on muutuja „keskmine päevade arv päringute vahel“, seda tihedamini erinevaid teenuseid kasutatakse, kus kliendi tausta ja krediidivõimelisust hinnatakse või näiteks on inimene sarilaenaja, kes ühe laenu tasumiseks võtab järgmise laenu, mis peaks andma indikatsiooni, millise käitumisharjumustega inimesega on tegu. Kui muutuja „maksimaalne päevade arv päringute vahel“ on väike, siis annaks see märku sellest, et päringud on toimunud tihti, vastupidisel juhul, kui vastav väärtus on suur, siis tähendab, et mingil pikemal perioodil pole vastava inimese kohta huvi tuntud. Muutuja „päevade arv, mis on möödas hiliseimast päringust“, kui vastav väärtus on väike, siis tähendaks seda, et viimane päring on alles toimunud ning inimene taotleb juba uut laenutoodet, vastupidisel

juhul tähendaks seda, et viimasel ajal pole päringuid isiku kohta tehtud. Kui muutuja „päevade arv, mis on möödas varaseimast päringust“ väärtus on väike, siis tähendaks seda, et alles hiljuti on selle isiku kohta huvi tuntud ja varasemalt pole see inimene ettevõtete huviorbiiti sattunud. Kui muutuja „minimaalne päevade arv päringute vahel“ on väike, siis annab see märku, et inimese kohta on tehtud tihedalt mingil perioodil päringuid.

Ülaltoodud kirjelduse põhjal loodi kokku 189 uut muutujat. Muutujate täpne nimekiri on toodud lisas 2.

3.4 Dimensionaalsuse vähendamine

Eraisikute andmestikus loodud 189 päringupõhise muutuja omavaheliste seoste uurimiseks arvutati tunnuste omavahelised korrelatsioonid. Korrelatsioonimaatriksi põhjal järeldus, et paljud ettevõtte grupi sisesed muutujad on omavahel tugevalt korreleeritud, mida oligi arvata. On näidatud, et kui klasteranalüüsi jaoks kasutatavad muutujad on kollineaarsed, saavad need muutujad suurema kaalu kui teised [18]. Selleks, et tugevalt korreleeritud muutujad ei moonutaks klasteranalüüsis tulemusi, eemaldati andmestikust muutujad, mis olid omavahel tugevalt korreleeritud. Kui kahe muutuja vahelise seose kirjeldamiseks kasutatud Spearmani korrelatsioonikordaja väärtus oli üle 0.7, siis üks muutujates eemaldati andmestikust. Peale antud sammu jäi Creditinfo eraisikute andmestikku 53 päringupõhist muutujat, seega kasutati klasteranalüüsis pea kolmandikku esialgse andmestiku muutujatest.

Alternatiivse võimalusena rakendati esialgsel 198 muutujal peakomponentide analüüsi, et saada uued mittekorreleeritud tunnused ja vähendada andmemahtu. Antud meetodit rakendati standardiseeritud andmestikul. Peakomponentide analüüsi korral üheks kõige enam kasutatavaks kriteeriumiks peakomponentide arvu määramisel on Kaiser-Guttmani kriteerium, mille korral soovitatakse kasutada peakomponente, millele vastavad omaväärtused on suuremad ühest [19].

3.5 Klasteranalüüsi rakendamine

Klasteranalüüsi üheks eesmärgiks oli andmeid visualiseerida ning hinnata, kas erineva päringute mustri korral on ka maksehäire osakaalud erinevad. Teiseks eesmärgiks oli saadud klastreid kasutada muutujatena krediidiriski mudelis.

Klasteranalüüs teostati Creditinfo eraisikute andmestikul. Esimese sammuna andmed normaliseeriti. Seejärel rakendati andmetel k-keskmiste meetodit, mille korral on oluline määrata klastrite arv. Optimaalse klastrite arvu leidmiseks, anti meetodile ette klastrite vahemik 2 kuni 65.

Optimaalse klastrite arvu hindamiseks kasutati meetrikutena lihtsustatud silueti meetodil põhinevat indeksit ja Davies-Bouldini indeksit.

Objektide vahelise kauguse mõõduna kasutati nii eukleidilist kaugust kui ka koosinuse sarnasust. Käesolevas töös kasutatav eraisikute andmestik esindab päringute loendusandmeid ja seega on andmetes palju 0 väärtuseid. Tulenevalt sellest on vaja sellist kaugusmõõtu numbriliste andmete jaoks, mis ignoreeriks null-vasteid [4]. Nii eukleidiline kaugus kui ka koosinuse sarnasus sobib kasutamiseks numbriliste suuruste korral, viimast kasutatakse tihti andmestike korral, kus on palju nulle [4]. K-keskmiste meetod eukleidilise

kaugusega andis optimaalseks klastrite arvuks. See näitab, et suure dimensiooni tõttu on kõik punktid üksteisest küllaltki kaugel ja eukleidilise kauguse mõttes ei tekkinud lähedaste punktide rühmi, mida klasterdamisel saaks edukalt kasutada. Arvestades antud töös kasutatava andmestiku omadusi on sobilikum kasutada koosinuse sarnasust.

K-keskmiste klasterdamiseks on vajalik määrata maksimaalne iteratsioonide arv. Vastav väärtus määrab ära, mitu korda klastrite tsentroide maksimaalselt ümber arvutatakse. Kui maksimaalne iteratsioonide arv on liiga väike, siis ei pruugi algoritm vastavate korduste jooksul koonduda. Antud töös on kasutatud maksimaalse iteratsioonide arvuna 100, kuid valdavalt toimus koondumine juba väiksema arvu iteratsioonide korral.

Lisaks eelnevale, tuleb valida algoritm, mida kasutatakse k-keskmiste klasterdamiseks. Käesolevad töös on kasutatud k-keskmiste++ algoritmi, mis on välja töötatud D.Arthur and S.Vassilvitskii poolt.

Viimase olulise aspektina, tuleb valida esialgne seeme (*initial seed*), millest lähtuvalt tekitatakse juhuarvud, mida kasutatakse keskpunktide valimisel. Kuna k-keskmiste meetodi tulemus on tundlik esialgse keskpunkti valikus, siis katsetati töös erinevate seemne väärtustega ja valiti väärtus, mille korral klastrite hindamise ja valideerimise indeksid andsid parima tulemuse.

Klasteranalüüsi käigus saadud klastrite alusel jaotati kogu Creditinfo eraisikute andmestik vastavatesse klastritesse ja igale klastrile leiti ajaloolised maksehäirete esinemissagedused.

3.6 Logistilise regressiooni rakendamine

Logistilise regressiooni mudeli leidmiseks kasutatakse LHV klientide kohta kogutud andmeid, lisaks on leitud igale laenutaotlejale aasta enne taotluse kuupäeva Creditinfosse tehtud päringute informatsioon, mille põhjal määratakse isikud eelnevalt defineeritud klastritesse (Creditinfo eraisikute andmestikul leitud klastrid). Selleks, et mõista, kas maksejõuetuse ja eraisikute kohta tehtud ajalooliste päringute informatsiooni andmete vahel on seos, teostatakse logistiline regressioonanalüüs. Logistilise regressiooni korral on uuritavaks tunnuseks maksejõuetuks muutumine laenu väljastamisele (st. ka päringute tegemise perioodile) järgneva aasta jooksul, kus 1 tähistab maksejõuetut klienti ja 0 maksevõimelist.

Treeningandmestikus, mille põhjal luuakse logistiline regressiooni mudel, on 80% esialgse andmestiku klientidest, ülejäänud 20% andmestikust kasutatakse testimiseks, et hinnata saadud mudeli täpsust. Andmestikud on jaotatud nii, et treening- ja testandmestikus on proportsionaalselt sama palju maksejõuetuks muutunud kliente.

Parima mudeli valikul kasutati ettepoole sammregressiooni (*forward stepwise regression*) lähenemist, kus alustatakse ainult vabaliikmega mudelist ja lisatakse järjest argumente juurde tunnuste hulgast [13]. Argumentide lisamine toimub Akaike informatsioonikriteeriumi (AIC) väärtuse alusel, protsessi jätkatakse seni kuni ühegi argumendi lisamine AIC väärtust oluliselt ei muuda.

Parimaks mudeliks võib logistilise regressiooni korral pidada mudelit, mille AIC väärtus on väikseim. Akaike informatsioonikriteerium on defineeritud järgnevalt:

$$AIC = -2\log(L) + 2p,$$

kus L on uuritava mudeli tõepärafunktsiooni väärtus ja p on mudeli parameetrite arv. [20]

Peale mudeli sobitamist ja parima leidmist, soovitakse näidata, kui palju mõjutab antud välise allika lisamine mudelisse ennustuste täpsust. Mudeli ennustuse täpsuse hindamiseks kasutatakse ROC-kõvera aluse pindala näitajat ja tõeselt positiivsete määra lävendi 0.05 korral ning võrreldakse, kui palju muutuvad vastavate näitajate väärtused uute muutujate kasutamisel logistilise regressiooni mudelis. Mudeli täpsuse hindamise meetrikud valiti LHV ärinõuetest lähtuvalt. Mudeli üldise täpsuse (AUC) kõrval on vähemalt sama oluliseks ka mudeli sensitiivsus, mis kirjeldab mudeli võimet õigesti klassifitseerida positiivsed andmepunktid. Laenuandmise puhul on valenegatiivse (väljastatud halva laenu) kulu pangale oluliselt suurem kui valepositiivse (väljastamata jäetud hea laenu) puhul.

3.7 **Implementeerimine**

Funktsioonid muutujate loomiseks kirjutati Pythonis ning kood pandi jooksma Amazoni EC2 virtuaalmasinas, kuna andmestiku suurusest tingituna oli vaja tunduvalt suuremat arvutusvõimsust kui lauaarvuti võimaldab. Pythoni kood muutujate loomiseks on toodud lisas 2.

Eelnevast tingituna ei olnud R tarkvaraga võimalik leida optimaalselt klastrite arvu. Kuna vastavad arvutused on sellise andmehulga juures väga mahukad ja nõuavad suurt mäluhulka ning R-i interpretaator on väga piiratud ressursikasutusega, siis oli ka klasteranalüüsi puhul vaja kasutada suurandmete töötlemiseks sobivamaid rakendusi. Edasine klasteranalüüs viidi läbi Microsoft Azure Machine Learning Studio keskkonnas.

Microsoft Azure Machine Learing on SaaS (*software as a service*) pilveteenus, mille kasutajaliides on brauseripõhine. Keskkond pakub masinaõppeks vajaminevat arvutusjõudlust ja mälu (kuni 56Gb). Azure ML Studios on lohista ja aseta (*drag and drop*) kasutajaliidesega veebirakendus, kus on võimalik masinaõppe mudeleid treenida, testida ja juurutada vastavaid lahendusi ilma koodi kirjutamata. Lisaks sellele toetab Azure ML Studio Pythoni ja R programmikoodide kasutamist. [21]

K-keskmiste meetodi rakendamiseks Azure ML Studios on kasutatud moodulit *K-means Clustering.* Vastava mooduli korral tuleb määrata järgmiste parameetrite väärtused nagu klastrite arv, klastrite vahelise kaugusmõõt (eukleidiline kaugus või koosinuse sarnasus), iteratsioonide arv, k-keskmiste klasterdamise algoritm, käesolevad töös on kasutatud *K-means++* meetodit, mis põhineb D.Arthur and S.Vassilvitskii algoritmil. Täpsem nimekiri parameetrite valikutest on leitav Azure ML Studio veebilehelt. [22]

Logistilise regressiooni mudeli loomiseks LHV laenusaajate andmetel kasutati tarkvara R funktsiooni glm (*general linear model*). Sammregressiooni rakendamiseks on funkstioon *step*, kus ettepoole lähenemise korral tuleb valida parameetri *direction* väärtuseks f*orward*. ROC-kõvera aluse pindala leidmiseks *ROCR* paketti.

4. Tulemused

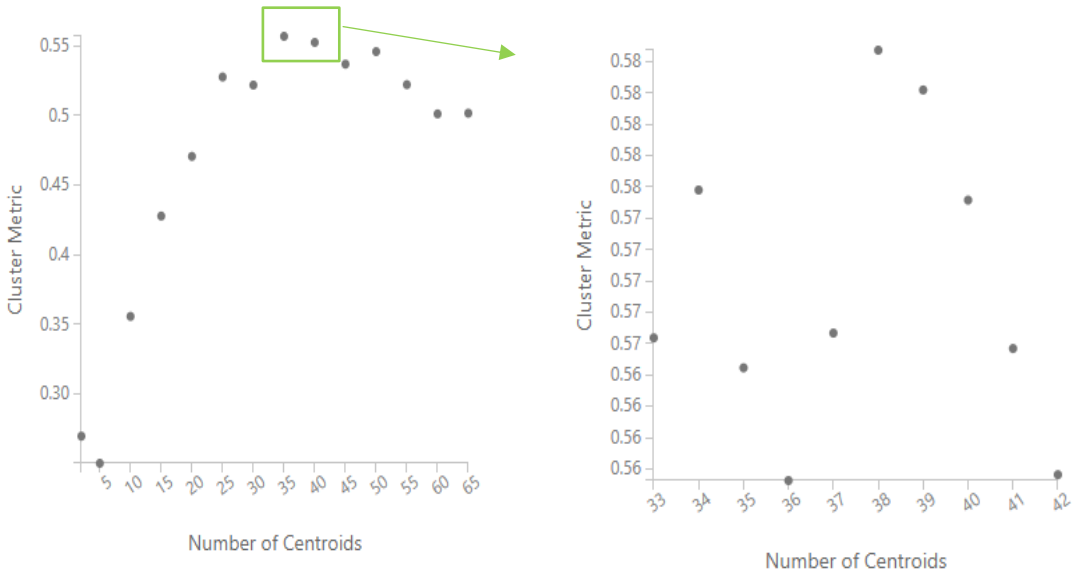
4.1 Optimaalse klastrite arvu leidmine Creditinfo päringute ajalugu kirjeldavate muutujatega andmestikul

Creditinfo eraisikute andmestikul, kus oli esialgu 198 muutujat eraisikute kohta tehtud päringute ajaloo kohta, valiti välja 53 muutujat, mis omavahel ei olnud tugevalt korreleeritud ning viidi läbi klasteranalüüs. K-keskmiste++ meetodi rakendamisel sobiva arvu klastrite leidmiseks anti algoritmile ette klastrite arvu väärtused vahemikus 2 kuni 65. Optimaalse klastrite arvu hindamisel kasutati sobivuse meetirikutena kahte erinevat meetodit - lihtsustatud silueti ja Davies-Bouldini indeksit. Tabelis 2 on toodud lihtsustatud silueti ja Davies-Bouldini indeksi väärtused erinevate klastrite arvu korral, lihtsustatud silueti indeks näitab, et optimaalseks klastrite arvuks on 38 ja Davies- Bouldini indeksi korral 39 klastrit.

Tabel 2 Lihtsustatud silueti ja Davies-Bouldini indeksi väärtused erinevate klastrite arvu korral. Roheliseks värvitud väärtus tähistab optimaalseimat väärtust. Lihtsustatud silueti indeksi korral on suurim väärtus optimaalseim, Davies-Bouldini indeksi korral näitab väikseim väärtus optimaalseimat klastrite arvu.

Klastrite arv	Lihtsustatud silueti	Davies-Bouldin
2	0.26981	1.16187
5	0.25035	1.48353
10	0.35584	1.42207
15	0.42787	1.27912
20	0.47071	1.29480
25	0.52771	1.21362
30	0.52190	1.04573
33	0.56636	1.02987
34	0.57580	1.02064
35	0.56445	1.04496
36	0.55726	1.13148
37	0.56667	1.12260
38	**0.58473**	1.02032
39	0.58219	**1.01497**
40	0.57516	1.03289
41	0.56569	1.04854
42	0.55763	1.06419
45	0.53694	1.02782
50	0.54595	1.03245
55	0.52224	1.11284
60	0.50122	1.14284
65	0.50173	1.18342

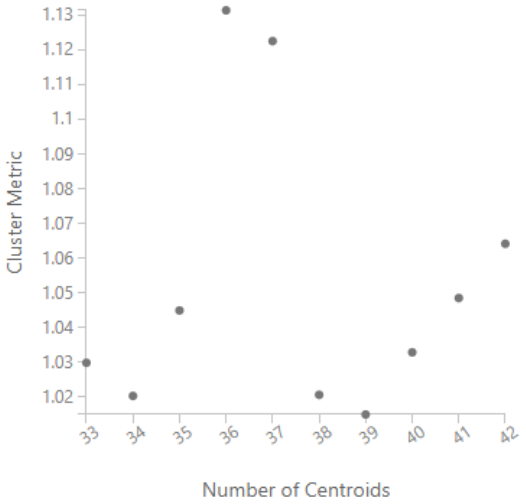
Joonis 4 illustreerib graafiliselt silueti meetodil hinnatud klastrite arvu Creditinfo eraisikute andmestikul. X-teljel on klastrite arv ja y-teljel on silueti meetodil põhinev indeks. Esimeselt graafikult võime näha, et väiksema arvu klastrite korral on lihtsustatud silueti väärtus väike, nagu teooria peatükis sai välja toodud, mida lähemal antud väärtus on 1-le, seda parem on olnud klastritesse määramine. Seega vaadatakse joonise 4 (b) graafikul lähemalt klastritesse jaotamist klastrite arvu 33 kuni 42 korral koos vastava lihtsustatud silueti meetodil põhineva indeksiga.



(a) (b)

Joonis 4 Lihtsustatud silueti meetodil hinnatud optimaalne klastrite arv Creditinfo andmestikul. (a) Võimalike klastrite arvu vahemik 2 kuni 65. (b) Täpsem sissevaade klastrite arvu 34 kuni 42 korral ja neile vastava lihtsustatud silueti indeksi väärtus.

Klastrite arvu ja Davies-Bouldini indeksi väärtus on graafiliselt toodud joonisel 5, optimaalseimaks klastrite arvuks võib pidada seda väärtust, mille korral on meetriku väärtus väikseim, ehk antud juhul 39 klastri korral.



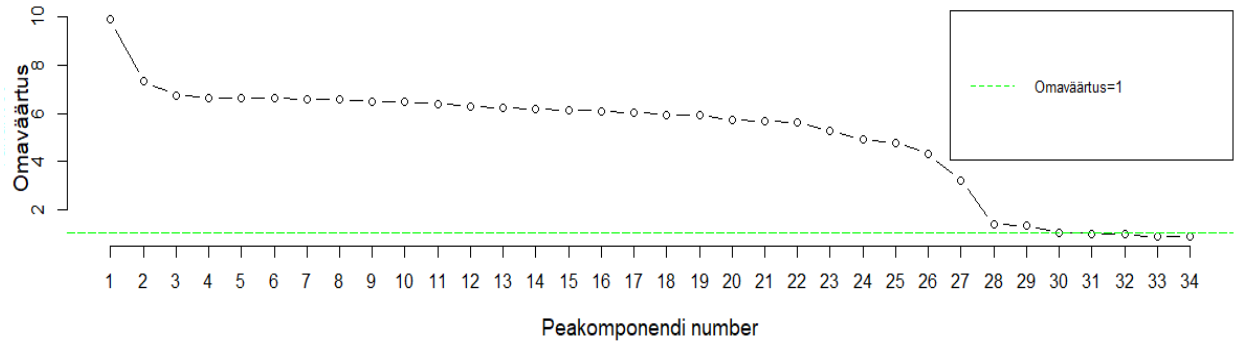
Joonis 5 Davies-Bouldini meetodil hinnatud optimaalne klastrite arv Creditinfo andmestikul. Väikseim väärtus on optimaalsem.

Saadud tulemustest lähtuvalt, jaotati Creditinfo eraisikute andmestik 38 klastriks. Antud andmestiku sisu arvesse võttes võib eksperthinnanguna pidada sellist klastrite arvu sobilikuks.

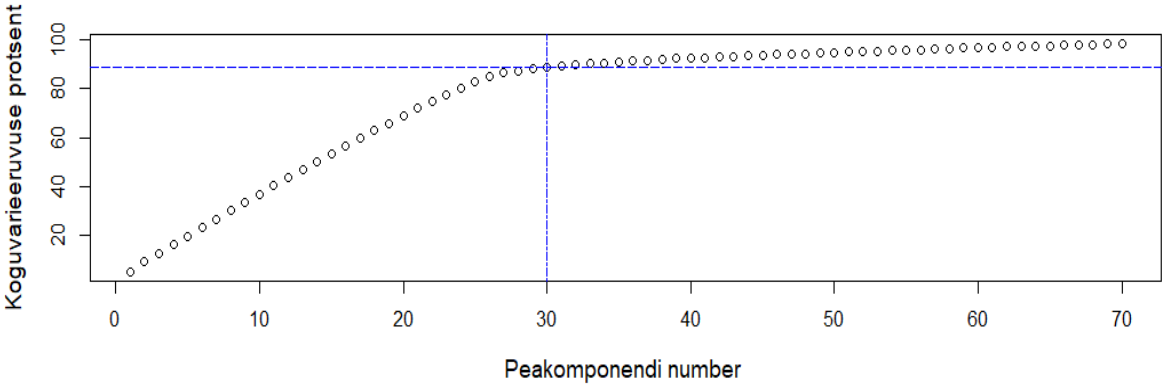
4.2 Creditinfo andmestiku dimensionaalsuse vähendamine peakomponentide analüüsiga ja transformeeritud andmestiku optimaalse klastrite arvu leidmine

Eelnevas peatükis leitud klastrid saadi andmestikul, mille dimensionaalsuse vähendamiseks oli kasutatud korrelatsioonimaatriksit, alternatiivina katsetati leida klastreid andmestikul, kus uued muutujad oli loodud peakomponentide meetodil. Antud meetodit katsetati eeldusel, et peakomponentide analüüsis saadud uute muutujate andmestiku põhjal leitud klastrid kirjeldavad andmeid paremini kui korreleeritud muutujate eelmaldamisel saadud andmestikul loodud klastrid.

Peakomponentide analüüsi eesmärgiks oli moodustada esialgsest 198 muutujast väiksem arv mittekorreleeritud uusi muutujaid, mille peal teostada klasteranalüüsi. Peakomponentide arvu määramisel kasutati Kaiser-Gutmani kriteeriumi, mille korral soovitatakse kasutada peakomponente, millele vastavad omaväärtused on suuremad ühest. Jooniselt 6 selgub, et antud kriteeriumi alusel osutub valituks 30 peakomponenti. Joonisel 7 on toodud kumulatiivne koguvarieeruvuse kirjeldatuse protsent ning nähtub, et 30 peakomponenti kirjeldab ära 89% tunnuste koguvarieeruvusest. Peakomponentide analüüsi tulemusena selgus, et esimene peakomponent kirjeldab 5.2% muutujate koguvarieeruvusest, teine peakomponent 3.8%, kolmas peakomponent 3.5%. Kuna soov oli vähendada andmestiku dimensionaalsust, aga samal ajal võimalikult suur osa andmete koguvarieeruvusest kirjeldada, siis valitud 30 peakomponendi korral vähendati andmestiku suurust 85%, kuid sealjuures andmete koguvarieeruvuse kirjeldatusest vähenes vaid 11%. Seega kasutati klasteranalüüsis esimest 30 peakomponenti.



Joonis 6 Peakomponentide arvu määramine Kaiser-Gutmanni kriteeriumi alusel. Peakomponendid, mille omaväärtus on suurem 1 soovitatakse kasutada.

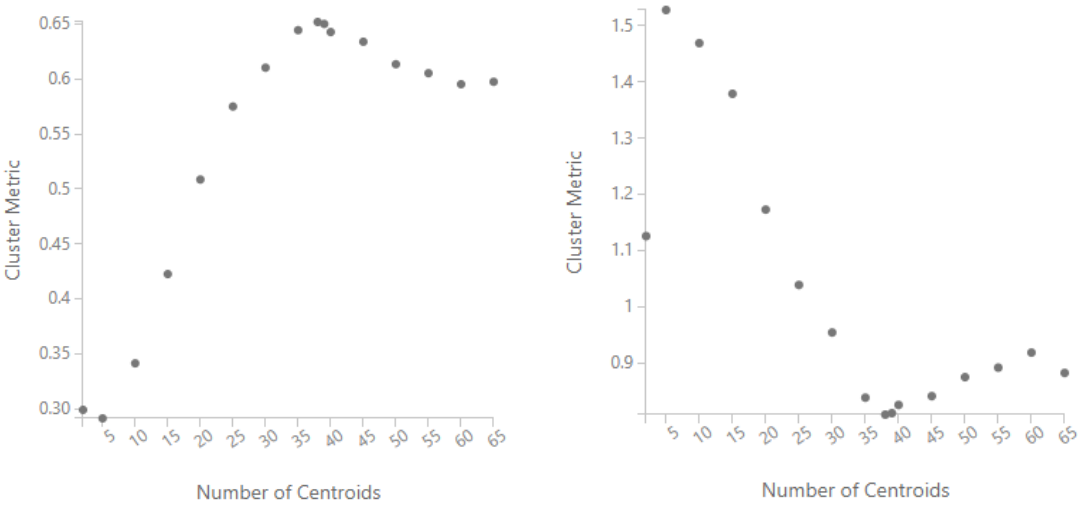


Joonis 7 Kumulatiivne koguvarieeruvuse kirjeldatuse protsent Creditinfo eraisikute andmestikul. 30 peakomponendi üldine kirjeldatuse protsent on 89%.

Sarnaselt eelnevas peatükis kirjeldatule, leiti ka Creditinfo eraisikute andmestikus leitud peakomponentide põhjal optimaalne klastrite arv, kasutades selleks lihtsustatud silueti ja Davies-Bouldini indeksit. Järgnevas tabelis 3 on esitatud kahe indeksi väärtused erinevate klastrite arvu korral, mõlemad indeksid viitavad sellele, et parim klastrite arv on 38. Graafiliselt on mõlema indeksi väärtused näidatud joonisel 8.

Tabel 3 Lihtsustatud silueti ja Davies-Bouldini indeksi väärtused erinevate klastrite arvu korral. Roheliseks värvitud väärtus tähistab parimat väärtust. Lihtsustatud silueti indeksi korral on suurim väärtus optimaalseim, Davies-Bouldini indeksi korral näitab väikseim väärtus optimaalseimat klastrite arvu.

Klastrite arv	Lihtsustatud silueti	Davies-Bouldin
2	0.2989	1.1260
5	0.2911	1.5279
10	0.3413	1.4690
15	0.4225	1.3790
20	0.5085	1.1731
25	0.5750	1.0392
30	0.6103	0.9547
35	0.6445	0.8388
38	**0.6519**	**0.8087**
39	0.6502	0.8113
40	0.6428	0.8259
45	0.6340	0.8415
50	0.6135	0.8753
55	0.6053	0.8922
60	0.5953	0.9189
65	0.5975	0.8828



(a) (b)

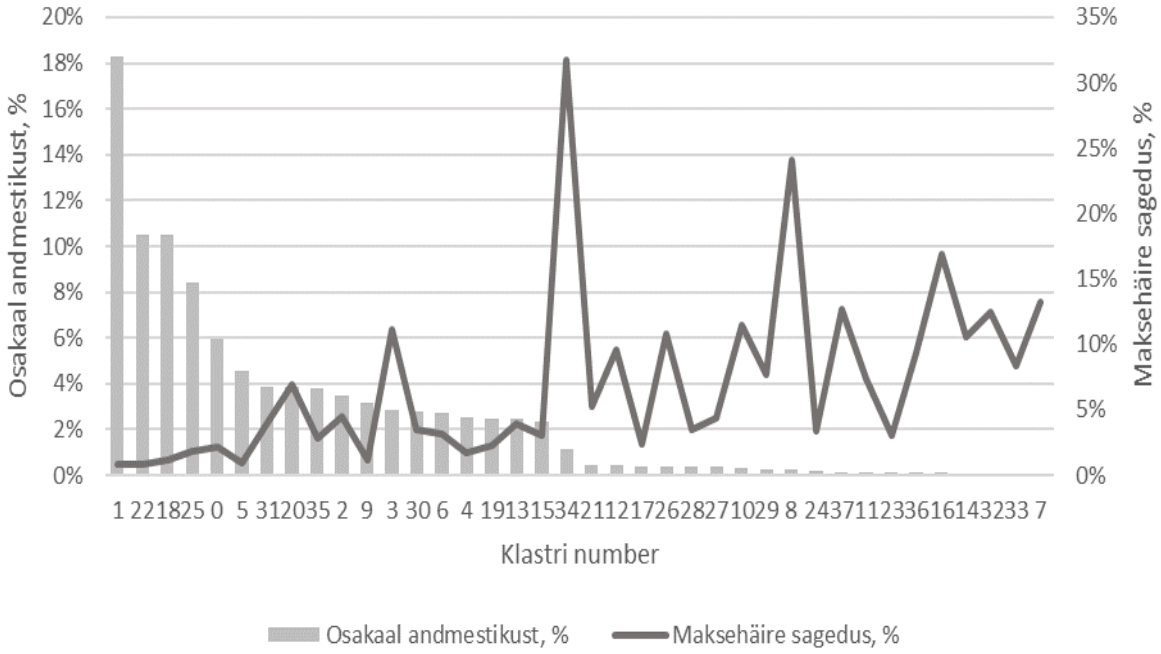
Joonis 8 Lihtsustatud silueti meetodil (a) ja Davies-Bouldini indeksi abil (b) hinnatud optimaalne klastrite arv Creditinfo peakomponentide andmestikul.

4.3 **Klastrite kirjeldus**

Creditinfo andmestiku korral, kus oli 53 eraisiku kohta tehtud päringutega seotud muutujat saadi optimaalseimaks klastrite arvuks 38 klastrit, sama tulemus oli ka peakomponentide andmestiku korral, kus sobivaim oli 38 klastrit. Tekkinud klastrite suurused 53 muutujaga eraisiku andmestiku korral on toodud järgnevas tabelis 4.

Tabel 4 K-keskmiste meetodil saadud klastrite suurused, osakaal andmestikust ja keskmine maksehäire sagedus Creditinfo 53 päringute ajalugu kirjeldava muutujaga andmestikul. Maksehäire sageduse protsent näitab, kui mitmel eraisikul antud klastris tekkis päringute tegemise perioodile järgneva aasta jooksul maksehäire.

Klastri number	Klastri suurus	Osakaal andmestikust, %	Maksehäire sagedus, %
0	46 888	6.0%	2.1%
1	143 938	18.3%	0.8%
2	27 426	3.5%	4.5%
3	22 634	2.9%	11.1%
4	19 920	2.5%	1.7%
5	36 027	4.6%	0.9%
6	21 383	2.7%	3.2%
7	234	0.0%	13.2%
8	1 899	0.2%	24.1%
9	24 850	3.2%	1.1%
10	2 724	0.3%	11.5%
11	1 180	0.2%	7.4%
12	3 325	0.4%	9.5%
13	19 226	2.4%	3.9%
14	603	0.1%	10.6%
15	18 615	2.4%	3.1%
16	852	0.1%	16.9%
17	3 086	0.4%	2.4%
18	82 354	10.5%	1.2%
19	19 564	2.5%	2.3%
20	30 380	3.9%	6.9%
21	3 333	0.4%	5.3%
22	82 737	10.5%	0.8%
23	1 099	0.1%	3.1%
24	1 400	0.2%	3.4%
25	66 044	8.4%	1.8%
26	2 911	0.4%	10.8%
27	2 763	0.4%	4.4%
28	2 881	0.4%	3.5%
29	2 078	0.3%	7.7%
30	22 052	2.8%	3.5%
31	30 416	3.9%	4.1%
32	290	0.0%	12.4%
33	262	0.0%	8.4%
34	9 136	1.2%	31.7%
35	29 666	3.8%	2.8%
36	914	0.1%	9.4%
37	1 195	0.2%	12.6%



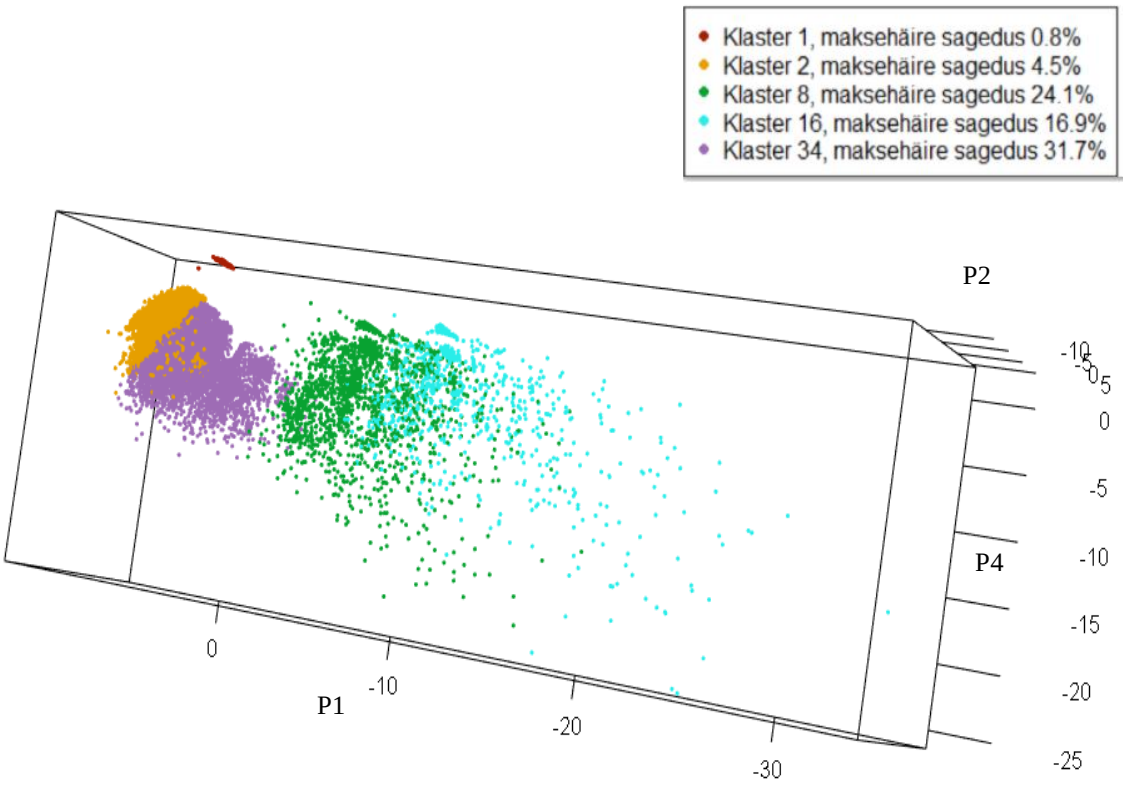
Joonis 9 Creditinfo andmestiku jaotus 38 klastriks, iga klastri protsentuaalne osakaal andmestikust ja keskmine maksehäire sagedus. Andmestiku keskmine makshäire osakaal on 2.9%.

Jooniselt 9 võib välja lugeda, et tekib üks väga suur klaster, kuhu kuulub veidi üle 18% kõigist eraisikutest. Järgnevasse nelja suurde klastrisse on koondunud pea 30% kõigist eraisikutest. Keskmise suurusega klastritesse, kus klastri suurus on umbes 10 000 - 50 000 eraisikut, kuulub ligikaudu 50% kõigist eraisikutest. Alla 3 500 eraisikuga klastreid on kokku 18 ning moodustavad need kokku umbes 2% tervest andmestikust, väikseimas klastris on veidi üle 200 eraisiku.

Arvestades, et tekkis üks väga suur klaster, kus selget ettevõtete gruppi ei tekkinud, kes antud klastri eraisikuid iseloomustaks, katsetati andmestiku jaotamist 39 klastrisse, mis Davies- Bouldini meetodi korral sai veidi parema indeksi väärtuse kui 38 klastri kasutamisel. Andmestiku jaotamisel 39 klastriks selgus, et tulemus ei muutunud, üks suur klaster jäi siiski alles ja pigem väiksemad klastrid jagunesid.

Creditinfo 53 päringupõhise muutujaga andmestiku visualiseerimiseks teostati andmestikul peakomponentide analüüs ning projitseeriti kolmemõõtmelises ruumis kolm peakomponenti. Kolmemõõtmelises ruumis klastrite visualiseerimine aitab paremini märgata mustreid ja näitlikustada tekkinud klastreid.

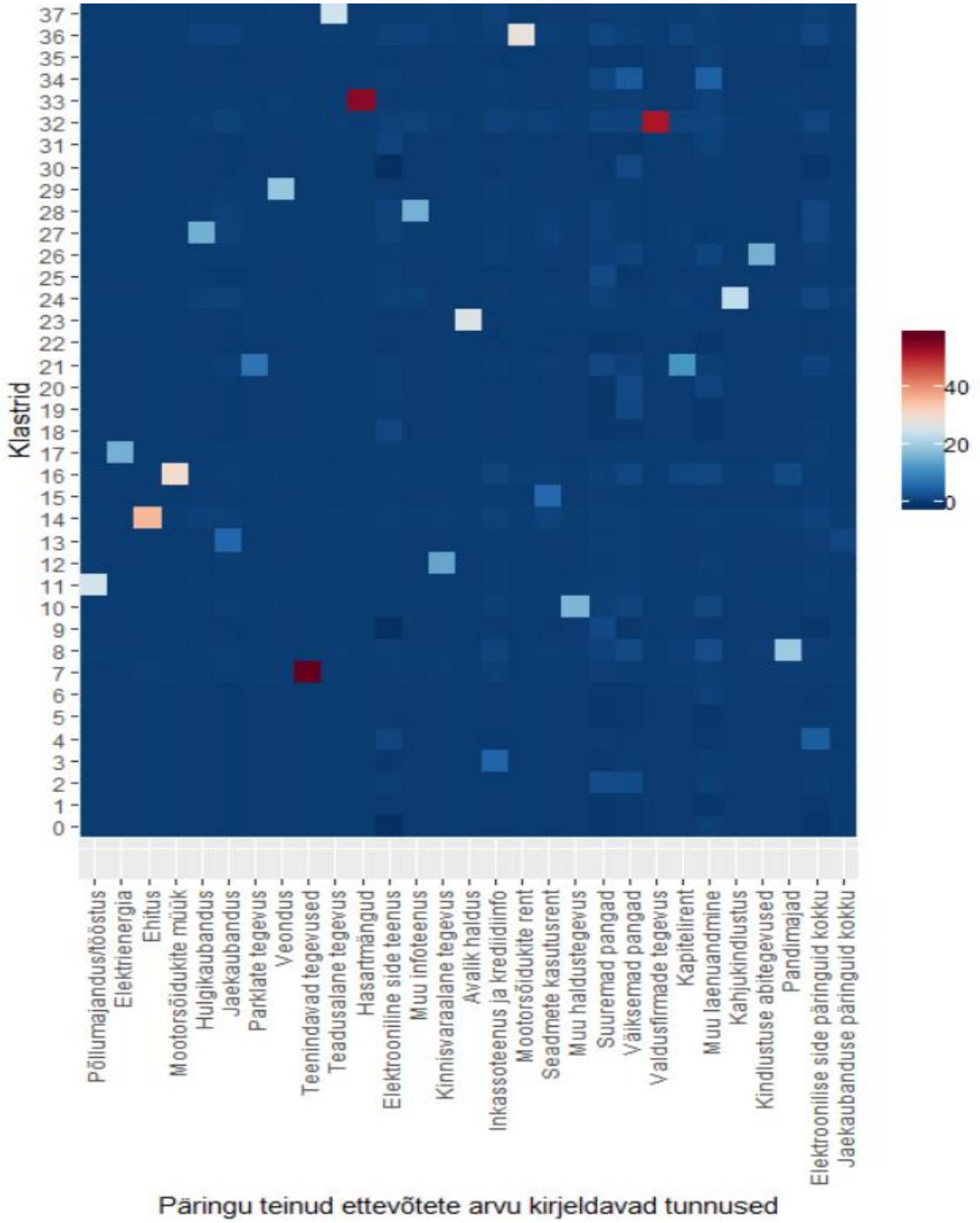
Joonisel 10 on graafiliselt illustreeritud 3-mõõtmelises ruumis Creditinfo andmestikul saadud 38 klastrist viis. Klaster numbriga 1, kuhu kuulus üle 140 000 eraisiku on antud joonisel toodud punasega ning on hästi kompaktne, antud klastri keskmine maksehäire sagedus on ka madalaim, vaid 0.8%. Ka klastrite 2 ja 34 korral on klastrid rohkem koondunud keskmise ümber. Seevastu klastrid 8 ja 16 on hajusad. Vaadates klastrite eraldatust, siis osad on üksteisele lähedal, näiteks klaster 2 ja klaster 34, klaster 1 on teistest väga eraldatud.



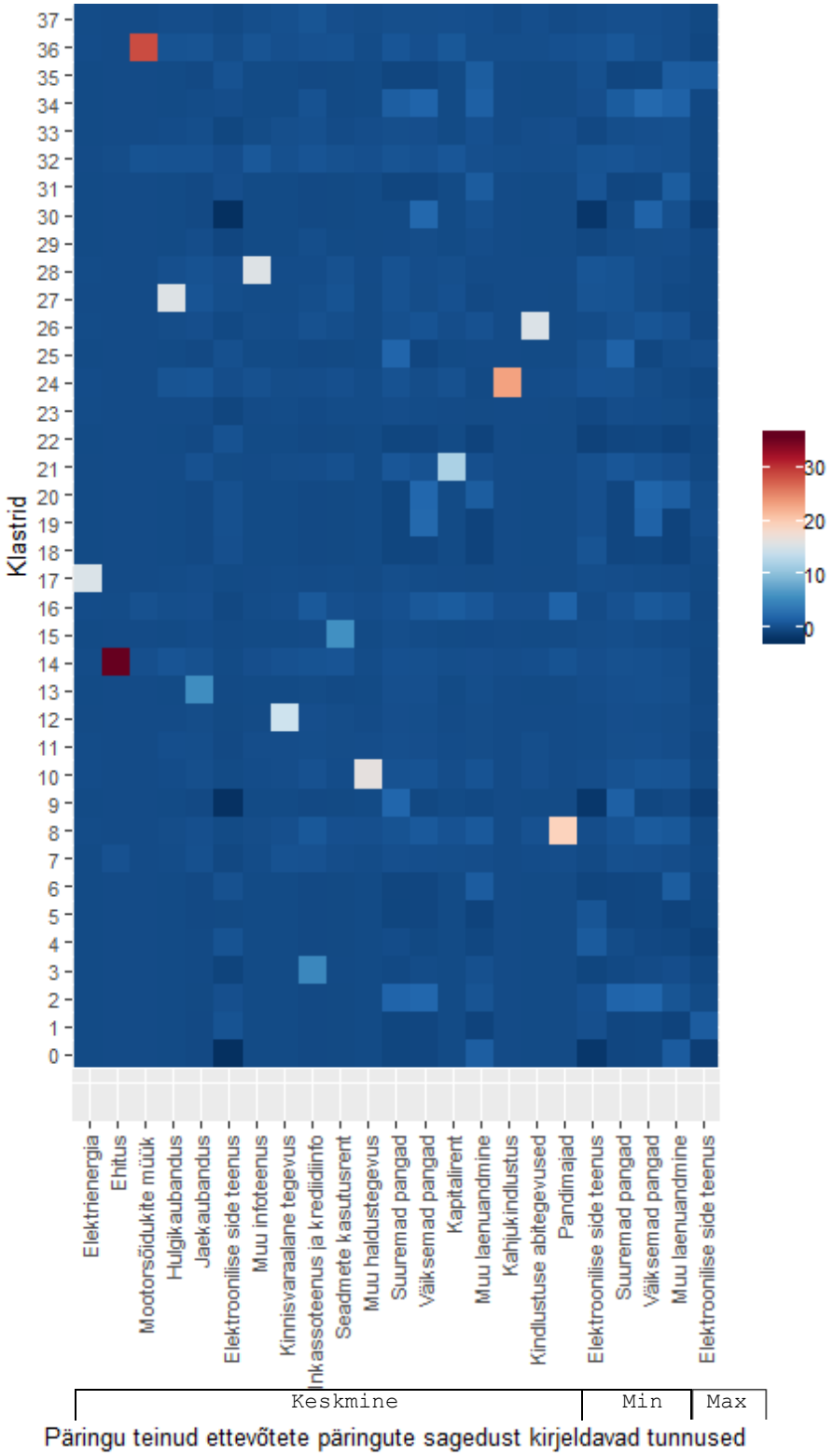
Joonis 10 Creditinfo andmestikul tekkinud klastrite graafiline esitlus esimese, teise ja neljanda peakomponendi kaudu kolmemõõtmelises ruumis. Graafiku selguse mõttes pole toodud kõiki Creditinfo andmestikul leitud 38 klastrit, vaid valitud on 5 klastrit. Samasse klastrisse kuuluvad eraisikud on tähistatud legendis toodud värvidega, lisaks on legendis toodud vastava klastri keskmine maksehäire sagedus.

Lisaks eelnevale jaotati eraisikute andmestik 38 klastrisse ka peakomponentide põhjal, kuna aga maksejõuetuse tõenäosuse hindamisel LHV andmestikul andis paremaid tulemusi siiski esimene variant, siis peakomponentide klasterdamisel saadud klastreid täpsemalt ei kirjeldata.

Klastrite paremaks kirjeldamiseks on kasutatud soojuskaarti (*heatmap*), mis on graafiline kujutis, kus maatriksis sisalduvad väärtused on esitatud värvidena, aitamaks lihtsamini võrrelda väärtusi ja kirjeldada seoseid. Joonis 11 illustreerib ettevõtete gruppide ja klastrite vahelist seost. Antud joonisel toodud maatriks on loodud standardiseeritud andmetel.



Joonis 11 Päringu teinud ettevõtete arv ettevõtete gruppides klastrite lõikes. Legendist võime välja lugeda, et mida tumedam punane on lahter, seda enam erinevaid ettevõtteid on päringu teinud antud ettevõtte grupi poolt vastavasse klastrisse kuuluvate inimeste korral võrreldes teistesse klastritesse kuuluvate inimestega. Sinine lahter tähendab seda, et antud klastrisse kuuluvate eraisikute kohta on antud ettevõtte grupi poolt päringu teinud väike arv ettevõtteid, võrreldes teistesse klastritesse kuuluvate eraisikutega.



Joonis 12 Ettevõtte gruppide päringute sagedused klastrite lõikes. Päringute sagedusi kirjeldatakse keskmise, minimaalse ja maksimaalse päringute tihenduse kaudu. Mida punasem on lahter, seda sagedamini on päringuid tehtud vastava ettevõtte gruppi poolt eraisikute kohta vastavas klastris. Sinine märgib seda, et päringuid on väga harva tehtud võrreldes teiste klastritega.

Joonistelt 11 ja 12 on selgelt näha, et eristuvad grupid, kus ühiseks teguriks on kindla ettevõtete grupi poolt tehtud päringute arv ja sagedus võrreldes teiste klastritega. Välja saab tuua järgmised klastrid: inkasso ja krediidiinfo, teenindavad tegevused (sh ka teadmata tegevusalaga ettevõtted), pandimajad, muu haldustegevus, põllumajandus, kinnisvara, jaekaubandus, ehitus, seadmete kasutusrent, mootorsõidukite müük, elektrienergia, parklate tegevus/kapitalirent, avalik haldus, kahjukindlustus, kindlustuste abitegevus, hulgikaubandus, muu infoteenused, veondus, valdusfirmade tegevus, hasartmängus, mootorsõidukite rent, teadustegevus.

Kokku loetleti eespool 22 klastrit, kus oli selgelt näha klastri moodustanud eraisikute korral peamist päringute teinud ettevõtete gruppi. Suurima klastri korral pole võimalik välja tuua üht kindlat ettevõtete gruppi, kes vastavaid eraisikuid iseloomustaks. Sarnaselt suurima klastriga on veel kaheksa klastrit, kus ettevõtete päringud eraisikute kohta on ühtlaselt jaotunud. Lisaks on seitse sellist klastrit, mille ühiseks jooneks on krediidiasutuste päringud, ehk siis suurte ja väiksemate pankade päringud ja muu laenuandmisega tegelevad ettevõtted.

Tulemustest on näha, et eraisikud jaotuvad Creditinfosse tehtud päringute põhjal rohkem või vähem konkreetselt defineeritud klastritesse. Enamus klastreid (kokku 29) saab väga selgelt defineerida ühe konkreetse päringuid teinud ettevõtete grupi kaudu. Ülejäänud klastrid on ebaspetsiifilisemad ning sinna paigutunud isikute kohta on kas väga vähe päringuid või puudub selge muster.

Edasise töö seisukohalt on eriti oluline välja tuua, et aasta jooksul tehtud päringute mustri põhjal loodud klastrid kirjeldavad üsna hästi inimeste finantskäitumist ja elustiili ning erinevate päringute mustritega eraisikute klastritel on ka olulised erinevused päringute tegemise perioodile järgneva 12 kuu jooksul tekkinud maksehäirete sagedustes. Graafikutelt võib näha ka seda, et sagedasemad ja ebaspetsiifilisemad päringute mustrid seostuvad madalama riskiga ning vähemlevinud päringute mustrid on seotud keskmise või keskmisest kõrgema maksehäire esinemise sagedusega. See kinnitab autori hüpoteesi, et minevikus tehtud päringud võivad aidata ennustada tulevast maksekäitumist ning selle info kasutamine LHV krediidiriski mudelis võib anda oluliselt täpsusema mudeli võrreldes praeguse mudeliga.

4.4 Päringute info kasutamine LHV krediidiriski mudelis logistilise regressiooni mudeli näitel

Logistilise mudeli loomiseks on iga laenusaaja kohta teada järgmised andmed:

1) LHV andmed, mis on kogutud taotluse kuupäeval, koosnevad sotsiaal-demograafilistest ja finantsandmetest, laenuspetsiifilistest muutujatest, lisaks maksekäitumise andmed, mis seotud maksuvõlgade ja –häiretega ning pärinevad krediidibüroodest ja panga enda ajaloolistel andmetel ;
2) Taotlusele eelneva aasta päringute info Creditinfo andmebaasist, mille põhjal on arvutatud ajaloolised päringupõhised muutujad. Päringupõhiste muutujate loomine on täpsemalt kirjeldatud alapeatükis 3.3;

3) Klastrite maksehäire sagedus meetod 1 ja meetod 2 korral. Iga LHV laenusaaja on taotlusele eelneva aasta päringute põhjal arvutatud muutujate põhjal määratud eelnevalt defineeritud klastritesse (klastrid, mis on leitud Creditinfo 53 päringupõhise muutujaga eraisikute andmestikul (edaspidi meetod 1) ja klastrid, mis on leitud Creditinfo peakomponentide andmestikul (edaspidi meetod 2)). Seega on igale laenusaajale leitud 2 erineva metoodikaga saadud klastri maksehäire sagedused.

Kasutades ettepoole sammregressiooni loodi 5 erinevat mudelit. Esimese mudeli loomisel kasutati ainult LHV muutujaid. Teise mudeli korral kasutati lisaks klastrite maksehäire sagedust, kus klastrid on leitud meetod 1 põhjal. Kolmanda mudeli puhul kaasati regressioonanalüüsi klastrite maksehäire sagedus, kus klastrid on defineeritud meetod 2 põhjal. Neljanda mudeli loomisel kasutati LHV muutujatele lisaks ka ajaloolisi päringupõhiseid muutujaid, lõplikku mudelisse jäid vaid olulised muutujad. Viienda mudeli korral kasutati lisaks päringupõhistel andmetel leitud peakomponente.

Eelpool kirjeldatud mudelite võrdlemiseks vaadatakse tõeselt positiivsete määra ja AUCi väärtust, piirmäärana (*cut-off*) kasutatakse väärtust 0.05, mis vastab üldjoontes LHV-s antud andmestikus olevate krediiditoodete piirmäärale. Tabelis 5 on toodud logistilise regressiooni mudeli rakendamisel testandmestikul saadud tulemused . Tulemustest on näha, et ajalooliste päringute informatsiooniga seotud muutujate kasutamisel mudeli AUC väärtus ja ka tõselt positiivsete määr paranevad.

Tabel 5 LHV andmete, päringupõhiste muutujate ja klastrite lisamisel loodud mudelite headuse näitajate tulemused

Mudel	**Mudeli kirjeldus**	**Tõeselt positiivsete määr**	**AUC**
Mudel1	LHV muutujad	0.151	0.751
Mudel2	LHV muutujad + klastri maksehäire sagedus (meetod 1)	0.278	0.802
Mudel3	LHV muutujad + klastri maksehäire sagedus (meetod 2)	0.278	0.793
Mudel4	LHV muutujad + ajaloolised päringupõhised muutujad	0.412	0.832
Mudel5	LHV muutujad + peakomponendid	0.388	0.820

Mudeli 1 korral on nii tõeselt positiivsete määr kui ka AUC näitaja kõige kehvem, vastavalt 0.151 ja 0.751, mida võis eeldada. Klastrite maksehäire sageduste lisamine tõstab oluliselt AUCi väärtust 0.802, seega mudeli täpsuse paranemine on märgatav. Ka tõeselt positiivsete määr on antud mudelis oluliselt kõrgem kui vaid LHV muutujaid sisaldavas mudelis. Klastrite maksehäire sageduse muutujad parandavad mudeli täpsust suhteliselt võrdselt,

veidi parema tulemuse AUC näitaja mõttes annab mudel 2 võrreldes mudeliga 3. Oluliselt täpsem mudel saadakse, kui kaasatakse mudeli loomisesse peakomponendid, mis on päringupõhistest muutujatest saadud. Tõeselt positiivsete määr on tõusnud 2 protsendipunkti võrra võrreldes esialgse mudeliga. Kõige täpsem mudel maksejõuetuse tõenäosuse hindamiseks sisaldab endas nii LHV muutujaid kui ka päringupõhiseid muutujaid. Antud mudeli korral on AUCi väärtus oluliselt parem kui esialgse mudeli korral, vastavaks tulemuseks on 0.832. Tõeselt positiivsete määr on 0.412.

Seega saame järeldada, et päringute informatsiooni klastrite lisamine parandab oluliselt mudeli täpsust, aga esialgsete päringupõhiste muutujate lisamine otse mudelisse annab isegi paremaid tulemusi eraisikute krediidiriski hindamisel.

5. Järeldused

Analüüsi käigus pandi palju rõhku eraisikute klastritesse jaotamisele kasutades selleks eraisikute kohta tehtud päringute informatsiooni andmeid. Saadud klastritel ning nende keskmiste maksehäirete sageduste infol on potentsiaali aidata krediidiasutustel ja teistel eraisikute krediidivõimelisust hindavatel ettevõtetel tuvastada täpsemalt klientide maksejõuetust. Lisaks on antud klastrid kasulikud uurimaks inimeste käitumis- ja tarbimismustreid ning maksekäitumise harjumusi.

Magistritöös võrreldi viite erinevat logistilise regressiooni mudelit, millest selgub, et ajalooliste päringute informatsiooni lisamine parandab oluliselt mudeli täpsust. Huvitaval kombel on ajalooliste päringute informatsiooni sisaldavate muutujatega mudeli täpsust parem kui mudelil, milles on kasutatud seletava tunnusena klastrite maksehäire sagedust. Võib spekuleerida, et LHV andmestikul toimisid transformeerimata muutujaid paremini kuna andmestik oli suur (70 000 laenutaotlust) ning populatsioonis esineb oluline kallutatus üldisest populatsioonist (mida esindab Creditinfo andmestik). Kallutatuse on siinkohal põhjustanud eelnev laenutaotluste krediidivõimelisuse hindamise protsess LHV-s.

Kui võrrelda ühe päringupõhise muutuja või klastri maksehäire sageduse lisamist mudelisse, siis kindlasti annab viimane parema tulemuse, kuna sarnased grupid on leitud terve andmestiku objektide vaheliste sarnasuste abil ning lisaks on igale klastrile leitud maksehäire osakaal, mis annab rohkem konteksti populatsiooni käitumise kohta. Samas tasub arvestada, et päringute põhjal saadud klastreid on katsetatud LHV panga krediidiriski mudelis ja võib anda mõne teise laenuandja mudeli korral erinevaid tulemusi. Võib spekuleerida, et klastritunnuse kasutamine päringupõhiste muutujate asemel toimib paremini mõne väiksema laenuandja mudelis, kellel on vähem andmeid ning populatsiooni väiksem kallutatus.

Klasterdamise poole pealt tasub tulevikus arvestada ka uute muutujate lisamise võimalust. Potentsiaalsed muutujad võivad olla näiteks vanus ja sugu. Lisaks võib mõelda ka ettevõtete grupeerimisel mõned grupid jätta eraldamata või tekitada lisa tasemeid juurde. Näiteks väiksemad klastrid, kus on ainult mõnisada inimest ja kelle profiili iseloomustab mingi spetsiifilise ettevõtete grupi päringud, aga hilisema maksehäire esinemise sagedus on väga sarnane populatsiooni keskmisele, ei lisa mingit uut lisainformatsiooni.

Lõpetuseks võib autor töö tulemustele tuginedes väita, et eraisiku kohta tehtud päringute info on väga väärtuslik krediidiasutustele maksevõimelisuse hindamiseks. Pärast käesolevas töös kirjeldatud muutujate ja klasteranalüüsi implementeerimist Creditinfo poolt saavad LHV ning teised Eestis tegutsevad krediidiasutused ligipääsu täiesti uudsele infoallikale.

6. Kokkuvõte

Käesolev magistritöö uurib eraisikute maksehäirete kohta tehtud päringute informatsiooni abil eraisikute gruppidesse jaotamist ja kas saadud gruppides eristuvad maksekäitumise harjumused. Teise olulise eesmärgina soovitakse uurida, kas laenutaotlusele eelneva aasta jooksul tehtud päringud eraisiku kohta aitavad prognoosida kliendi maksejõuetust järgneva 12 kuu jooksul. Antud töös püstitatud eesmärkide saavutamiseks kasutati Creditinfo ettevõtete ja eraisikute andmestikku inimeste klastritesse jagamiseks ning logistiline regressiooni mudeli prognoosimaks kliendi maksejõuetust loodi LHV laenusaajate andmestikul.

Esimeseks etapiks oli eraisikute kohta päringu teinud ettevõtete grupeerimine, et luua mõistlik arv ettevõtete tasemeid, mille alusel leida päringu teinud ettevõtete arvu ja päringute sagedust, lisaks oleks ainult ühe ettevõtte põhised muutujad ajas väga ebastabiilsed. Kasutades Creditinfosse päringu teinud ettevõtete andmestikku klassifitseeriti ettevõtted EMTAK koodi ja aastakäibe alusel 27 gruppi. Saadud ettevõtete gruppide alusel jaotati eraisiku kohta teinud ettevõtted ka eraisikute andmestikus.

Antud töö üheks eesmärgiks oli klasterdada eraisikud minevikus tehtud päringute alusel, mis peegeldaks nende finantskäitumist ja harjumusmustreid. Tulemused näitasid, et päringute põhjal eristuvad grupid üldjuhul väga selgesti, sealjuures klastritel olid väga erinevad järgneva 12 kuu maksehäire esinemise sagedused. Analüüsi ja tulemuste aluseks olevad eraisikute kohta tehtud päringute andmed pärinesid Creditinfo andmebaasist ja vaatlusperioodiks oli 30.09.2016 kuni 30.09.2017 tehtud päringud. Antud andmestik koosnes ca 800 000 erisiku anonümiseeritud isikukoodist, päringute kuupäevadest ja anonümiseeritud ettevõtete registrikoodidest. Selleks, et antud andmestik oleks rakendatav statistiliste meetodite jaoks loodi kokku 198 päringutepõhist muutujat – ettevõtete arvu, päringute arvu ja päringute sagedust sisaldavad muutujad. Lähtuvalt andmestiku suurusest ja muutujate arvust vähendati edasise analüüsi jaoks andmestiku dimensionaalsust. Selleks kasutati kahte lähenemist – korrelatsioonimaatriksit ja peakomponentide meetodit. Saadud tulemuste peal rakendati klasteranalüüsi.

Klastrite leidmiseks kasutati k-keskmiste meetodit. Klastrite täpsus sõltub suuresti sellest, kui hästi on klastrite arv valitud. Klastrite arvu valik pakkus töö käigus korralikku väljakutset, lisaks sellele muutis andmemahu suurus arvutused aeganõudvaks ja osade tarkvarade korral võimatuks. Katsetati erineva arvu klastritega vahemikus 2 kuni 65. Sobiva arvu klastrite valimiseks kasutati lihtsustatud silueti ja Davies-Bouldini indeksit. Lõpptulemusena jaotati eraisikute andmestik 38 klastrisse.

Peamiseks motivatsiooniks päringute informatsiooni töötlemisel ja klasterdamisel oli saadud tulemuste kasutamine LHV krediidiriski mudelis, et näidata, kas laenutaotlejat on võimalik hinnata mineviku päringute põhjal. Seega teiseks eesmärgiks oli näidata, et laenutaotlusele eelneva aasta jooksul tehtud päringute põhjal on võimalik ennustada taotlusele järgneva aasta maksekäitumist. Kuna saadud klastrite puhul olid gruppidel erinevad maksehäire sagedused, andis see lootust, et antud tulemuste kaasamine maksejõuetuse tõenäosuse hindamisel võib mudeli täpsust parandada. Erinevate Creditinfosse tehtud päringute ajaloo põhjal loodud muutujate ja LHV andmete kombinatsioonidel saadi lootustandvaid tulemusi.

Esialgses mudelis, mis sisaldas vaid LHV muutujaid oli mudeli AUC näitaja väärtus 0.751. Klastrite maksehäire sageduse kaasamine mudelisse parandas antud väärtust, andes tulemuseks 0.802. Üllatavalt hästi töötasid krediidiriski mudelis aga esialgsed ajalooliste

päringute põhjal loodud muutujad, mille korral AUC oli 0.832. Sama tendents oli ka tõeselt positiivsete määra korral. Kui esialgse mudeli korral oli tõeselt positiivsete määr 0.15, klastri lisamisel 0.278 ja päringupõhiste muutujate lisamisel 0.412. Et seda näitlikustada taotlejate arvuga, siis 70 000 laenutaotluse korral suudab esimene mudel 147 maksejõuetut eraisikut õigesti klassifitseerida, teise mudeli korral, kus ka klastrite maksehäire sagedus on sees, suudab 270 ja kolmanda mudeli korral, kus päringute informatsioon on vastavaks suuruseks 400 laenutaotlejat, erinevused on märkimisväärsed. Nagu selgus töö analüüsitulemustest, siis ajalooliste päringute info kasutamine LHV krediidiriski mudelis suurendas mudeli täpsust märgatavalt.

7. Viidatud kirjandus

[1] Han, J., Kamber, M., Pei, J. (2012) *Data Mining: Concepts and Techniques, Third Edition.* MorganKaufmann Publishers.

[2] Capo, M., Perez, A., Lozano, J. A. An efficient K-means algorithm for Massive Data. (2016) *Knowledge-Based Systems 117*, pp. 56-69. doi:10.1016/j.knosys.2016.06.031

[3] James, G., Witten, D., Hastie, T., , Tibshirani, R. (2013) *An Introduction to statistical Learning: with Application R.* Springer.

[4] Kassambara, A. (2017) *Practical Guide To Cluster Analysis in R: Unsupervised Machine Learning.* STHDA.

[5] Arthur, D., Vassilvitskii, S. k-means++: The Advatages of Careful Seeding. (2017) *Society for Industrial and Applied Mathematics Philadelphia*, pp. 1027–1035.

[6] Amorim, R.C., Hennig, C. Recovering the number of clusters in data sets with noise features using feature rescaling factors. (2015) *Information Sciences*, 324, pp. 126-145. doi:10.1016/j.ins.2015.06.039

[7] Arbelaitz,O., Gurrutxaga, I., Muguerza, J., Perez, J. M., Perona, I. An extensive comparative study of cluster validity indices. (2013) *Pattern Recognition*, vol. 46, no. 1, pp. 243–256.

[8] Wang, F., Penya, H., Kelleher, J.D., Pugh, J., Ross, R. An Analysis of the Application of SimplifiedSilhouette to the Evaluation of k-means Clustering Validity. (2017) *Machine Learning and Data Mining in Pattern Recognition. 13th International Conference, MLDM 2017*, vol. 10358, pp. 291-305. doi:10.1007/978-3-319-62416-7_21

[9] Davies, D.L., Bouldin, D.W. A Cluster Separation Measure. (1979) *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.1, no.2, pp. 224–227. doi:10.1109/TPAMI.1979.4766909

[10] Al-Anazi, S., AlMahmoud, H., Al-Turaiki, I. Finding similar documents using different clustering techniques. (2016) *Procedia Computer Science* 82, pp. 28-34. doi:10.1016/j.procs.2016.04.00

[11] Traat, I. (2013) M*itmemõõtmeline analüüs.* Loengukonspekt. Tartu Ülikool, matemaatilise statistika instituut.

[12] Käärik, E. (2013) *Andmeanalüüs II*. Loengukonspekt. Tartu Ülikool, matemaatilise statistika instituut.

[13] Hosmer, D.W., Lemeshow, S. (2000) *Applied Logistic Regression, Second Edition*. John Wiley & Sons, Inc.

[14] Creditinfo blogi. Creditinfo koduleheküIg. https://blog.creditinfo.ee/krediidinfost/ (19.04.2019)

[15] Introducing JSON. https://www.json.org/(19.04.2019)

[16] LHV koduleheküIg. https://www.lhv.ee/et/ettevottest (19.04.2019)

[17] Zheng, A., Casari, A. (2018) *Feature Engineering for Machine Learning: Principles and Techniques for Data Scientists.* O'Reilly Media, Inc.

[18] Sambandam, R. Cluster Analysis Gets Complicated. (2003) *Marketing Research*, vol. 15, no. 1.

[19] Kaiser, H.F. The application of electronic computers to factor analysis. (1960) *Educational and Psychological Measurement*, 20, pp. 141-151.

[20] Akaike, H. A new look at statistical model identification. (1974) *IEEE Transactions on Automatic Control* AU–19, pp. 716–722.

[21] What is Azure Machine Learning Studio? (2017) https://docs.microsoft.com/en-us/azure/machine-learning/service/overview-what-is-azure-ml (22.04.2019)

[22] K-Means Clustering. (2019) https://docs.microsoft.com/en-us/azure/machine-learning/studio-module-reference/k-means-clustering (22.04.2019)

Lisad

Lisa 1

```
#leiame ettevõtte koodile vastava ettevõtete grupi

def comp_clust_Obj(obj):
    try:
        global companies
        clust_dict=dict()
        obj = ast.literal_eval(obj)
        unique_clusters = set()
        for key, value in obj.items():
            try:
                cluster_id = companies.loc[key, 'company_cluster']
                unique_clusters.add(cluster_id)
            except KeyError:
                cluster_id = 9
                unique_clusters.add(cluster_id)
        unique_clusters = list(unique_clusters)
        for cluster in unique_clusters:
            clust_dict[cluster] = {}
        for key, value in obj.items():
            cluster_id = companies.loc[key, 'company_cluster']
            clust_dict[cluster_id][key] = value
        return clust_dict
    except:
        return None

# iga ettevõtete grupi kohta, mitu unikaalset ettevõtet on päringuid teinud

def entitiesPerClusterQueried(data):
    try:
        obj = data['reg_code']
        obj = comp_clust_Obj(obj)
        clusters = obj.keys()
        data['object_parsed'] = obj
        for cluster in clusters:
            no_of_companies = len(obj[cluster])
```

```
            colname = 'no_of_companies_queried_in_cluster_' + str(clus-
ter)
            data[colname] = no_of_companies
        return data
    except:
        return data

# iga ettevõtete grupi kohta, mitu päringut on tehtud ettevõtete poolt

def queriesPerCluster(data):
    try:
        obj = data['reg_code']
        obj = comp_clust_Obj(obj)
        clusters = obj.keys()
        #data['object_parsed'] = obj
        for cluster in clusters:
            cluster_data = obj[cluster]
            unique_queries = 0
            for key, value in cluster_data.items():
                unique_dates = set()
                for query_time in value[0]:
                    query_date = pd.to_datetime(query_time[0]).date()
                    unique_dates.add(query_date)
                unique_dates = list(unique_dates)
                unique_dates_count = len(unique_dates)
                unique_queries += unique_dates_count
            colname = 'no_of_unique_queries_in_cluster_' + str(cluster)
            data[colname] = unique_queries
        return data
    except:
        return data

# Ettevõtte päringute listi kohta, jäta alles ainult unikaalsed kuupäevad

def getUniqueDatesPerCompany(obj):
    unique_dates = set()
    for query_time in obj:
        query_time = query_time[0]
        query_date = pd.to_datetime(query_time).date()
        unique_dates.add(query_date)
```

```
    unique_dates = list(unique_dates)
    #result = [unique_dates]
    return unique_dates


# Keskmine päevade arv päringute vahel, päevade arv möödas hiliseimast
#päringust, päevade arv möödas varaseimast päringust, minimaalne päevade
#arv päringute vahel, maksimaalne päevade arv päringute vahel,
def avgTimeBetweenQueries(data):
    dataset_date = datetime(2017, 9, 1)
    obj = data['reg_code']
    obj = comp_clust_Obj(obj)
    try:
        for cluster, cluster_data in obj.items():
            days_between_queries = []
            query_times = set()
            for company, company_data in cluster_data.items():
                for date in getUniqueDatesPerCompany(company_data[0]):
                    query_times.add(date)
            query_times = list(query_times)
            query_times.sort(reverse=True)
            min_of_days_from_query_date    =    dataset_date.date()    -
query_times[0]
            min_of_days_from_query_date                                  =
min_of_days_from_query_date.days
            max_of_days_from_query_date    =    dataset_date.date()    -
query_times[-1]
            max_of_days_from_query_date                                  =
max_of_days_from_query_date.days
            for i in range(0, len(query_times)):
                try:
                    days_diff = query_times[i] - query_times[i+1]
                    days_between_queries.append(days_diff.days)
                except IndexError:
                    pass
            try:
                avg_days_between_queries     =     sum(days_between_que-
ries)/len(days_between_queries)
            except ZeroDivisionError:
                avg_days_between_queries = 0
            try:
                max_days_between_queries = max(days_between_queries)
                min_days_between_queries = min(days_between_queries)
            except ValueError:
```

```
                try:
                    max_days_between_queries = days_between_queries[0]
                    min_days_between_queries = days_between_queries[0]
                except IndexError:
                    max_days_between_queries = 0
                    min_days_between_queries = 0
            colname_1 = (f"cluster_{cluster}_avg_days_between_queries")
            colname_2                =                (f"cluster_{clus-
ter}_min_of_days_from_query_date")
            colname_3                =                (f"cluster_{clus-
ter}_max_of_days_from_query_date")
            colname_4 = (f"cluster_{cluster}_max_days_between_queries")
            colname_5 = (f"cluster_{cluster}_min_days_between_queries")
            data[colname_1] = avg_days_between_queries
            data[colname_2] = min_of_days_from_query_date
            data[colname_3] = max_of_days_from_query_date
            data[colname_4] = max_days_between_queries
            data[colname_5] = min_days_between_queries
    except:
        print(data)
    return data
```

Lisa 2

Muutuja nimetus	Kirjeldus	Muutuja kasutamine klasteranalüüsis
no_of_companies_queried_in_cluster_1	Esimese ploki muutuja, unikaalsete ettevõtete arv vastavas ettevõtte grupis	JAH
no_of_companies_queried_in_cluster_10		JAH
no_of_companies_queried_in_cluster_11		JAH
no_of_companies_queried_in_cluster_12		JAH
no_of_companies_queried_in_cluster_13		JAH
no_of_companies_queried_in_cluster_14		JAH
no_of_companies_queried_in_cluster_15		JAH
no_of_companies_queried_in_cluster_16		JAH
no_of_companies_queried_in_cluster_17		JAH
no_of_companies_queried_in_cluster_18		JAH
no_of_companies_queried_in_cluster_19		JAH
no_of_companies_queried_in_cluster_2		JAH
no_of_companies_queried_in_cluster_20		JAH
no_of_companies_queried_in_cluster_21		JAH
no_of_companies_queried_in_cluster_22		JAH
no_of_companies_queried_in_cluster_23		JAH
no_of_companies_queried_in_cluster_24		JAH
no_of_companies_queried_in_cluster_25		JAH
no_of_companies_queried_in_cluster_26		JAH
no_of_companies_queried_in_cluster_27		JAH
no_of_companies_queried_in_cluster_3		JAH
no_of_companies_queried_in_cluster_4		JAH
no_of_companies_queried_in_cluster_5		JAH
no_of_companies_queried_in_cluster_6		JAH
no_of_companies_queried_in_cluster_7		JAH
no_of_companies_queried_in_cluster_8		JAH
no_of_companies_queried_in_cluster_9		JAH
no_of_unique_queries_in_cluster_1	Teise ploki muutuja, päringute arv vastavas ettevõtte grupis	EI
no_of_unique_queries_in_cluster_10		EI
no_of_unique_queries_in_cluster_11		EI
no_of_unique_queries_in_cluster_12		JAH
no_of_unique_queries_in_cluster_13		EI
no_of_unique_queries_in_cluster_14		EI
no_of_unique_queries_in_cluster_15		EI
no_of_unique_queries_in_cluster_16		EI
no_of_unique_queries_in_cluster_17		EI
no_of_unique_queries_in_cluster_18		EI
no_of_unique_queries_in_cluster_19		EI
no_of_unique_queries_in_cluster_2		EI
no_of_unique_queries_in_cluster_20		EI
no_of_unique_queries_in_cluster_21		EI
no_of_unique_queries_in_cluster_22		EI
no_of_unique_queries_in_cluster_23		EI
no_of_unique_queries_in_cluster_24		EI
no_of_unique_queries_in_cluster_25		EI
no_of_unique_queries_in_cluster_26		EI
no_of_unique_queries_in_cluster_27		EI
no_of_unique_queries_in_cluster_3		EI
no_of_unique_queries_in_cluster_4		EI
no_of_unique_queries_in_cluster_5		EI
no_of_unique_queries_in_cluster_6		JAH

no_of_unique_queries_in_cluster_7		EI
no_of_unique_queries_in_cluster_8		EI
no_of_unique_queries_in_cluster_9		EI
cluster_10_avg_days_between_queries	Kolmanda ploki muutujad. 1)keskmine päevade arv päringute vahel vastavas ettevõtte grupis; 2)maksimaalne päevad arv päringute vahel vastavas ettevõtte grupis; 3)päevade arv, mis möödas varaseimast päringust vastavas ettevõtte grupis; 4)minimaalne päevade arv päringute vahel vastavas ettevõtte grupis; 5)päevade arv, mis möödas hiliseimast päringust vastavas ettevõtte grupis	EI
cluster_10_max_days_between_queries		EI
cluster_10_max_of_days_from_query_date		EI
cluster_10_min_days_between_queries		EI
cluster_10_min_of_days_from_query_date		EI
cluster_11_avg_days_between_queries		EI
cluster_11_max_days_between_queries		EI
cluster_11_max_of_days_from_query_date		EI
cluster_11_min_days_between_queries		EI
cluster_11_min_of_days_from_query_date		EI
cluster_12_avg_days_between_queries		JAH
cluster_12_max_days_between_queries		EI
cluster_12_max_of_days_from_query_date		JAH
cluster_12_min_days_between_queries		EI
cluster_12_min_of_days_from_query_date		JAH
cluster_13_avg_days_between_queries		JAH
cluster_13_max_days_between_queries		EI
cluster_13_max_of_days_from_query_date		EI
cluster_13_min_days_between_queries		EI
cluster_13_min_of_days_from_query_date		EI
cluster_14_avg_days_between_queries		JAH
cluster_14_max_days_between_queries		EI
cluster_14_max_of_days_from_query_date		EI
cluster_14_min_days_between_queries		EI
cluster_14_min_of_days_from_query_date		EI
cluster_15_avg_days_between_queries		EI
cluster_15_max_days_between_queries		EI
cluster_15_max_of_days_from_query_date		EI
cluster_15_min_days_between_queries		EI
cluster_15_min_of_days_from_query_date		EI
cluster_16_avg_days_between_queries		JAH
cluster_16_max_days_between_queries		EI
cluster_16_max_of_days_from_query_date		EI
cluster_16_min_days_between_queries		EI
cluster_16_min_of_days_from_query_date		EI
cluster_17_avg_days_between_queries		JAH
cluster_17_max_days_between_queries		EI
cluster_17_max_of_days_from_query_date		EI
cluster_17_min_days_between_queries		EI
cluster_17_min_of_days_from_query_date		EI
cluster_18_avg_days_between_queries		JAH
cluster_18_max_days_between_queries		EI
cluster_18_max_of_days_from_query_date		EI
cluster_18_min_days_between_queries		EI
cluster_18_min_of_days_from_query_date		EI
cluster_19_avg_days_between_queries		JAH
cluster_19_max_days_between_queries		EI
cluster_19_max_of_days_from_query_date		EI
cluster_19_min_days_between_queries		EI
cluster_19_min_of_days_from_query_date		EI
cluster_1_avg_days_between_queries		EI
cluster_1_max_days_between_queries		EI
cluster_1_max_of_days_from_query_date		EI
cluster_1_min_days_between_queries		EI
cluster_1_min_of_days_from_query_date		EI

cluster_20_avg_days_between_queries		JAH
cluster_20_max_days_between_queries		EI
cluster_20_max_of_days_from_query_date		EI
cluster_20_min_days_between_queries		EI
cluster_20_min_of_days_from_query_date		JAH
cluster_21_avg_days_between_queries		JAH
cluster_21_max_days_between_queries		EI
cluster_21_max_of_days_from_query_date		EI
cluster_21_min_days_between_queries		EI
cluster_21_min_of_days_from_query_date		JAH
cluster_22_avg_days_between_queries		EI
cluster_22_max_days_between_queries		EI
cluster_22_max_of_days_from_query_date		EI
cluster_22_min_days_between_queries		EI
cluster_22_min_of_days_from_query_date		EI
cluster_23_avg_days_between_queries		JAH
cluster_23_max_days_between_queries		EI
cluster_23_max_of_days_from_query_date		EI
cluster_23_min_days_between_queries		EI
cluster_23_min_of_days_from_query_date		EI
cluster_24_avg_days_between_queries		JAH
cluster_24_max_days_between_queries		EI
cluster_24_max_of_days_from_query_date		EI
cluster_24_min_days_between_queries		EI
cluster_24_min_of_days_from_query_date		JAH
cluster_25_avg_days_between_queries		JAH
cluster_25_max_days_between_queries		EI
cluster_25_max_of_days_from_query_date		EI
cluster_25_min_days_between_queries		EI
cluster_25_min_of_days_from_query_date		EI
cluster_26_avg_days_between_queries		JAH
cluster_26_max_days_between_queries		EI
cluster_26_max_of_days_from_query_date		EI
cluster_26_min_days_between_queries		EI
cluster_26_min_of_days_from_query_date		EI
cluster_27_avg_days_between_queries		JAH
cluster_27_max_days_between_queries		EI
cluster_27_max_of_days_from_query_date		EI
cluster_27_min_days_between_queries		EI
cluster_27_min_of_days_from_query_date		EI
cluster_2_avg_days_between_queries		JAH
cluster_2_max_days_between_queries		EI
cluster_2_max_of_days_from_query_date		EI
cluster_2_min_days_between_queries		EI
cluster_2_min_of_days_from_query_date		EI
cluster_3_avg_days_between_queries		JAH
cluster_3_max_days_between_queries		EI
cluster_3_max_of_days_from_query_date		EI
cluster_3_min_days_between_queries		EI
cluster_3_min_of_days_from_query_date		EI
cluster_4_avg_days_between_queries		JAH
cluster_4_max_days_between_queries		EI
cluster_4_max_of_days_from_query_date		EI
cluster_4_min_days_between_queries		EI
cluster_4_min_of_days_from_query_date		EI
cluster_5_avg_days_between_queries		JAH
cluster_5_max_days_between_queries		EI
cluster_5_max_of_days_from_query_date		EI

cluster_5_min_days_between_queries		EI
cluster_5_min_of_days_from_query_date		EI
cluster_6_avg_days_between_queries		JAH
cluster_6_max_days_between_queries		EI
cluster_6_max_of_days_from_query_date		EI
cluster_6_min_days_between_queries		EI
cluster_6_min_of_days_from_query_date		EI
cluster_7_avg_days_between_queries		EI
cluster_7_max_days_between_queries		EI
cluster_7_max_of_days_from_query_date		EI
cluster_7_min_days_between_queries		EI
cluster_7_min_of_days_from_query_date		EI
cluster_8_avg_days_between_queries		EI
cluster_8_max_days_between_queries		EI
cluster_8_max_of_days_from_query_date		EI
cluster_8_min_days_between_queries		EI
cluster_8_min_of_days_from_query_date		EI
cluster_9_avg_days_between_queries		EI
cluster_9_max_days_between_queries		EI
cluster_9_max_of_days_from_query_date		EI
cluster_9_min_days_between_queries		EI
cluster_9_min_of_days_from_query_date		EI

Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks

Mina, Triin Ree,

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

Krediidibüroosse eraisikute kohta tehtud päringute informatsiooni kasutamine panga krediidiriski mudelis,

mille juhendajad on Raul Kangro ja Karl Märka,

reprodutseerimiseks eesmärgiga seda säilitada, sealhulgas lisada digitaalarhiivi DSpace kuni autoriõiguse kehtivuse lõppemiseni.

2. Annan Tartu Ülikoolile loa teha punktis 1 nimetatud teos üldsusele kättesaadavaks Tartu Ülikooli veebikeskkonna, sealhulgas digitaalarhiivi DSpace kaudu Creative Commonsi litsentsiga CC BY NC ND 3.0, mis lubab autorile viidates teost reprodutseerida, levitada ja üldsusele suunata ning keelab luua tuletatud teost ja kasutada teost ärieesmärgil, kuni autoriõiguse kehtivuse lõppemiseni.

3. Olen teadlik, et punktides 1 ja 2 nimetatud õigused jäävad alles ka autorile.

4. Kinnitan, et lihtlitsentsi andmisega ei riku ma teiste isikute intellektuaalomandi ega isikuandmete kaitse õigusaktidest tulenevaid õigusi.

Triin Ree

15.05.2019