TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Reimo Loopere 154826IABB

KLIENDI HAARATUSE MÕÕTMINE JA ANALÜÜS ÄRIPÄEVAS

Bakalaurusetöö

Juhendaja: Ants Torim
PhD

Tallinn 2018

Autorideklaratsioon

Kinnitan, et olen koostanud antud lõputöö iseseisvalt ning seda ei ole kellegi teise poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on töös viidatud.

Autor: Reimo Loopere

21.05.2018

Annotatsioon

Käesoleva töö esimeseks eesmärgiks on uurida seoseid digitellimuste pikendamiste ning kasutajate aktiivsuse andmete vahel. Eesmärgi saavutamiseks võetakse kasutajate andmed tellimuse viimase 30 päeva kohta ning proovitakse ennustada ainult kasutusandmete põhjal tellimuste pikendamist. Ennustamiseks kasutatakse tuntumaid masinõppe algoritme andmekaeve tarkvaraga Weka. Valitud andmestikus ei olnud seoseid tellimuste pikendamiste ning kasutaja aktiivsuse vahel Äripäeva veebilehel www.aripaev.ee.

Lisaks on töö eesmärkideks saada parem ülevaade Äripäeva digitellimustest ning luua tellimuste pikendamisi ennustavad mudelid. Eesmärkide saavutamiseks analüüsitakse alustuseks tellimustega seotud andmeid andmeanalüüsi tarkvaraga Qlikview. Seejärel uuritakse erinevate algoritmide omaduste ning tulemustega, et valida eesmärgi täitmiseks sobivaim. Lõpuks luuakse kaks otsustuspuud, mis ennustavad digitellimuste pikendamisi.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 26 leheküljel, 4 peatükki, 19 joonist, 6 tabelit.

Abstract

Measuring and analysing client engagement at Äripäev

The first purpose of this thesis is to explore the patterns between the extensions of digital subscriptions and users' activity data. To achieve the goal, user data is taken from the last 30 days of the subscriptions and predictions are done using only information about users' activity. Well-known machine learning algorithms are used to predict order renewals with data mining software Weka. The results concluded that there were no patterns found between the extensions of digital subscriptions and users' activity at Äripäev's website www.aripaev.ee

Additional aims of this thesis are to gain a better overview of the Äripäev's digital subscriptions and to create predictive models for order renewals. In order to achieve the goals, Qlikview data analysis software is used to get an improved understanding about the data. Then, the results of different algorithms are examined to determine the most suitable for the purpose. Finally, two decision trees are created to predict digital subscription renewals.

The thesis is in Estonian and contains 26 pages of text, 4 chapters, 19 figures, 6 tables.

Lühendite ja mõistete sõnastik

Andmekaeve	*Data mining*. Andmete töötlemine/sorteerimine eesmärgiga leida mustreid ja seoseid.
Algoritm	Protsess, kus tuleb täita mingi arv protsesse, et saavutada soovitud tulemus.
SQL	*Structured Query Language*. Struktuurpäringukeel. Enimkasutatav päringukeel, mida toetavad kõik klient-server keskkonnale projekteeritud relatsiooniandmebaasid.
Andmestik	*Dataset*. Andmete kogumik.
CSV	*Comma Separated Values*. Komaeraldusega väärtused. Failivorming, kus kirjed on üksteisest eraldatud komadega.
Drag and drop	Hiirega objekti või teksti teisaldamine.

Sisukord

Jooniste loetelu

Tabelite loetelu

1 Sissejuhatus

AS Äripäev on Eesti ärimeedia turuliider ning teda tuntakse 1989. aastast ilmuva ajalehe Äripäev järgi. Tänaseks on Äripäeva tootevalik laienenud, antakse välja arvukalt erinevaid raamatuid, käsiraamatuid ja infolehti, toimetatakse erinevaid teemaveebe ning samuti ollakse Eesti suurim ärikonverentside korraldaja. Äripäeva omanikuks on Skandinaavia suurim meediakontsern Bonnier Grupp [1].

Antud töös uuritakse andmeid, mis on seotud Äripäeva digitellimustega, sest töö üheks alameesmärgiks on uurida omavaheliseid seoseid tellimuste pikendamise ja veebis kasutajate haaratuse vahel. Digitellimus annab ligipääsu arvutis, tahvlis ja mobiilis ainult tellijatele mõeldud lugudele. Samuti juurdepääsu Äripäeva TOPidele ja arhiividele [2].

Telekommunikatsiooni firmadel on üldiselt korralik ülevaade, millised kliendid tõenäoliselt pikendavad tellimusi ning millised mitte. Sarnaselt on võimalik andmekaeve meetodeid kasutada, et ennustada meediaettevõtetes tellimuste pikendamist. Selleks tuleb esialgu tunda põhjalikult olemasolevaid andmeid, uurida korrelatsiooni erinevate atribuutide vahel ja seejärel teha ennustusi [3, lk 32].

1.1 Töö eesmärgid

Esimeseks töö eesmärgiks on analüüsida klientide haaratust ehk seoseid digitellimuste pikendamiste ning kasutajate aktiivsusega Äripäeva veebileheküljel www.aripaev.ee.

Lisaks on töö eesmärkideks saada parem ülevaade Äripäeva digitellimustest ja seejärel luua tellimuste pikendamist ennustavad mudelid kasutades tuntumaid ning ennast tõestanud masinõppe algoritme. Leides tellimused, mille lõpetamine on tõenäolisem ning tunnused, mis soodustavad tellimuse katkestamist, on võimalik tõsta pikendajate osakaalu pakkudes tellijatele vastavalt erinevaid lisaväärtusi. Näiteks suunata passiivsemaid kasutajaid rohkem lugema, eeldades et aktiivsemad kasutajad pikendavad tellimusi suurema tõenäosusega.

1.2 Metodoloogia

Töö esimeses pooles selgitatakse töö teoreetilisi aluseid. Töö teises pooles luuakse antud eesmärkide täitmiseks sobivamad mudelid, mis ennustaks digitellimuste pikendamisi. Eelnevalt tutvutakse andmetega, kasutades selleks andmeanalüüsi tarkvara Qlikview ning seejärel uuritakse erinevate masinõppe algoritmide efektiivsust ja sobivust probleemi lahendamisel tarkvaraga Weka. Töö lõpus vaadatakse lähemalt kahe otsustuspuu kirjeldusi ning edastatakse töö järeldused.

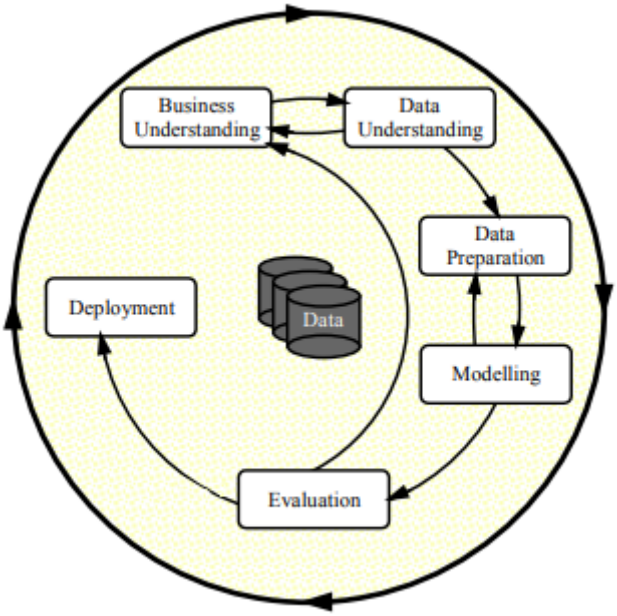
Töö ülesehitusel järgitakse CRISP-DM andmekaeve meetodit. Erialaste mõistete ning lühendite kirjeldamiseks või tõlkimiseks kasutatakse e-teatmiku [4].

2 Teoreetilised alused

Antud peatükis kirjeldatakse töös kasutatud teoreetilisi teadmisi.

2.1 CRISP-DM

CRISP-DM (*Cross Indrusty Process for Data Mining*) on andmekaeve meetod, mis kirjeldab tervikliku andmekaeve projekti loomist. Projekti ülesehitus meetodi järgi koosneb kuuest etapist: arusaamine ärist, arusaamine andmetest, andmete ettevalmistamine, modelleerimine, tulemuste hindamine ja kasutuselevõtt. Joonis 1 esitab CRISP-DM etappide käiku [5].



Joonis 1. CRISP-DM protsessimudeli tsüklid.

Järgnevalt kirjeldatakse CRISP-DM kuute etappi:

- Arusaamine ärist

Esimene etapp keskendub projekti ärilistele eesmärkidele ning nõuetele. Äriliste nõuete mõistmine on oluline, et püstitada lahendatav probleem ja suunata ülejäänud projekt selle lahendamisele [5].

- Arusaamine andmetest

Teine etapp koosneb esialgsete andmete kogumisest ja tegevustest, mis aitavad andmetest täiuslikumat ülevaadet saada [6].

- Andmete ettevalmistamine

Andmete ettevalmistamise etappi võivad kuuluda tegevused nagu andmete valik, korrastamine, uute atribuutide tuletamine ning andmestiku sobitamine kasutatava tarkvaraga. Järelikult kolmandas etapis tehakse andmetega tegevused, mis tagavad, et andmed sobiksid andmekaeve tarkvarale ning nendest on võimalik midagi tuletada [5].

- Modelleerimine

CRISP-DM modelleerimise faasis kasutatakse erinevaid masinõppe algoritme, et luua ennustavaid mudeleid. Vastavalt tulemustele ning eesmärkidele valitakse neist parim kasutuselevõtuks [7, lk 14].

- Hindamine

Eelmises etapis loodud mudelid tuleb enne kasutuselevõttu põhjalikult üle vaadata ning hinnata, kas nad täidavad soovitud eesmärki. Hindamise faas katab kõik tegevused, mis näitavad, et mudel ennustab täpselt ning ei kannata üle- või alasobitamise all [7, lk 15].

- Kasutuselevõtt

Masinõppe mudelid luuakse, et lahendada esimeses etapis defineeritud ärilised probleemid. Viimases faasis tuleb integreerida masinõppe mudel tööprotsessidega [7, lk 15].

2.2 Kasutatud tarkvara

Peatükis tutvustatakse spetsiifilisemaid tarkvarasid, mida töös kasutati.

2.2.1 Weka

Weka on erinevate masinõppe algoritmide kogum, mida saab kasutada andmekaeveks. Põhilisteks tööriistadeks Wekas on andmete eeltöötlemine, liigitamine, regressioon,

klasterdamine, seosete loomine ning visualiseerimine. Seetõttu öeldakse, et Weka on hulk erinevaid programme ühes kasutajaliideses [8].

Näiteks on Wekas võimalik andmestikule rakendada masinõppe meetod ja analüüsida tulemusi, et saada sügavam arusaam andmetest. Teine võimalus on kasutada treenitud mudeleid, et ennustada uute andmete põhjal mingeid tulemusi [9, lk 7-22].

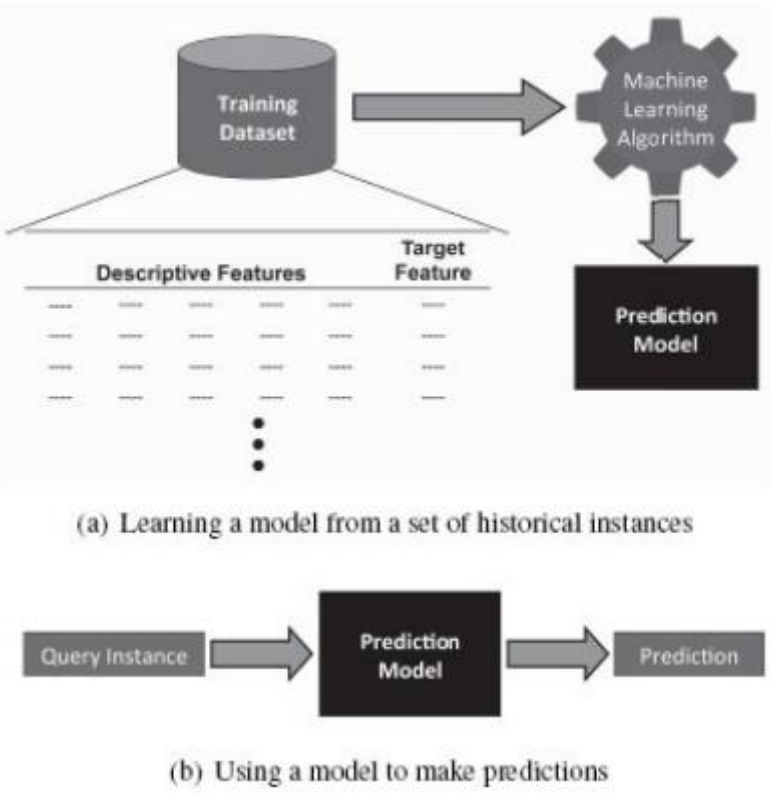
2.2.2 Qlikview

Qlikview on ärianalüüsi tarkvara, mille eelisteks on kiire ja mugav andmete ühendamine ning visualiseerimine. Tihti on vaja luua seoseid andmetega erinevatest allikatest, näiteks andmebaasidest, Exceli tabelitest, APIdest ja mujalt. Qlikviewis kasutatakse SQL-le sarnast süntaksit andmete eeltöötlemiseks [10].

Pärast soovitud andmete kättesaamist võimaldab Qlikview kuvada lihtsamaid tulemustabeleid kasutades *drag and drop* funktsionaalsust ning keerulistemate jaoks tuleb kasutada süntaksit, mis on justkui segu SQL-st ja Microsoft Exceli funktsioonidest [10].

2.3 Masinõpe

Automaatset protsessi, mille käigus otsitakse sarnaseid mustreid andmestikest nimetatakse masinõppeks. Andmeanalüüsi rakendustes kasutavate ennustavate mudelite koostamiseks kasutatakse juhitud masinõpet (*supervised machine learning*). Selleks õpib mudel erinevate atribuutide seoseid ajalooliste andmete põhjal ning kasutab loodud mudelit uute tulemuste ennustamiseks (Joonis 2) [7, lk 3].



Joonis 2. Juhitud masinõppe sammud.

Standardselt kasutatakse ühel andmestikul masinõppe mudeli veamäära hindamiseks kihilist kümnekordset ristvalideerimist (*stratified tenfold cross-validation*), kus andmestik on jagatud kümneks osaks. Seejärel sooritatakse masinõppe protseduuri kümme korda, kus iga kord jäetakse üks osa välja ning treenitakse üheksa osa peal. Veamäär arvutatakse välja jäetud osa põhjal. Lõpuks arvutatakse kümne veamäära keskmine, et leida täielik veamäär [11, lk 126].

Ennustamismudelite loomisel tuleb vältida ala- ja ülesobitamist. Alasobitamine tähendab, et loodud ennustusmudel on liiga lihtne, et väljendada andmestiku atribuutide ja ennustatava välja vahelisi seoseid. Ülesobitamine tähendab, et loodud mudel on liiga keerukas ning tundlik mürale andmestikus [7, lk 10].

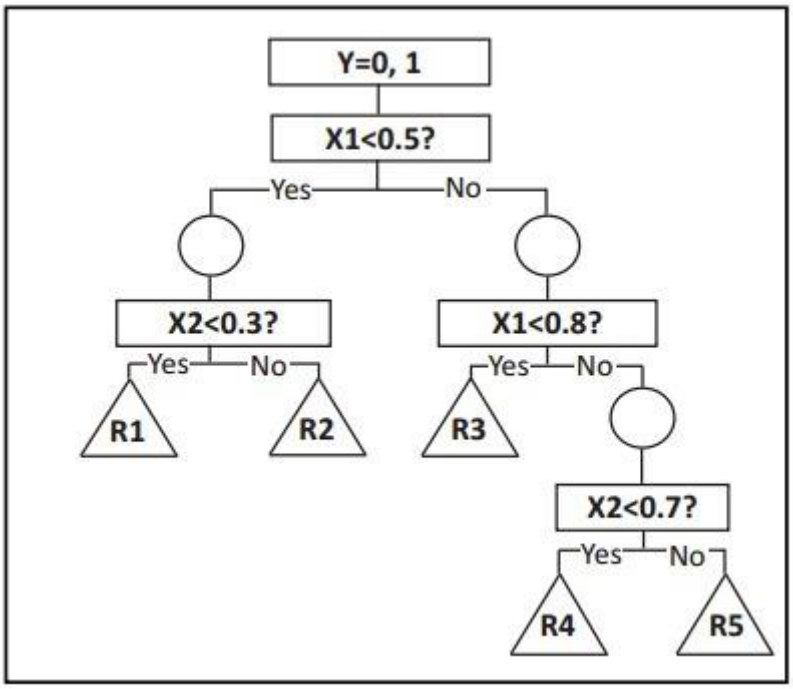
2.4 Algoritmide kirjeldused

Antud peatükis kirjeldatakse tuntumaid ning ennast tõestanud algoritme. Kuigi kõiki kirjeldatud algoritme ei kasutata põhjalikumalt valitud andmestike peal, siis on oluline aru saada erinevatest töö põhimõtetest, et valida probleemi lahendamiseks sobivaim algoritm.

- Otsustuspuu

Otsustuspuu annab selge ülevaate graafilise mudelina, kuidas tehakse otsus mingi järelduseni jõudmiseks. Otsustuspuu koosneb sõlmedest ja harudest. Iga sõlm tähistab otsust ning harud võimalikke väärtuseid [12].

Sõlmi on kolme liiki. Otsustussõlm tähistab valikut, mille tulemusena jagatakse andmestik kaheks või enamaks alamhulgaks. Sündmussõlm on üks võimalik valik antud puustruktuuris. Terminaalsõlm tähistab puu lõpptulemust, kuhu erinevate valikute tulemusena jõutakse [13]. Joonis 3 esitab otsustuspuu ülesehitust.



Joonis 3. Otsustuspuu ülesehitus.

Otsustuspuude peamine eelis on see, et nad on lihtsasti tõlgendatavad. Visuaalse pildi pealt on kerge välja lugeda, miks lõpptulemuseni jõuti. Otsustuspuu võib muutuda liiga suureks ning tundlikuks mürale, kui andmestik sisaldab palju kirjeldavaid välju [7, lk 167-168].

Antud töös kasutatakse Weka otsustuspuu algoritme HoeffidingTree[14], RandomTree[15], REPTree[16] ning J48[16]. Täpsemalt uuritakse kahe viimase mudeleid, mõlemad on loodud populaarse C4.5[17] otsustuspuu algoritmi põhjal.

- Otsustusmets

Nagu nimestki võib välja lugeda, loob otsustusmets erinevaid otsustuspuid ning ühendab need kokku, et saada kõige täpsem ennustus. Otsustusmetsa saab kasutada nii klassifikatsiooni kui ka regressiooni probleemi lahendamiseks. Eeliseks otsustuspuu ees on väiksem tundlikkus ülesobitamisele [19]. Töös uuritakse Weka otsustusmetsa algoritmi RandomForest[20] tulemusi.

- Lähima naabri algoritm

Lähima naabri mudeli treenimine koosneb ainult kõikide treeningandmete mällu salvestamisest ning on seega väga lihtne. Standardses algoritmiga ennustamise faasis kõrvutatakse ennustatava andmerea atribuudid treeningmudelis olevatega ning arvutatakse välja, milline rida on kõige sarnasem ennustatavale [7, lk 179-186].

Kuna standardse valemi järgi toimub ennustus vaid ühe mudelis oleva andmerea järgi, siis on ennustamine üsna tundlik mürale. Müra mõju vähendamiseks kasutatakse k-lähima naabri algoritmi, kus võetakse ennustamisel arvesse lähimad *k* naabrit [11, lk 72-73]. Lähima naabri algoritmil põhinevad töös kasutatud KStar[21] ja IBk[22] masinõppe meetodid Wekas.

- Naiivne Bayes

Naiivse Bayesi meetod põhineb Bayesi teoreemil ning eeldab sündmuste iseseisvust. Klassifitseerimismudelit on lihtne ehitada, mis muudab ta kasulikuks suurte andmemahtudega mudeli loomiseks. Olenemata oma lihtsusele suudab Naiivne Bayes üsna täpselt ennustada [23]. Wekas uuritavaks algoritmiks on NaiveBayes[24].

- Logistiline regressioon

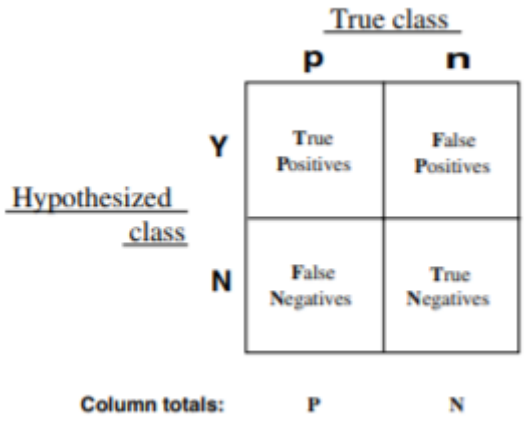
Logistilise regressiooni puhul on ennustusel ainult kaks võimalikku väärtust. Näiteks 1 või 0, Jah või Ei, *True* või *False*. Ennustuseni jõudmiseks analüüsitakse ühte või mitut atribuuti andmestikus. Logistilist regressiooni kasutatakse, et näidata seoseid erinevate atribuutide ning ennustava binaarse välja vahel [25]. Töös vaadatakse Weka algoritmide Logistic[26] ja SimpleLogistic[27] tulemusi.

- Otsustustabel

Otsustustabel visualiseeritakse, mille tõttu on võimalik saada hea ülevaade, kuidas järelduseni jõutakse. Lihtne struktuur teeb algoritmi poolt loodud mudeli arusaadavaks. Otsustustabelis uuritakse atribuutide põhjal, millised atribuutide väärtused viivad tulemuseni *x*. Saadud andmete põhjal lihtsustatakse ning välistatakse reeglid, kui on piisavalt palju näiteid, mis käituvad vastupidiselt [28]. Weka DecisionTable[29] algoritmi kasutatakse töös ennustamiseks.

2.5 Mudelite hindamine

Klassifitseerimis probleemil on kaks võimalikku väljundit, seega võimalikke tulemeid on neli. Antud töö näite põhjal on digitellimuse võimalikeks väljunditeks kas tellimus on pikendatud või mitte (*True* ja *False*). Tulemuste põhjal tekib veamaatriks, mille igas ruudus tähistatakse ruudus märgitud tulemuste arvu. Joonis 4 näitab, kuidas erinevad tulemused jaotuvad. *True Positives* tähistab andmeridade arvu, mille korral ennustati positiivne tulemus õigeks. *True Negative* näitab õigesti ennustatud negatiivsete, *False Positives* valesti ennustatud negatiivsete ning *False Negatives* valesti ennustatud positiivsete arvu [30].



Joonis 4. Veamaatriks.

Täpsus (*precision*), saagis (*recall*) ning *F-measure* on sagedasti kasutatud hindamismeetodid, mida on võimalik arvutada veamaatriksi põhjal. Täpsus näitab, kui kindel saab olla selles, et mudel ennustas õigesti kõik positiivselt klassifitseeritud tulemused. Saagis näitab, kui suur on tõenäosus, et positiivne ennustus on õige.

$$täpsus = \frac{TP}{(TP + FP)}$$

$$saagis = \frac{TP}{(TP + FN)}$$

F-skoor (*F-measure*) on täpsuse ja saagise harmooniline keskmine.

$$F - skoor = \ 2 \ \times \frac{(täpsus \times saagis)}{(täpsus + saagis)}$$

Nimetatud hindamismeetodid on kõige efektiivsemad kui otsitakse mudelit, mis ennustab kõige täpsemini positiivseid tulemusi [7, lk 414-417].

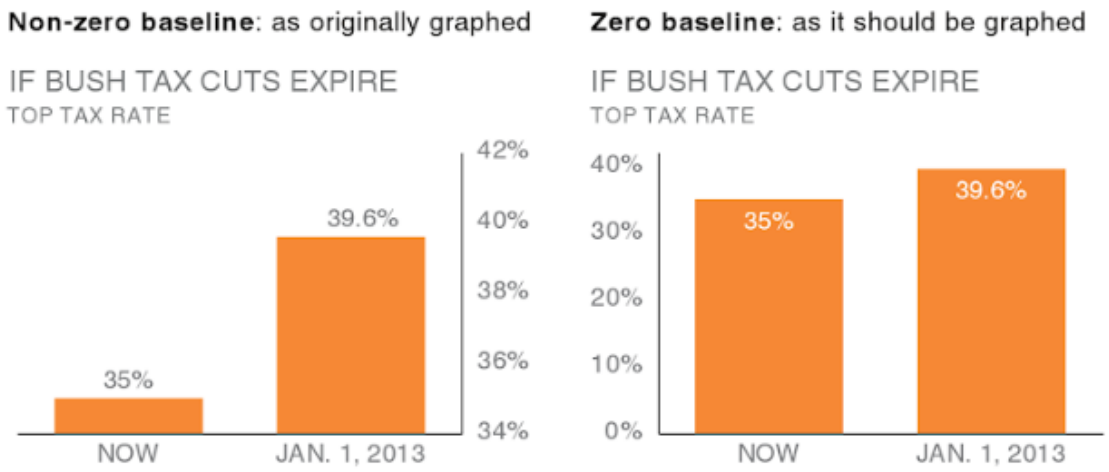
Ruutkeskmise vea juur (*root mean squared error*) aitab määrata mudeli efektiivsust ennustamisel. Valem avaldub kujul

$$root\ mean\ squared\ error = \sqrt{\frac{\sum_{i=1}^{n} abs(t_i - M(d_i))^2}{n}}$$

kus $t_1 \ldots t_n$ on *n* oodatud väärtust ning $M(d_i) \ldots M(d_n)$ on *n* ennustatud väärtust testitavate andmeridade $d_1 \ldots d_n$ kohta [7, lk 443-444].

2.6 Andmete visualiseerimine

Visualiseerimine aitab suurtes andmestikkudes ära tunda mustreid ja seoseid, mille põhjal on võimalik hiljem järeldusi teha. Kui tabelite puhul on lihtsam järgida täpseid numbreid ning neid võrrelda, siis visualiseerides andmeid on võimalused andmete tõlgendamiseks tunduvalt suuremad. Näiteks on võimalik graafikutel skaalasid muutes manipuleerida vaatlejaid (Joonis 5). Mõlemad graafikud esindavad samu andmeid. Kui vasakul graafikul on y-skaala alguseks valitud 34%, siis tunduvad kahe tulba vahed mitmekordsed. Seega andmete tõeseks esitamiseks peaks skaala alati algama nullist, nagu paremal graafikul [31, lk 10-31].



Joonis 5. Graafikutega manipuleerimine.

Tabeleid „loetakse“, mis tähendab, et liigutakse mööda ridu ja veerge võrreldes väärtuseid. Seega on tabelid kasulikud, kui presenteeritakse andmeid erinevatele sihtrühmadele ning nad leiavad endale olulise rea vaatlemiseks. Tabelite disainimisel peaks põhirõhk jääma andmetele, mitte piirjoontele või muudele detailidele [31, lk 40-41].

Graafikuid „vaadatakse“, seega informatsioon jõuab kiiremini adressaadini. Antud töös kasutatakse tulpdiagramme. Kuna tulpdiagrammid on harilikud, siis neid välditakse tihti. Samas säästab levinud kasutamine graafikust arusaamise aega ning olulise informatsiooni leiab koheselt, ilma et peaks aega kulutama graafikute õppimiseks [31, lk 50-59].

3 Tellimuste pikendamise ennustamine

Käesolevas peatükis täidetakse CRISP-DM etapid 2-5. Kuuenda etapi asemel tehakse järeldused, kuna kasutuselevõtt pole autori pädevuses. Järelduste peatükis analüüsib autor tehtud töö tulemusi ning annab edasi enda mõtteid, kuidas tööd võiks edasi arendada ja rakendada.

3.1 Algandmed

Analüüsitavateks andmeteks on võetud Äripäeva digitellimustega seotud andmed. Täpsemalt tellimused, mis on lõppenud vahemikus 01.09.2017 – 28.02.2018. Ajavahemik on valitud selliselt, et oleks võimalik tagasivaatavalt uurida kasutaja viimase 30 päeva lugemisharjumusi tellimuse lõppkuupäevast alates ning vaadata, kas tellimust on pikendatud 1 kuu jooksul pärast tellimuse lõppu.

Kuna uuritavad andmed asuvad erinevates allikates, siis kasutati esialgseks andmete analüüsiks ning ühendamiseks tarkvara Qlikview.

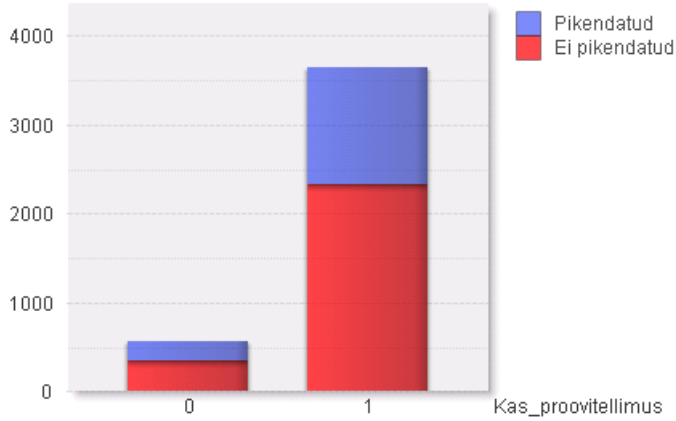
- Otsesed andmed

Otseselt tellimuste või klientidega seotud andmete nimetused ning kirjeldused on välja toodud tabelis 1.

Tabel 1. Otsesed andmed.

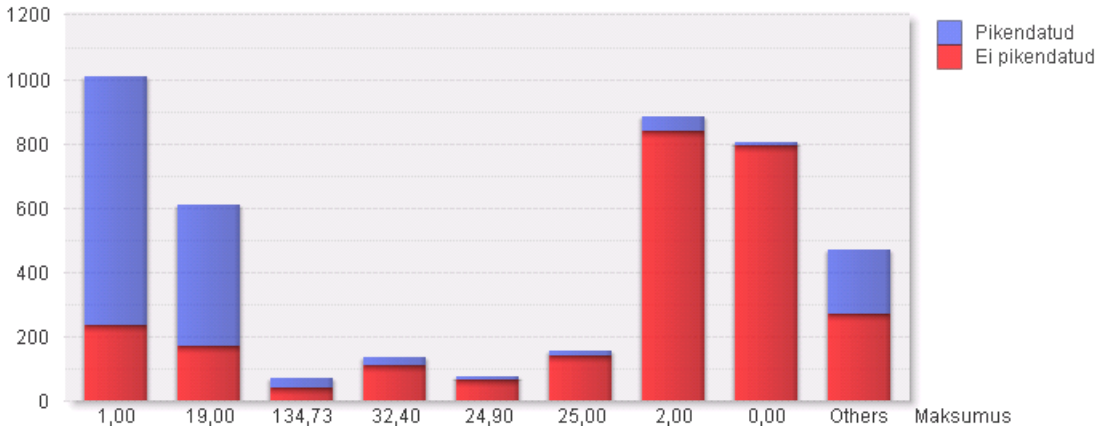
Atribuut	**Kirjeldus**
Digioigusi	Tellimusele antud digiõiguste arv.
Kas_pikendustellimus	Kas tellimus on pikendustellimus.
Kas_proovitellimus	Kas tellimus on proovitellimus.
Kampaania_kood	Kampaaniakood, millega tellimus seotud on.
Maksumus	Tellimuse hind.
Periood	Tellimuse pikkus kuudes.
Makseviis	Tellimuse makseviis.
Sugu	Füüsiliste isikute sugu.
Juriidiline	Kas maksja on juriidiline isik.
Sektor	Juriidilise isiku sektor.
Kas_Tmo_Keeld	Kas klient soovib telemarketingi.

Esialgseks uurimiseks oli 4206 tellimuse andmed. Kokku pikendati valitud ajavahemikus digitellimusi 1543 korda ehk 36.7%. Proovitellimusi oli 3639, pikendatud tellimuste osakaal ligikaudu 36% ning jätkutellimusi pikendati 40% juhtudest (Joonis 6).

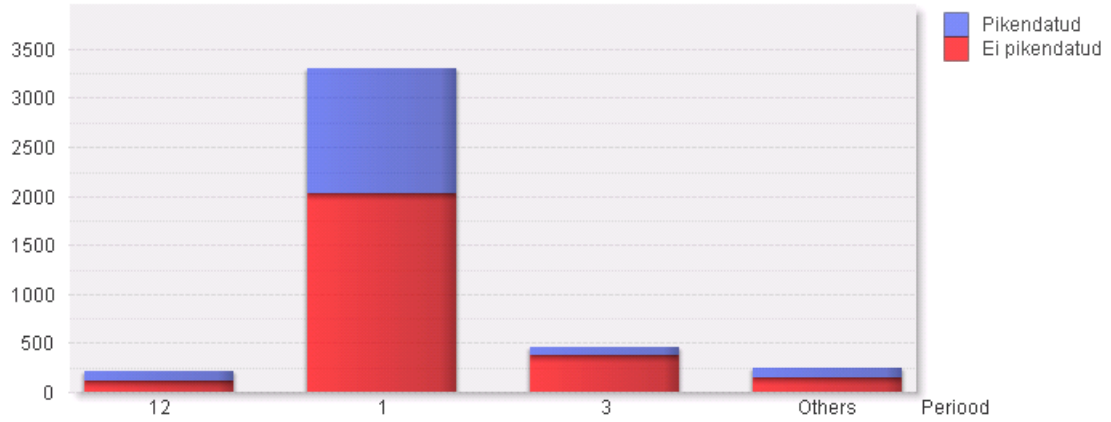


Joonis 6. Atribuudi „Kas_proovitellimus“ jaotused.

Kõige enam, 76,5%, pikendati ühe eurose maksumusega tellimusi. Tellimusi, mille hind oli 19 eurot pikendati 442 korda 611st. Tasuta tellimusi pikendati vaid 1,4% (Joonis 7). Parim pikendamiste protsent oli aastastel tellimustel (Joonis 8).

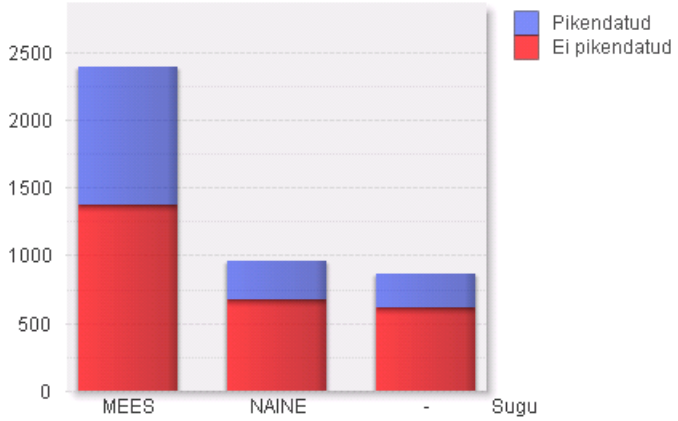


Joonis 7. Tellimuste maksumuste jaotused.

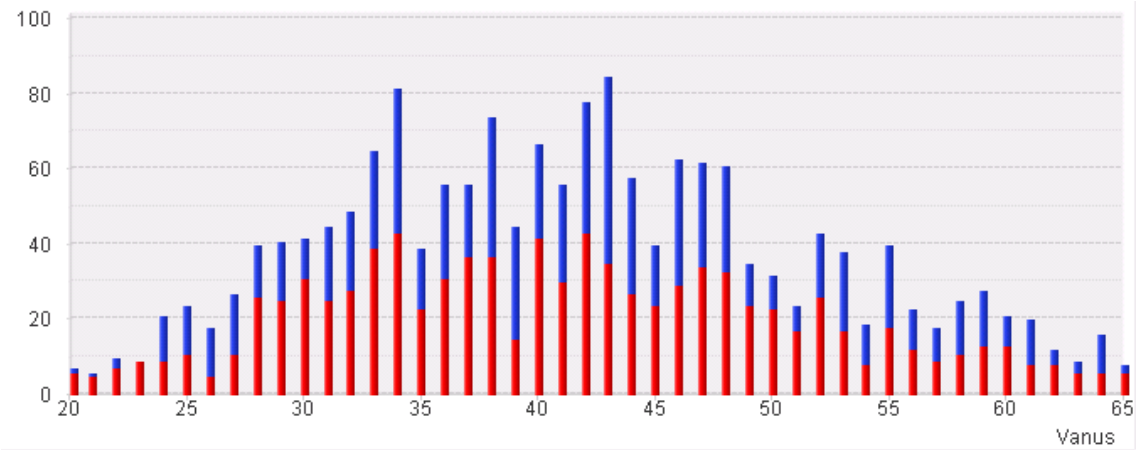


Joonis 8. Tellimuste perioodide jaotused.

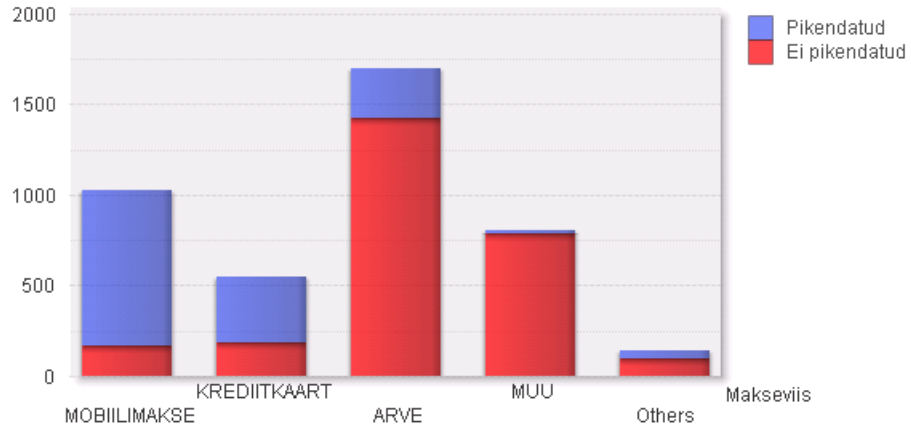
Eraisikutest oli meeste osakaal digitellimuste pikendamisel 42,5% ja naiste osakaal 30% (Joonis 9). Juriidilistest isikutest maksjaid oli kokku 689, kellest pikendasid 27,7%. Suurem osa digitellimuste kliente jäid vanuse vahemikku 33-48 (Joonis 10). Kõige rohkem pikendati mobiili ning krediitkaardiga makstud tellimusi, vastavalt 83,2% ja 66,2%. Arvega tasutud 1694st tellimusest pikendati vaid 273, ehk 16,1% (Joonis 11).



Joonis 9. Atribuudi „Sugu“ jaotused.



Joonis 10. Eraisikust tellijate vanuste jaotus.



Joonis 11. Makseviiside jaotus.

- Kasutusandmed

Tellimustega seotud klientide kasutusandmed veebileheküljel www.aripaev.ee on kirjeldatud tabelis 2.

Tabel 2. Kasutusandmed.

Atribuut	**Kirjeldus**
Ad campaign hit	Mitu korda on kasutaja tellimuse viimase 30 päeva jooksul reklaamikampaaniale klikanud.
Click on related article	Mitu korda on kasutaja tellimuse viimase 30 päeva jooksul klikanud seotud artiklile.
Clicked e-mail link	Mitu korda on kasutaja tellimuse viimase 30 päeva jooksul klikanud e-maili lingile.
Logged in	Mitu korda on kasutaja tellimuse viimase 30 päeva jooksul sisse loginud Äripäeva veebileheküljele.
Search engine hit	Mitu korda on kasutaja tellimuse viimase 30 päeva jooksul külastanud Äripäeva veebilehte otsingumootori kaudu.
Viewed article	Mitu korda on kasutaja tellimuse viimase 30 päeva jooksul lugenud artiklit äripäeva veebilehel.
Visited site	Mitu korda on kasutaja tellimuse viimase 30 päeva jooksul äripäeva veebilehte.

Kasutusandmete puhul jääb silma, et 20% kasutajatest pole tellimuse lõpust viimase 30 päeva jooksul ühtegi tegevust. Selle põhjuseks võib olla reklaamifiltrite (näiteks uBlock) kasutamine. Samal põhjusel võib olla osa kasutusandmeid puudu, kui näiteks kasutatakse

filtreid arvutis ning telefonis mitte. Tabel 3 näitab, mitmel protsendil kasutajatel ei tuvastatud ühtegi vasakul veerus nimetatud tegevust tellimuse viimase 30 päeva jooksul.

Tabel 3. Kasutusandmete puudumine.

Atribuut	**% = 0**
Ad campaign hit	47,8%
Click on related article	81,9%
Clicked e-mail link	54,5%
Logged in	72,3%
Search engine hit	73,4%
Viewed article	30,3%
Visited site	28,6%

- Tuletatud andmed

Tuletatud andmed on leitud proovimaks tuvastada seoseid varasema kliendikogemuse ning uute tellimuste pikendamiste vahel (Tabel 4). Lisaks arvutatakse välja ennustatav väli „Pikendatud“ ja klientide vanus.

Tabel 4. Tuletatud andmed.

Atribuut	**Kirjeldus**
Vanus	Arvutatud vanus aastates. Tänasest päevast lahutatud sünnikuupäev. Juriidilistel isikutel firma asutamise kuupäev.
Tellimusi_AP	Mitu Äripaeva tellimust on maksja teinud 5 aasta jooksul enne tellimuse algust.
Maksnud_AP	Kui palju on maksja tasunud Äripäeva tellimuste eest 5 aasta jooksul enne tellimuse algust.
Pikkus_AP	Mitu päeva on maksjal olnud kehtiv Äripäeva tellimus 5 aasta jooksul enne tellimuse algust.
Proovi_AP	Mitu korda on maksjal olnud kehtiv proovitellimus 5 aasta jooksul enne tellimuse algust.
Pikendatud	Kas tellimuse maksjal on ühe kuu jooksul pärast tellimuse lõppu alustatud uus Äripäeva tellimus.

Andmete päringu päringuks kasutati skripti tarkvaras Qlikview (Lisa 1).

3.2 Andmete eeltöötlemine

Andmete eeltöötlemise käigus eemaldati andmestikust müra ning tehti andmed loetavaks masinõppe tarkvarale Weka. Alustuseks eksporditi andmestik Excelisse ning eemaldati andmed, mille puhul ei ole äriliselt mõistlik ennustada pikendamist ja mis võivad teisi ennustusi segada. Näiteks toimetuse poolt tasuta jagatud digiõigused kaasautoritele.

Weka loeb vaikimise ARFF formaadis faile. ARFF sarnaneb CSV failile, kuid nõuab mõningaid täiendusi. Alustuseks asendati tühjad väljad väärtusega „?“. Seejärel lisati tekstiredaktoris andmestiku nimi *@relation* märgendiga, atribuutide info *@attribute* märgenditega ning *@data* silt komaga eraldatud andmete ette [7, lk 17].

Kõige lõpuks eemaldati atribuut „Kampaania_kood“, sest kampaaniakoode tuleb ajas juurde ning seetõttu on ta pigem müraks andmestikus. Lisaks on kampaaniakood üldiselt kombinatsioon hinnast, perioodist ning makseviisist ja atribuudi kaotamine ei muuda ennustamismudelite efektiivsuseid halvemaks. Kokku jäi alles 4112 andmerida 23 atribuudiga.

3.3 Mudelite treenimine

Sobiva algoritmi leidmisel seati ootused, et mudel oleks võimalikult täpne, et mudelit oleks võimalik ajas täiustada ning et mudelist oleks võimalik leida põhjuseid, miks inimesed lõpetavad oma tellimusi. Kuna töös oli tegemist klassifitseerimis ülesandega (Kas tellimus pikendatakse = Jah/Ei), siis sobivad antud probleemi lahendamiseks klassifitseerimisalgoritmid.

Kuna äriliselt oluline on leida nii tellimuste pikendajad ja lõpetajad võimalikult täpselt, siis mudelite hindamiseks kasutati eelkõige õigesti ennustatud tellimuste protsenti. Mudel, mis ennustab kõik tellimused mitte pikendatuks, saavutaks täpsuseks ligikaudu 63%. Seega peaks treenitavate mudelite täpsus olema märgatavalt kõrgem.

Esialgseks mudelitest ülevaate saamiseks kasutati Weka Experiment Environment’i, millega saab jooksutada korraga erinevaid algoritme ühel andmestikul ning hiljem võrrelda tulemusi. Igat valitud algoritmi treeniti 10 korda kasutades kihilist kümnekordset

ristvalideerimist (*stratified tenfold cross-validation*), et nende tulemustel oleks võimalik valida parimad algoritmid, mida hiljem täpsemalt uurida.

Erinevate algoritmide ennustamiste tulemused on näha tabelis 5. *Percent_correct* näitab täpselt ennustatud tellimuste protsenti, *number_correct* näitab mitu tellimust ennustati õigesti. *True_positive_rate* näitab, kui suure täpsusega ennustati pikendatud tellimused pikendatuks. Ruutkeskmise vea juur (*Root_mean_squared_error*) aitab kontrollida kui stabiilselt algoritm suutis ennustada, kuna vead pannakse enne keskmise arvutamist ruutu, siis mõjutavad suured vead tulemust märgatavalt.

Tabel 5. Algoritmide tulemused.

Algoritm	**Percent_correct**	**Number_correct**	**True_positive_rate**	**Root_mean_squared_error**
functions.SimpleLogistic	84,98	349,42	0,84	0,33
trees.RandomForest	84,96	349,36	0,82	0,33
functions.Logistic	84,53	347,60	0,85	0,34
trees.J48	84,41	347,09	0,80	0,35
rules.DecisionTable	84,40	347,04	0,84	0,34
trees.REPTree	84,09	345,78	0,83	0,34
bayes.NaiveBayes	83,12	341,80	0,83	0,38
lazy.Kstar	81,32	334,38	0,77	0,40
trees.RandomTree	80,55	331,24	0,73	0,42
trees.HoeffidingTree	80,27	330,07	0,71	0,38
lazy.Ibk	80,21	329,82	0,79	0,44

Järgmiseks uuriti, kui edukalt suudavad algoritmid ennustada kasutades andmestikus ainult kasutusandmeid (Tabel 2. Kasutusandmed.). Joonisel 12 on näha kaheksa erineva algoritmi ennustamistäpsus kasutades Äripäeva veebilehe kasutusandmeid pikendamiste ennustamiseks. Ennustamistäpsust hinnatakse õigesti arvatud tellimuste protsendi järgi, mis on näidatud joonise keskel. Joonise all legendil on kirjeldatud, millisele masinõppe algoritmile esitatud tulemus kuulub.

Tulemustest on näha, et ükski mudel ei suuda keskmise täpsusega ületada mudelit, mis ennustaks kõik tellimused mitte pikendatuks (63%). Põhjuseks võib olla andmete puudumine suures hulgas (Tabel 3). Lisaks sisaldab valitud andmestik palju püsimakse tellimusi. Tulemuste järeldusena otsustati kasutusandmeid edasisel mudelite treenimisel mitte kasutada, et vähendada mudelite ülesobitamist.

```
Tester:     weka.experiment.PairedCorrectedTTester -G 4,5,6 -D 1 -R 2 -S 0.05 -result-matrix "weka.experiment.R
Analysing:  Percent_correct
Datasets:   1
Resultsets: 8
Confidence: 0.05 (two tailed)
Sorted by:  -
Date:       7.05.18 22:03


Dataset                   (1) function |  (2) trees (3) funct (4) trees (5) trees (6) rules (7) bayes (8) lazy.
-------------------------------------------------------------------------------------------------------------------
wap                      (100)   63.66 |   61.25 *   63.63     63.24     58.96 *   63.77     55.76 *   62.82
-------------------------------------------------------------------------------------------------------------------
                               (v/ /*) |   (0/0/1)   (0/1/0)   (0/1/0)   (0/0/1)   (0/1/0)   (0/0/1)   (0/1/0)


Key:
(1) functions.SimpleLogistic '-I 0 -M 500 -H 50 -W 0.0' 7397710626304705059
(2) trees.RandomForest '-P 100 -I 100 -num-slots 1 -K 0 -M 1.0 -V 0.001 -S 1' 1116839470751428698
(3) functions.Logistic '-R 1.0E-8 -M -1 -num-decimal-places 4' 3932117032546553727
(4) trees.J48 '-C 0.25 -M 2' -217733168393644444
(5) trees.RandomTree '-K 0 -M 1.0 -V 0.001 -S 1' -9051119597407396024
(6) rules.DecisionTable '-X 1 -S \"BestFirst -D 1 -N 5\"' 2888557078165701326
(7) bayes.NaiveBayes -output-debug-info 5995231201785697655
(8) lazy.KStar '-B 20 -M a' 332458330800479083
```

Joonis 12. Kasutusandmetega ennustamise tulemused.

Esialgsetest algoritmide tulemustest (Tabel 5) on näha, millised algoritmid esinesid kihilise ristvalideerimisega kõige paremini. Otsustuspuud sobivad kõige tõhusamini täitma mudelitele seotud ootuseid ning kasutatakse üldiselt tellimuste pikendamistega seotud ennustusmudelite loomisel [7, lk 477]. Kuna otsustuspuude ja teiste algoritmide efektiivsuses märkimisväärseid erinevusi ei olnud, siis otsustati lähemalt uurida otsustuspuid. Lisaks loobuti otsustusmetsa sügavamast uurimisest, kuna Weka ei paku baasversioonis nimetatud algoritmile kirjeldavaid ega visualiseeritud võimalusi.

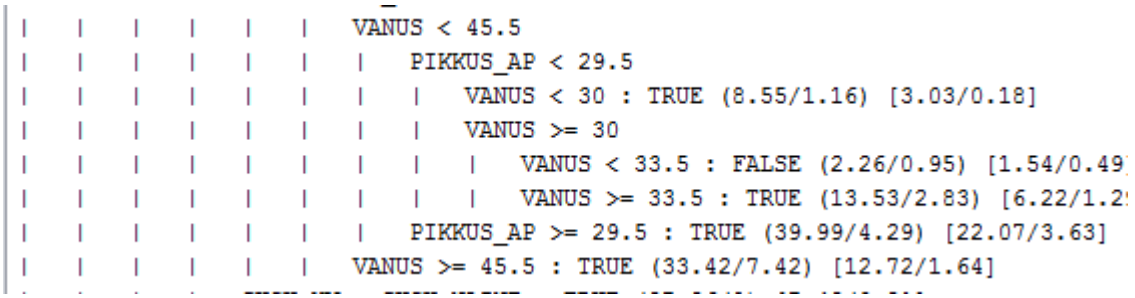
Pärast kasutusandmete ja kampaaniakoodide eemaldamist andmestikust jäi alles 16 atribuuti, mille põhjal ennustada. Kuigi üleliigsete andmete välja jätmisel paranesid mudelite tulemused, oli näha endiselt ülesobitamist mudelites. Seda tõestasid nii puude suurused, kui ka puude sõlmed täpsemal uurimisel.

Algoritmi REPTree puu kirjeldavast mudelist joonisel 13 on näha, et puu kaugemad oksad määravad tõenäoliselt liiga täpselt vanuse seost pikendamisega. Veamaatriksid joonistel 14 ja 15 näitavad, et algoritm J48 ennustas täpsemini pikendatud tellimusi ning REPTree

pikendamata tellimusi. Tabelis 6 on välja toodud algoritmide täpselt ennustatud tellimuste protsent ning loodud otsustuspuude suurused.

Tabel 6. Otsustuspuude treenimine.

Algortim	**Protsent õige**	**Puu suurus**
tree.J48	84,87	61
tree.REPTree	84,82	51
tree.RandomTree	81,57	2503



Joonis 13. Algoritmi REPTree ülesobitamine.

```
=== Confusion Matrix ===

    a    b   <-- classified as
 1214  275 |    a = TRUE
  347 2276 |    b = FALSE
```

Joonis 14. J48 veamaatriks.

```
=== Confusion Matrix ===

    a    b   <-- classified as
 1205  284 |    a = TRUE
  340 2283 |    b = FALSE
```

Joonis 15. REPTree veamaatriks.

3.4 Hindamine

Mudelite täpsemaks hindamiseks ning sobitamiseks võeti lisaks andmestik aprillikuus lõppenud tellimustest. Kui ennustusmudeli loomiseks kasutatavas andmestikus oli pikendatud tellimusi vaid 36,7%, siis aprillis lõppenud tellimustest pikendati 896st tellimusest 668, ehk koguni 74,6%. Osakaalu suur muutumine näitab kui väga võivad andmed ajas muutuda ning kui oluline on ennustusmudeli täiendamine tulevikus.

Esialgsed mudelid suutsid ennustada aprillikuu digitellimuste pikendamist järgnevalt: REPTree 77,26% ja J48 75,57%. Arvestades, et pikendamiste osakaal ja andmed erinesid märgatavalt algandmetest, siis võib tulemusi lugeda üsna täpseteks.

Samas oli näha, et ülesobitamist vähendades võib mudeli ennustamise täpsust tõsta. Et vähendada müra mõju loodud mudelitele piirati puude sügavusi.

Joonisel 16 on algoritmi REPTree tulemused ja joonisel 17 algoritmi J48 tulemused pärast sügavuste piiramist. Tellimusi klassifitseeriti õigesti vastavalt REPTree 77,4% ning J48 77,5%. REPTree algoritmi loodud mudel ennustas vaid ühe digitellimuse rohkem õigesti ning täpsuste (*Precision*) ja saagiste (*Recall*) vahed olid minimaalsed. Lisaks ei olnud erinevust algoritmide F-skooridel.

```
=== Summary ===

Correctly Classified Instances         684               77.3756 %
Incorrectly Classified Instances       200               22.6244 %
Kappa statistic                          0.3077
Mean absolute error                      0.3143
Root mean squared error                  0.4042
Relative absolute error                 55.3025 %
Root relative squared error             69.5944 %
Total Number of Instances              884

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0,917    0,650    0,807      0,917    0,858      0,324    0,684     0,828     TRUE
                 0,350    0,083    0,586      0,350    0,438      0,324    0,684     0,516     FALSE
Weighted Avg.    0,774    0,507    0,751      0,774    0,752      0,324    0,684     0,749

=== Confusion Matrix ===

   a   b   <-- classified as
 606  55 |   a = TRUE
 145  78 |   b = FALSE
```

Joonis 16. Otsustuspuu REPTree tulemused.

```
Time taken to test model on supplied test set: 0 seconds

=== Summary ===

Correctly Classified Instances         685               77.4887 %
Incorrectly Classified Instances       199               22.5113 %
Kappa statistic                          0.3052
Mean absolute error                      0.3244
Root mean squared error                  0.4077
Relative absolute error                 57.0914 %
Root relative squared error             70.2062 %
Total Number of Instances              884

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0,921    0,659    0,806      0,921    0,860      0,324    0,666     0,818     TRUE
                 0,341    0,079    0,594      0,341    0,433      0,324    0,666     0,493     FALSE
Weighted Avg.    0,775    0,513    0,752      0,775    0,752      0,324    0,666     0,736

=== Confusion Matrix ===

   a   b   <-- classified as
 609  52 |   a = TRUE
 147  76 |   b = FALSE
```

Joonis 17. Otsustuspuu J48 tulemused.

Järgnevalt vaadeldi täpsemalt REPTree (Joonis 18) ning J48 (Joonis 19) algoritimide loodud kirjeldavaid mudeleid. Olenemata otsustuspuude peaaegu identsetest ennustamistäpsustest on näha esialgu otsustuspuude kirjeldustes erinevusi. Kuigi mõlema algoritmi poolt loodud mudelid alustavad tellimuste klassifitseerimist makseviisidest ning ennustavad mobiilimaksega seotud tellimused automaatselt pikendatuks, siis ülejäänud harud erinevad üksteisest märgatavalt. Kirjeldustest avaldub, et tellimuste pikendamiste seosed lahknevad makseviiside alusel ning enamik ennustamisest toimub tellimuse andmete põhjal. Väiksemas osas prognoositakse pikendamisi kliendi või kliendi eelnevate tellimuste andmetel.

Kuigi REPTree vastupidiselt J48 mudelile proovib täpsustada krediitkaardiga tasutud tellimuste pikendamist, siis on näha, et puu sügavamas osas ennustatakse kõik peale väga täpse vahemiku pikendatuks. „Muu“ makseviisiga tellimused on tasuta tellimused. Andmete analüüsist tuli välja, et nendest pikendatakse vaid 1,4%. J48 on ennustanud kõik tasuta tellimused mitte pikendatuks, samas REPTree proovib leida need üksikud tellimused, mis pikendatakse.

Joonisel 18 on näha, et REPTree mudelis arvega makstes pikendatakse tellimusi, millel on üle ühe digiõiguse ning maksavad vähemalt 34 eurot. Joonisel 19 kirjeldab J48 algoritmiga loodud mudel arvega makstud tellimusi täpsemalt. Proovitellimused prognoositakse mitte pikendatuks, aga pikendustellimuste puhul on tuvastatud mõningad

seosed. Näiteks on leitud, et kõik üle 32 aastased naised pikendavad pikendustellimusi üsna tõenäoliselt, meeste puhul arvestatakse pikendajateks need, kelle vanus jääb vahemikku 32 – 57.

```
MAKSEVIIS = TEL_MAKSEVIIS.ARVE
|   MAKSUMUS < 33.5 : FALSE (845/53) [428/30]
|   MAKSUMUS >= 33.5
|   |   DIGIOIGUSI < 1.5 : FALSE (240/94) [117/44]
|   |   DIGIOIGUSI >= 1.5 : TRUE (7/0) [2/0]
MAKSEVIIS = TEL_MAKSEVIIS.MOBIILIMAKSE : TRUE (675/114) [333/48]
MAKSEVIIS = TEL_MAKSEVIIS.MUU
|   JURIIDILINE = 0 : FALSE (482/2) [251/6]
|   JURIIDILINE = 1
|   |   PROOVI_AP < 1.5 : FALSE (37/2) [19/0]
|   |   PROOVI_AP >= 1.5 : TRUE (2/1) [1/0]
MAKSEVIIS = TEL_MAKSEVIIS.KREDIITKAART
|   MAKSNUD_AP < 5.5
|   |   MAKSUMUS < 1.5 : TRUE (194/61) [94/25]
|   |   MAKSUMUS >= 1.5
|   |   |   MAKSNUD_AP < 4.5 : TRUE (81/39) [40/19]
|   |   |   MAKSNUD_AP >= 4.5 : FALSE (5/0) [1/0]
|   MAKSNUD_AP >= 5.5 : TRUE (87/16) [35/12]
MAKSEVIIS = TEL_MAKSEVIIS.EA_PML : FALSE (49/20) [28/11]
MAKSEVIIS = TEL_MAKSEVIIS.PANGALINK
|   MAKSUMUS < 1.5 : FALSE (16/0) [4/0]
|   MAKSUMUS >= 1.5
|   |   TELLIMUSI_AP < 0.5 : FALSE (12/1) [7/2]
|   |   TELLIMUSI_AP >= 0.5
|   |   |   VANUS < 45.5 : TRUE (4.5/0) [5/2]
|   |   |   VANUS >= 45.5 : FALSE (4.5/0.5) [6/0]

Size of the tree : 27
```

Joonis 18. REPTree otsustuspuu visuaalne kirjeldus.

```
MAKSEVIIS = TEL_MAKSEVIIS.ARVE
|   KAS_PROOVITELLIMUS = 0
|   |   KAS_PIKENDUSTELLIMUS = 0: FALSE (156.0/50.0)
|   |   KAS_PIKENDUSTELLIMUS = 1
|   |   |   VANUS <= 32
|   |   |   |   SUGU_XV = SUGU.MEES
|   |   |   |   |   JURIIDILINE = 0
|   |   |   |   |   |   MAKSUMUS <= 120: TRUE (2.25/0.81)
|   |   |   |   |   |   MAKSUMUS > 120: FALSE (5.24/1.81)
|   |   |   |   |   JURIIDILINE = 1
|   |   |   |   |   |   PROOVI_AP <= 0
|   |   |   |   |   |   |   DIGIOIGUSI <= 1: FALSE (41.38/17.67)
|   |   |   |   |   |   |   DIGIOIGUSI > 1: TRUE (2.32)
|   |   |   |   |   |   PROOVI_AP > 0: FALSE (12.24/3.1)
|   |   |   |   SUGU_XV = SUGU.NAINE: FALSE (18.48/6.91)
|   |   |   VANUS > 32
|   |   |   |   SUGU_XV = SUGU.MEES
|   |   |   |   |   VANUS <= 57: TRUE (11.51/0.97)
|   |   |   |   |   VANUS > 57: FALSE (2.3/1.11)
|   |   |   |   SUGU_XV = SUGU.NAINE: TRUE (5.26/0.16)
|   KAS_PROOVITELLIMUS = 1: FALSE (822.0/52.0)
MAKSEVIIS = TEL_MAKSEVIIS.MOBIILIMAKSE: TRUE (679.0/116.0)
MAKSEVIIS = TEL_MAKSEVIIS.MUU: FALSE (533.0/7.0)
MAKSEVIIS = TEL_MAKSEVIIS.KREDIITKAART: TRUE (353.0/113.0)
MAKSEVIIS = TEL_MAKSEVIIS.EA_PML
|   ON_TMO_KEELD = 0
|   |   JURIIDILINE = 0: FALSE (37.0/11.0)
|   |   JURIIDILINE = 1
|   |   |   MAKSNUD_AP <= 483: FALSE (15.0/6.0)
|   |   |   MAKSNUD_AP > 483: TRUE (4.0)
|   ON_TMO_KEELD = 1: TRUE (2.0)
MAKSEVIIS = TEL_MAKSEVIIS.PANGALINK: FALSE (40.0/8.0)

Number of Leaves  :     19

Size of the tree :      33
```

Joonis 19. Otsustuspuu J48 visuaalne kirjeldus.

3.5 Järeldused

Otsustuspuude kirjelduste uurimisel tundus esialgu, et puud on üsnagi erinevad. Siiski on näha, et otsustuspuude erinevaid osi kasutatakse väiksema arvu tellimuste ennustamisel, mille tõttu on ka ennustamistulemused sarnased. Võib järeldada, et ainult tellimuste makseviisi teades saaks üsna täpse ennustuse. Mudeleid on võimalik mõneti kasutada ennustamisel ning seoste leidmisel, kuid klassifitseerimiseks olid kindlad seosed kliendi harjumuste ja pikendamiste vahel nõrgad.

Täpsemate mudelite loomiseks võiks abiks olla andmete täpsem jaotamine, näiteks võiks vaadata eraldi tasuta ning tasulisi tellimusi, samuti juriidilisi- ning eraisikuid. Ühtlasi võiks abiks tulla suurema andmestiku kasutamine, mille rakendamist antud töös takistas eesmärk uurida seoseid kasutajate aktiivsuse veebis ning tellimuse pikendamise vahel.

Kuna kindlaid seoseid pikendajate ja kasutaja harjumuste andmetega oli vähe, siis aitaks ärilisi probleeme lahendada numbriline ennustamine, kus igale tellimusele arvutatakse pikendamise tõenäosus. Töös kasutati klassifitseerimis mudeleid, sest nende abil on lihtsam andmetest aru saada ning näha, milliste otsuste tagajärjel tulemuseni jõutakse.

Tasuta tellimuste 1,4 protsendiline pikendamise protsent viitab nende proovitellimuste ebaefektiivsusele. Seega on nende tellimuste pikkuste ennustamise vajalikkus kaheldav, vaid tuleks leida probleemid kogu tasuta tellimuste jagamise protsessis ning järeltegevustes.

Püsimaksega (mobiili- ja krediitkaardiga makstud) tellimusi pikendati valitud ajavahemikus kõige parema protsendiga. Samas võib otsustuspuu kirjeldusi analüüsides järeldada, et nende tellimuste puhul on seoste leidmine kõige keerulisem. Kui muude makseviiside puhul tuleb enne ülekande sooritamist teha otsus, kas tellimus on endiselt raha väärt, siis püsimakse puhul toimub maksmise protsess automaatselt kui klient pole tellimust tühistanud vahepeal, mis võib olla põhjenduseks nõrgemate seoste vahel.

4 Kokkuvõte

Esimeseks töö eesmärgiks oli uurida seoseid digitellimuste pikendamiste ning kasutajate aktiivsuse andmete vahel. Eesmärgi saavutamiseks võeti kasutajate kasutusandmed (Tabel 2) tellimuse viimase 30 päeva kohta ning prooviti ennustada ainult nende põhjal tellimuste pikendamist. Ennustamiseks kasutati andmekaeve tarkvara Weka. Ükski kasutatud algoritmidest ei suutnud ennustada soovitud täpsusega, seega valitud andmestikus ei olnud seoseid tellimuste pikendamiste ning kasutaja aktiivsuse vahel Äripäeva veebilehel www.aripaev.ee. Võimalike põhjustena toodi välja andmete puudumine suurel hulgal (Tabel 3) ning püsimaksetega tellimuste osakaalu valitud andmestikus.

Lisaks olid töö eesmärkideks saada parem ülevaade Äripäeva digitellimustest ning luua tellimuste pikendamisi ennustavad mudelid. Eesmärkide saavutamiseks analüüsiti alustuseks tellimustega seotud andmeid andmeanalüüsiga Qlikview. Andmete kirjeldamiseks ja visualiseerimiseks kasutati tabeleid ning tulpdiagramme. Järgnevalt tehti andmekaeve tarkvarale andmestik loetavaks. Seejärel tutvuti erinevate algoritmide omaduste ning tulemustega, et valida eesmärgi täitmiseks sobivaim mudel. Lõpuks loodi kaks otsustuspuud, kasutades Weka masinõppe algoritme J48 ja REPTree, mis ennustavad digitellimuste pikendamisi. Otsustuspuude kirjelduste analüüsimisel selgus, et kõige tugevamad seosed tellimuste pikendamisega leiti makseviisist, mille põhjal on küllaltki täpselt võimalik ennustada.

Kahe erineva andmestiku kasutamine mudelites näitas, kui palju võivad andmed ning seetõttu ka ennustamise efektiivsus ajas muutuda. Järelikult on oluline otsustuspuid aja jooksul pidevalt täiendada.

Kasutatud kirjandus

[1] „Firmast“, *Äripäev*. [Võrgumaterjal]. Available at: http://firma.aripaev.ee/firmast/. [Vaadatud: 17.04.2018].

[2] „Äripäeva tellimine“. [Võrgumaterjal]. Available at: https://www.aripaev.ee/tellimisviisid/. [Vaadatud: 17.04.2018].

[3] M. Lindsay, X. V. Leeuwe, ja M. V. D. Peppel, *How To Succeed in the Relationship Economy: Make Data Work for You, Empathise with Customers, Grow Valuable Relationships*. Charleston, SC: Advantage Media Group, 2017.

[4] „e-Teatmik: IT ja sidetehnika seletav sõnaraamat“. [Võrgumaterjal]. Available at: http://vallaste.ee/index.htm. [Vaadatud: 20.05.2018].

[5] R. Wirth, „CRISP-DM: Towards a standard process model for data mining“, *Proceedings of the Fourth International Conference on the Practical Application of Knowledge Discovery and Data Mining*, 2000, lk 29–39.

[6] A. Azevedo ja M. F. Santos, „KDD, SEMMA AND CRISP-DM: A PARALLEL OVERVIEW“, lk 6.

[7] J. D. Kelleher, B. M. Namee, ja A. D’Arcy, *Fundamentals of Machine Learning for Predictive Data Analytics: Algorithms, Worked Examples, and Case Studies*, 1 edition. Cambridge, Massachusetts: The MIT Press, 2015.

[8] P. C. J. Navas, Y. C. G. Parra, ja J. I. R. Molano, „Big Data Tools: Haddop, MongoDB and Weka“, *Data Mining and Big Data*, 2016, lk 449–456.

[9] E. Frank, M. A. Hall, ja I. H. Witten, *The WEKA Workbench. Online Appendix for „Data Mining: Practical Machine Learning Tools and Techniques“*, Fourth. 2016.

[10] RJ Podeschi, *Experiential Learning using QlikView Business Intelligence Software*. 2014.

[11] I. H. Witten ja E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*, 1 edition. San Francisco, Calif: Morgan Kaufmann, 1999.

[12] S. Eriksen ja L. R. Keller, „Decision trees“, *Encyclopedia of Operations Research and Management Science*, Dordrecht: Kluwer Academic Publishers, 2001, lk 202–205.

[13] Y. SONG ja Y. LU, „Decision tree methods: applications for classification and prediction“, *Shanghai Arch. Psychiatry*, kd 27, nr 2, lk 130–135, apr 2015.

[14] „HoeffdingTree“. [Võrgumaterjal]. Available at: http://weka.sourceforge.net/doc.dev/weka/classifiers/trees/HoeffdingTree.html. [Vaadatud: 18.05.2018].

[15] „RandomTree“. [Võrgumaterjal]. Available at: http://weka.sourceforge.net/doc.dev/weka/classifiers/trees/RandomTree.html. [Vaadatud: 18.05.2018].

[16] „REPTree“. [Võrgumaterjal]. Available at: http://weka.sourceforge.net/doc.dev/weka/classifiers/trees/REPTree.html. [Vaadatud: 18.05.2018].

[17] „J48“. [Võrgumaterjal]. Available at: http://weka.sourceforge.net/doc.dev/weka/classifiers/trees/J48.html. [Vaadatud: 18.05.2018].
[18] S. L. Salzberg, „C4.5: Programs for Machine Learning by J. Ross Quinlan. Morgan Kaufmann Publishers, Inc., 1993“, *Mach. Learn.*, kd 16, nr 3, lk 235–240, sept 1994.
[19] N. Donges, „The Random Forest Algorithm“, *Towards Data Science*, 22-veebr-2018. [Võrgumaterjal]. Available at: https://towardsdatascience.com/the-random-forest-algorithm-d457d499ffcd. [Vaadatud: 06.05.2018].
[20] „RandomForest“. [Võrgumaterjal]. Available at: http://weka.sourceforge.net/doc.dev/weka/classifiers/trees/RandomForest.html. [Vaadatud: 18.05.2018].
[21] „KStar“. [Võrgumaterjal]. Available at: http://weka.sourceforge.net/doc.dev/weka/classifiers/lazy/KStar.html. [Vaadatud: 18.05.2018].
[22] „IBk“. [Võrgumaterjal]. Available at: http://weka.sourceforge.net/doc.dev/weka/classifiers/lazy/IBk.html. [Vaadatud: 18.05.2018].
[23] „Naive Bayesian“. [Võrgumaterjal]. Available at: http://www.saedsayad.com/naive_bayesian.htm. [Vaadatud: 18.05.2018].
[24] „NaiveBayes“. [Võrgumaterjal]. Available at: http://weka.sourceforge.net/doc.dev/weka/classifiers/bayes/NaiveBayes.html. [Vaadatud: 18.05.2018].
[25] P. Chandrayan, „Machine Learning Part 3 : Logistic Regression“, *Towards Data Science*, 26-aug-2017. [Võrgumaterjal]. Available at: https://towardsdatascience.com/machine-learning-part-3-logistics-regression-9d890928680f. [Vaadatud: 06.05.2018].
[26] „Logistic“. [Võrgumaterjal]. Available at: http://weka.sourceforge.net/doc.dev/weka/classifiers/functions/Logistic.html. [Vaadatud: 18.05.2018].
[27] „SimpleLogistic“. [Võrgumaterjal]. Available at: http://weka.sourceforge.net/doc.dev/weka/classifiers/functions/SimpleLogistic.html. [Vaadatud: 18.05.2018].
[28] R. Kohavi, „The Power of Decision Tables“, *Proc. Eur. Conf. Mach. Learn.*, sept 1997.
[29] „DecisionTable“. [Võrgumaterjal]. Available at: http://weka.sourceforge.net/doc.dev/weka/classifiers/rules/DecisionTable.html. [Vaadatud: 18.05.2018].
[30] T. Fawcett, „An introduction to ROC analysis“, *Pattern Recognit. Lett.*, kd 27, nr 8, lk 861–874, juuni 2006.
[31] C. N. Knaflic, *Storytelling with Data: A Data Visualization Guide for Business Professionals*, 1 edition. Hoboken, New Jersey: Wiley, 2015.

Lisa 1 – Andmete päringu skript

```
Tellimused:
SQL SELECT t.Tellimuse_kood, t.VaIndeks, Periood, TellimKpv, Maksja_kood,
       t.Algus, t.Lopp, Maksumus, Digioigusi, k.Kampaania_kood,
       Kas_Proovitellimus, Kas_Pikendustellimus, Makseviis,
       Vana_Tellimuse_kood, Automaatne_pikenemine, Annulleeritud, Sugu,
       Juriidiline, On_Tmo_Keeld, Sektor, SSO_email, SSO.Algus AS SSO_Algus,
       SSO.Lopp AS SSO_Lopp, Hash,
       ROUND((SYSDATE - Synni_Kpv)/365, 1) AS Vanus,
       (SELECT COUNT(*)
       FROM Tellimus AS t_valim
       WHERE t_valim.Maksja_kood=t.Maksja_kood
              AND t_valim.Algus < t.Algus
              AND t_valim.Algus > ADD_MONTHS(t.Lopp, -12*5)
              AND t_valim.VaIndeks IN ('WAP', 'AP', 'APM', 'APPAB')
              AND t_valim.Kinnitatud = 1
              AND t_valim.Annulleeritud IS NULL) AS Tellimusi_AP,
       (SELECT SUM(Maksumus)
       FROM Tellimus AS t_valim
       WHERE t_valim.Maksja_kood=t.Maksja_kood
              AND t_valim.Algus < t.Algus
              AND t_valim.Algus > ADD_MONTHS(t.Lopp, -12*5)
              AND t_valim.VaIndeks IN ('WAP', 'AP', 'APM', 'APPAB')
              AND t_valim.Kinnitatud = 1
              AND t_valim.Annulleeritud IS NULL ) AS Maksnud_AP,
       (SELECT SUM(Lopp-Algus)
       FROM Tellimus AS t_valim
       WHERE t_valim.Maksja_kood=t.Maksja_kood
              AND t_valim.Algus < t.Algus
              AND t_valim.Algus > ADD_MONTHS(t.Lopp, -12*5)
              AND t_valim.VaIndeks IN ('WAP', 'AP', 'APM', 'APPAB')
              AND t_valim.Kinnitatud = 1
              AND t_valim.Annulleeritud IS NULL ) AS Pikkus_AP,
       (SELECT COUNT(*)
       FROM Tellimus AS t_valim LEFT JOIN Kampaania AS k_valim ON
              t_valim.Kampaania_kood = k_valim.Kampaania_kood
       WHERE t_valim.Maksja_kood=t.Maksja_kood
              AND t_valim.Algus < t.Algus
              AND t_valim.Algus > ADD_MONTHS(t.Lopp, -12*5)
              AND t_valim.VaIndeks IN ('WAP', 'AP', 'APM', 'APPAB')
              AND t_valim.Kinnitatud = 1
              AND t_valim.Annulleeritud IS NULL
              AND k_valim.kas_Proovitellimus = 1) AS Proovi_AP
FROM Tellimus AS t
       LEFT JOIN Kampaania AS k ON t.Kampaania_kood = k.Kampaania_kood
```

```
	LEFT JOIN Klient AS kl ON t.Maksja_kood = kl.Kliendi_kood
	LEFT JOIN SSO ON t.Tellimuse_kood = SSO.Tellimuse_kood
WHERE t.VaIndeks = ('WAP')
	AND Kinnitatud = 1
	AND Annulleeritud IS NULL
	AND t.Lopp >= '2017-09-01'
	AND t.Lopp < '2018-03-01';

Proov_pikendus:
SQL WITH valim AS (
	SELECT Digikonto_id, Tellimuse_kood, t.Lopp, t.Algus, t.VaIndeks
	FROM Tellimus AS t LEFT JOIN Kampaaniak ON t.Kampaania_kood =
		k.Kampaania_kood
	WHERE t.VaIndeks IN ('AP','WAP','APM', 'APPAB')
		AND t.Kinnitatud = 1
		AND t.Annulleeritud IS NULL
		AND t.Lopp >= '2017-11-01'
), proovid AS (
	SELECT t1.Digikonto_id, t1.Tellimuse_kood AS Vana_Tellimuse_kood,
		t1.VaIndeks AS VanaToode, t1.Lopp AS VanaLopp,
		MIN(t2.Tellimuse_kood) AS Uus_Tellimuse_kood
	FROM valim AS t1 LEFT JOIN valim AS t2 ON t1.Digikonto_id =
		t2.Digikonto_id
	WHERE t1.Lopp >= '2017-11-01'
		AND t1.Lopp < SYSDATE
		AND t2.Algus > t1.Lopp
		AND ADD_MONTHS(t1.Lopp, 1) >= t2.Algus
	GROUP BY t1.Digikonto_id, t1.Tellimuse_kood, t1.VaIndeks, t1.Lopp
)
SELECT Vana_Tellimuse_kood,
	Uus_Tellimuse_kood
FROM proovid;

Pikendatud:
LOAD Vana_Tellimuse_kood AS Tellimuse_kood,
	If(len(Uus_Tellimuse_kood)>0, 1) AS Pikendatud
Resident Proov_pikendus;

DROP TABLE Proov_pikendus;

Events:
LOAD UPPER(Hash) AS Hash,
	Timestamp,
	EventType
FROM [C:\Qlikview\Desing\Reimo\Kasutusandmed.qvd]
(qvd)
WHERE SiteCode = 'EA' OR ISNULL(SiteCode) ;

RIGHT JOIN (Events)
LOAD DISTINCT Tellimuse_kood,
	Hash,
```

```
       Algus AS Tellimuse_Algus,
       Lopp AS Tellimuse_Lopp
RESIDENT Tellimused;

NOCONCATENATE
ResultTable:
 LOAD Tellimuse_kood, EventType,
       COUNT(Hash) AS Kordi_30p
RESIDENT Events
WHERE Timestamp <= Tellimuse_Lopp
       AND Timestamp >= (Tellimuse_Lopp - 30)
GROUP BY Tellimuse_kood, EventType;

DROP TABLE Events;

GENERIC LOAD * Resident ResultTable;

DROP TABLE ResultTable;
```