TARTU ÜLIKOOL
Loodus- ja täppisteaduste valdkond
Arvutiteaduse instituut
Andmeteaduse õppekava

Martti Praks

Masinõppe rakendamine makseviivituse tõenäosuse hindamisel

Magistritöö (15 EAP)

Juhendajad: Markus Kängsepp, MSc
Meelis Kull, PhD
Kuldar Kõiv, MSc

Tartu 2022

Lühikokkuvõte

Masinõppe rakendamine makseviivituse tõenäosuse hindamisel

Makseviivituse tõenäosuse hindamine on finantsasutusel üheks võtmetegevuseks krediidiriski hindamisel. Makseviivituse tõenäosus on oluline tunnus, mille pealt otsustatakse kas ja mis tingimustel krediiti anda ning jälgitakse kogu krediiditoodete portfelli kvaliteeti. Üldistatult saab kasutatavad mudelid jagada kaheks: statistilised lähenemised ja masinõppe tehnikad. Magistritöö peamisteks tulemusteks on võrdlus logistilise regressiooni ja teiste masinõppemeetoditega loodud mudelite vahel, kasutades AS LHV Group'i reaalseid andmeid.

Töös demonstreeritakse erinevate meetoditega saavutatud makseviivituse hindamismudeli tulemeid ja arutletakse erinevate meetodite eeliste üle. Parima tulemuse saavutas mõõdikute alusel otsustuspuu algoritmil põhinev otsustusmets. Töös rakendatakse erinevaid meetodeid otsustusmetsa mudeli seletamiseks toetades selle meetodi rakendamist praktikas. Arvestades viimasel ajal erinevate otsustuspuu meetodil põhinevate masinõppemeetodite edukust paljudes valdkondades, ei ole saavutatud tulemused üllatuslikud. Otsustusmetsa ennustusi seletatakse läbi mudeli üldiste seoste andmestiku tunnustega ja konkreetsemalt näitlikustatakse erinevate näidete ennustuse kujunemist. Kas need tulemid on piisavad, et praktikas otsustusmetsa kasutada, jäetakse lõppkasutaja otsustada.

Võtmesõnad

masinõpe, makseviivitus, logistiline regressioon, tugivektormasin, otsustusmets, tehisnärvivõrk, LIME, tõenäosuste kalibreerimine

CERCS

P176 - Tehisintellekt

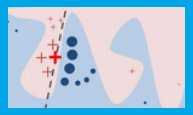
Masinõppe rakendamine makseviivituse tõenäosuse hindamisel

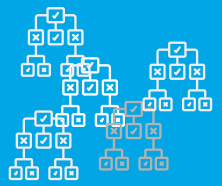
Praktiline eksperiment levinud meetodite võrdlemiseks AS LHV Group'i andmetel
Andmeteaduse õppekava magistritöö (15 EAP) visuaalne kokkuvõte


tehisnärvivõrk


logistiline regressioon


tõenäosuste kalibreerimine


LIME


otsustusmets
xgboost


otsustuspuu

LHV

krediidilepingud

10779 vaatlust

88 makseviivitust

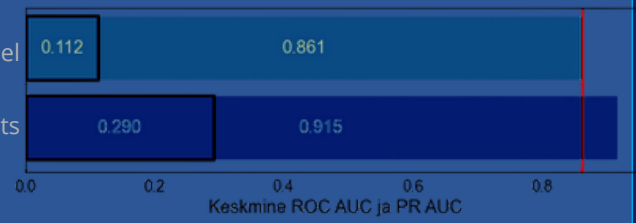
25 tunnust

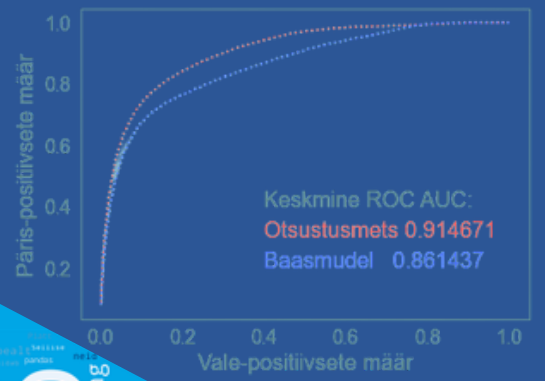
Meetodid

Andmed

Tulemused

Kirjalik töö





Autor:
Martti Praks

Juhendajad: **Markus Kängsepp**, MSc, Arvutiteaduse instituut, Tartu Ülikool
Meelis Kull, PhD, Arvutiteaduse instituut, Tartu Ülikool
Kuldar Kõiv, MSc, AS LHV group

#UniTartuCS

TARTU ÜLIKOOL
arvutiteaduse instituut

LHV

Abstract

Using Machine Learning for Default Prediction

Default prediction is one of the key activities for a financial institution when estimating credit risks. The likelihood of default is an important indicator to decide if and with what conditions credit can be given and how the whole credit portfolio is performing. In general, used models are divided into two domains: statistical approaches and machine learning techniques. The main result of the thesis is a comparison between models created with logistic regression and other main machine learning techniques using actual data from AS LHV Group.

The thesis displays different default prediction models and includes discussions over benefits provided by different machine learning techniques. Best results are achieved with the random forest method. Different methods are used to explain the decision-making mechanisms of the random forest model to support using it in practice. Considering recent successes of decision tree-based models, the results are not surprising. Random forest results are explained by feature influences and concrete examples of why some probabilities were provided. Whether these methods are enough to use the random forest in practice, is left up to decide by the end-user.

Keywords

machine learning, default, logistic regression, support vector machine, random forest, neural network, LIME, probability calibration

CERCS

P176 - Artificial intelligence

Sisukord

Jooniste loend

1 Sissejuhatus

Makseviivituse tõenäosus on oluliseks indikaatoriks finantsasutusele, kas väljastatud laen makstakse tagasi. See koondab kokku erinevad andmepunktid ja annab hea ülevaate kui tõenäoline on väljastatud laenu tagasi maksmine. Masinõpe on viimasel ajal kiirelt arenev valdkond, mis aina enam mõjutab igapäeva elu. Olgu selleks rämpsposti vähendamine, võõrkeelse teksti tõlkimine nutitelefoni kaamera abil või õnnetusele kiirem reageerimine, sest häirekeskuse ennustus oli edukas. Masinõppe valdkonnas avaldatakse palju uurimustöid ja leiutatakse uusi meetodeid, mida rakendada. See magistritöö paneb kaks valdkonda kokku ja rakendab viimase aja tulemusi masinõppes makseviivituse tõenäosuse arvutamiseks.

Töö kasutab reaalseid andmeid AS LHV Group finantslepingutest, valmistab need ette töötluseks, kirjeldab peamisi rakendatavaid meetodeid, rakendab erinevaid masinõppe meetodeid makseviivituse tõenäosuse hindamiseks ja võrdleb erinevate meetodite saavutatud tulemusi. Seega töö peamine väärtus on teooria praktiline rakendamine ja lisaväärtuseks sissejuhatus valdkonda.

Makseviivituse ja masinõppe valdkonnad on mahukad, millest tingitult tehti valikuid sobiliku alamosa leidmiseks. Fookus langes portfellimudelitele, millega finantsasutus hindab regulaarselt kõigi kehtivate finantseerimislepingute makseviivituste tõenäosust. Finantseerimisotsuse raames makseviivituse tõenäosuse hindamist, millele rakenduvad keerukamad regulatsioonid, töös ei käsitleta. Andmete piirangutest lähtuvalt kasutatakse väikse ja keskmise suurusega ettevõtete kindlaid laenutooteid. Hoidmaks tööd avalikuna ja kaitsmaks AS LHV Group'i ärisaladust piirdutakse minimaalse andmestiku kirjeldusega, mis omakorda piirab töös tehtavaid järeldusi üldisele tasemele.

Kirjeldatud piirangute ja valikute vahel sai töö peamiseks panuseks erinevate algoritmide rakendamine ja nende tulemuste võrdlemine erinevates olulistes aspektides, mis peaksid mõjutama otsust, millist algoritmi rakendada. Selline tulemus aitab luua intuitsiooni ja põhimõtteid makseviivituse valdkonnas algoritmide praktikas rakendamisel.

Töö koosneb teoreetilisest taustast, eksperimendi ülesehitusest, tulemuste kirjeldusest ja arutelust. Lisade all on toodud välja täiendav ja detailsem abimaterjal. Teoreetiline taust annab ülevaate peamistest meetoditest, tehes lühikokkuvõtte ja andes viited täpsematele materjalidele. Need kirjeldused on loodud mõeldes valdkonnas vähemalt baasteadmisi omavatele lugejatele. Eksperimendi ülesehitus kirjeldab kogu valdkonna tausta lähtudes CRISP-DM meetodi raamistikust. See peatükk seob kokku lahendatava probleemi tausta ja kuidas seda probleemi lahendati. Tulemuste peatükk kirjeldab peamisi saadud tulemusi keskendudes baasalgoritmi ja parimaid mõõdetavaid tulemusi pakkunud algoritmi võrdlusele. Mahu piirangust tingitult on joonistel kajastatu ainult autori poolt hinnatult olulisem ja lisade alt on leitav osa ülejäänud tehtud töö detailidest. Arutelu osas kirjeldatakse tulemustest tehtud üldistusi. Jätkutööd, mis autori hinnangul annavad olulist lisandväärtust tööle, aga mis eelkõige ajalise piirangu tõttu on kõrvale jäänud, on loetletud kokkuvõttes. Lisade alt on leitav sõnastik, mis kirjeldab olulisemaid

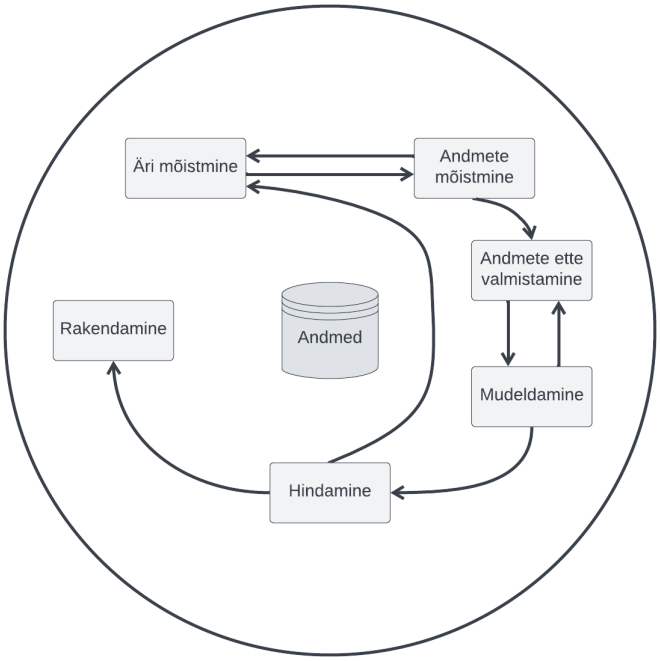
termineid ja kasutatud lühendeid. Masinõppe valdkonnas on hetkel eestikeelse materjali hulk vähene ja paljudel terminitel on tõlked puudulikud või kohmakad. Toetamaks võõrkeelsete terminitega harjunud lugejal materjali mõistmist on peamistel terminitel esmakordsel kasutamisel ja sõnastikus välja toodud inglise keelne termin.

2 Teoreetiline taust

Toetamaks tehtud töö sisu mõistmist, annab järgneva peatükk lühiülevaate peamiste töös rakendatud meetodite sisust ja miks need meetodid valiti. Kuidas need töös kokku sobituvad ja mis tulemused nendega saavutati, on kirjeldatud järgmistes peatükkides.

2.1 CRISP-DM standard

Kogu järgnev CRISP-DM standardi kirjeldus ja joonis on kokkuvõte selle metoodika alusdokumendist (Chapman *et al.* 2000).



Joonis 1: CRISP-DM peamised faasid ja nende seosed

CRISP-DM on standard organiseerimaks andmekaeve tegevusi. See on kirjeldatud läbi hierarhilise protsessi mudeli, kus on loetletud erinevad faasid, nende peamised ja detailsemad tegevused ning faaside väljundid. Selle standardi loomise eestvedajateks olid ettevõtted DaimlerChrysler, SPSS (*Statistical Product and Service Solutions*) ja NCR (*National Cash Register*), kes alustasid metoodikat arendusega 1996 ning avalikustasid esimese versiooni 2000. aastal. Nad ise rõhutavad, et metoodika on arendatud tuginedes praktilistele kogemustele, kuidas andmekaeve projekte teostada. Hea ülevaate selle standardi sisust saab peamiste faaside ja nendes sisalduvate tegevuste nimekirjast:

1. **Äri mõistmine** - taust, ärilised eesmärgid, edu kriteerium, ressursside inventar, nõuded, eeldused ja piirangud, riskid ja ettenägematused, terminoloogia, kulud ja tulud, andmekaeve eesmärgid, andmekaeve edu kriteerium, projekti plaan

2. **Andmete mõistmine** - esialgne andmete kogumise raport, andmete kirjelduse raport, andmete kvaliteedi raport, uurimusliku analüüsi raport

3. **Andmete ette valmistamine** - andmed ja andmete kirjeldus

4. **Mudeldamine** - disaini testimine, mudelid, parameetrite seadistus, mudeli kirjeldus, tulemuste hindamine

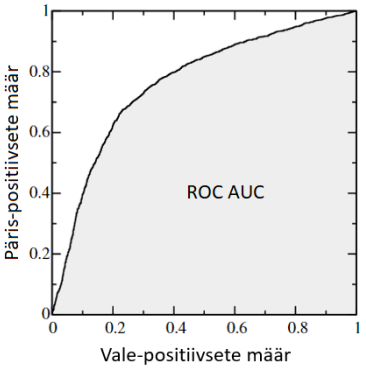
5. **Hindamine** - hindamine lähtudes ärilistest edu eesmärkidest, protsessi ülevaatamine, järgmised sammud

6. **Rakendamine** - kasutuselevõtukava, hoolduskava, lõplik raport ja esitlemine, kogemuse dokumenteerimine

Nende peamiste faaside omavahelised seosed on kirjeldatud joonisel 1. Tänaseks päevaks on see metoodika edasi arenenud saades erinevaid täiendusi ja täpsustusi, aga on üldiselt jätkuvalt praktikas laialdaselt rakendatud metoodika. Üheks heaks näiteks metoodika edasi arendustest on CASP-DM (Martínez-Plumed *et al.* 2017), mis adresseerib masinõppe ja andmekaeve väljakutseid konteksti ja mudeli taaskasutatavuses.

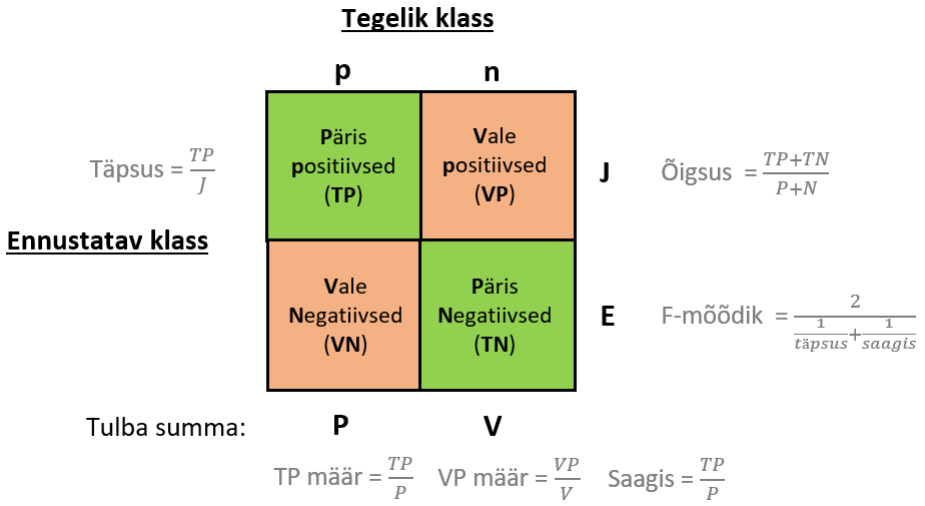
2.2 ROC AUC mõõdik tulemuste võrdlemiseks

ROC graafik (*ROC graph*, pikemalt *receiver operating characteristics graph* ehk otsetõlkes vastuvõtja tööomaduste graafik) on tehnika klassifikaatori tulemuste visualiseerimiseks, organiseerimiseks ja valimiseks soorituse alusel (Fawcett 2005). See metoodika töötati välja 20. sajandi keskpaigas signaali tuvastamise teoorias ja on nüüdseks leidnud laialdast kasutamist masinõppe kogukonnas. ROC graafikul on mitmeid eeliseid lihtsalt mudeli õigsuse (*accuracy*) ees ja neid eelistatakse eriti andmestike puhul, kus binaarsel ennustamisel on klasside näidised ebaproportsionaalselt erinevad või klassidel on erinev oodatav eksimisviga, nagu näiteks makseviivituse tõenäosuse hindamine.



Joonis 2: ROC graafiku näide

ROC graafik visualiseerib mudeli ennustuste segadusmaatriksit *(confusion matrix)*. Segadusmaatriksis kujutatakse mudeli päris-positiivsete *(true positive)*, vale-positiivsete



Joonis 3: Segadusmaatriks ja levinud mudeli soorituse hindamise mõõdikud

(false positive), vale-negatiivsete *(false negative)* ja päris-negatiivsete *(true negative)* ennustuste koguarvu. Selle erinevate veergude ja tulpade pealt arvutatakse soorituse hindamise mõõdikuid nagu õigsus *(accuracy)*, täpsus *(precision)*, saagis *(recall)* ja F-mõõdik *(f-measure)*. Nende mõõdikute sisu ja seos segadusmaatriksiga on nähtav joonisel 3. ROC graafik on kahe-dimensiooniline graafik, kus Y-teljel kujutatakse päris-positiivsete määra ja X-teljel vale-positiivsete määra. ROC graafiku näide on nähtav joonisel 2. Kasutades mudeli ennustuses kajastuvaid tõenäosusi on võimalik mudeli tõenäosuse hinnangute erinevatel tasemetel hinnata päris-positiivsete ja vale-positiivsete määra ning joonistada need ROC graafikule. Ühendades punktid järjestikku tekib nii nimetatud ROC kõver *(ROC curve)*. Sellisel ROC kõveral on huvitav ja kasulik omadus - see on sõltumatu klasside vahelisest jaotusest, sest kasutab ainult positiivse klassi näidiseid ja ei sõltu selle klassi määrast kogu andmestikku. See omadus teeb ROC kõvera kasulikuks ebaproportsionaalsete klasside jaotuse korral.

ROC AUC (ROC kõvera alune pindala, pikemalt *Receiver Operating Characteristic Area Under Curve* ehk otsetõlkes vastuvõtja tööomaduste kõvera alune pindala) on ROC graafiku üldistus ühele numbrilisele väärtusele. Täpsemalt on see ROC graafikuga tekkinud ROC kõvera alla jääva ala pindala. Kuna ROC graafik on kahe 0 ja 1 vahele jääva muutuja visualisatsioon, siis jääb ROC AUC alati samasse vahemikku. Seda omakorda piirab märkus, et juhusliku arvamisega tekkival joonel on ROC AUC väärtus 0,5. Sellest tulenevalt ei tohiks ühegi reaalse mudeli ROC AUC kunagi olla alla 0,5. Kõik need omadused kokku muudavad ROC AUC heaks makseviivituse tõenäosuse hindamise mudeli mõõdikuks.

Lisaks ROC AUC väärtusele kasutatakse töös sarnast täpsus-saagis kõverat (lühendatult PR-kõver), kus päris-positiivsete ja vale-positiivsete määrad on asendatud täpsuse

ja saagise väärtustega. Numbriliseks hinnanguks on sellise graafiku alla jääva ala pindala, lühendatult PR AUC. See ilmestab täpsuse ja saagise suhet erinevatel künnistel ja aitab mõista mudeli efektiivsust.

2.3 Kasutatud mudelid

Antud töö üks fookustest on rakendada erinevaid mudeleid, mis masinõppe algoritmide ja etteantud treeningandmete abil on õppinud hindama tõenäosust, et leping satub järgmise aasta jooksul makseviivituse. Uuritav probleem on sõnastatud kahe klassiga, kas satub makseviivitusse või mitte, binaarseks klassifitseerimisprobleemiks. Seega nimetatakse kasutatud algoritme klassifitseerimisalgoritmideks. Algoritmide valikul on lähtutud tingimusest, et need pakkuvad tõenäosusi oma hinnangutele, sest mudeli tulemustest kasutatakse eelkõige väljastatud tõenäosusi. Kasutatud algoritmid on valitud tuginedes kirjandusest leitud loetelule (Kim, Cho ja Ryu 2020), aga juurde on lisatud mudeleid, mis autoril õpingute raames on näidanud häid tulemusi sarnastel probleemidel.

Mudeliga lahendatava probleemi matemaatiliseks defineerimiseks on kasutada kategoriseeritud andmestik $\{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$, kus N on andmestikus olevate kirjete arv, $\mathbf{x}_i$ on D-dimensiooniga reaalarvudega tunnuste vektor iga kirje $i = 1, ..., N$ kohta ja y_i on ennustuse eesmärk numbrilise klassina kujul 0 või 1, milles 0 kirjeldab negatiivset klassi ning 1 positiivset klassi ehk makseviivitusse sattumist. Lahendatav probleem on leida funktsioon $f(\mathbf{x})$, millega ennustada tundmatut y, kasutades etteantud vektori $\mathbf{x}$ väärtusi. Masinõppe valdkonnas kasutatakse funktsioonis tihtipeale muutujaid kaalud (*weight*) $\mathbf{w}$ ja nihe (*bias*) b, mistõttu otsitavale funktsioonile on sobilikumaks kujuks $f_{\mathbf{w},b}(\mathbf{x})$.

2.3.1 Logistiline regressioon

Vastupidiselt oma nimele on logistiline regressioon *(logistic regression)* klassifitseerimise, mitte regressiooni algoritm. See on statistiline mudel, mille juured ulatuvad juba 19. sajandi esimesse poolde ja mille arendamisega on tegelenud mitmed erinevad matemaatikud ja statistikud (Cramer 2004). Logistilise regressiooni meetodi sisuks on logistiline funktsioon, millega modelleeritakse binaarset sõltumatut tunnust. Binaarset regressiooni, mille üheks meetodiks on logistiline regressioon, on makseviivituse ennustamiseks kasutatud juba 1980. aastast (Kim, Cho ja Ryu 2020). Tänaseks päevaks on logistiline regressioon valdkonnas laialdaselt levinud tänu heale seletatavusele ja üldistamisvõimele.

Lahendamaks püstitatud probleemi, otsitakse logistilise regressiooniga mudelit kujul:

$$f_{\mathbf{w},b}(\mathbf{x}) := \frac{1}{1 + e^{-(\mathbf{w}\mathbf{x}+b)}} \tag{1}$$

kus $\mathbf{w}$ on dimensiooniga D parameetrite vektor ja b on reaalarv (Burkov 2019, leheküljed 25-27).

Leidmaks parimat lahendust maksimeeritakse logistiline regressiooni puhul mudeli alusel treeningandmestiku tõepära. Seega eesmärgiks on leida suurim tõepära, et etteantud treenimisandmed oleksid leitud mudeliga kirjeldatud. Selleks maksimeeritakse:

$$\max \prod_{i=1...N} f_{\mathbf{w},b}(\mathbf{x}_i)^{y_i}(1 - f_{\mathbf{w},b}(\mathbf{x}_i))^{(1-y_i)} \tag{2}$$

Praktikas kasutatakse pigem selle funktsiooni logaritmilist kuju vältimaks suurte numbritega tekkivaid arvutuslikke probleeme. Selle funktsiooni optimaalseks lahendamiseks kasutatakse protseduuri – gradientlaskumine *(gradient descent)*.

2.3.2 Tugivektormasin

Tugivektormasina *(support vector machine)* algoritmi kandvaks ideeks on kaardistada sisendvektor mitte-lineaarselt kõrgdimensioonilisesse tunnuste ruumi ja luua selles ruumis lineaarne otsustuspind, mis eristab ennustatavaid klasse. See meetod modifitseeritult toimib ka juhul kui andmestik ei ole selles ruumis täielikult eristatav. Meetod kirjeldati esmalt 1995. aastal (Cortes ja Vapnik 1995). Makseviivituse hindamiseks on esmaseid viiteid 2005. aastast (Kim, Cho ja Ryu 2020).

Tugivektormasina puhul on püstitatud probleemi lahendamiseks otsitav mudel kirjeldatud kujul (Burkov 2019, leheküljed 30-31):

$$f_{\mathbf{w},b}(\mathbf{x}) = sign(\mathbf{w}\mathbf{x} - b) \tag{3}$$

Parima lahenduse leidmiseks minimeeritakse tugivektormasina puhul järgnevat funktsiooni:

$$\min \frac{1}{2}||\mathbf{w}||^2, \text{ nii et } y_i(\mathbf{w}\mathbf{x}_i - b) - 1 \geq 0, i = 1, ..., N \tag{4}$$

Sellise funktsiooniga tekivad probleemid mürase ja mitte-lineaarse sisendi puhul, seega praktikas kasutatakse niinimetatud kerneli trikki, kus siis optimeeritakse hoopis järgnevat võrrandit maksimeerides:

$$\max_{\alpha_1...\alpha_N} \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{k=1}^{N} y_i \alpha_i (\mathbf{x}_i \mathbf{x}_k) y_k \alpha_k, \tag{5}$$

mis alluvad piirangutele:

$$\sum_{i=1}^{N} \alpha_i y_i = 0 \text{ ja } \alpha_i \geq 0, i = 1, ..., N,$$

kus α_i nimetatakse Lagrange'i kordajateks. Kerneleid on erinevaid, aga töös rakendatud mudel piirdub RBF *(Radial Basis Function)* meetodiga.

2.3.3 Otsustuspuu

Otsustuspuu *(decision tree)* on atsükliline graaf, mille abil saab teha otsuseid (Burkov 2019, leheküljed 27-30). Igal graafi hargnemisel võrreldakse mingit tunnuste vektori väärtust määratud künnisega otsustamaks kumba haru jätkata. Leht, milleni nii jõutakse kirjeldab ära, millisesse klassi ennustatav näide kuulub.

Otsustuspuu algoritmil on erinevaid kujusid, aga järgnevalt näitlikustatakse ID3 alusel (Burkov 2019, leheküljed 27-30). Selles optimeerimiseks minimeeritakse:

$$\min \frac{1}{N} \sum_{i=1}^{N} [y_i ln f_{ID3}(\mathbf{x}_i) + (1 - y_i) ln(1 - f_{ID3}(\mathbf{x}_i))] \tag{6}$$

kus f_{ID3} on otsustuspuu.

Üldistatult töötab ID3 algoritm järgnevalt. Esmalt koosneb otsustuspuu ainult ühest sõlmest, mille alusel ennustab mudel sama tulemust kõigile näidetele. Seejärel vaadatakse läbi kõik tunnused ja künnised ning jagatakse kogum kaheks, mis moodustavad uued lehed. Seda tehes valitakse parim tunnuste ja künniste paar. Parima hindamiseks valitakse jaotus, mis minimeerib entroopiat, kus uue jaotuse puhul mõõdetakse keskmiseks entroopiaks kogumite kaalutud keskmine entroopia. Algoritm lõpetab oma töö kui juhtub üks järgnevatest:

- Kõik näited on lehtedes korrektselt klassifitseeritud.
- Enam ei leita tunnust, millel jaotust teha.
- Jaotusel tekkiv entroopia vähenemine on alla määratud ϵ väärtust.
- Puu jõuab eelnevalt määratud maksimaalse sügavuseni.

Otsustuspuu pügamise käigus otsitakse hargnemisi, mis ei panusta märkimisväärselt vea vähenemisse ja need asendatakse lihtsalt lehega.

Otsustuspuud peetakse väga hästi seletatavaks, sest seda on üldiselt lihtne jälgida ja mõista, mis reegli alusel otsus tehti. Teisalt on selle algoritmi miinuseks üle sobitumine treenimisandmetele või mitte optimaalsete lahenduste leidmine.

2.3.4 Otsustusmets

Otsustusmets (*Random Forest*, otsetõlkes juhuslik mets) on klassifitseerimisalgoritm, mis sisaldab otsustuspuu struktuuriga klassifitseerijate kollektsiooni $h(x, \Theta_k), k = 1, ...$ kus (Θ_k) on sõltumatud identselt jaotunud juhuslikud vektorid ja iga puu annab hääle kõige populaarsema klassi ennustamiseks sisendi x puhul, mille alusel mudeli otsuseks saab kõige enim hääli saanud klass. Algoritmi ideed esitleti aastal 2001 (Breiman

2001). Otsustuspuude makseviivituse ennustamiseks kasutamisest on viiteid 2012 aastast (Kim, Cho ja Ryu 2020).

Teisiti öeldes, otsustusmets koosneb juhuslikult koostatud sõltumatutest vektoritest, milles ehitatakse otsustuspuu algoritmi alusel puu. Ennustamiseks valib iga puu sisendandmete pealt ennustatava klassi ja mudeli ennustuseks saab kõige populaarsem klass kõikide puude ennustustes. Need puud koostatakse *bootstrapi* meetodil alusandmete pealt uusi andmestikke koostades, kusjuures olemasolevad andmeread võivad korduda ühe puu jaoks kasutatavas andmestikus korduvalt. Otsustuspuu treenimisel omakorda valitakse juhuslikult sellest loodud andmestikust treenimiseks kasutatavad andmed.

Otsustusmets kuulub ansambli meetodite hulka ja on kasvanud välja otsustuspuu algoritmist. See on efektiivne ansambli meetod ennustamisel tänu sellele, et juhuslikult koostatud puudest head puud nõustuvad sama tulemuse osas, aga ülejäänud ei hääleta ühtselt ja jaotavad nii oma hääled erinevate variantide vahel.

2.3.5 Tehisnärvivõrk

Nimetuse tehisnärvivõrk (*artificial neural netowrk*) all mõeldakse erinevaid inimõppe ideedest inspireeritud algoritme (Vicente ja Roy 2021). See meetod kasutab tehislikke neuroneid, mis neisse saabuvate signaalide tugevuse teatava määra ületamisel edastavad oma signaali teistele seotud tehisneuronitele. Neid tehisneuroneid erinevatesse kihtidesse grupeerides on võimalik luua struktuure, mis on võimelised sisendparameetrite pealt hinnanguid andma. Sellise mudeli tulemust korrigeeritakse tagasilevi *(backpropagation)* meetodiga, mis kokku annab sellisele mudelile õppimisvõime, et üldistada õpitud andmestiku pealt uutele andmetele otsitavaid tunnuseid. Erinevaid võimalikke struktuure, kuidas tehisnärvivõrke edukalt rakendada, on palju ja neid kasutatakse erinevate probleemide lahendamiseks.

Esimene arvutuslik tehisnärvivõrgu mudel loodi juba 1943. aastal (Mcculloch ja Pitts 1943). Makseviivituse ennustamise valdkonna üks esimestest tulemuslikest uurimustest tehisnärvivõrkudega avaldati 1999 (Yang, M. B. Platt ja H. D. Platt 1999), samas kui katseid tehti juba eelneval kümnendil. Tuginedes makseviivituse valdkonnas tehtud uurimustele ja olles arvutuslikult piiratud, siis selles töös kasutatakse eelkõige mitmekihilist närvivõrku, mida nimetatakse mitmekihiliseks pertseptroniks.

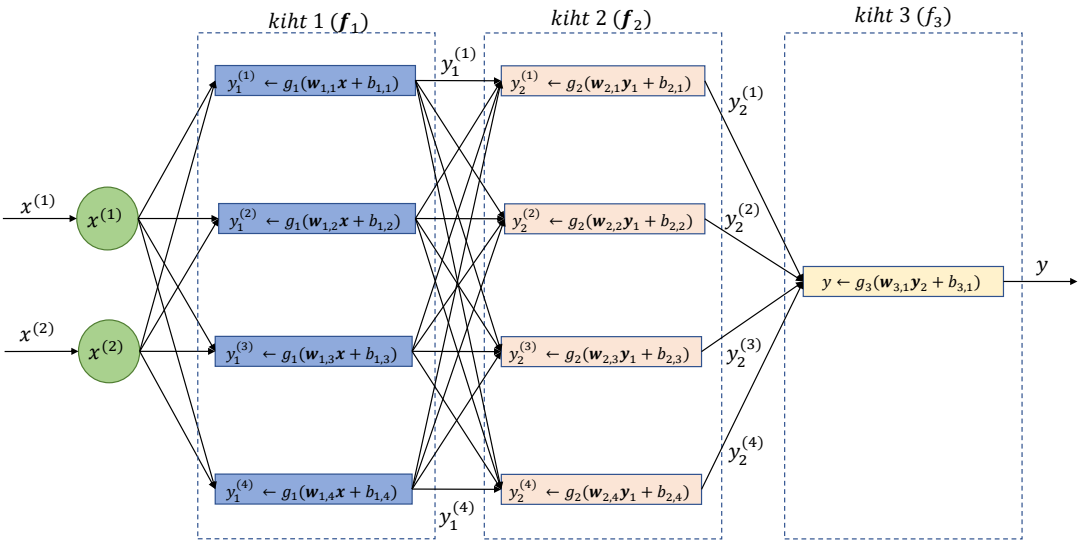
Tõstatatud probleemi lahendamiseks luuakse tehisnärvivõrgus järgnev mudel (Burkov 2019, leheküljed 61-62):

$$y = f_{NN}(\mathbf{x}) \tag{7}$$

kus funktsioon f_{NN} sisaldab kihile vastavaid funktsioone. Näiteks 3-kihlise tehisnärvivõrgu puhul:

$$y = f_{NN}(\mathbf{x}) = f_3(\mathbf{f_2}(\mathbf{f_1}(\mathbf{x}))) \tag{8}$$

Ühe tavapärase kihi funktsioon on vektorfunktsioon kujul:



Joonis 4: Kolmekihilise tehisnärvivõrgu näide

$$\mathbf{f}_l(\mathbf{z}) := \mathbf{g}_l(\mathbf{W}_l\mathbf{z} + \mathbf{b}_l), \tag{9}$$

kus l on kihi index ja mille iga kihi jaoks vajalikud parameetrid $\mathbf{W}_l$ (maatriks) ja $\mathbf{b}_l$ (vektor) on õpitud kasutades meetodit gradientlaskumine *(gradient descent)* ja optimeerides sõltuvalt ülesandest erinevat kaofunktsiooni *(loss function)*. Mitmekihilise pertseptroni näide on visualiseeritud joonisel 4 kolmekihilise pertseptronina, eesmärgiga anda ülevaate erinevate kihtide seostest. Sel visualiseeritud närvivõrgul on kaks tunnust sisendiks, kaks kihti nelja tehisneuroniga ja üks väljundkiht ühe tehisneuroniga. Iga kihi neuronite loogika on antud valemiga, mis näitab ära sellele neuronile sisendiks antud signaalide arvutuskäigu.

2.3.6 XGBoost

XGBoost (*Extreme Gradient Boosting*, otsetõlkes ekstreemne gradient võimendamine) on optimeeritud versioon gradient puu võimendamise *(Gradient Tree Boosting)* algoritmist, mille alged on otsustuspuu algoritmis (Chen ja Guestrin 2016a). See on arendatud tehes praktilisest kogemusest lähtudes väikseid kohendusi, eesmärgiga muuta algoritmi õppimist kiiremaks. Täpsemalt kirjeldades loob XGBoost otsustuspuude ansambli, mis erinevalt otsustuspuust summeerib erinevate puude lehtede ennustused saamaks mudeli

ennustust.

Püstitatud probleemi lahendamiseks otsib XGBoost algoritm lahendust järgnevale funktsioonide summale:

$$\hat{y}_i = \phi(\mathbf{x}_i) = \sum_{i=1}^{K} f_k(\mathbf{x}_i), f_k \in F \tag{10}$$

kus, $F = \{f(\mathbf{x}) = w_{q(\mathbf{x})}\}(q : \mathbb{R}^m \rightarrow T, w \in \mathbb{R}^T)$ on regressiooni puude ruum. q tähistab siin iga puu struktuuri, mis kaardistab sisendi vastava lehe indeksiga. T on lehtede arv igas puus. Iga f_k vastab sõltumatu puu struktuurile q ja lehtede kaaludele w. w_i on pidev i-nda lehe skoor. Õppimaks parimat funktsiooni, minimeeritakse järgnevat regulariseeritud eesmärki:

$$\min \sum_i l(\hat{y}_i, y_i) + \sum_k \Omega(f_k) \tag{11}$$

$$\text{kus } \Omega(f) = \gamma T + \frac{1}{2}\lambda ||\mathbf{w}||^2$$

Siin l on kaofunktsioon, mis mõõdab ennustuse $\hat{y}_i$ ja eesmärgi y_i vahet. Teine liidetav Ω karistab mudeli keerukust, mis aitab vähendada ülesobitust. XGBoost väljatöötamisel teisendati funktsioon 11 optimeerimiseks sobilikumale kujule, kus sammukaupa arvutades valitakse igal sammul nii öelda ahnuse printsiibi alusel lisa liige, mis kõige enam edendab algset funktsiooni.

2.4 Andmestiku tasakaalustamise meetod SMOTE

SMOTE (*Synthetic Minority Over-sampling Technique*, otsetõlkes sünteetiline vähemuse ülevalimise tehnika) on lähenemine klassifikaatorite loomiseks puuduliku tasakaaluga andmestikus. Kui ennustatavat klassi on märkimisväärselt vähem või rohkem kui alternatiivi, siis see võib tekitada erinevatel standardsetel klassifitseerimisalgoritmidel efektiivsuse probleeme, sest need pööravad rohkem tähelepanu enamusklassile (Faris *et al.* 2020). Praktikas kasutatakse klassifitseerimisalgoritme vähese tõenäosusega sündmuste ennustamiseks, mis tähendab, et neid näiteid on enamasti vähem kui muid andmeid. Sellega hakkama saamiseks on erinevaid meetmeid, nagu alavalimise *(undersampling)* või ülevalimise *(oversampling)* tehnikad, kus kas vähendatakse teisi näiteid või luuakse juurde ennustatavat klassi. Nende tehnikatega on halval juhul võimalik mõjutada mudeli õppimist vales suunas või jätta kõrvale oluline info. SMOTE idee on kombineerida neid mõlemaid, ülevalides vähemust ja alavalides enamik-klassi (Chawla *et al.* 2002). Kirjanduse alusel on SMOTE teistest meetoditest paremate ROC AUC tulemustega ja seetõttu valitud töö eksperimentides peamiseks andmestiku tasakaalustamise meetodiks (Veganzones ja Séverin 2018).

SMOTE vähemusklassi ülevalimise metoodika sisuks on sõltuvalt vajalike näidete loomise mahust liita k-lähimat naabrit, kus k on muudetav parameeter (Chawla *et al.* 2002). Nendest k-lähimast naabrist valitakse vastavalt loomist vajavate näidete mahule, osa naabritest, millest iga valitud naabri kohta luuakse uus näidis. Iga näidise loomisel lahutatakse vaatluse all olevast näite tunnusvektorist üks valitud naaber, saadud vektor korrutatakse juhusliku arvuga 0 ja 1 vahel ning lisatakse tulemus tunnusvektorite hulka. Seega sisuliselt valitakse valitud näidiste vahel juhuslik punkt, mis peaks vähemusklassi otsustus regiooni suurendama ja seega seda rohkem üldistama. Lisaks rakendatakse klassikalist alavalimist enamusklassi jaoks, mis peaks vähendama enamusklassi mõju mudelile.

2.5 Mudeli tõenäosuste kalibreerimine

Eluliste probleemide puhul on oluline, et ennustatav klass ei oleks mitte ainult täpne, vaid annaks aimu, millal ennustus ei ole korrektne. Selleks kasutatakse tõenäosust, et ennustus vastavasse klassi kuulub. Paljudel algoritmidel on selline väljastatav tõenäosus osa mudeli arhitektuurist. Mudeli tõenäosuste kalibreerimisega hinnatakse ja vajadusel korrigeeritakse mudeli väljastatud tõenäosusi, et need ühtiksid tegelike hinnangutega (Guo *et al.* 2017). Sellist tulemust nimetatakse kalibreeritud usaldusväärsuseks *(calibrated confidence)*. Kalibreeritud usaldusväärsus on oluline osa mudeli tõlgendatavusest, sest see aitab inimesel mõista ja usaldada mudeli tehtud ennustust.

Mudel on hästi kalibreeritud kui selle ennustused ühtivad andmete jaotusega. Tõenäosusi väljastava klassifitseerimisalgoritmi peetakse hästi kalibreerituks kui saadud testnäidete tõenäosusvektor p klasside jaotus on ligikaudselt sama p-ga (Kull, Silva Filho ja Flach 2017).

Mudeli kalibreerituse visualiseerimiseks kasutatakse usaldusväärsuse diagrammi *(reliability diagram)* (Guo *et al.* 2017). Sellel kujutatakse oodatavat näite õigsust usaldusväärsuse funktsioonina. Kalibreerituse ühte näitajasse kokku võtmiseks kasutatakse näiteks oodatavat kalibreerimisviga *(ECE - expected calibration error)* või maksimaalset kalibreerimisviga *(MCE - maximum calibration error)*. Kalibreerimiseks on loodud erinevaid meetodeid nagu histogrammi binnimine *(histogram binning)* või Platt-skaleerimine *(Platt scaling)*.

Osad algoritmid on disainilt hästi kalibreeritud, nagu otsustuspuud ja logistiline regressioon (Kull, Silva Filho ja Flach 2017). Teiste tulemused üldjoontes vajavad kalibreerimist, nagu tugivektormasin või tehisnärvivõrk.

Lisaks mudeli tõenäosuse kalibreerimisele rakendatakse makseviivituse tõenäosuse mudeli väljastatud tõenäosustele teisi kalibratsioone, nagu näiteks majanduslangusega seoses (Finantsinspektsioon ja EBA 2019). Selles töös käsitletakse kalibreerimise mõiste all ainult mudeli tõenäosuse kalibreerimist. Selle eesmärgiks on veenduda või saavutada mudelite hea kalibreeritus ja seeläbi suurendada mudeli usaldusväärsust.

2.6 Mudelite seletatavus

Makseviivituse hindamiseks, nagu enamike teiste masinõppe rakendusvaldkondade puhul, on oluline mudeli tulemuste seletatavus. Seletatavust saab jagada kaheks: miks mingi mudel mingi konkreetse näite puhul sellise ennustuse tegi ja mis tunnustele tuginedes mudel üldisemalt otsuseid teeb (Ribeiro, Singh ja Guestrin 2016). Need on olulised aspektid usaldamaks mudeli tulemusi, sest usalduse puudumisel ei ole mõeldav mudeli rakendamine. Iga mudeli puhul jääb mingi usalduse määr, mida mudeli kasutaja peab arvestama. Tihti see eeldab valdkonna ja rakendatava mudeli sügavamat mõistmist. Seega on oluline arvestada, kellele selgitust antakse, sest sõltuvalt taustast võib selgitamine olla väga erineva mahuga ülesanne. Töös rakendatud meetodite puhul lähtutakse eeldusest, et mudeli tulemuse tõlgendajal on baasteadmised matemaatikast, masinõppest ja statistikast, kõik, mida omandatakse näiteks Tartu Ülikooli Andmeteaduse magistriõppekaval. Seega rakendatud seletatavuse meetmed pigem vähendavad vajadust konkreetset mudelit hästi tunda, aga ei paku täielikku absoluutset usaldust.

Osade mudelite puhul on nende ennustuste sisu lihtsasti seletatav. Näiteks logistilise regressiooni puhul mudeli erinevatele tunnustele rakendatavad kordajad on läbi teisenduse tõlgendatavad selle tunnuse mõjuga ja individuaalse ennustuse puhul piisab andmete asendamisest, et täpsemat mõju näha (Molnar 2022). Keerukamate mudelite puhul see nii lihtne ei ole. Järgnevalt kirjeldatakse erinevaid meetmeid, mida töös rakendati, et säilitada või tekitada keerukamatele mudelitele piisav seletatavus.

2.6.1 PDP

PDP (*Partial Dependency Plot*, otsetõlkes osalise sõltuvuse graafik) on standardiseeritud meetod arvutamaks mudelis kasutatavate erinevate tunnuste olulisuse skoore (Molnar 2022). PDP abil on võimalik visualiseerida erinevate tunnuste seoste funktsiooni ennustatava tulemusega ja tekitades nii parema arusaama, kuidas erinevad tunnused tulemust mõjutavad.

PDP algoritm on järgnev (Greenwell, Boehmke ja Mccarthy 2018):
Olgu $z_s = x_1$ ennustamiseks kasutatud muutuja unikaalsete väärtustega $\{x_{11}, x_{12}, ..., x_{1k}\}$. Vastuse osalise sõltumise x_1 muutujasse saab algoritmiga 1.

PDP peamisteks tugevusteks on joonise intuitiivsus ja selle selge tõlgendatavus (Molnar 2022). Miinusteks eeldus, et tunnused on omavahel sõltumatud ja oma lihtsusega võib see peita keerukamaid efekte. Selle meetodi abil seletatakse ja seeläbi võrreldakse magistritöös tunnuste mõju mudelite ennustatud tõenäosusele.

2.6.2 LIME

LIME (*Local Interpretable Model-Agnostic Explanations*, otsetõlkes lokaalsed tõlgendatavad mudelist sõltumatud selgitatavused) on metoodika, millega selgitada klassifikaatorit läbi lokaalselt ühtiva selgitatava mudeli (Ribeiro, Singh ja Guestrin 2016).

Algorithm 1 PDP koostamine

Sisend: unikaalsed ennustaja väärtused $x_{11}, x_{12}, ..., x_{1k}$
Väljund: hinnangulised osalise sõltuvuse väärtused $\overline{f}_1(x_{11}), \overline{f}_1(x_{12}), ...,$ $\overline{f}_1(x_{1k})$
for $i \in \{1, 2, ..., k\}$ **do**
(1) kopeeri treeningandmed ja asenda originaalsed x_1 väärtused konstantiga x_{1i};
(2) arvuta ennustatavate väärtuste vektor muudetud treeningandmete koopiast;
(3) arvuta ennustuste keskmine saamaks $\overline{f}_1(x_{1i})$
end for
PDP x_1 kohta saadakse joonestades paarid $\{x_{1i}, \overline{f}_1(x_{1i}\}$ iga $i = 1, 2, ..., k$

Lihtsamalt öeldes mudeli ennustuste pealt luuakse selgitatavate omadustega mudel, mis rakendub ainult ennustatava näite lähiümbruses, aga on piisav selgitamaks selle konkreetse näite seoseid.

Matemaatiliselt saab seda mudelit kirjeldada järgnevalt (Ribeiro, Singh ja Guestrin 2016):

$$selgitus(\mathbf{x}) = \underset{g \in G}{\operatorname{argmin}} L(f, g, \pi_{\mathbf{x}}) + \Omega(g) \tag{12}$$

Selgitus mudel näite $\mathbf{x}$ kohta on mudel g, mis minimeerib kadu *(loss)* L (Molnar 2022). Kadu mõõdab kui lähedal selgitus on originaalsele mudel f ennustusele, samal ajal kui mudeli keerukust $\Omega(g)$ hoitakse madalana. G on võimalike selgituste perekond, näiteks kõikvõimalikud lineaarse regressiooni mudelid. Lähedusmõõt $\pi_{\mathbf{x}}$ defineerib kui suurt naabruskonda $\mathbf{x}$-i ümbruses selgitamiseks vaadatakse.

3 Eksperimendi ülesehitus

Praktilise töö tegemiseks kasutati Pythoni (Van Rossum ja Drake Jr 1995) programmeerimiskeelt ja Jupyter märkmiku (Kluyver *et al.* 2016) formaati. Peamiseks töövahendiks valiti JetBrains ettevõtte integreeritud arendamiskeskond DataSpell (JetBrains s.r.o 2021). Peamised teegid, mida töös rakendati:

- pandas (team 2020)
- numpy (Harris *et al.* 2020)
- matplotlib (Hunter 2007)
- seaborn (Waskom 2021)

Töö teostamine planeeriti lähtudes CRISP-DM standardist, mis on üks peamisi tunnustatud andmekaeve projektide tööprotsessi standardeid. Töö ei kirjelda rakendamise etappi, sest seda ei teostatud magistritöö raames. Organiseerimaks töö praktilises osas teostatud eksperimente, kasutatakse kirjeldamisel CRISP-DM metoodika samme samade etappide kaupa.

3.1 Äri mõistmine

Makseraskus on tähtajaks täitmata kohustus. Makseviivitus *(default)* on pikaajaline makseraskus, mis viivitab võetud kohustuste täitmist. Selle töö kontekstis on viivitus finantskohustuse tagasimaksmisel. See on enamikel juhtudel tingitud võlgu jäämisest, kus võlg on püsinud 90 päeva (Euroopa Parlament ja Euroopa Liidu Nõukogu 2013). Makseviivitusele võib järgneda maksevõimetus *(insolvency)*, kus ei olda võimeline võlga tagasi maksma (Varusk 2008) või kehvematel juhtudel maksejõuetus *(bankruptcy)* (Justiitsministeerium 2021).

Makseviivituse tõenäosuse hindamine on võtmetähtsusega tegevus krediiditeenust osutava finantsettevõtte jaoks. Krediidi või lihtsustatult laenu välja andmisel on oluline olla maksimaalselt veendunud, et laenu saaja selle tagasi maksab. Selleks hinnatakse muude aspektide hulgas kliendi makseviivituse tõenäosust. Seda laenuotsuse tegemise protsessi reguleerivad rohked rahvusvahelised ja riiklikud standardid ning regulatsioonid (Finantsinspektsioon 2022). Oluliseks aspektiks on Euroopa parlamendis vastuvõetud isikuandmete kaitse üldmäärus *(GDPR)* (Euroopa Parlament ja Euroopa Liidu Nõukogu 2016). Kui kliendile tehakse positiivne laenuotsus ja laen väljastatakse on edaspidi vajalik pidev jälgimine ja hindamine, kas laenul on tõenäosus makseviivitusse langeda. See on ühelt poolt vajalik mõistmaks, kas laen makstakse tagasi, aga suuremas pildis oluline mõistmaks, kas finantsasutus on suuteline toimima. Jälgimine on reguleeritud näiteks rahvusvaheline finantsinstrumentide standardiga IFRS9 (IFRS Foundation 2022). Üldjoontes hinnatakse kogu finantsasutuse aktiivsete krediidilepingute portfellis olevaid lepinguid ja hinnatakse näiteks kui suur on tõenäosus, et kliendi leping järgneva aasta jooksul langeb makseviivitusse. Selliseid makseviivituse hindamise mudeleid

nimetatakse portfellimudeliteks.

Portfellimudeliga leitakse igale lepingule tõenäosus makseviivitusse langeda. Seda hinnangut rakendatakse otsuse tegemisel, mis ulatuses laekunud krediidi tagasimakseid saab finantsasutus kasumiks lugeda või mis on selle asutuse laenuportfelli kvaliteet.

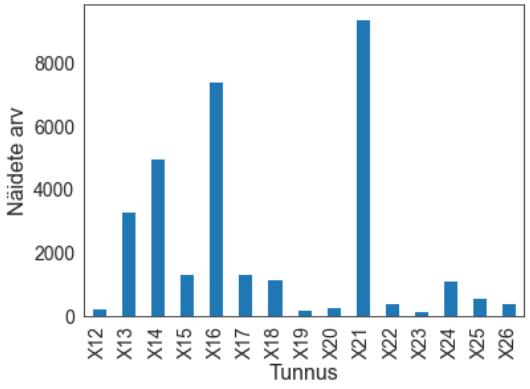
Lõputöös keskendutakse kogu sellest mahukast valdkonnast ühele kitsale osale - mudel, millega hinnatakse kõigi finantsasutuse portfellis olevate lepingute makseviivitusse langemise tõenäosust järgneva aasta jooksul. Veel täpsemalt rakendatakse erinevaid mudeli loomise algoritme ja võrreldakse nende tulemusi valdkonnas laialdaselt levinud logistilise regressiooni algoritmiga loodud mudeli tulemustega.

3.2 Andmete mõistmine

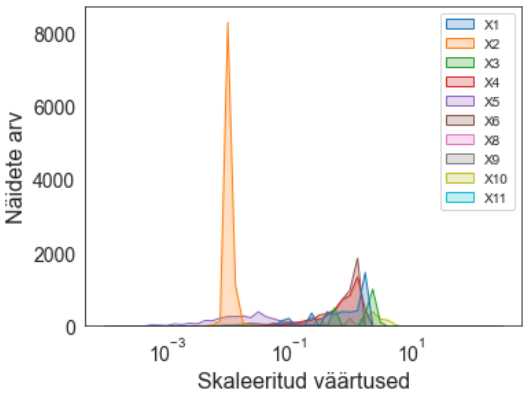
Selles lõputöös kasutatud andmestik on AS LHV Group'i krediidilepingute portfellist. Kaitsmaks ärisaladust ja vähendamaks riske, et lekitatakse tundlikku infot, siis magistritöö piirdub töödeldud andmestiku üldise kirjeldusega. Arvestades eesmärki, siis selle saavutamiseks ei rakendata erinevaid andmekaeve meetmeid paremate näidete või tunnuste leidmiseks, vaid keskendutakse eelkõige meetodite rakendamisele ja tulemuste võrdlemisele. Kõik võrreldavad meetodid on rakendatud samadel alusandmetel, et hoida meetodite tulemused võrreldavana. Alusandmete sisul on kindlasti mõju tehtud järeldustele, aga kuna valitud tunnused on tüüpilised, mis finantsasutusel on oma klientide kohta kasutada, siis need järeldused peaksid olema praktikas kasutatavad.

Alusandmestik koosneb 25 erinevast tunnusest, mille hulgas on tunnus, kas lepingul toimus andmete kogumise hetkest järgmise 12 kuu jooksul makseviivitus. Tunnustes 10 on numbrilised ja 15 kategoorsed, mille hulgas on mõned järjestatavad kategooriad. Neis sisalduvad erinevad finantsnäitajad nagu võlasummad, võlaperioodid, võetud krediidikohustuste mahud ja ettevõtte finantsnäitajad. Kokku sisaldab töö tegemiseks kasutatud andmestik üle 10 000 lepingu näite, mis on võetud mitme aastase perioodi kohta. See tähendab, et osad lepingud esinevad andmestikus mitmekordselt. Makseviivitust ennustatakse 12 kuu kohta ja andmed on korjatud 12 kuu pikkuse sammuga. Selline metoodika välistab ennustusperioodide kattumise ja minimeerib ebasobilikku mõju mudelile, samas võimaldades kasutada rohkem andmepunkte ennustamiseks. Kui lepingul tuvastatakse makseviivitus, siis järgmisel perioodil seda andmestikus enam ei kajastata. Sellistel lepingutel ei ole põhjust ennustada, kas juba makseviivituses olev leping sinna edaspidi satub.

Ennustatav tunnus, makseviivitus järgmise 12 kuu jooksul, on binaarne, olles andmekogumise hetkest järgmise 12 kuu jooksul juhtunud või mitte. Tulenevalt AS LHV Group'i madalast riskitasemest ja sellega kaasnevast madalast makseviivituste arvust, on andmestikus ainult 88 kirjet, kus leping on järgmise 12 kuu jooksul makseviivitusse sattunud. See on alla 1% kogu kirjete arvust. Seega on andmestik suuresti kaldus lepingute suunas, mis ei ole makseviivitust kogenud.



Joonis 5: Ühega kodeeritud tunnused ja nende tunnustega näidete arv kogu andmestikust



Joonis 6: Arvuliste tunnuste jaotus kogu andmestikust

Andmete paremaks mõistmiseks kasutati erinevaid visualiseerimismeetodeid, mida andmete kaitsmiseks siin kirjatöös avalikult välja ei tooda ja piirdutakse ainult üldise ülevaatega.

3.3 Andmete ettevalmistamine

Andmete ettevalmistamiseks oli vajalik nende laadimine arhiivist, formaadi korrigeerimine ja liigsete tunnuste eemaldamine, nagu näiteks kasutusel oleva mudeli parameetrid. Selline baastasemel ettevalmistamine oli märkimisväärne töö, mille detailidel ei peatuta.

Valmistamaks ette andmete kasutamist masinõppe mudelites, tehti järgnevad transformatsioonid:

1. Kategooriliste andmete numbriliseks muutmine. Kasutati meetodeid: ühega kodeerimine (*one hot encoding*) ja järguline kodeerimine (*ordinal encoding*).
2. Eritunnuste ümberkodeerimine numbriliseks.
3. Puuduvate andmetega kirjete eemaldamine. Valiti andmete imputeerimise asemel, sest selliseid kirjeid oli alla 1%.
4. Numbrite skaleerimine samale skaalale (standardne skaleerimine).
5. Treenimis-, valideerimis- ja testandmete eristamine (70%, 10%, 20%) nii, et ennustatava klassi jaotus püsib sama igas alamandmestikus.

Andmetes kasutatakse ainult väikese ja keskmise suurusega ettevõtete lepinguid ning piirdutakse ainult valitud toodetega. Selline otsus tagas madala puuduvate andmete hulga. Treenimis-, valideerimis- ja testandmete jaotuses on kasutusel ainult mudelite

kalibreerimiseks. Teistes sammudes rakendati mudeli treenimisel parimate parameetrite otsimisel meetodit ristvalideerimine (*cross validation*), olles treenimiseks eraldanud 80% ja jättes 20% testimiseks. Testimisandmestiku suurust piirab ennustatava klassi kirjete vähesus, mistõttu väiksem protsent muutuks rohkem juhuslikkusest sõltuvaks.

Andmetes on väga vähe makseviivitusega kirjeid, jäädes alla 1% koguandmestikust. Selle tasakaalustamiseks kasutatakse SMOTE tehnikat, mis kirjanduses oli välja toodud kui kõige efektiivsem (Faris *et al.* 2020).

Peale töötlust jääb mudelite õpetamiseks kasutatavasse andmestiku 26 tunnust, millest 15 on ühega kodeeritud, 1 kategooriline ja 10 numbrilist. Sel ainukesel kategoorilisel tunnusel on 3 väärtust jaotusega 80,8%, 13,9% ja 5,3%. Teiste tunnuste jaotus on kujutatud joonistel 5 ja 6. Kogu näidete arv on 9896, millest kõik 88 algses andmestikus olnud järgmise aasta jooksul makseviivitusse sattunud kirjet on alles.

3.4 Mudeldamine

Töö baasmudeliks on logistiline regressioon, mis on makseviivituse ennustamisel laialdaselt kasutatud. Baasmudeli puhul ei teostata hüperparameetrite otsimist (*hyperparameter search*) ja piirdutakse mudelil teegis vaikimisi seatud parameetritega. Baasmudel on loodud võrdluseks, et näha, kas erinevad mudelid ja meetodid, mida rakendatakse, suudavad anda märkimisväärseid eeliseid. Lisaks paremale ennustamisvõimele on oluline mudeli seletatavus ja jõudlus. Seega keerukamad mudelid, mille tulemuste seletatavus on keerulisem, on selle võrra kehvemad lihtsamatest mudelitest ja peavad muude eelistega silma paistma. Samas rakendatakse töös parimatele mudelitele erinevaid seletatavuse parandamise võimalusi eesmärgiga tagada piisav seletatavus, et mudel oleks kasutatav.

Ülejäänud mudelid, mida rakendati on peamiselt valitud erinevate uurimustööde põhjal neis välja toodud eeliste alusel:

1. Tugivektormasin
2. Otsustuspuu
3. Otsustusmets
4. Tehisnärvivõrk
5. XGBoost

Töö käigus rakendati erinevate mudelite parimate hüperparameetrite *(hyperparamter)* leidmiseks võrguotsingut *(grid search)* koos ristvalideerimine meetodiga. Mudeli loomisel kasutati peamiselt *sklearni* teeki (Pedregosa *et al.* 2011), mis on paindlik ja mugav töövahend enamike laialt levinud andmeteaduse algoritmide ja andmete ettevalmistamise meetodite kasutamiseks. Mudelite loomisel leidsid rakendamist veel järgmised teegid:

1. pyodbc (Kleehammer 2021)
2. imblearn (Lemaître, Nogueira ja Aridas 2017)

3. SciPy (Virtanen *et al.* 2020)
4. netcal (Küppers *et al.* 2020)
5. XGBoost (Chen ja Guestrin 2016b)
6. LIME (Ribeiro, Singh ja Guestrin 2016)

3.5 Hindamine

Töös kasutatav andmestik, on väga kaldus *(imbalanced)*, sisaldades ainult ühe protsendi jagu makseviivitusse sattuvaid lepinguid. See on makseviivituse hindamisel levinud probleem ja seega ei ole ennustuste õigsuse hindamisel mingit sisulist väärtust. Mudeli ennustamise puhul on palju olulisem selle makseviivitusse sattuvate lepingute tõenäosuse õige ennustamine. Makseviivitusse sattumine on haruldane ja mõne protsendine tõenäosus on märkimisväärne risk, millele tähelepanu pöörata. Top 5% protsenti kuuluvate lepingute maksimaalne makseviivituse tõenäosus on oluliseks loogiliseks kontrolliks jälgimaks mudeli ennustuste üldist trendi.

Hindamaks mudeli headust kasutatakse peamise mõõdikuna ROC AUC väärtust, mis ei sõltu ennustatavate klasside proportsionaalsusest. Paremaks näitlikustamiseks visualiseeritakse see graafiliselt. Samamoodi arvutatakse täpsus-saagis kõver koos selle AUC väärtusega. Kalibreeritust vaadeldakse ja vajadusel parandatakse baasmudelil ja parima mõõdetava tulemusega mudelil. Kalibreeritus ei mõjuta ROC AUC tulemust, aga aitab kaasa mudeli pakutud tulemuste tõenäosuste paremale mõistmisele.

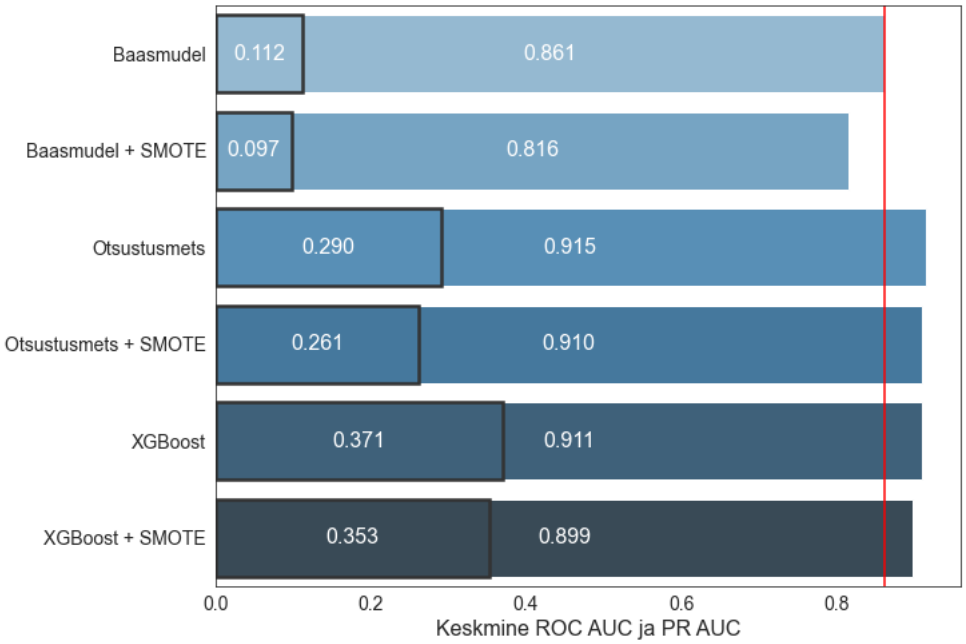
Tagamaks mudeli piisav seletatavus on kasutatud meetmeid PDP ja LIME. Nende tulemustega näitlikustatakse parima mudeli selgitatavust ja võrreldakse seda baasmudeliga.

Vähendamaks juhuslikkuse mõju tulemustele korrati eksperimenti 1000 korda ja võeti kõikide katsete keskmine lõplikuks mõõdikute väärtuseks. Iga katse puhul tehti uus andmestiku jaotus treening ja test andmeteks, teostati vajalikud andmete transformatsioonid, treeniti uuesti kõik kasutatud mudelid treening andmetel ja hinnati loodud test andmestikul kõikide mudelite mõõdikuid. Saamaks kinnitust baasmudeli ja parima mudeli tulemuste mõõdikute statistiliselt olulise erinevuse kohta, teostati paarikaupa t-test ja normaaljaotusele vastavuse test. Paarikaupa t-testi abil kontrollitakse, kas kahe normaaljaotusega valimi keskmised on võrdsed või omavad statistiliselt olulist erinevust (Student 1908).

4 Tulemused

Praktilise töö tulemuste kirjeldamisel keskendutakse eelkõige kasutatud mudelite tulemuste võrdlusele erinevates mõõdetud aspektides, jättes kõrval erinevate andmestiku tunnuste mõju detailse kirjeldamise, mis tavapäraselt on olulise tähtsusega. Sellise valiku eesmärgiks on kaitsta kasutatud andmestikku ja keskenduda töö põhifookusele.

4.1 Algoritmide võrdlus



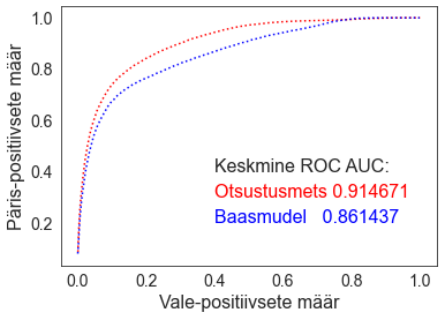
Joonis 7: Edukamate mudelite 1000 katse keskmine ROC AUC tulemus, kus punane joon tähistab baasmudeli tulemust ja tumedama joonega on tähistatud iga mudeli keskmine PR AUC

Baas algoritm logistiline regressioon saavutas tulemusteks: ROC AUC 86.1%, PR AUC 11,2%. Mõõdikute alusel oli parimaks algoritmiks otsustusmets tulemustega: ROC AUC 91,5%, PR AUC 29%. See on paarikaupa t-test alusel statistiliselt parem baasmudelist ($\alpha < 0.01$). Parimate mudelite tulemused on nähtavad joonisel 7, ülejäänud mudelid on leitavad lisade alt joonisel 14. Väike erinevus nendel joonistel samade algoritmide tulemuste vahel on tingitud kahest katsest, kus alguses vaadati kõiki mudeleid 10 katsel, valiti välja parimad ja seejärel korrati katset 1000 korda ainult parimatel meetoditel. Tulemustest on märgata, et SMOTE kasutamine enamasti ei aidanud ennustamist parandada. See oli võrreldavalt hea ainult otsustusmetsa ja XGBoosti puhul, enamikel

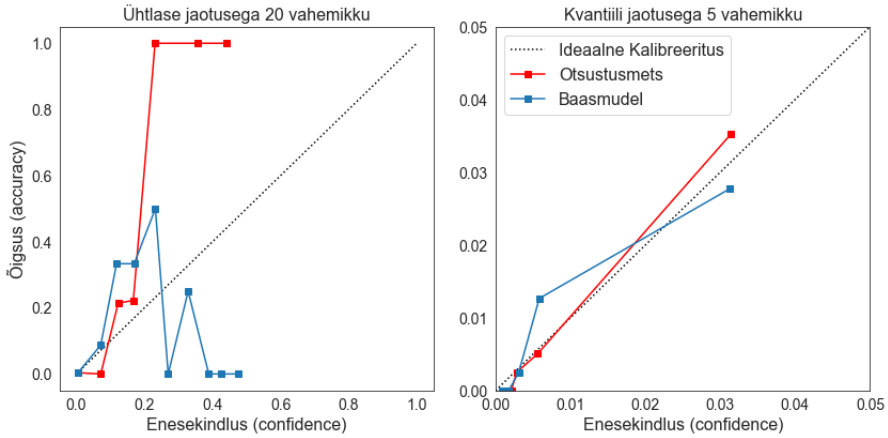
teistel juhtudel muutis mudeli üldistamisvõimet kehvemaks. Ekstreemseks näiteks on tugivektormasin, kus ROC AUC vähenes umbes 10% võrra SMOTE meetodil töödeldud treenimisandmestikku kasutades.

Tulemuste erinevuse mõistmiseks on hea vaadata, kuidas erinevad mudelite ROC graafikud, mida on võimalik näha joonisel 8. Otsustusmetsa ja baasmudeliks oleva logistilisest regressiooni peamine erinevus on madalate ja keskmiste vale-positiivse määra juures, kus otsustusmets natuke paremini toimis. Muus osas on graafiku erinevus väike.

Treenitud mudelite kalibreeritus oli kõrge, mida otsustuspuu ja logistilise regressiooni puhul võib eeldada. Testandmestikul mõõdetud ECE oli otsustusmetsal 0,0082 ja logistilisel regressioonil 0.0038. Platt-skaleerimise meetodil kalibreerimine seda tulemust sisuliselt ei muutnud. Peamiste meetodite usaldusdiagrammid on visualiseeritud joonisel 9. Ühtlase jaotusega joonisel on andmed jagatud 20 vahemikku, millest punktiga on tähistatud ainult need vahemikud, kus on näiteid. Mudeli ennustatud tõenäosused on üksikutel juhtudel 0,4-st suuremad ja valdav enamus näidetest on tõenäosusega alla 0,01. Kvantiili jaotusega joonisel on täpsemaks eristamiseks kujutatud ainult esimest 5% joonise telgedest ja vahemike arvuks valitud ainult 5, et vältida suurema grupi kuhjumist 0 lähedusse. Lisade all on joonisel 15 võrdluseks võimalik näha 10 vahemikuga jooniseid.



Joonis 8: Peamiste mudelite keskmised ROC graafikud

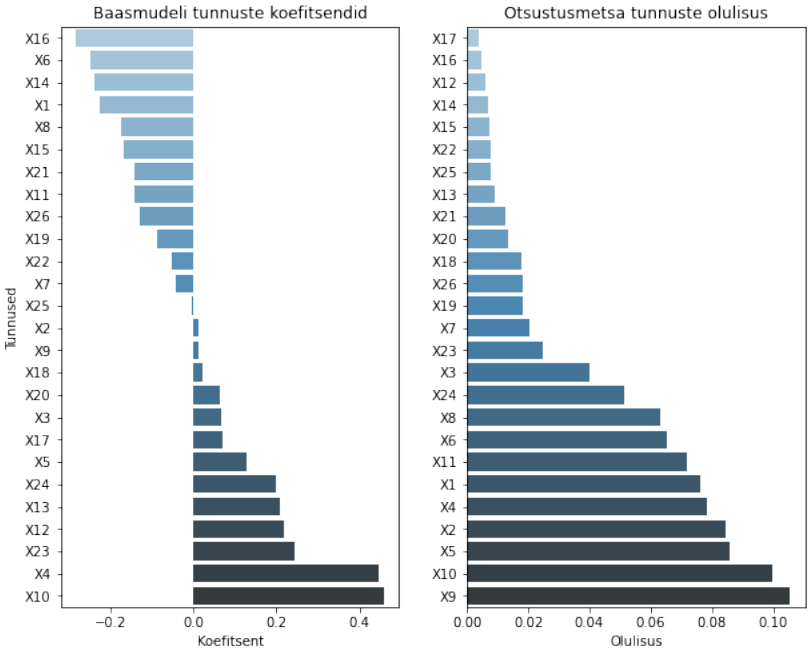


Joonis 9: Otsustusmetsa ja logistilise regressiooni usaldusdiagrammid erinevate vahemike arvuga

Mudelite treenimiseks kulunud ajal ei olnud märkimisväärset vahet. Logistiline reg-

ressioon oli kiirem, aga mõlemal juhul jäi treenimise aeg alla poole minuti. Treenimiseks kasutati tavalist äriklassi sülearvutit, 12 GB muutmäluga ja ilma graafika kaardita. Mudelite mälukasutuse maht on baasmudelil 0.001 ja 0.3 megabaiti otsustusmetsal.

4.2 Seletatavus



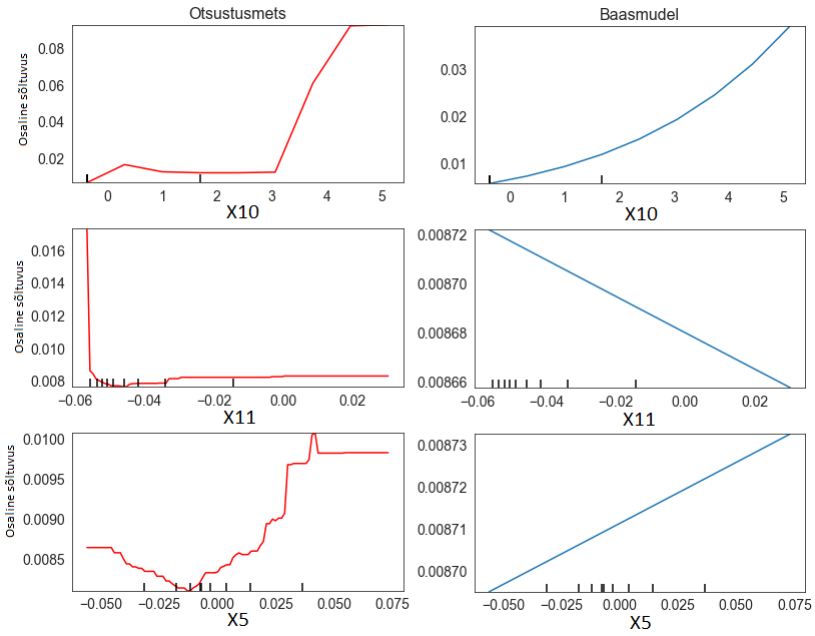
Joonis 10: Baasmudeli ja otsustusmetsa tunnuste mõju

Mudeli tulemuste seletatavuse puhul on kirjeldatud ühe juhuslikult valitud eksperimendi tulemused, ilmestamaks meetodi toimimist ja lähtudes sellest, et lõpuks valitakse üks mudel, mida rakendatakse.

Täpne mõjude ülevaade on leitav joonisel 10. Baasmudeli tulemusi mõjutavad kõige enam tunnused X10 ja X4. Samas kui otsustusmetsal on olulisemateks mõjutajateks X9 ja X10. X10 puhul on tegemist võlainfoga seotud tunnusega, mille hea ennustusvõime on tavapärane.

4.2.1 Mudeli tunnuste mõju seletamine - PDP

Suurima mõjutajate PDP graafikud on nähtavad joonisel 11, kus võrreldakse baasmudeliks olevat logistilist regressiooni ja otsustusmetsa. Kõigi mitte binaarsete tunnuste PDP graafikud on leitavad lisade alt, viitega 16. Tunnuse X10 puhul on otsustusmetsa ja baasmudeliks oleva logistilise regressiooni mõju sarnane, suurima vahega väärtuste



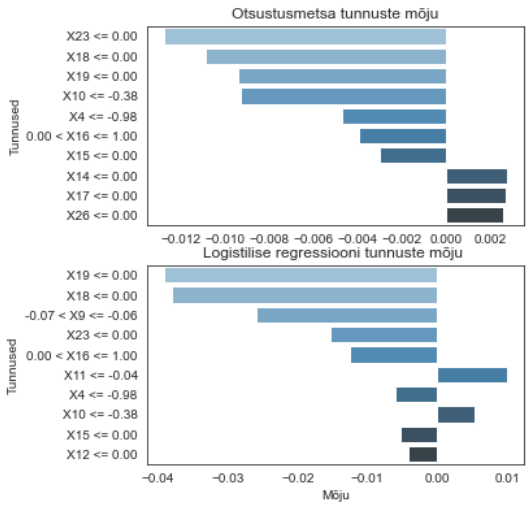
Joonis 11: Suurima mõjuga tunnuste PDP graafikud

3 ja 4 juures, kus otsustusmetsas on tunnuse mõju järsemate muutustega, baasmudelil laugem. Tunnuse X11 puhul on otsustusmetsal mõju muutust näha ainult väärtuse -0.04 ümbruses ja muude väärtuste puhul on mõju sama. Baasmudelil on sama tunnuse mõju ühtlaselt langev. X5 tunnus näitlikustab otsustusmetsa ebaühtlasemalt muutuvat mõju võrreldes ühtlase logistilise regressiooni mõju muutusega.

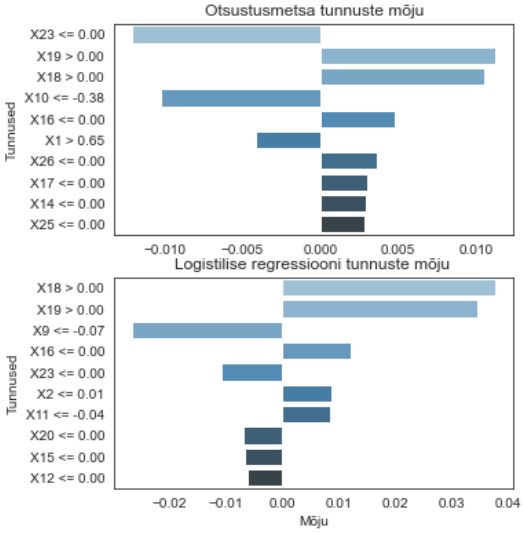
4.2.2 Konkreetse ennustuse seletamine - LIME

LIME meetodi tulemuste näitlikustamiseks kasutatakse kahte näidet. Esimene on leping, mis ei langenud makseviivitusse ning mida baasmudel ja otsustusmets ennustasid edukalt kui väga madala tõenäosusega makseviivitusse langev. Baasmudeli tõenäosuse hinnang oli $< 0.006\%$ ja otsustusmetsa ennustus $< 0.009\%$. Näite erinevate tunnuste väärtuste täpsem mõju tõenäosuse hinnangule on nähtav joonisel 12. Tulemustest on näha, et peamine mõjutaja erineb mudelite vahel, aga näiteks tunnused X18 ja X19 on mõlemal juhul olulise mõjuga, miks mudelid hindavad, et näide ei lange makseviivitusse.

Teine näide on lepingust, mis langes makseviivitusse järgneva aasta jooksul. Selle ennustamisel on otsustusmetsa ennustus parem, aga mõlemal juhul on tõenäosus mõne protsendi juures. Baasmudeli tõenäosuse hinnang oli 2.7% ja otsustusmetsa ennustus 8.5%. Näite erinevate tunnuste mõju tõenäosuse hinnangule on nähtav joonisel 13. Antud näite puhul on taaskord erinevad tunnused peamisteks mõjutajateks, aga ühisosaks



Joonis 12: Baasmudeli ja otsustusmetsa tunnuste mõju tavalisel lepingul, mis ei sattunud makseviivitusse



Joonis 13: Makseviivitusse sattunud näite baasmudeli ja otsustusmetsa tunnuste mõju

on taas tunnused X18 ja X19. Tunnus X18 on seotud võlainfoga.

5 Arutelu

Otsustusmetsa mudeli ennustustel mõõdetud tulemused on statistiliselt paremad baasmudeliks kasutatud logistilise regressiooni mudeli tulemustest. Seda nii ROC AUC kui PR AUC mõõdikutel. Seega võib otsustusmetsa mudelit pidada peamise mõõdetava parameetri alusel paremaks. Kas see paremus on reaalselt kasutatav?

Muudes aspektides, mudeli tehniliste parameetrite mõistes, ei ole neil kahel algoritmil märkimisväärset vahet. Treenimiseks kuluv aeg ja mudeli maht on küll erinev, aga suurusjärkude mõistes võrreldav. Seega neid algoritme võib tehnilise parameetrite osas pidada samaväärseteks. Sama moodi on mõlemal mudelil olemas tugi laialt levinud teekides, mis muudab nende mudelite rakendamise keerukuse samaväärseks.

Peamine miinus otsustusmetsal on keerulisem seletatavus kui logistilisel regressioonil, sest see algoritm sisaldab keerukamat metoodikat lõpphinnanguni jõudmiseks. Seega paremuse saavutamine taandub eelkõige hinnangule, kas kasutatud meetodite abil on võimalik otsustusmetsa seletatavust viia piisavale tasemele, et õigustada väikest paranemist mudeli mõõdetud tulemustes.

Seletatavuse parandamiseks kasutati meetodeid, mis aitavad mõista, kuidas erinevad tunnused mõjutavad mudeli väljastatavaid tõenäosuse hinnanguid või kuidas konkreetsetel näidetel erinevad tunnused tõenäosuse hinnangut kujundasid. Need meetodid võrreldes baasmudeliks kasutatud logistilise regressiooni tulemustega annavad võimaluse erinevate meetodite rõhuasetusi mõista.

Visualiseerides erinevaid tunnuseid ja nende mõju klassifitseerimisalgoritmi hinnangutele tekib võimalus algoritmi valikul eelistada algoritme kasutatud tunnusete alusel. Näitena on osad tunnused tavapärasest rohkem vea altid, sest näiteks sõltuvad mingist välisest osapoolest või käsitsi teostatavatest protsessidest, mis võivad tekitada hooletusvigu. Makseviivituse tõenäosuse hindamisel sellistele tunnustele tuginemine võib tekitada loodud mudeli rakendamisel ootamatuid tulemusi ja eksimisi ebasoodsas suunas. Makseviivituse tõenäosuse regulaarsel hindamisel on sellised ootamatused potentsiaalselt väga halva mõjuga. Tulenevalt alusandmestiku ärisaladuse kaitsmisest sellist algoritmi kasutatud tunnuste analüüsi töös ei teostatud. Aga rakendades erinevaid algoritme piisava edukusega on võimalus lõpliku otsuse tegemisel kaaluda sellist aspekti.

6 Kokkuvõte

Makseviivituse tõenäosuse hindamist on võimalik edukalt teha masinõppemeetoditel. Reaalelus on selle probleemi lahendamiseks tarvis lahendada mitmeid keerukusi, nagu vähene ennustatava klassi näidete arvukus, kasutatud algoritmi tunnuste mõju seletatavus või konkreetse näite ennustuse seletatavus. Kõiki neid keerukusi lahendati selle magistritöö raames. Logistilise regressiooni algoritm võimaldab edukalt makseviivituse tõenäosust hinnata, aga teised masinõppealgoritmid saavad selle probleemi lahendamisega võrreldavalt hakkama. Keerukamatel algoritmidel on puuduseid, eelkõige seletatavuse näol, aga need pakuvad täiendavaid võimalusi andmestiku seoste ja tulemuste mitmekülgsemaks mõistmiseks.

Selle magistritöö aluseks oleva AS LHV Group'i väike ja keskmise suurusega ettevõtete lepingute pealt õpitud otsustusmets suutis edukalt konkureerida logistilise regressiooni tulemustega, saavutades mõõdikutes statistiliselt olulisemaid tulemusi. Keerukama seletatavuse tasakaalustamiseks sai rakendatud erinevaid meetmeid, mida on võimalik rakendada teistele masinõppe algoritmidele kui need juhtuvad paremini toimima konkreetse andmestiku peal. Mudelite rakendamisel on erinevaid eesmärke ja lõplik otsus, mis meetodit kasutada, tuleb teha lähtuvalt eelkõige peamistest eesmärkidest ja olukorrast.

Jätkutööna saaks laiendada valitud algoritmide nimekirja. Kõige olulisema täiendusena tuleks katsetada keerukamaid närvivõrke kui mitmekihiline pertseptron. Näidetena tooksin välja kirjanduses mainitud rekurrentse tehisnärvivõrgu *(recurrent neural network)*, mille puhul saaks ära kasutada andmete perioodilisust. Selleks saab laiendada alusandmestikku 12 kuu pikkuse sammu asemel näiteks ühe kuu pikkusele sammule, mis suurendab andmestikku märkimisväärselt. Kui selline algoritm osutub paremaks, siis tuleb tööd laiendada seletatavuse osas. Sellest tööst jäi selline katsetus välja töö mahu kontrolli all hoidmiseks ja kasutatud andmestiku puhul oleks efekt arvatavasti liiga väike, et keerukuse kasvu õigustada.

Teise suunana saaks tööd laiendada katsetades erinevaid andmete transformatsioone. See muudaks mudeli otsuste tegemise seletatavust keerukamaks, aga võib ära tasuda mudeli paremate peamiste mõõdetud tulemustega. Sellised katsetused jäid välja töömahu ja seatud ajaraami hoidmiseks.

Viited

Breiman, Leo (oktoober 2001). "Random forests". *Machine Learning* 45.1, lk. 5–32. ISSN: 08856125. DOI: 10.1023/A:1010933404324.

Burkov, A. (2019). *The Hundred-page Machine Learning Book*. Andriy Burkov. ISBN: 9781999579500. URL: https://books.google.ee/books?id=ZF3KwQEACAAJ.

Chapman *et al.* (2000). "CRISP-DM 1.0: Step-by-step data mining guide".

Chawla, Nitesh V *et al.* (2002). "SMOTE: Synthetic Minority Over-sampling Technique". *Journal of Artificial Intelligence Research* 16, lk. 321–357.

Chen, Tianqi ja Carlos Guestrin (2016a). "XGBoost: A Scalable Tree Boosting System". URL: https://github.com/dmlc/xgboost.

— (2016b). "XGBoost: A Scalable Tree Boosting System". Teoses: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '16. San Francisco, California, USA: ACM, lk. 785–794. ISBN: 978-1-4503-4232-2. DOI: 10.1145/2939672.2939785. URL: http://doi.acm.org/10.1145/2939672.2939785.

Cortes, Corinna ja Vladimir Vapnik (september 1995). "Support-Vector Networks". *Machine Learning* 20.3, lk. 273–297. ISSN: 15730565. DOI: 10.1023/A:1022627411411. URL: https://dl.acm.org/doi/abs/10.1023/A%3A1022627411411.

Cramer, J. S. (detsember 2004). "The early origins of the logit model". *Studies in History and Philosophy of Science Part C: Studies in History and Philosophy of Biological and Biomedical Sciences* 35.4, lk. 613–626. ISSN: 1369-8486. DOI: 10.1016/J.SHPSC.2004.09.003.

Euroopa Parlament ja Euroopa Liidu Nõukogu (juuni 2013). *Euroopa Parlamendi ja Nõukogu määrus (EL) nr 575/2013*. URL: https://eur-lex.europa.eu/legal-content/ET/TXT/HTML/?uri=CELEX:32013R0575&from=ET#d1e15148-1-1.

— (aprill 2016). *Euroopa Parlamendi ja Nõukogu määrus (EL) 2016/679 - Isikuandmete kaitse üldmäärus*. URL: https://eur-lex.europa.eu/legal-content/ET/TXT/PDF/?uri=CELEX:32016R0679&from=EN.

Faris, Hossam *et al.* (märts 2020). "Improving financial bankruptcy prediction in a highly imbalanced class distribution using oversampling and ensemble learning: a case from the Spanish market". *Progress in Artificial Intelligence* 9.1, lk. 31–53. ISSN: 21926360. DOI: 10.1007/S13748-019-00197-9.

Fawcett, Tom (2005). "An introduction to ROC analysis". DOI: 10.1016/j.patrec.2005.10.010. URL: www.elsevier.com/locate/patrec.

Finantsinspektsioon (2022). *Juhendid ja märgukirjad*. URL: https://fi.ee/et/juhendid/pangandus-ja-krediit.

Finantsinspektsioon ja EBA (märts 2019). *Guidelines on LGD estimates under downturn conditions*. URL: https://www.fi.ee/sites/default/files/

2019-08/Guidelines%20on%20LGD%20estimates%20under%20downturn%20conditions_ET.pdf.

Greenwell, Brandon M, Bradley C Boehmke ja Andrew J Mccarthy (2018). "A Simple and Effective Model-Based Variable Importance Measure". URL: https://arxiv.org/abs/1805.04755.

Guo, Chuan *et al.* (2017). "On Calibration of Modern Neural Networks".

Harris, Charles R. *et al.* (september 2020). "Array programming with NumPy". *Nature* 585.7825, lk. 357–362. DOI: 10.1038/s41586-020-2649-2. URL: https://doi.org/10.1038/s41586-020-2649-2.

Hunter, J. D. (2007). "Matplotlib: A 2D graphics environment". *Computing in Science & Engineering* 9.3, lk. 90–95. DOI: 10.1109/MCSE.2007.55.

IFRS Foundation (2022). *IFRS 9 Financial Instruments*. URL: https://www.ifrs.org/issued-standards/list-of-standards/ifrs-9-financial-instruments/.

JetBrains s.r.o (2021). *JetBrains DataSpell: The IDE for Data Scientists*. URL: https://www.jetbrains.com/dataspell/.

Justiitsministeerium (november 2021). *Maksejõuetus*. URL: https://www.just.ee/era-ja-avalik-oigus/tsiviilmenetlus/maksejouetus.

Kim, Hyeongjun, Hoon Cho ja Doojin Ryu (august 2020). *Corporate default predictions using machine learning: Literature review*. DOI: 10.3390/SU12166325.

Kleehammer, Michael (2021). *pyodbc*. URL: https://pypi.org/project/pyodbc/.

Kluyver, Thomas *et al.* (2016). "Jupyter Notebooks – a publishing format for reproducible computational workflows". Teoses: *Positioning and Power in Academic Publishing: Players, Agents and Agendas*. Toim. F. Loizides ja B. Schmidt. IOS Press, lk. 87–90.

Kull, Meelis, Telmo M. Silva Filho ja Peter Flach (2017). "Beyond Sigmoids: How to obtain well-calibrated probabilities from binary classifiers with beta calibration". *Electronic Journal of Statistics* 11.2, lk. 5052–5080. ISSN: 19357524. DOI: 10.1214/17-EJS1338SI.

Küppers, Fabian *et al.* (juuni 2020). "Multivariate Confidence Calibration for Object Detection". Teoses: *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

Lemaître, Guillaume, Fernando Nogueira ja Christos K. Aridas (2017). "Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning". *Journal of Machine Learning Research* 18.17, lk. 1–5. URL: http://jmlr.org/papers/v18/16-365.html.

Martínez-Plumed, Fernando *et al.* (2017). "Context Aware Standard Process for Data Mining". URL: http://www.casp-dm.org.

Mcculloch, Warren S ja Walter Pitts (1943). “A logical calculus of the ideas immanent in nervous activity”. *Bulletin of methematical biophysics* 5.

Molnar, Christoph (2022). *Interpretable Machine Learning. A Guide for Making Black Box Models Explainable*. 2. väljaanne. URL: https://christophm.github.io/interpretable-ml-book.

Pedregosa, F. *et al.* (2011). “Scikit-learn: Machine Learning in Python”. *Journal of Machine Learning Research* 12, lk. 2825–2830.

Ribeiro, Marco Tulio, Sameer Singh ja Carlos Guestrin (2016). “"Why Should I Trust You?"Explaining the Predictions of Any Classifier”. DOI: 10.1145/2939672.2939778. URL: http://dx.doi.org/10.1145/2939672.2939778.

Student (1908). “The Probable Error of a Mean”. *Biometrika* 6.1, lk. 1–25. ISSN: 00063444. URL: http://www.jstor.org/stable/2331554 (vaadatud 06.05.2022).

team, The pandas development (veebruar 2020). *pandas-dev/pandas: Pandas*. Versioon latest. DOI: 10.5281/zenodo.3509134. URL: https://doi.org/10.5281/zenodo.3509134.

Van Rossum, Guido ja Fred L Drake Jr (1995). *Python reference manual*. Centrum voor Wiskunde en Informatica Amsterdam.

Varusk, Merike (2008). “Maksejõuetus-mis see on?” URL: https://www.just.ee/sites/www.just.ee/files/merike_varusk._maksejouetus_-_mis_see_on.pdf.

Veganzones, David ja Eric Séverin (august 2018). “An investigation of bankruptcy prediction in imbalanced datasets”. *Decision Support Systems* 112, lk. 111–124. ISSN: 01679236. DOI: 10.1016/j.dss.2018.06.011.

Vicente, Raul ja Kallol Roy (2021). *Õppeaine Tehisnärvivõrgud*. URL: https://courses.cs.ut.ee/LTAT.02.001/2021_spring/uploads/Main/Lecture1_NN21.pdf. (alates slaid 31, vaadatud: 18.11.2021).

Virtanen, Pauli *et al.* (2020). “SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python”. *Nature Methods* 17, lk. 261–272. DOI: 10.1038/s41592-019-0686-2.

Waskom, Michael L. (2021). “seaborn: statistical data visualization”. *Journal of Open Source Software* 6.60, lk. 3021. DOI: 10.21105/joss.03021. URL: https://doi.org/10.21105/joss.03021.

Yang, Z. R., Marjorie B. Platt ja Harlan D. Platt (veebruar 1999). “Probabilistic Neural Networks in Bankruptcy Prediction”. *Journal of Business Research* 44.2, lk. 67–74. ISSN: 0148-2963. DOI: 10.1016/S0148-2963(97)00242-7.

Sõnastik

***bootstrapi* meetod** Juhuslik näidete valimine asendusega. 15

AS LHV Group Suurim eestimaine finantskontsern, mis tegutsenud alates aastast 1999. 2, 4, 7, 22, 32

AUC (Eng: *Area Under Curve*) Kõvera alune pindala. 25

CRISP-DM (Eng: *Cross-Industry Standard Process for Data Mining)* Standardne meetod andmekaeve efektiivseks teostamiseks. 5, 9, 21

DataSpell Ettevõtte JetBrains loodud integreeritud arendamiskeskkond andmeteaduse tegemiseks (JetBrains s.r.o 2021). 21

F-mõõdik (Eng: *f-measure*) Mõõdik, mis on harmooniline keskmine täpsuse ja saagise mõõdikutest. 11

gradientlaskumine (Eng: *gradient descent*) Optimiseerimisalgoritm leidmaks diferentseeritava funktsiooni lokaalset miinimumi. 13, 16

IFRS9 (Eng: *International Financial Reporting Standard, version 9*) Rahvusvaheline finantsinstrumentide standard. (IFRS Foundation 2022). 21

imblearn Pythoni teek, mis pakub erinevaid meetodeid andmestikus näidete tasakaalu taastamiseks. (Lemaître, Nogueira ja Aridas 2017). 24

Jupyter märkmik Vabavaraline ja vaba standarditega tarkvara teostamaks andmeteadust ja teaduslikku arvutamist erinevates programmeerimiskeeltes. (Kluyver *et al.* 2016). 21

järguline kodeerimine (Eng: *ordinal encoder*) Kategooriliste andmete numbriliseks teisendamine säilitades kategooriate järjestatuse. 23

kaofunktsioon (Eng: *loss function*) Funktsioon, mis väärtustab ennustuse tulemuse mingi numbrilise kao väärtusega, mis ilmestab selle ennustuses tehtud kadu. 16, 17

LIME (Eng: *Local Interpretable Model-agnostic Explanations*) Tehnika mudelist sõltumatuks tulemuste tõlgendamiseks (Ribeiro, Singh ja Guestrin 2016). 19, 25

logistiline regressioon (Eng: *logistic regression*) Klassifitseerimisalgoritm, mis maksimeerib tõepära, et testandmed on kirjeldatud logistilise funktsiooniga. 12, 13, 24

makseviivitus (Eng: *default*) Pikaajaline makseraskus, mis viivitab kohustuste täitmist. 2, 7, 10–13, 15, 18, 19, 21, 22, 24, 25, 29, 32

matplotlib Pythoni teek liidese kaudu jooniste loomiseks. (Hunter 2007). 21

netcal Pythoni teek mudeli kalibreerituse kontrollimiseks ja parandamiseks. (Küppers *et al.* 2020). 25

numpy Pythoni teek massiivide ja maatriksite mugavamaks töötlemiseks ning matemaatiliste funktsioonide rakendamiseks. (Harris *et al.* 2020). 21

otsustusmets (Eng: *random forest*) Klassifitseerimisalgoritm, mis kasutab juhuandmetelt koostatud otsustuspuude ansamblit klassi kuulumise tõenäosuse hindamiseks. 2, 14, 15, 24, 26–29, 31, 32

otsustuspuu (Eng: *decision tree*) Klassifitseerimisalgoritm, mis kirjeldab andmete pealt reeglid klassi kuulumise tõenäosuse hindamiseks. 2, 14–16, 18, 24, 27

paarikaupa t-test Statistiline test võrdlemaks kahe normaaljaotusega tunnuse keskväärtuste erinevust. 26

pandas Pythoni teek andmete töötlemiseks ja analüüsiks. (team 2020). 21

PDP (Eng: *Partial Dependence Plot*) Osalise sõltuvuse graafik, millega ilmestatakse erinevate tunnuste mõju ennustusele. 19, 25

PR AUC (Eng: *Precision Recall Area Under Curve*) Täpsus-saagis kõvera alla jääva graafiku ala pindala, mida kasutatakse PR-kõvera hindamiseks. 26, 31

pyodbc Pythoni teek OBDC standardiga andmebaasile ligipääsemiseks (Kleehammer 2021). 24

Python Kõrgetasemeline ja üldiseks kasutamiseks mõeldud programmeerimiskeel (Van Rossum ja Drake Jr 1995). 21

ristvalideerimine (Eng: *cross validation*) Meetod andmestikus parima mudeli leidmiseks, kus x grupiks jagatud andmestiku ühte gruppi kasutatakse tulemuste hindamiseks ja teisi treenimiseks ning protsessi korratakse nii, et iga grupp saab olla hinnatavaks. 24

ROC AUC (Eng: *Receiver Operating Characteristic Area Under Curve*) ROC-graafiku alla jääva ala pindala, mida kasutatakse ROC-graafiku hindamiseks. 5, 10, 11, 25, 26, 31

saagis (Eng: *recall*) Mõõdik, mis näitab õigesti positiivseks hinnatud tulemuste määra kõigist tegelikult positiivsetest näidetest. 11

SciPy Pythoni teek, mis sisaldab erinevaid teadusarvutuse algoritme (Virtanen *et al.* 2020). 25

seaborn Pythoni kõrgetasemeline teek liidese kaudu jooniste loomiseks (Waskom 2021). 21

segadusmaatriks (Eng: *confusion matrix*) Klassifitseerimise probleemides ennustatud tulemuste tabeli kujul kirjeldamine, kus kajastatakse iga ennustuse tegelikku ja ennustatud klassi eri telgedel. 10

SMOTE (Eng: *Synthetic Minority Over-Sampling Technique)* Tehnika sünteetiliste andmete loomiseks läbi üle-näidustamise (Chawla *et al.* 2002). 5, 17, 18, 24

standardne skaleerimine (Eng: *standard scaling*) Meetod andmestiku numbriliste väärtuste viimiseks samale skaalale läbi keskväärtuse lahutamise ja standardhälbega jagamise. 23

tehisnärvivõrk (Eng: *neural network*) Klassifitseerimisalgoritm, mis õpib etteantud kihtides ära iga tunnusega seotud signaali mõju ennustusele. 24

tugivektormasin (Eng: *support vector machine*) Klassifitseerimisalgoritm, mis ehitab sisendvektori peale kõrg-dimensioonilise tunnuste ruumi, kus leiab otsustuspinna, mis eristab ennustatavaid klasse. 2, 13, 18, 24, 27

täpsus (Eng: *precision*) Mõõdik, mis näitab õigesti positiivseks hinnatud tulemuste määra kõigist positiivseks ennustatud ennustustest. 11

täpsus-saagis kõver (Eng: *precision recall curve*) Mõõdik, mis näitab täpsuse ja saagise suhet erinevatel künnistel. 11

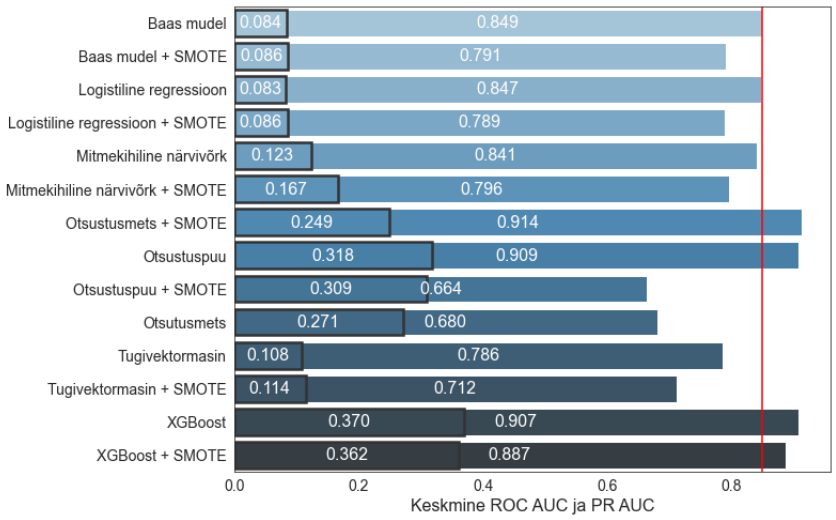
XGBoost (Eng: *XGBoost*) Klassifitseerimisalgoritm, mis kasutab otsustuspuude ansamblit, kus iga otsustuspuu lehe tulemust arvestatakse kogu mudeli ennustuses (Chen ja Guestrin 2016b). 16, 17, 24, 25

õigsus (Eng: *accuracy*) Mõõdik, mis näitab kui suur hulk ennustustest olid õiged. 11, 25

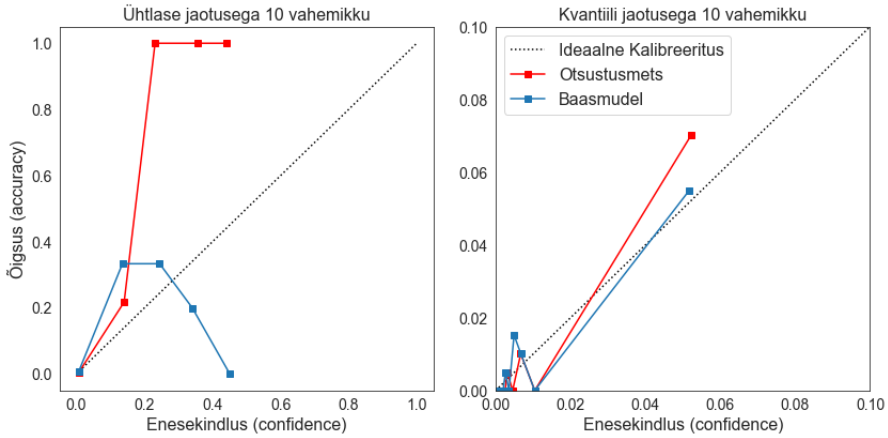
ühega kodeerimine (Eng: *one-hot encoding*) Kategooriliste andmete numbriliseks teisendamine luues iga tunnuse kohta uue tunnuse ja väärtustades selle numbriga 0 või 1, vastavalt kas see kategooria on näitel olemas. 23

Lisad

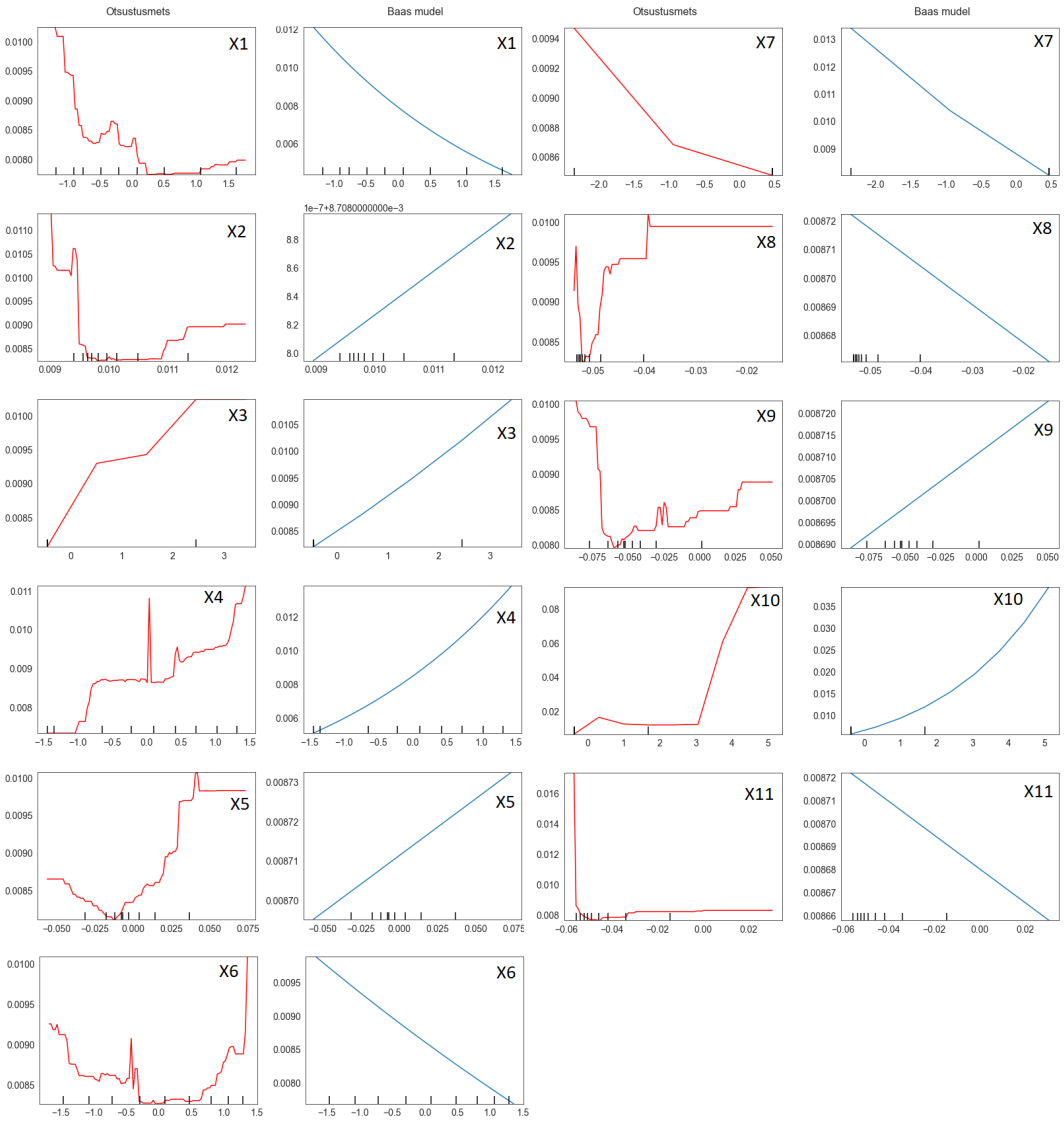
I. Täiendavad joonised



Joonis 14: Kõikide mudelite 10 katse keskmine ROC AUC tulemus, kus punane joon tähistab baasmudeli tulemust ja tumedama joonega on tähistatud iga mudeli keskmine PR AUC



Joonis 15: Otsustusmetsa ja logistilise regressiooni usaldusdiagrammid 10 vahemikuga



Joonis 16: Kõikide mitte binaarsete tunnuste PDP joonised

II. Litsents

Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks

Mina, **Martti Praks**,

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

   Masinõppe rakendamine makseviivituse tõenäosuse hindamisel,

   mille juhendajad on Markus Kängsepp, Meelis Kull ja Kuldar Kõiv,

   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.

Martti Praks
10.05.2022