Ensemble learning -menetelmät luokitteluongelmissa

Juha Kavka

LuK-tutkielma
Turun yliopisto

Huhtikuu 2020

Sisältö

1 Johdanto

Ensemble learning -menetelmät ovat tulleet suosituiksi koneoppimisen eri sovelluksissa. Termi viittaa menetelmiin, joissa aineiston mallinnuksessa käytetään usean mallin joukkoa yhden mallin sijaan. Mallin lopullinen ennuste muodostetaan yhdistämällä osamallien ennusteet. Lopullinen ennuste muodostetaan yleensä ennustejoukon keskiarvosta tai painotetusta keskiarvosta. Näitä menetelmiä käytetään yleensä sen vuoksi, että niillä saadaan parannettua ennustetarkkuutta yksittäiseen malliin verrattuna. Esimerkiksi *puumalleissa* voidaan käyttää *bagging-* ja *boosting*-menetelmiä parantamaan mallin ennustetarkkuutta. Enseble learning -menetelmiä voidaan käyttää sekä *regressio-* että luokitteluongelmissa.

Tässä tutkielmassa tarkastellaan ensemble learning -menetelmiä ja sovelletaan niitä esimerkkiaineston luokitteluongelmaan. Toisessa luvussa esitetään menetelmän perusperiaatteet. Luvun lähteenä on käytetty Lior Rokachin teosta *Pattern Classification Using Ensemble Methods* ja Dr. Robi Polikarin artikkelia *Ensemble learning* Scholarpedia sivustolla. Kolmannessa luvussa tarkastellaan yleisiä ensemble learning -sovelluksia. Luvun päälähteenä on käytetty Trevor Hastien, Robert Tibshiranin ja Jerome Friedmanin teosta *The Elements of Statistical Learning*. Lisäksi neljännessä luvussa tarkastellaan esimerkkiaineistolla, miten enseble learning -menetelmät vaikuttavat tilastollisen mallin suorituskykyyn. Esimerkkiaineiston mallinnuksessa on käytetty R-ohjelman *Super Learner* -mallia. Lopuksi esitetään tutkielman johtopäätökset.

2 Ensemble learning -menetelmät

2.1 Tausta

Ensemble learning -menetelmän periaatteena on muodostaa tilastollinen malli, joka koostuu usean mallin joukosta. Etuna on se, että mallien joukon yhdistetty ennuste on todennäköisesti tarkempi kuin satunnaisesti valittu yksittäinen ennuste. Ensemble learning -menetelmä jäljittelee ihmisen luontaista

tarvetta tutkia useita eri vaihtoehtoja ennen lopullista päätöstä [1]. Kirjallisuudessa viitataan usein ensemble learning -termillä menetelmiin, joissa tilastollinen malli muodostetaan saman mallin varioidusta joukosta, kuten esimerkiksi random forest -puumalli. Ensemble learning -algoritmeja voidaan kuitenkin rakentaa myös täysin eri malliperheisiin kuuluvista malleista.

Ensemble learning -menetelmät ovat hyödyllisiä tilanteissa, joissa on käytettävissä liian paljon tai liian vähän aineistoa. Yksittäisen mallin opettaminen voi olla hankalaa, jos aineistoa on todella paljon. Tällöin on olemassa muun muassa ylisovittumisen vaara. Tällaisessa tilanteessa aineisto voidaan jakaa osa-aineistoihin ja sovittaa niihin malli niihin. Jos taas aineistoa on liian vähän, voidaan aineistosta ottaa useampi satunnaisotos bootstrap-menetelmällä ja sovittaa malli käyttämällä näitä otoksia. [2]

2.2 Historia

Rokach kirjoittaa Ensemble learning -menetelmien historiasta kirjassaan *Pattern Classification Using Ensemble Methods*. Menetelmän juuret ulottuvat vuoteen 1977, jolloin Tukey yhdisti kaksi lineaarista regressiomallia samaan aineistoon. Hän sovitti yhden mallin havainnoille ja toisen mallin residuaaleille. Vuonna 1979 Dasarathy ja Sheela jakoivat aineiston eri luokkiin ja sovittivat niihin omat mallinsa. 1990-luvulla menetelmä otti suureen harppauksen eteenpäin, kun Hansen ja Salamon sovittivat useasta neuroverkosta koostuvan mallin aineistoon. Samaan aikaan Shapire loi perustan palkittuun AdaBoost -algoritmiin.[1] Algoritmi laajentaa boosting -menetelmän moniluokkaisiin regressio-ongelmiin. AdaBoostin myötä ensemble learning -menetelmien kehittyminen on ollut voimakasta. Menetelmistä on kehitetty monia edistyksellisiä sovelluksia, kuten *composite classifier systems, mixture of experts, stacked generalization systems, combinations of multiple classifier, dynamic classifier systems, classifier fusion, classifier ensembles* ja monia muita. [2]

2.3 Filosofia

Rokachin mukaan ensemble learning -menetelmän filosofinen ajatus perustuu siihen, että useasta lähteestä kerätyn tiedon perusteella tehty päätös on todennäköisemmin oikeassa kuin yhden lähteen perusteella. Englantilainen filosofi ja tilastotieteilijä Sir Francis Galton (1822-1911) teki kokeen, jossa pyysi ihmisiä arvaamaan härän painon. Hän huomasi, että arvauksien keskiarvo oli lähellä härän todellista painoa, vaikka kukaan ei arvannut painoa täysin oikein. Hän päätteli, että arvauksien keskiarvo on todennäköisemmin lähempänä oikeaa arvoa kuin mikään yksittäinen arvaus. Galton havaitsi, että yhdistämällä monia eri ennusteita saavutetaan tarkempi ennuste.

Amerikkalainen taloustoimittaja Surowiecki (1967-) on kirjoittanut teoksen *The Wisdom of Crouds: Why Many Are Smarter Than the Few and How Collective Wisdom Shapes Business, Economies, Societys and Nations*. Surowiecki väittää, että tiettyjen olosuhteiden vallitessa useaan mielipiteeseen perustuva päätös on usein parempi kuin yhteen mielipiteeseen perustuva päätös. Tietenkään kaikki eivät ole välttämättä viisaita. Surowiecki mukaan ollakseen viisas ryhmän tulee täyttää tietyt kriteerit. Mielipiteiden tulee olla monimuotoisia. Tämä tarkoittaa sitä, että jokaisella tulee olla henkilökohtainen tieto asiasta. Lisäksi jokaisen tulee muodostaa mielipide itsenäisesti. Mielipiteiden täytyy olla myös paikallisia, koska jokainen on paras asiantuntija omalla alueellaan. Näiden lisäksi on oltava jokin menetelmä, joka koostaa yksityiset päätökset yleiseksi. [1]

2.4 Mallin valinta

Mallin valinnan ongelma on eräs yleinen syy käyttää ensemble learning -menetelmiä. Usein mallinnusta suunniteltaessa herää tärkeä kysymys, mikä malli on paras kyseessä olevaan ongelmaan. Yleensä valinta tapahtuu testiaineiston ennustevirheen perusteella, mutta kuinka voidaan olla varmoja, että kyseessä oleva malli ennustaa parhaiten myös uutta aineistoa. Onkin mahdollista, että ei valita parhaiten uutta aineistoa ennustavaa mallia. Tätä riskiä voidaan hallita ensemble learning -menetelmillä. Tällöin muodostetaan ennusteet usean mallin perusteella. Lopullinen ennuste muodostuu yleensä

näiden keskiarvosta tai painotetusta keskiarvosta. Malli ei ole välttämättä parempi kuin paras yksittäinen malli, mutta todennäköisesti parempi kuin satunnaisesti valittu malli. Jotta menetelmä olisi tehokas, mallien tulisi olla riittävän erilaisia. [2]

2.5 Ennusteiden yhdistämissäännöt

Ensemble learning -mallien ennusteiden muodostamisessa voidaan käyttää erilaisia yhdistämissääntöjä. Lopullinen ennuste muodostetaan yleensä ennusteiden keskiarvosta tai painotetusta keskiarvosta. Lisäksi on muita tapoja yhdistää ennusteet. Yhdistäminen voidaan toteuttaa myös erillisellä mallilla, kuten esimerkiksi logistisella regressiolla. Yhdistämisalgoritmi käyttää syötteenä osamallien ennusteita, jotka voivat olla diskreettejä tai jatkuvia muuttujia. Jatkuvat muuttujat voidaan tulkita todennäköisyyksiksi kuulua tiettyyn luokkaan. Yksinkertaisimmillaan syöte on vain diskreetti muuttuja, joka määrittelee mihin luokkaan havainto todennäköisimmin kuuluu. Toinen tapa on määrittää luokille järjestysasteikollinen arvo, joka perustuu todennäköisyyden kuulua eri luokkaan. Syöte voi myös olla jatkuva-arvoinen todennäköisyyden estimaatti. Jatkuva-arvoinen ennuste voi olla luokkakohtainen todennäköisyyden estimaatti tai luokkakohtainen uskottavuusestimaatti. On myös muita laskennallisia tapoja muodostaa lopullinen ennuste, kuten esimerkiksi mediaanisäätö, tulosääntö. Vaihtoehtoisesti voidaan myös minimoida tai maksimoida erilaisia funktioita ennusteiden yhdistämisessä. [2]

2.6 Diversiteetti

Ensemble learning -mallin tehokkuus perustuu kykyyn korjata osamallien virheitä. Tämän takia on tärkeää, että mallit ovat riittävän erilaisia toisiinsa nähden. Tällöin yhden mallin väärin luokiteltu havainto voidaan korjata toisilla saman havainnon paremmin luokittelevilla malleilla. Tämän johdosta mallin ennustevirhettä voidaan saada pienemmäksi. Menetelmä tarvitsee erityisesti luokittelijoita, joilla on erilaiset päätösrajat. Luokittelijoiden diversiteetti voidaan saavuttaa monin eri tavoin. Yleisin tapa opettaa malli on käyttää eri satunnaisotosta opetusaineistosta kullekin osamallille. Satun-

naisotokset voidaan muodostaa bootstrap-menetelmällä, jonka vuoksi osa-aineistossa voi olla sama havainto useampaan kertaan. Tämä lisää mallin diversiteettiä. Diversiteettiä voidaan kasvattaa myös käyttämällä satunnaisesti eri selittäjämuuttujia, kuten random forest -mallissa. Yksi tapa kasvattaa diversiteettiä on yhdistää ensemble -malliin eri malliperheiden malleja. Diversiteetin kannalta on optimaalinen tilanne, jos luokittelijoiden ennusteet ovat luokkaehdollisesti riippumattomia ja korreloimattomia. [2]

3 Ensemble learning -sovelluksia

3.1 Bagging

Bagging (*bootstrap aggregating*) -menetelmä on yksi vanhimmista ja yksinkertaisimmista ensemble learning -menetelmistä. Mallin diversiteetti muodostetaan satunnaisotoksilla. Siinä aineistosta otetaan usea satunnaisotos bootstrap -menetelmällä. Satunnaisotos suoritetaan siten, että otetaan joukko satunnaisotoksia takaisinpanolla. Näin ollen otoksiin voi tulla sama havainto useampaan kertaan. Menetelmällä voidaan tarkentaa estimaattia tai ennustetta. Bootstrap-menetelmän ja Bayes-mallien välillä on yhteys. Bootstrap -keskiarvo on approksiivisesti posteriori-jakauman keskiarvo. [3] Bagging on tehokas menetelmä pienentää estimaatin harhaa. Vaikka menetelmä on yksinkertainen, sen on havaittu parantavan merkittävästi mallien suorituskykyä.

Bagging-menetelmää käytetään esimerkiksi parantamaan puumallien (bagged tree) ennustetarkkuutta. Siinä otetaan monta satunnaisotosta bootstrap-menetelmällä, esimerkiksi 500 otosta. Jokaiseen otokseen sovitetaan oma puumalli. Lopullinen ennuste muodostetaan kaikkien mallien keskiarvosta. On havaittu, että puumallit ovat herkkiä satunnaisotokselle. Mallit voivat olla hyvin erilaisia eri otoksille. Usealla satunnaisotoksella saadaan pienenettyä satunnaisuuden vaikutusta ja pienenettyä mallin varianssia. Tästä johtuen bagged tree -mallit tuottavat tarkempia ennusteita yksittäiseen puumalliin verrattuna. Bagging-menetelmä toimii erityisen hyvin malleissa, joissa on suuri varianssi, mutta pieni harha. Tällaisia ovat esimerkiksi epäli-

neaariset puumallit.

3.2 Random forest

Random forest on bagging- menetelmän laajennus. Siinäkin sovitetaan puumalleja joukolle bootstrap-otoksia. Erona on se, että malleissa käytetään vain satunnaisesti valittua joukkoa selittäviä muuttujia. Ideana on muodostaa joukko keskenään korreloimattomia puumalleja. Yleinen hyväksi havaittu sääntö on valita selittäjämuuttujien määräksi neliöjuuri alkuperäisten selittäjien määrästä jokaisessa satunnaisotoksessa. Random forest -menetelmä on tullut suosituksi hyvän suorituskyvyn ja helpon optimoinnin vuoksi. Kaikki mallit eivät hyödy tällaisesta aineiston 'sekoittamisesta'. On havaittu, että epälineaariset mallit, kuten puumallit, hyötyvät eniten tästä menetelmästä. [3]

3.3 Boosting

Boosting on yksi tehokkaimmista viime vuosikymmeninä kehitetyistä ensemble learning -menetelmistä. Menetelmä kehitettiin aluksi luokitteluongelmiin, mutta siitä on kehitetty sovelluksia myös regressio-ongelmiin. Menetelmän ideana on kehittää joukko heikkoja luokittimia, joiden yhdistelmästä muodostetaan vahva luokitin. Boosting-menetelmä eroaa oleellisesti bagging-menetelmästä. Boosting luo iteratiivisesti painoarvon jokaiselle eri havainnolle aineistossa. Painoarvo on suuri huonosti luokitetuille havainnolle ja pieni hyvin luokitetuille havainnolle. Kun aineistosta otetaan uusi satunnaisotos, painoarvot vaikuttavat havaintojen todennäköisyyteen tulla valituksi uuteen satunnaisotokseen. Jokaisella iteraatiolla päivitetään painokertoimia. Menetelmä siis pakottaa algoritmin keskittymään huonosti luokiteltuihin havaintoihin seuraavalla iteraatiolla. [3] Menetelmä opettaa algoritmia pikkuhiljaa parhaaseen ennustekykyyn optimoimalla havaintojen painoarvoa satunnaisotoksissa.

Boosting-menetelmästä on kehitetty useita variaatioita. *Gradiet boosting* on edelleen kehitetty versio boosting -menetelmästä. Ero on siinä, kuinka väärin luokitellut havainnot poimitaan. Boosting-menetelmässä luokittelun

onnistuminen määritetään havaintojen painokertoimilla, kun taas gradient boosting käyttää mittana tappiofunktiota. Menetelmän etuna on se, että mallin tappiofunktiota voidaan optimoida parhaan ennustetarkkuuden saavuttamiseksi. Boosting-menetelmästä on kehitetty monia edistyksellisiä algoritmeja, kuten tarkasti approksimoiva *XGboost*, jatkuva-arvoista ennustetta käyttävä *Real AdaBoost* ja logististista regressiota käyttävä *LogitBoost*.

3.4 Bucket of models

Bucket of models on varsin yksinkertainen menetelmä. Siinä ideana on sovittaa aineistoon joukko tilastollisia malleja yksi kerrallaan. Jokaisen mallin ennustevirhe lasketaan esimerkiksi ristiin validoimalla. Kyseiseen ongelmaan valitaan malli, joka tuottaa alhaisimman ennustevirheen. Menetelmä on hyvin yksinkertainen ja suoraviivainen tapa valita ennustevirheeltään paras malli tiettyyn ongelmaan. Menetelmän heikkous on se, että ei voida olla varmoja ennustaako valittu malli parhaiten myös uutta aineistoa.

3.5 Bayes-luokittelijat

Bayes-mallit voidaan myös luokitella ensemble learning- menetelmiin. *Bayes Optimal Classifier* on havaittu tehokkaaksi menetelmäksi luokitella uusia havaintoja. Sen perustana on Bayesin teoria, joka perustuu estimoitavien parametrien ja havaintojen ehdolliseen yhteisjakaumaan (*posteriori*). Todennäköisin luokka löydetään yhdistämällä kaikki hypoteesit painotettuna posteriorijakaumalla. Malli siis etsii todennäköisintä hypoteesia tietyn ennusteen sijaan. Menetelmä on laskennallisesti raskas, joten siitä on kehitetty yksinkertaistettuja versioita, kuten *Gibbsin* algoritmi ja *Naiivi Bayes* -malli.

Naiivi Bayes on yksinkertaistettu malli oletuksella, jonka mukaan havainnot ovat ehdollisesti riippumattomia luokasta. Epärealistisesta oletuksesta huolimatta, Naiivi Bayes -malli on hyvin käyttökelpoinen monissa tilanteissa. Bayes-malleista on kehitetty myös edistyneempiä sovelluksia, kuten *Bayesian model averaging (BMA)* ja siitä edelleen kehitetty *Bayesian model combination (BMC)*.

3.6 Stacking

Ensemble learning -malli voidaan koota myös täysin eri malliperheisiin kuuluvista malleista. Menetelmää kutsutaan termillä *stacking*. Menetelmä pyrkii määrittämään, mitkä luokittelijat ovat luotettavia ja mitkä eivät. Menetelmässä algoritmi kootaan eri malliperheisiin kuuluvista malleista. Malli voitaisiin koota esimerkiksi tukivektorikoneesta, lähinaapurin mallista, naiivista Bayes-mallista ja puumallista. Jokainen malli opetetaan opetusaineiston eri satunnaisotoksesta. Otokset muodostetaan usein bootstrap-menetelmällä. Seuraavaksi kootaan uusi opetusaineisto luokittelijoiden ennusteista. Koostettu aineisto syötetään algoritmille, joka muodostaa syötteistä lopullisen ennusteen. Tehtävässä käytetään usein logistista regressiota. Kehittyneissä stacking-algoritmeissa päästään parempaan ennustustarkkuuteen kuin parhaan yksittäisen mallin. Esimerkiksi R-ohjelmistosta löytyy Super Learner -paketti, josta löytyy työkalut stacking-mallin rakentamiseen.

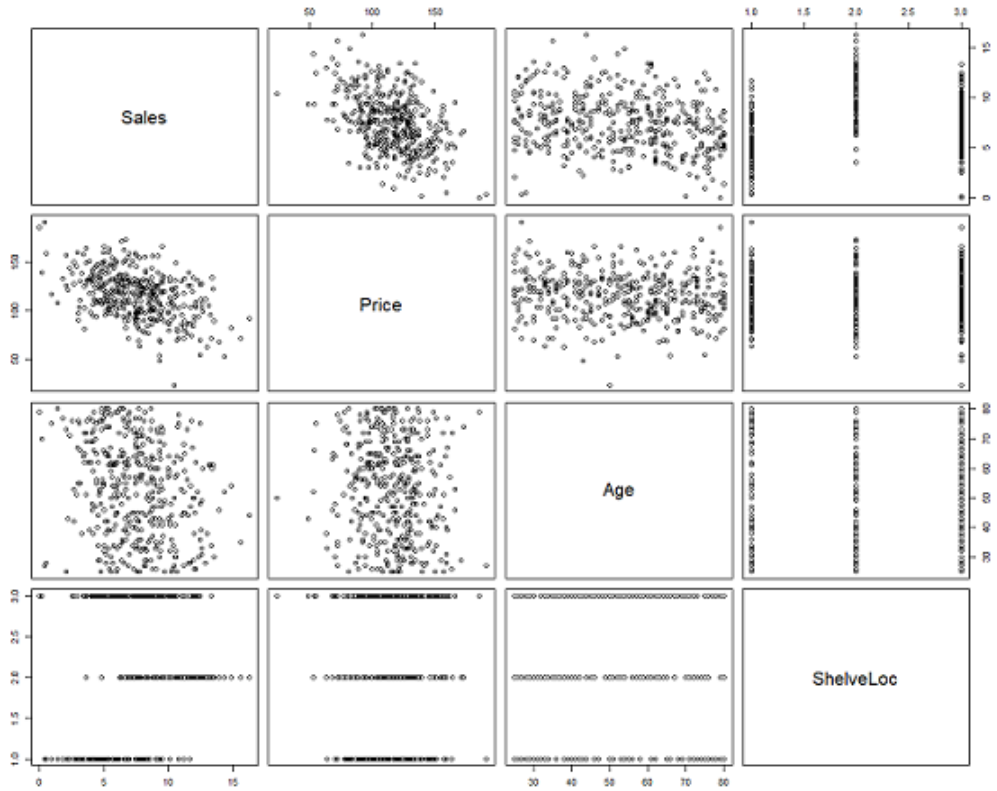
4 Empiirinen aineisto

Havaintoaineistona käytän R-ohjelman *ISLR*-paketista löytyvää *Carseat* -aineistoa. Tehtävänä on ennustaa auton istuinten myynnin tasoa taustamuuttujien perusteella eri alueilla. Aineisto sisältää 400 havaintoa, jotka on jaettu satunnaisesti 200 havainnon opetus- ja testiaineistoon. Aineistoon on lisätty kaksiluokkainen muuttuja High, joka kuvaa auton istuinten myynnin tasoa. High saa arvon No, jos myynnin taso on alle 8000 kappaletta tietyllä alueella ja Yes, jos myynnin taso on yli 8000 kappaletta. Taulukossa 1 esitetään aneiston muuttujat.

Carseats- data (muuttujat ja selite)

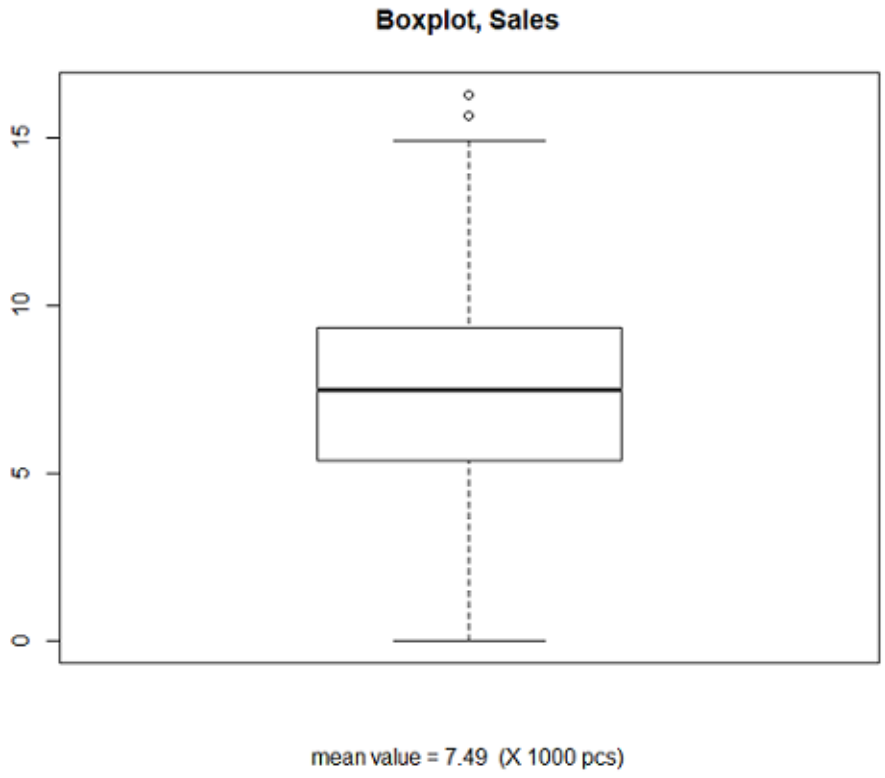
High	Myytyjen istuinten lukumäärä (yli/ alle 8000 kpl)
Sales	Myytyjen istuinten lukumäärä myymälässä (tuhat kpl)
CompPrice	Kilpailijan hinta auton istuimelle (dollaria)
Income	Alueen tulotaso (tuhat dollaria)
Advertising	Paikallinen markinointibudjetti (tuhat dollaria)
Population	Alueen asukasluku (tuhat ihmistä)
Price	Istuimen hinta tietyllä alueella (dollaria)
ShelveLoc	Istuimen sijainti myynihyllyssä (bad, good, medium)
Age	Paikallisten asukkaiden keski-ikä
Education	Paikallisten asukkaiden koulutustaso
Urban	Sijaitseeko kauppa kaupunki- vai maaseudulla (Yes/No)

Taulukko 1: Carseat -aineiston muuttujat



Kuva 1: Muuttujien suhteita

Kuvasta 1 havaitaan, että istuimen hinnan kasvu laskee selvästi myyntiä. Alueen keski-iällä ja myynnillä näyttäisi olevan lievä yhteys. Keski-iän kasvaessa myynnin taso laskee hieman. Kaupoissa, joissa istuimet on sijoitettu parhaaseen paikkaan hyllyssä, on parempi myynti kuin kaupoissa, joissa istuimet on sijoitettu huonommille hyllypaikoille. Kuvassa 1 tarkastellaan osaa selittävistä muuttujista. Tilastollinen mallinnus tehdään kaikille taulukon 1 muuttujille.

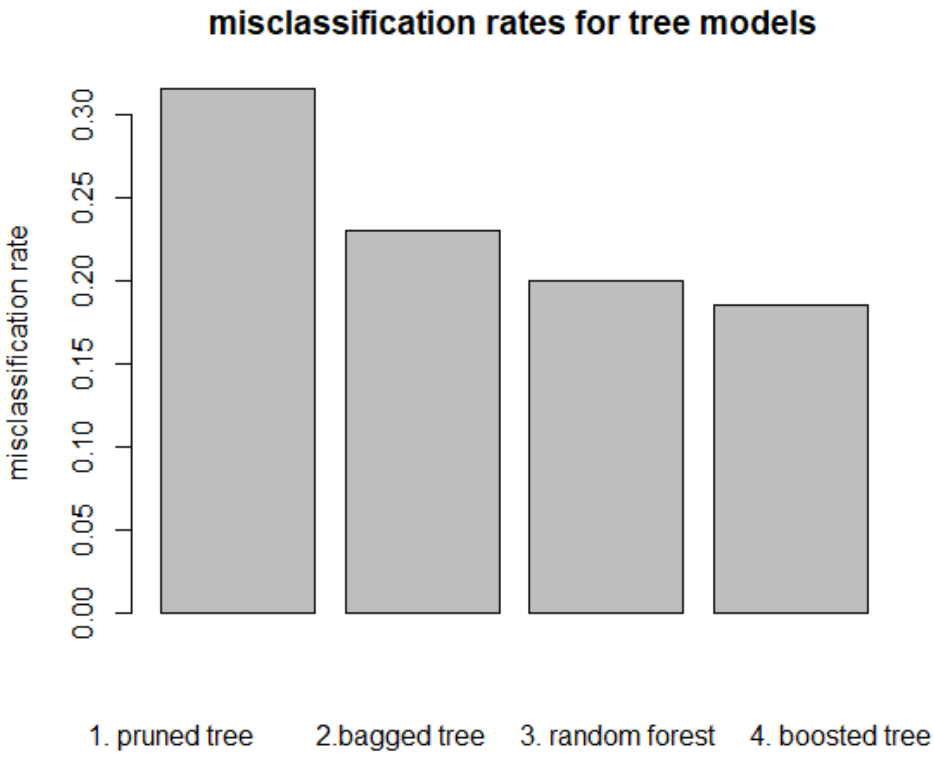


Kuva 2: Laatikko-janakuva Sales -muuttujalle

Vastemuuttuja Sales on laatikko-janakuvan (kuva 2) mukaan symmetrisesti jakautunut. Muuttujan mediaani on 7,49. Alakvartiili on 5,39 ja yläkvartiili on 9,32. Myynnin keskiarvo on 7496 myytyä istuinta myymälää kohti. Aineistossa on kaksi poikkeavaa havaintoa, jotka nähdään kuvassa 2.

4.1 Puumallit

Ennustetaan auton istuinten myynnin tasoa käyttämällä enseble learning -menetelmiä. Sovitetaan ensiksi aineistoon puumallit käyttämällä bagging-, random forest- ja boosting-menetelmiä.



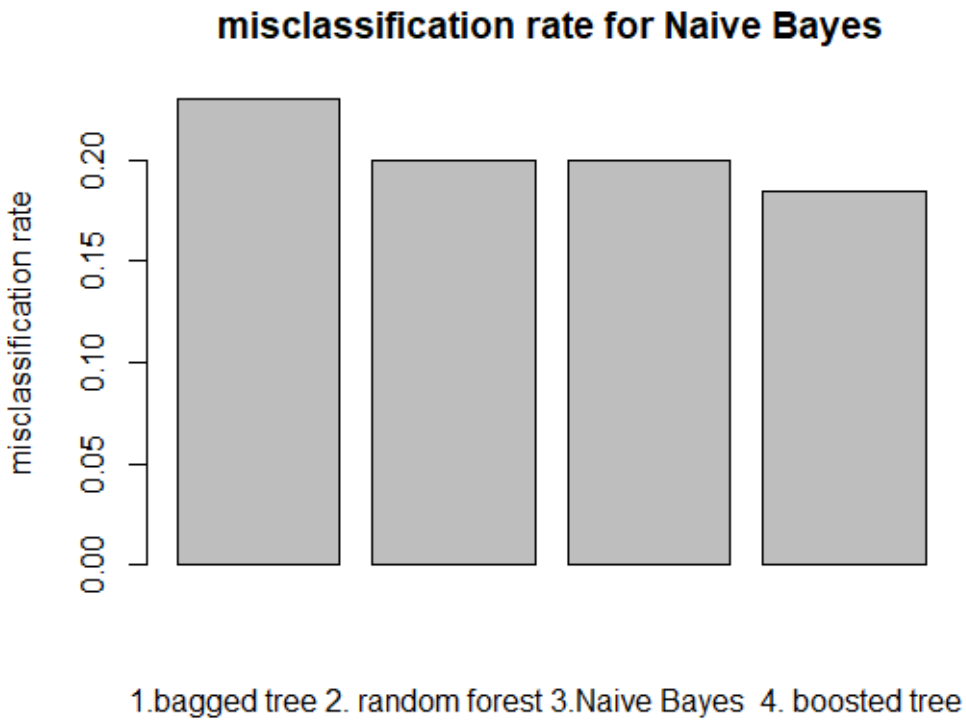
Kuva 3: Pylväsdiagrammi puumallien luokitteluvirheille

Mallinnus toteutettiin 1000 puun ensemble-malleilla. Random forest -mallissa valittiin kolme satunnaista selittäjää puuta kohti. Parhaan ennusteen myynnin tasolle tuottaa boosted tree -malli, jolla on alhaisin ennustevirhe 0,185. Toiseksi parhaan tuloksen tuottaa random forest, jonka ennustevirhe on 0,2. Bagged tree -mallilla on joukon suurin ennustevirhe 0,23. Kaikki enseble learning -mallit suoriutuivat luokitteluongelmasta paremmin kuin yksittäinen puumalli, jonka koko oli optimoitu ristiinvalidoimalla. Mallin ennustevirhe oli kaikista suurin 0,315. Ennustevirhe on määritelty laskemalla väärin ennustettujen havaintojen suhteellinen osuus. Kuvan 3 tuloksista voi-

daan nähdä, että ensemble learning -menetelmät parantavat selvästi ennustetarkkuutta. Boosted tree -mallin ennustetarkkuus on 41 prosenttia parempi kuin yksittäisen puun. Puumallien ennustetarkkuudet esitetään kuvassa 3.

4.2 Naiivi Bayes -malli

Sovitetaan Naiivi Bayes -malli aineistoon. Malli suoriutuu luokitteluongelmasta yhtä hyvin kuin random forest -puumalli 0,2 ennustevirheellä.



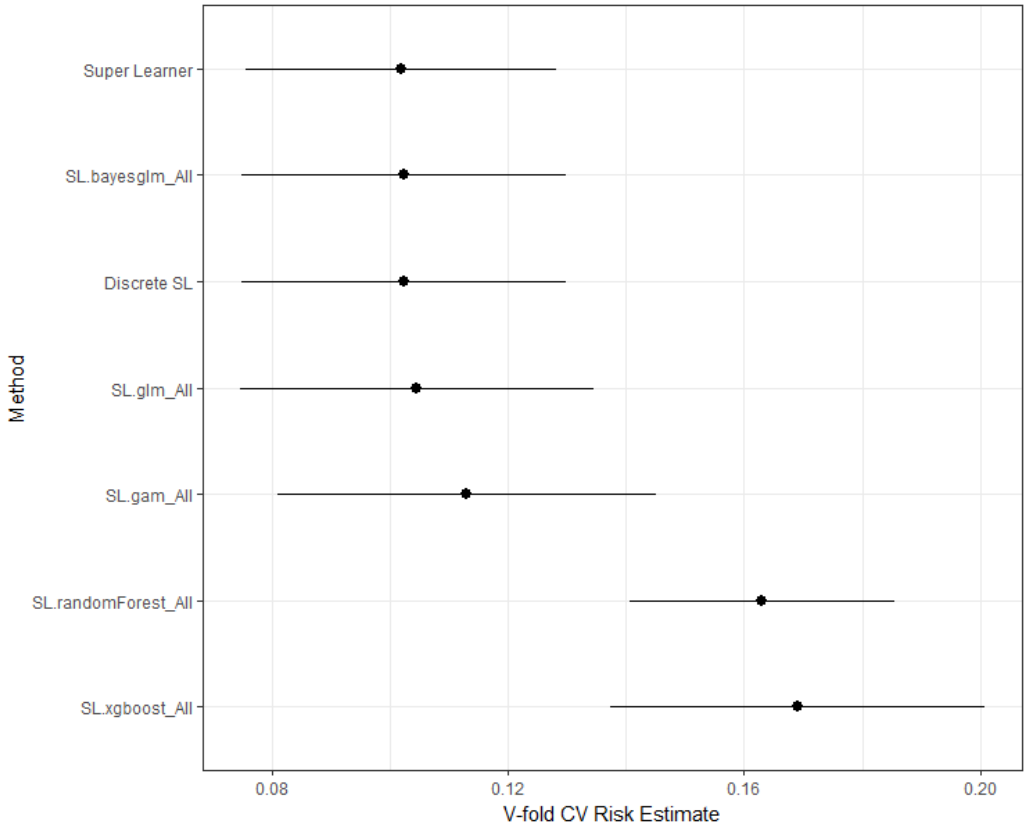
Kuva 4: Pylväsdiagrammi mallien ennustevirheille

Naiivin Bayes -mallin ennustetarkkuus on hyvin lähellä puumallien ennustetarkkuutta. Erot mallien ennustetarkkuuden välillä ovat pieniä. Kuvassa 4 esitetään ennustetarkkuudet Naiivin Bayes -mallin ja puumallien välillä.

4.3 Super Learner

Super Learner on R-ohjelman *CRAN*-paketin sisältämä algoritmi, joka mahdollistaa usean eri mallin yhdistämisen ensemble learning -malliksi. Algoritmi käyttää ristiinvalidointia mallin ennustetarkkuuden arvioimiseen.

Sovitetaan seuraavaksi Super Learner -algoritmi empiiriseen aineistoon. Rakennetaan useasta mallista koottu ennustemalli, joka sisältää mallit, bayesilainen logististen regressio (SLbayes glm All), logistinen regressio (SL glm All), genaralized additive -malli (Sl gam All), random forest (SL randomForest All) ja Xboost algoritmi (Sl xboost All). Xboost on optimoitu gradient boost -puumalli. SuperLearner sisältää hyödyllisen funktion, jolla voidaan estimoida mallin ennustetarkkuutta testiaineistossa ristiinvalidoimalla.



Kuva 5: Ennustevirheet Super Learner -mallille

Super Learner tuottaa tarkemman ennusteen kuin mikään yksittäinen malli. Mallin ennustevirheen estimaatti on 0,1018 ja estimaatin keskivirhe 0,0135. Ennustetarkkuudessa pääse hyvin lähelle bayesilainen logistinen regressio, joka on tarkin yksittäinen luokittelija. Mallin ennustevirhe on 0,1023 ja estimaatin keskivirhe 0,0141. Muiden mallien ennustevirheiden estimaatit ovat selvästi suuremmat.

Algorithm	ave	se	min	max
Super Learner	0.10181	0.013472	0.024365	0.16657
Discrete SL	0.10236	0.014023	0.016686	0.16701
SL.randomForest All	0.16302	0.011450	0.095971	0.24218
SL.glm All	0.10448	0.015321	0.010463	0.17151
SL.gam All	0.11295	0.016375	0.016306	0.19413
SL.xgboost All	0.16911	0.016206	0.105171	0.27409
SL.bayesglm All	0.10236	0.014023	0.016686	0.16701

Taulukko 2: Super Learner- mallin ennustevirheiden estimaatit

Taulukossa 2 esitetään Super Learner -algoritmin ja sen sisältävien mallien ennustevirheiden estimaatit. Estimaatit ovat ristiinvalidointikierroksen keskiarvoja. Lisäksi taulukossa esitetään estimaattien keskivirheet sekä virheiden pienimmät ja suurimmat arvot. Discrete SL kuvaa parhaan yksittäisen mallin, tässä tapauksessa bayesilaisen logististisen regression ennustetarkkuutta.

5 Johtopäätökset

Ensemble learning -menetelmät ovat tehokkaita keinoja parantaa mallin ennustarkkuutta. Mallin suorituskyvyn paraneminen on usein merkittävää, kuten random forest ja boosting-menetelmät näyttivät. Super Learner osoittautui joukon tarkimmaksi ennustemalliksi esimerkkiaineistossa. Mallin ennustetarkkuus on parempi kuin minkään sen sisältämän yksittäisen mallin.

Ensemble learning -menetelmät ovat hyvin käyttökelpoisia myös tilanteissa, joissa käytettävissä on hyvin paljon tai hyvin vähän aineistoa. Menetelmät ovat käteviä myös silloin, kun parhaan mallin valinta on vaikeaa. Tällöin voidaan pienentää huonon mallin valinnan riskiä. Menetelmän heikkoutena voidaan pitää sitä, että monimutkaiset mallit ovat vaikeita tulkita. Tämän vuoksi enseble learning -malleja käytetään yleensä luokitteluun tai ennustamiseen. Monimutkaiset mallit vaativat myös paljon laskentatehoa tietokoneelta, koska ne ovat laskennallisesti raskaita ja hitaita mallintaa.

Lähteet

[1] R. Rokach: *Pattern Classification Using Enseble Methods*. Machine Perception and Artificial Intelligence, Danvers, 2010

[2] R. Polikar: *Ensemble learning*. Scholarpedia, 2009

[3] T. Hastie, R. Tibshirani, J. Friedman: *The Elements of Statistical Learning*. Springer, California, 2017, 2nd edition