AUTOMATISOIDUT KONEOPPIMISMENETELMÄT (AUTOML) – CASE: FORMULA 1 -OSAKILPAILUSIJOITUSTEN ENNUSTAMINEN

Lappeenrannan–Lahden teknillinen yliopisto LUT

Kauppatieteiden kandidaatintutkielma

2023

Renea Laine

Tarkastaja: Tutkijatohtori Jyrki Savolainen

TIIVISTELMÄ

Lappeenrannan–Lahden teknillinen yliopisto LUT

LUT-kauppakorkeakoulu

Kauppatieteet

Renea Laine

Automatisoidut koneoppimismenetelmät (AutoML) – Case: Formula 1 -osakilpailusijoitusten ennustaminen

Kauppatieteiden kandidaatintutkielma

2023

37 sivua, 4 kuviota, 5 taulukkoa ja 5 liitettä

Tarkastaja: Tutkijatohtori Jyrki Savolainen

Avainsanat: koneoppiminen, automatisoitu koneoppimien, Formula 1 -sijoitusten ennustaminen, TPOT

Tässä kandidaatintutkielmassa tutkittiin automatisoitua koneoppimista (AutoML) ja pyrittiin ennustamaan Formula 1 -osakilpailusijoituksia sen avulla. Tutkimuksessa perehdyttiin ensin koneoppimisen ja automatisoidun koneoppimisen teoriaan. Tutkimus toteutettiin kvantitatiivisena tutkimuksena, jossa luotiin ennustusmalli Pythonin AutoML-työkalun, TPOT:n, avulla. Tutkimuksessa hyödynnetty data kerättiin virallisesta Formula 1 -datastreamista.

Tutkimuksen tulokset osoittivat, että ennustusmalli pystyi ennustamaan Formula 1 -osakilpailusijoituksia kohtalaisesti. Tutkimuksessa ennustusmallin tärkeimmät osakilpailusijoitukseen vaikuttavat tekijät yhtyvät aikaisempiin tutkimuksiin. AutoML-ennustusmallin toimivuuden suoriutumismittarit ovat maltillisia, mikä alentaa tuloksien merkittävyyttä. Tutkimus osoittaa kuitenkin AutoML:n potentiaalin sijoituksien ennustamisessa sekä Formula 1:ssä että muussa moottoriurheilussa.

ABSTRACT

Lappeenranta–Lahti University of Technology LUT

LUT Business School

Business Administration

Renea Laine

Automated machine learning (AutoML) – Case: predicting Formula One driver standings

Bachelor's thesis

2023

37 pages, 4 figures, 5 tables and 5 appendices

Examiner: Postdoctoral researcher Jyrki Savolainen

Keywords: machine learning, automated machine learning, predicting Formula One driver standings, TPOT

This bachelor's thesis examined automated machine learning (AutoML) and its possibilities to predict Formula One driver standings. The research commenced by devolving into the theory of machine learning and AutoML. The research was carried out as a quantitative study where a prediction model was created using Pythons AutoML tool, TPOT. The dataset utilized in this research was collected from the official Formula One data stream.

The results of this research demonstrated a moderate predictive capability of the developed model for predicting Formula One driver standings. According to the prediction model the significant factors influencing the driver standings align with the existing research. The performance measures of AutoML prediction model are moderate which lowers the significance of the results. However, the research underscores the potential of AutoML techniques in predicting standings not only in Formula One but also in other motorsport disciplines.

Sisällysluettelo

Tiivistelmä

Abstract

1 Johdanto ... 6
1.1 Tutkimuksen tavoitteet ja tutkimuskysymykset ... 7
1.2 Tutkimusmenetelmät ja -aineisto ... 8
1.3 Tutkimuksen rajaukset ja rakenne ... 8
2 Teoreettinen viitekehys ... 10
2.1 Koneoppiminen ... 10
2.1.1 Ohjattu oppiminen ... 11
2.1.2 Ohjaamaton oppiminen ... 12
2.1.3 Koneoppimisen vaiheet ... 12
2.2 Automatisoitu koneoppiminen ... 13
2.3 Formula 1 -aineiston kuvaus ... 15
2.4 Aikaisempi tutkimus Formula 1 -osakilpailusijoitusten ennustamisesta ... 16
3 Data ja metodologia ... 19
3.1 AutoML työkalut ... 19
3.2 Tree-based Optimization Tool (TPOT) ... 20
3.3 Kuvaus datasetistä ... 23
4 Tutkimuksen toteutus ... 25
4.1 Tutkimusaineiston esikäsittely ... 25
4.2 Mallin sovitus ... 27
5 Tutkimustulokset ... 29
6 Yhteenveto ja johtopäätökset ... 34
6.1 Vastaukset tutkimuskysymyksiin ... 34
6.2 Tutkimuksen luotettavuus ja jatkotutkimus ... 36
Lähteet ... 38
Liitteet ... 42

Liitteet

Liite 1. Bahrainin osakilpailun ennustetut sijoitukset

Liite 2. Espanjan osakilpailun ennustetut sijoitukset

Liite 3. Italian osakilpailun ennustetut sijoitukset

Liite 4. Abu Dhabin osakilpailun ennustetut sijoitukset

Liite 5. Britannian osakilpailun ennustetut sijoitukset

1 Johdanto

Koneoppiminen on kohdannut yhä enemmän haasteita sen nopean leviämisen, datan räjähdysmäisen kasvun sekä datatietelijöiden asiantuntijuuspulan takia. Alalta vaaditaan koko ajan haastavampia innovaatioita pysyäkseen mukana nopeatempoisissa kehitys- ja käyttöönottomallien elinkaarissa. (Yakolev et al. 2020) Datan uuden aikakauden tuomat haasteet ovat luoneet tarpeen määritellä vanhoja ratkaisuja uudelleen. Siispä haasteita on alettu käsittelemään automaattisen koneoppimisen (engl. Automated machine learning) avulla. (Celik, Singh & Vanschoren 2022)

Automatisoidut koneoppimismenetelmät (AutoML) voidaan määritellä automaation ja koneoppimisen yhdistelmäksi. Toisin sanoen, AutoML automatisoi koneoppimisputken rakentamisen rajoitetuilla laskennallisilla resursseilla. (He et al. 2021) AutoML mahdollistaa koneoppimismallien rakentamisen myös sellaisille henkilöille, joilla on vähän tai ei ollenkaan tietotaitoa koneoppimismallien rakentamisesta. (He, Zhao & Chu 2021; Karmaker et al. 2022) AutoML säästää asiantuntijoilta aikaa hoitamalla osan koneoppimisputken rakentamisesta ja mahdollistaa korkeatasoisten mallien rakentamisen huomattavasti tehokkaammin. (Celik, Singh & Vanschoren 2022)

Formula 1 (F1) on yksi maailman suosituimmista moottoriurheilun muodoista. Viime vuosien aikana datan ja teknologian käyttö Formula 1:ssä on noussut yhä tärkeämmäksi ja laji työntääkin jatkuvasti teknisiä rajojaan löytääkseen nopeimman reitin maaliviivalle. (FIA 2023b) Kun otetaan huomioon teknologian ja datan merkitys Formula 1:ssä, ei ole yllättävää, että koneoppimismenetelmät ovat nousseet lupaavaksi työkaluiksi. F1:n datatieteilijät hyödyntävät Amazonin tarjoamia syväoppimismalleja suorituskykytilastojen analysoimisessa tarkkojen kilpailuennusteiden tekemiseksi. (AWS 2018)

Formula 1 -sarjan merkittävyydestä huolimatta tieteellisiä tutkimuksia AutoML-menetelmien hyödyntämisestä osakilpailujen tuloksien ennustamisessa on vaikeasti löydettävissä. Vaikka erilaisia tutkimuksia muiden urheilumuotojen tuloksien ennustamisesta on löydettävissä, niin Hendersonin ja Kirranen (2018) sekä Patil et al. (2023) mukaan lajin suosiosta

huolimatta julkaistuja tilastollisia analyyseja Formula 1:n tuloksista on harvassa. Formula 1:n luonne lukuisine muuttujineen ja tekijöineen, jotka voivat vaikuttaa kilpailun lopputulokseen asettaa ainutlaatuisen haasteen AutoML-algoritmeille. Tästä syystä lisätutkimusta tarvitaan AutoML:n potentiaalin selvittämiseksi Formula 1 -tulosten ennustamisessa.

1.1 Tutkimuksen tavoitteet ja tutkimuskysymykset

Tässä kandidaatintutkielmassa tutkitaan AutoML-menetelmiä ja voidaanko niiden avulla ennustaa Formula 1 -osakilpailujen tuloksia. Tutkimuksessa tarkastellaan Pythonin AutoML-työkalun, TPOT:n, suoriutumista Formula 1 -osakilpailujen sijoitusten ennustamisessa. Tämän tutkielman tavoitteena on tarkastella AutoML:n sovelluksia ja hyödyntää niitä Formula 1 -osakilpailusijoitusten ennustamisessa ja antaa näkemyksiä tämän lähestymistavan mahdollisista eduista ja rajoituksista tutkimuskysymysten avulla. Tutkimuksen yleisenä tavoitteena on myötävaikuttaa kasvavaan urheiluanalytiikan alaan tutkimalla AutoML:n potentiaalia F1-kilpailujen sijoitusten ennustamisessa. Tutkimuksessa tutkimusongelmaan pyritään vastaamaan seuraavien tutkimuskysymysten avulla:

Mitä on AutoML ja millaisia sovelluksia sillä on kirjallisuuden mukaan?

Mitkä ovat Formula 1 -osakilpailusijoitusten ennustamiseen sopivat AutoML-algoritmit ja millaisia tuloksia niiden avulla on saavutettavissa?

Ensimmäisen tutkimuskysymyksen tarkoituksena on keskittyä teoriaan AutoML:n taustalla ja auttaa tunnistaman sen sovituksia. Toisen tutkimuskysymyksen avulla on tarkoitus tunnistaa eri AutoML-algoritmeja sekä ennustaa niiden avulla Formula 1 -osakilpailujen sijoituksia. Tutkimukselle on muodostettu kaksi hypoteesia perustuen aikaisempiin tutkimuksiin. Aikaisempia tutkimuksia käydään tarkemmin läpi tutkielman teoriaosuudessa, kappaleessa 2.4. Tutkimukselle on muodostettu seuraavat hypoteesit.

H1: Lähtösijoitus vaikuttaa suuresti maaliintulojärjestykseen

H2: Tiettyyn talliin kuuluminen vaikuttaa maaliintulojärjestykseen

Hypoteesien tavoitteena on tutkia mahdollisia tekijöitä ja muuttujia, jotka voivat vaikuttaa sijoituksiin Formula 1 -osakilpailussa. Ensimmäinen hypoteesi viittaa siihen, että lähtöpaikalla olisi merkittävä rooli osakilpailusijoituksien määrittämisessä. Toinen hypoteesi

ehdottaa, että tietylle tallille ajaminen voisi vaikuttaa osakilpailusijoitusjärjestykseen. Hypoteesien avulla tutkimus voi saada käsityksen tekijöistä, jotka vaikuttavat F1-kilpailujen tuloksiin.

1.2 Tutkimusmenetelmät ja -aineisto

Kandidaatintutkielma toteutetaan määrällisenä eli kvantitatiivisena tutkimuksena. Tutkimusaineistona käytetään Pythonin FastF1-kirjastoa, joka sisältää historialliset kilpailutiedot kaikista Formula 1 -osakilpailuista vuosilta 1950–2022. Data sisältää erilaisia muuttujia, kuten kierrosajat, kuljettajatiedot, tallitiedot sekä sääolosuhteet. Tutkimuksessa FastF1-aineiston avulla muodostetaan kahdeksan uutta datasettiä kahdeksasta eri Formula 1 -osakilpailusta. Dataa kerätään kaikista osakilpailuihin liittyvistä tapahtumista eli harjoituksista, aikaajoista sekä kilpailusta. Tutkimustyökaluksi, jonka avulla tutkimus toteutetaan, valikoitui Pythonin AutoML-työkalu: TPOT (Tree-based Pipeline Optimization Tool). TPOT:n avulla tutkimuksessa muodostetaan ennustusmalli, jota testataan kaikkiin kahdeksaan eri datasettiin.

Tutkimusaineistossa saattaa olla puutteita, mistä syystä tämä tutkimus ei välttämättä edusta koko Formula 1 -populaatiota, sillä tietojen saatavuus voi rajoittaa otosta. Tutkimuksessa ei pystytä huomioida kaikkia osakilpailusijoitukseen mahdollisesti vaikuttavia tekijöitä, kuten kuljettajan käyttäytymistä tai formuloihin tehtyjä muutoksia. Näistä puutteista huolimatta tämä tutkimus on tärkeä askel kohti AutoML:n potentiaalin ymmärtämistä moottoriurheilun alalla ja tuloksia voidaan käyttää ohjaamaan tulevaa tutkimusta ja kehitystä tällä alalla.

1.3 Tutkimuksen rajaukset ja rakenne

Tutkimuksen empiriaosuudessa käytetään vain yhtä AutoML-työkalua Formula 1 -osakilpailusijoituksien ennustamisessa. Tutkimuksessa tutkitaan osakilpailusijoituksia AutoML:n rakentaman ennustusmallin avulla, jolloin muita ennustamistapoja tuodaan esille vain tarvittaessa. Teoria rajataan pääpiirteittäin siten, että se on yhtenevää tutkimuksen kanssa. Vaikka neuroverkot ovat modernin koneoppimisen kannalta erittäin tärkeässä roolissa (Zhang et al. 2018), rajataan ne pois tutkimuksesta. Rajauksien tarkoituksena on estää tutkimuksen liiallinen laajentuminen.

FastF1-tutkimusaineisto rajataan vastaamaan osakilpailukohtaisia datasettejä. Tutkimuksessa ennustusmallin rakentamisessa hyödynnetään dataa vain vuosilta 2018–2022 ja mallin avulla pyritään ennustamaan vuoden 2022 osakilpailujen toteutuneita sijoituksia 1–10. Ennustusmallin aikaväli on rajoitettu vuosille 2018–2022, sillä aikaisempien vuosien data käyttää eri ohjelmointirajapintaa (engl. Application Programming Interface, API). Tästä syystä aikaisempien vuosien tiedot eivät ole välttämättä yhteensopivia, mikä saattaa vaikuttaa tutkimuksen pätevyyteen. Näin voidaan varmistaa, että käytetty data on johdonmukaista ja yhteensopivaa tutkimuksessa käytetyn ohjelmointirajapinnan kanssa, joka parantaa tutkimuksen luotettavuutta. Formula 1 -osakilpailuissa pisteitä jaetaan vain sijoituksille 1–10 (FIA 2023d). Henderssonin ja Kirranen (2018) tutkimuksen mukaan tulokset voivat olla parempia, kun ennustusmalli katkaistaan tietylle sijoitukselle. Lisäksi vain pistesijojen tulokset ovat ensisijaisesti kiinnostavia, eikä ylimenevien sijoitusten järjestyksellä ole väliä pisteiden tai mestaruuden kannalta. Siten tutkimuksessa on perusteltua keskittyä ennustamaan vain sijoituksia 1–10.

Tutkimus lähtee liikkeelle teoriaosuudesta, jossa syvennytään koneoppimiseen, sen kahteen pääkategoriaan sekä koneoppimisprosessin vaiheisiin. Lisäksi teoriaosuudessa käydään tarkemmin läpi automatisoitua koneoppimista ja selvitetään sen vaikutuksia erilaisten ennustusmallien luomisessa ja hyödyntämisessä. Teoriaosuudessa esitellään myös Formula 1 -case-aineisto ja syvennytään tarkemmin koneoppimisen hyödyntämiseen aikaisemmissa tutkimuksissa liittyen Formula 1 -kisojen ennustamiseen. Näistä kokonaisuuksista muodostuu tutkimuksen teoreettinen viitekehys. Tutkimuksen empiriaosuudessa tarkastellaan erilaisia AutoML-malleja ja syvennytään tarkemmin tutkimuksessa käytettyyn malliin. Mallien esittelyä seuraa datasettien kuvaus, jossa perehdytään tarkemmin tutkimuksessa rakennettuihin datasetteihin. Lopuksi tutkielmassa analysoidaan ennustusmallia ja sen antamia tuloksia ja ennustetarkkuuksia. Johtopäätöksissä sovitetaan tutkimustulokset vastaamaan tutkimuksessa esitettyjä tutkimuskysymyksiä ja hypoteeseja sekä arvioidaan tutkimuksen luotettavuutta ja pohditaan jatkotutkimuksien mahdollisuuksia.

2 Teoreettinen viitekehys

Tutkimuksen teoreettinen viitekehys muodostuu koneoppimisesta ja automatisoidusta koneoppimisesta. Lisäksi viitekehyksessä syvennytään tarkemmin case-aineistoon sekä käydään läpi aiheeseen liittyvää aikaisempaa tutkimusta. Teoriaosuudessa tuodaan ensin esille teoriaa koneoppimisesta ja sen kahdesta pääkategoriasta sekä tarkastellaan koneoppimisen eri vaiheita. Koneoppimisen jälkeen viitekehyksessä siirrytään automatisoituun koneoppimiseen. Lopuksi esitellään Formula 1 -aineisto ja tarkastellaan aikaisempia tutkimuksia sekä niiden tuloksia liittyen Formula 1 -kisojen ennustamiseen.

2.1 Koneoppiminen

Ihmiset ovat olleet jo pitkään ennen tietokoneita kiinnostuneita löytämään erilaisia kaavoja datasta. Digitaalisten tietokoneiden rakentaminen 1900-luvun puolessa välissä mahdollisti data-analyysitekniikoiden automatisoinnin. (Biamonte et al. 2017) Koneoppiminen on eräs tekoälytekniikka, joka opettaa koneita oppimaan kokemuksesta (MathWorks 2023c). Koneoppiminen on ollut olemassa jo 1970-lvuvulla, kun ensimmäiset algoritmit kehitettiin. Suurimmat muutokset alalla ovat tapahtuneet laskentatehossa, mikä on mahdollistanut entistä monimutkaisempien ongelmien ratkaisemisen koneoppimisen avulla. (Louridas & Ebert 2016)

Koneoppimismenetelmien avulla tietokoneet oppivat suorittamaan erilaisia tehtäviä tutkimalla aikaisempia tietoaineistoja. Koneoppimista käytetään tunnistamaan erilaisia kaavoja ja luomaan sääntöjä data-analytiikkaa ja datan käsittelyä varten. (Louridas & Ebert 2016) Koneoppimisalgoritmit käyttävät laskennallisia menetelmiä tiedon oppimiseen nojautumatta määrättyihin yhtälöihin. (MathWorks 2023c) Perusideana on tapauspohjainen päättely, joka on ongelmanratkaisuprosessi viittaamalla vastaaviin aikaisempiin tapauksiin. Aikaisempia tapauksia pidetään harjoitusesimerkkeinä ja ongelman ratkaisuprosessia harjoituksien perusteella kutsutaan oppimiseksi. (Jo 2021, 3)

Koneoppiminen voidaan jakaa kahteen pääkategoriaan: ohjattuun oppimiseen (engl. Supervised learning) ja ohjaamattomaan oppimiseen (engl. Unsupervised learning). Ohjatussa

oppimisessa koneoppimisalgoritmi rakentaa mallin, joka tekee ennusteita syötedatasta ja datan tuloksista ja kouluttaa mallin luomaan järkeviä ennusteita uudelle datalle. Ohjaamaton oppiminen löytää piilossa olevia kaavoja tai toistuvuuksia datasta ilman valmiita vastauksia. (MathWorks 2023c)

2.1.1 Ohjattu oppiminen

Ohjattua oppimista pidetään usein ensimmäisenä koneoppimisen tyyppinä. Ohjatussa oppimisessa oletetaan, että kaikki aikaisemmat esimerkit eli harjoitusdata on merkitty. Ohjatussa oppimisessa siis kehitetään kykyä ratkaista ongelmat, jotka perustuvat aikaisempaan dataa eli harjoitusdataan. (Jo 2021, 3) Ohjattu oppiminen käyttää harjoitusdataa algoritmin kouluttamiseen. Harjoitusdata koostuu syötedatasta ja niitä vastaavista tunnetuista vastauksista. Tavoitteena on käyttää tätä merkittyä dataa opettamaan algoritmi tunnistamaan kuvioita ja tekemään ennusteita täysin uudesta datasta. (MathWorks 2023a) Ohjatun oppimisen algoritmi mittaa koneoppimismallin tarkkuutta, kunnes sen virhemarginaali minimoituu. (IBM 2023a)

Ohjattu oppiminen voidaan jakaa kahteen luokkaan: luokittelu (engl. Classification) ja regressio. Luokittelu käyttää algoritmia, joka määrittää datan tarkasti tiettyihin luokkiin. Se tunnistaa tietyt kokonaisuudet datasta ja pyrkii tekemään johtopäätöksiä siitä, kuinka eri kokonaisuudet tulisi määritellä. (IBM 2023a) Yleisimmät luokittelu algoritmit ovat: logistinen regressio, tukivektorikoneet (engl. Support-vector machines, SVM), neuroverkot (engl. Neural networks), päätöspuut (engl. Decision trees), lähimmän naapurin menetelmä (engl. Nearest neighbor) ja satunnainen metsä (engl. Random forest). (IBM 2023a; MathWorks 2023a) Regressiota käytetään tutkimaan riippuvien ja riippumattomien muuttujien välistä suhdetta sekä tekemään ennusteita muuttujien avulla. (IBM 2023a) Yleisimmin käytetyt regressioalgoritmit pitävät sisällään lineaarisen regression, epälineaarisen regression, päätöspuut sekä neuroverkot. (IBM 2023a; MathWorks 2023a)

Tässä tutkimuksessa koneoppimismalli muodostetaan valvotun oppimisen avulla. Valvottu oppiminen on hyödyllinen lähestymistapa tässä tutkimuksessa, koska sen avulla voidaan käyttää olemassa olevaa dataa aikaisemmilta vuosilta mallin kouluttamiseen. Analysoimalla eri ominaisuuksien ja kilpailutulosten välisiä suhteita malli oppii tunnistamaan kaavoja ja tekemään ennusteita niiden perusteella.

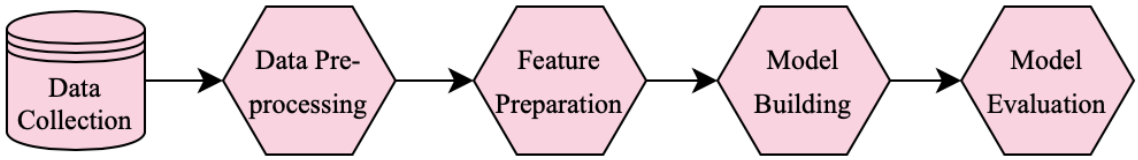
2.1.2 Ohjaamaton oppiminen

Ohjaamaton oppiminen on koneoppimisen tyyppi, jossa koulutustiedoissa ei ole tunnettuja vastauksia. Sen sijaan algoritmi yrittää löytää kaavoja ja erilaisia suhteita datassa käyttämällä samankaltaisuusmetriikkaa. (Jo 2021, 3) Algoritmin pyrkimys löytää piilotettuja kuvioita tai tietojoukkoja tapahtuu täysin ilman ihmisen väliintuloa (IBM 2023b; MathWorks 2023b).

Klusterointi on yleisimmin käytetty ohjaamaton oppimistekniikka. Klusteroinnissa klusterointimentelmiä sovelletaan datasetteihin piilotettujen kuvioiden tai ryhmittymien löytämiseksi. (MathWorks 2023b) Prosessi voi siis auttaa tunnistamaan muuttujien välisiä suhteita ja ryhmittämään samankaltaista dataa yhteen ilman olemassa olevia tunnisteita. IBM:n (2023b) mukaan klusterointi voidaan kategorisoida neljään päätyyppiin: eksklusiivinen, päällekkäinen, hierarkkinen ja todennäköisyyspohjainen klusterointi.

2.1.3 Koneoppimisen vaiheet

Tyypillisen koneoppimisprosessin vaiheet (engl. Machine learning pipeline) koostuvat viidestä eri vaiheesta (kuvio 1). Koneoppimisputken tarkoituksena on auttaa koneoppimisen työnkulun automatisoinnissa, jotta suurista datajoukoista saadaan käyttökelpoisia erilaisiin tehtäviin. Tavoitteena on siis kouluttaa tarkka malli ratkaisemaan taustalla oleva ongelma (Rafatirad, Homayoun, Chen, Dinakarrao 2022, 5).



Kuvio 1: Koneoppimisen vaiheet (mukaillen Olson & Moore 2016)

Ennen minkäänlaisien analyysien tekoa, kerätty data täytyy esikäsitellä eli valmistella. Kerätty data saattaa sisältää virheitä, poikkeamia (engl. outliers) ja puuttuvia arvoja. Esikäsittelyvaiheessa (engl. Data Preprocessing) nämä epäjohdonmukaisuudet korjataan siivoamalla dataa. Sen lisäksi data voi myös vaatia normalisointia tai standardointia ja sen muokkaamista

(Rafatirad et al. 2022, 5–6) Esikäsittelyn läpikäynyt data on puhdasta, ja sillä voidaan jatkaa koneoppimisprosessin läpikäyntiä.

Ominaisuuksien valmistelu -vaiheeseen (engl. Feature Preparation) kuuluvat ominaisuussuunnittelu ja ominaisuuksien valinta (engl. Feature engineering, feature selection). Vaiheessa valitaan tärkeät ominaisuudet koneoppimismallin koulutukseen datan avulla. Olemassa olevia ominaisuuksia voidaan muokata tai luoda uusia, jotta ne sopisivat paremmin mallin kouluttamiseen. Ominaisuuksien valinnassa korostuu ylisovituksen (engl. over-fitting) vähentäminen, mallin tarkkuuden parantaminen sekä harjoitusajan lyhentäminen. (Rafatirad et al. 2022, 5–6) Vaihe on siten ratkaiseva mallin suorituskyvyn parantamiseksi.

Mallin rakentamisvaiheessa (engl. Model Building) valmisteltu data ositetaan harjoitus- ja testijoukoiksi. Harjoitusdataa käytetään kouluttamaan koneoppimisalgoritmi koneoppimismallin rakentamiseksi. (Rafatirad et al. 2022, 5–6) Mallin rakentamisvaiheessa valitaan mallille sopiva koneoppimisalgoritmi. Erilaisia koneoppimisalgoritmeja on laaja valikoima, mukaan lukien lineaarinen regressio, päätöspuut ja tukivektorikoneet. Rakentamisvaiheeseen kuuluu myös hyperparametrien optimointi, jossa valitun koneoppimisalgoritmin parametrit viritetään parhaan suorituskyvyn saavuttamiseksi. Optimointiin on useita menetelmiä kuten ruudukkohaku, satunnaishaku ja Bayesilainen optimointi. (Raschka & Mirajalili 2017)

Mallin arviointi (engl. Model Evaluation) viittaa prosessiin, jossa arvioidaan koulutetun mallin suorituskykyä uudella, ennenäkemättömällä datalla. Erilaiset arviointitekniikat perustuvat valittuun malliin. Esimerkiksi regressiomallin arviointitekniikat lukevat mukaansa keskineliövirheen ja selitysasteen (engl. mean squared error, R-squared). (Raschka & Mirajalili 2017) Kun malli on testattu toimivaksi, voidaan sitä käyttää ennusteiden tekemiseen uusista dataseteistä.

2.2 Automatisoitu koneoppiminen

Pohjimmiltaan jokaisen tehokkaan koneoppimismallin on ratkaistava perusongelmat liittyen koneoppimisalgoritmin valintaan tietyssä datasetissä, oli kyseessä sitten ominaisuuksien esikäsittely tai hyperparametrien asettelu. (Feurer et al. 2015) Tehokkaiden koneoppimismallien luominen ja suunnittelu on usein työlästä ja vaatii huomattavaa kokemusta koneoppimisalgoritmeista. Del Bimbon (2021) mukaan koneoppimisputken vaiheista monimutkaisin

on ominaissuunnittelu. Datatieteilijät pitävät vaihetta ajallisesti pisimpänä koko koneoppimismallin rakentamisesta vaatien erityisosaamista. Koneoppimissovelluksien kehittämistarpeiden kasvu suhteessa datatieteilijöiden määrän kasvuun tekee manuaalisten koneoppimismallien luomisen mahdottomaksi (Olson & Moore 2016). AutoML:n ensisijainen tavoite on vähentää koneoppimistekniikoiden vaatimaa manuaalista työtä automatisoimalla tiettyjä koneoppimisvaiheita (Karmaker 2022)

AutoML-menetelmät määritetään usein automaation ja koneoppimisen yhdistelmäksi. Käytännössä AutoML hoitaa manuaalisen koneoppimismallin luomisen huomattavasti pienemmällä määrällä resursseja. (He et al. 2021) Se pyrkii automatisoimaan käytetystä algoritmista riippuen ainakin ominaisuuksien valmistelun ja mallin rakentamisen. (Das & Cakmak 2018) Valmis automatisoitu koneoppimisjärjestelmä rakentaa dynaamisen kombinaation eri algoritmitekniikoita muodostaakseen käyttäjäystävällisen päästä-päähän koneoppimisputken. (He et al. 2021)

Koneoppimisprosessin automatisoinnissa voidaan käyttää lukuisia erilaisia AutoML-tekniikoita (Romano et al. 2021). On olemassa useita erilaisia menetelmiä, jotka tarjoavat lähestymistavan koneoppimisprosessin automatisointiin. Tässä tutkimuksessa käydään lyhyesti läpi niistä yleisimmät: ruudukkohaku, bayesilainen optimointi, metaoppiminen ja geneettinen ohjelmointi.

Ruudukkohaku (engl. Grid search) on yleisimmin käytätetty hyperparatmetrien optimoinnin muoto, jossa käytetään raakavoimaa (engl. Brute force) (Olson et al. 2016). Raakavoiman käytön tarkoituksena on tutkia laajaa valikoimaa parametrikombinaatioita, jotta voidaan löytää se hyperparametrijoukko, jonka avulla koneoppimismalli toimii parhaiten (Priyadarshini & Cotton 2021).

Bayesilainen optimointi (engl. Bayesian optimization) on tekniikka, jota käytetään optimoimaan black-box -funktiota (Luong et al. 2021). Se on iteratiivinen algoritmi, jossa on kaksi keskinäistä osaa: todennäköisyysmalli ja Acquisition-funktio. Todennäköisyysmallin toiminta perustuu ennustamiseen arvioimalla aikaisempaa dataa. Malli kehittyy sitä mukaa kun arvioita tehdään. (Hutter et al. 2019) Acquisition-funktion tarkoituksena on määrittää, mitä vaihtoehtoa mallin tulisi arvioida seuraavaksi. Funktio tasapainottaa tarvetta tutkia uusia vaihtoehtoja ja tarvetta hyödyntää aiemmin hyvin toimineita vaihtoehtoja. (Luong et al. 2021)

Metaoppimisessa (engl. Meta-learning) hyödynnetään aikaisempia oppimiskokemuksia tulevaisuuden oppimisen tehostamiseksi. Metaoppimisessa kerätään metatietoja, jotka kuvaavat aikaisempia oppimistehtäviä ja aiemmin opittuja malleja. (Olson et al. 2016) Mallit sisältävät erilaista dataa esimerkiksi tarkkoja algoritmikonfiguraatioita, joita on käytetty mallin kouluttamiseen sekä mallin arviontien tuloksia. Metaoppimistekniikat voidaan luokitella sen mukaan, minkä tyyppistä tietoa ne hyödyntävät. Aikaisemmista metatiedoista voidaan poimia ja siirtää tietoa, joka ohjaa optimaalisten mallien etsimistä. (Hutter et al. 2019)

Geneettinen ohjelmointi (engl. Genetic programming) perustuu matemaattisiin funktiopuihin, jotka on optimoitu suhteessa johonkin mittariin kuten luokittelutarkkuuteen (Romano et al. 2021). Optimointiprosessiin kuuluu uusien puusukupolvien luominen ja kehittäminen tekemällä satunnaisia mutaatioita joko puun rakenteeseen tai jokaisessa solmussa suoritettaviin toimintoihin (Wan, Yu & Guo 2019). Prosessi jatkuu useiden sukupolvien ajan, kunnes syntyy optimaalinen puu (Romano et al. 2021).

AutoML on noussut lupaavaksi ratkaisuksi koneoppimisen popularisoimiseksi ja koneoppimisputken muodostamisen sekä käytön helpottamiseksi ei-asiantuntijoille. Vaikka AutoML on vielä kehittyvä ala, on sillä suuri potentiaali vähentää tehokkaiden koneoppimismallien rakentamiseen tarvittavaa aikaa ja asiantuntemusta, mikä nopeuttaa edistymistä monissa eri sovelluksissa.

2.3 Formula 1 -aineiston kuvaus

Formula 1 -maailmanmestaruussarjan järjestää voittoa tavoittelematon kansainvälinen autoliitto FIA (Féderation Internationale de l'Automobile) (FIA 2023c). Maailman arvostetuimpana moottoriurheilukilpailuna tunnettua Formula 1 -maailmanmestaruussarjaa on järjestetty vuodesta 1950 alkaen. Siitä lähtien lajissa on tapahtunut merkittäviä innovaatioita, ja sitä pidetäänkin teknologian edelläkävijänä autotekniikan kehityksessä. (FIA 2023b)

Formula 1 -kisakausi muodostuu sarjasta osakilpailuja (engl. Grand Prix), jotka järjestetään eri puolella maailmaa aina maaliskuusta marraskuuhun (Gutiérrez & Lozano 2020). Maailmanmestaruudesta Formula 1:ssä kilpaillaan sekä kuljettajien että valmistajien kesken. Formula 1 maailmanmestaruus -titteli palkitaan kuljettajalle, joka on saanut eniten pisteitä kisakauden aikana. Kuten myös kuljettajan titteli, Formula 1 -valmistajien maailmanmestaruus

palkitaan eniten pisteitä saaneelle tallille. Valmistajien maailmanmestaruuspisteissä otetaan huomioon tallien molempien kuljettajien keräämät pisteet. (FIA 2023d)

Formula 1 -osakilpailut koostuvat viidestä eri tapahtumakokonaisuudesta: kisasta, aika-ajoista sekä harjoituksista. Osakilpailukokonaisuuteen kuuluu yhteensä kolme eri harjoitusta. FIA:n säännösten mukaan jokaisen kuljettajan tulee osallistua aika-ajoihin, joista nopein kuljettaja palkitaan osakilpailun lähtösijoituksella yksi (”paalupaikka”). (FIA 2023d)

Aikaisempien tutkimusten mukaan Formula 1 -osakilpailusijoituksiin vaikuttavat monet tekijät: lähtöjärjestys, auton ominaisuudet, kuljettajan taidot, ratastrategiat sekä monet muut (Gutiérrez, Lozano 2020). Formula 1 -osakilpailuissa ei kuitenkaan ole pelkästään kysymys nopeimmasta autosta, vaan tasapainosta teknologian, pääoman ja henkilöresurssien välillä. Vuosien saatossa teknologian ja sen kehittämisen merkitys on noussut kokonaan uudelle tasolle. (Jenkins 2010) Formula 1 on monimutkainen urheilulaji, joka sisältää yhdistelmän useita erilaisia tekijöitä. Aiempien osakilpailujen ja tapahtumien analysoinnin avulla voidaan tunnistaa malleja ja trendejä, joita voidaan käyttää tulevien osakilpailujen ennustamiseen.

2.4 Aikaisempi tutkimus Formula 1 -osakilpailusijoitusten ennustamisesta

Formula 1 -aika-ajoissa nopein kuljettaja palkitaan parhaalla lähtöpaikalla. (FIA 2023d) Palkitsemismenettelyn takia voidaan olettaa, että kuljettaja hyötyisi tästä lähtösijoituksesta. Muehlbauer (2017) tutki suhdetta lähtösijoituksen ja maaliintulosijoituksen välillä 70 eri Formula 1 -kisassa neljän vuoden aikaperiodilla. Tutkimus toteutettiin analysoimalla korrelaatiota sijoituksien 1–8 välillä. Mahdollisuudet kisan voittoon kasvoivat suuresti, mikäli kuljettaja lähtee kisaan sijalta yksi (engl. Pole position). Tarkemmin todennäköisyys kisan voittoon oli 59,9 % lähtösijoitukselta yksi. Kuitenkin lähempänä lähtösijoituksia seitsemän ja kahdeksan todennäköisyys tippui lähelle 0 %. Lisäksi pääsy palkintosijoille (engl. Podium) oli todennäköisintä, mikäli lähtösijoitus oli jokin kolmesta ensimmäisestä. (Muehlbauer 2017) Tutkimus siis indikoi, että parempi lähtösijoitus voi vaikuttaa merkittävästi kisan lopputulokseen.

Henderson ja Kirrane (2018) vertailivat tutkimuksessaan neljää varianttia Plackett-Luce-mallista testatakseen niiden suorituskykyä ennustaa Formula 1 -kisojen tuloksia. Plackett-

Luce-mallia käytetään tilastolliseen mallintamiseen luokitteludataa (Schäfer & Hüllermeier 2018) Mallien avulla pyrittiin vertailemaan Formula 1 -sijoituksia yhteensä 77:lle eri kilpailuille vuosilta 2010–2013. Tutkimuksessa ennustettiin mallien avulla voittajaa, palkintosijoja ja pistesijoja sekä maailmanmestaruuden voittajaa ja kilpailusijoitusjärjestyksiä. Tutkimuksessa kävi ilmi, että parempia tuloksia voidaan saada aikaan eri malleilla kiinnostavien ominaisuuksien osalta, kuten osakilpailun tai mestaruuden voittaja, kun malli katkaistaan sijalle r. Tutkimuksessa osoitettiin myös, että aikaisempien tulosten painottaminen voi parantaa mallien ennusteita. (Hendersson & Kirrane 2018) Tutkimuksessa pääpaino oli eri Plackett-Luce -mallien vertaileminen eikä tulosten analysoiminen. Kutenkin tutkimuksen lopputuloksesta voidaan nostaa esille mallin ennustusotoksen rajaamisen merkitys tuloksien tarkkuudessa.

Sicoien (2022) tekemässä tutkimuksessa tavoitteena oli tutkia koneoppimisen kykyä ennustaa Formula 1 -maailmanmestaruutta sekä osakilpailuvoittoja. Tutkimuksessa hyödynnettiin kolmea eri ohjatun oppimisen koneoppimisalgoritmia: Satunnaisen metsän regressio, Garduent Boosting -regressio sekä Tukivektoriregressio. Tutkimuksessa havaittiin, että dominoiva ominaisuus oli tiettyyn talliin kuuluminen. Kuten Muehlbaurer (2017) tutkimuksessaan havaitsi parhaan lähtösijoituksen merkittävyyden liittyen loppusijoitukseen, myös Sicoien (2022) tutkimus osoitti lähtösijoituksen olevan ominaisuutena olennainen. Lisäksi merkittäväksi ominaisuudeksi nousu kuljettajan ikä. Vaikka alkutulokset eivät olleet tyydyttäviä, kun tulokset laskettiin koko kauden ajalta, osoittivat ne korkean korrelaation todellisten sijoitusten kanssa. (Sicoie 2022) Tutkimuksen viitekehyksen havaittiin siis ennustavan Formula 1 -mestaruussijoituksia huomattavissa määrin. Lisäksi tutkimus nostaa esille Formula 1 -tallien keskeisen roolin osakilpailuissa suoriutumisessa.

Myös van Kesterenin ja Bergkampin (2022) tutkimuksessa havaittiin Formula 1 -tallien merkittävyys menestystekijänä osakilpailuissa. Tutkimuksessa analysoitiin Formula 1 -osakilpailujen tuloksia Bayesilaisella monitasoisella Beta-regressio metodilla, jolla mallinnettiin hybridiaikakauden (2014–2021) osakilpailuissa voittaneiden kuljettajien osuutta. Tutkimuksessa pääpaino oli erityisesti kuljettajien taidoilla ja tallin tuottamalla edulla. Tutkimuksen perusteella keskimäärin sekä talli että formula-auto ovat kuljettajaa tärkeämpi kilpailutuloksessa. Tutkimuksesta selvisi, että kolme parasta tallia olivat selvästi parempia kuin muut tallit hybridiaikakaudella. (van Kesteren & Bergkamp 2022) Vuonna 2021 Formula 1:ssä talleille tuli voimaan budjettikatto, jonka tarkoituksena on pienentää kilpailijoiden

suorituskykyjen välisiä eroja (FIA 2023a). Budjettikaton takia voidaankin olettaa, että tallit vaikuttavat osakilpailuissa suoriutumiseen.

Keertish ja Preethi (2023) tutkivat ristiinluokiteltuja monitasomalleja (engl. Cross-classified multilevel models) ymmärtääkseen paremmin tekijöitä, jotka vaikuttavat Formula 1-kilpailun suorituskykyyn. Tutkimuksen tarkoituksena oli kehittää menetelmä, joka käyttää koneoppimistekniikoita ennustamaan kuljettajan menestystä Formula 1 -sarjassa kuljettajan urapolun perusteella. Tutkimuksessa käytettiin valmista datasettiä, johon testattiin kuutta eri koneoppimismallia. Tutkimuksessa mallien tarkkuudet kuljettajien ja tallien suorituksille olivat kaikki 90 % molemmin puolin. Tutkimus antoi myös osviittaa hyvistä ristiinvalidointituloksista, jotka olivat jokaisella mallilla yli 0,85. (Keertish & Preethi 2023)

Aikaisempien tutkimuksien perusteella yksi Formula 1 -osakilpailusijoituksiin vaikuttava tekijä on osakilpailun lähtöjärjestys. Toinen merkittävä tekijä, joka nostettiin tutkimuksissa esille, on tallien merkittävyys koko Formula 1 -sarjassa. Siten aikaisempien tutkimuksien tulosten ja löytöjen perusteella tälle tutkielmalle muodostettiin hypoteesit H1 ja H2.

Formula 1 -sarjan merkittävyydestä huolimatta tieteellisiä tutkimuksia, joissa hyödynnetään AutoML-menetelmiä osakilpailujen tulosten ennustamisessa, on vaikeasti löydettävissä. Useissa löydetyissä tutkimuksissa saatuja tuloksia ja niihin vaikuttavia tekijöitä ei tarkasteltu näkökulmista, jotka olisivat olleet oleellisessa asemassa tämän tutkimuksen kannalta. Henderssonin ja Kirranen (2018) sekä Patil et al. (2023) mukaan tutkimuksia Formula 1 -osakilpailujen tulosten ennustamisesta on hyvin rajallinen määrä, vaikka muiden urheilulajien tulosten ennustamisesta tieteellisiä tutkimuksia on laajalti tarjolla. Tästä syystä on tärkeää selvittää AutoML:n potentiaalia Formula 1 -sarjan että muiden merkittävien moottoriurheilulajien ennusteiden tekemisessä

3 Data ja metodologia

Tässä luvussa esitellään yleisesti käytettyjä AutoML-työkaluja ja perehdytään tarkemmin siihen työkaluun, joka valitaan tutkimuksessa ennustusmallin rakentamiseen. Luvussa esitellään myös tutkimuksessa käytetty aineisto, johon kuuluu kahdeksan eri datasettiä, jotka on muodostettu Formula 1 -osakilpailuista.

3.1 AutoML työkalut

Pythonille on saatavilla useita suosittuja AutoML-kirjastoja. Kirjastot tarjoavat tehokkaita työkaluja koneoppimisen eri vaiheiden automatisointiin datan esikäsittelystä aina hyperparametrien optimointiin asti. Dasin ja Cakmakin (2018) mukaan datatieteiden yhteisössä yleisesti käytettyjä Python -kirjastoja ovat muun muassa: Featuretools, Auto-sklearn, MLBox ja TPOT. Kirjastot esitellään ensin lyhesti ja ne ovat koottu tiivistetysti taulukkoon 1.

Avoimen lähdekoodin Python-kirjasto Featuretools toimii parhaiten koneoppimisen vaiheista ominaisuussuunnittelun automatisoinnissa, kun kyseessä on relaatio- tai tapahtumadata. (Das & Cakmak 2018) Featuretools -kirjaston taustalla oleva ominaisuuksien luomisalgoritmi Deep Feature Synthesis (DFS) tuottaa monimutkaisia pinottuja ominaisuuksia, jotka ovat usein erittäin ennakoivia koneoppimismalleissa. (Badithe 2022)

Auto-sklearn on oivallinen ohjelmointirajapinta koneoppimismallien kehittämiseen. (Das, Cakmak 2018) Auto-Sklearn perustuu suosittuun Pythonin scikit-learn-kirjastoon. (Waring, Lindvall, Umeton 2020) Kirjasto käyttää bayesilaista optimointia löytääkseen ideaalin kombinaation ominaisuuksia ja hyperparametrejä maksimoidakseen luokittelutarkkuuden. (Olson et al. 2016)

MLBox tukee hajautettua tietojenkäsittelyä ja huippuluokan algoritmeja kuten LightGBM. Kirjasto tukee myös mallien pinoamista (engl. Model stacking), jonka avulla voidaan yhdistää tietojoukot (engl. Information ensemble of models) luodakseen uuden mallin, jonka tavoitteena on parempi suorituskyky. (Das & Cakmak 2018)

TPOT (Tree-Based Pipeline Optimization Tool), käyttää geneettistä ohjelmointia löytääkseen parhaiten toimivia koneoppimisprosesseja. Kirjasto on rakennettu scikit-learnin päälle,

mikä tekee siitä erittäin käyttäjäystävällisen. (Das & Cakmak 2018) TPOT:n ominaisuuksiin kuuluu myös hyperparametrien optimointi, jonka AutoML-algoritmi suorittaa datan koulutuksen aikana. (Romano et al. 2021)

Taulukko 1: AutoML kirjastot tiivistettynä

Kirjasto	**Kehittäjä**	**AutoML tyyppi**	**Avainominaisuudet**
Featuretools	Feature Labs (2017)	Avoin lähde-koodi	automatisoitu ominais-suunnittelu, aikasarjadata
Auto-Sklearn	Feurer et al. (2015)	Avoin lähde-koodi	ominaissuunnittelu, hyper-parametrien viritys
MLBox	Aronio De Rom-blay (2017)	Avoin lähde-koodi	mallien pinoaminen, omi-naisuuksien valinta
TPOT	Olson & Moore (2016)	Avoin lähde-koodi	geneettinen ohjelmointi, hyperparametrioptimointi

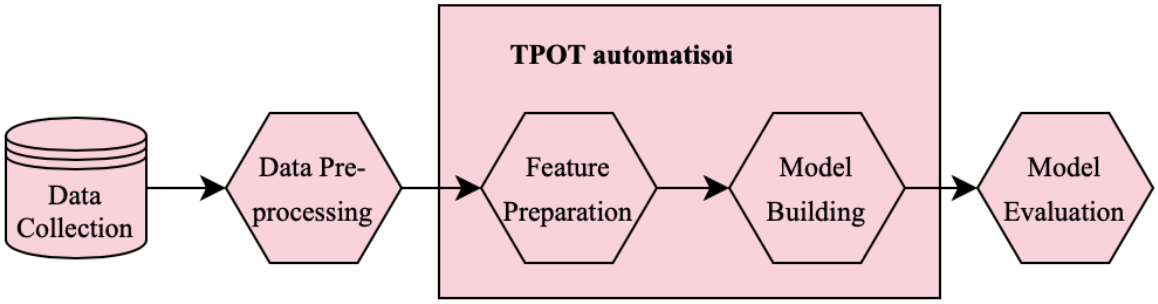
AutoML-kirjastot pyrkivät automatisoimaan koneoppimisprosessin, mikä tekee laadukkaiden mallien rakentamisesta helpompaa ja nopeampaa. Kirjastot tarjoavat erilaisia ratkaisuja, joiden avulla voidaan säästää huomattavasti aikaa alan ammattilaisilta. Kätevyydestään huolimatta AutoML-kirjastoilla on myös joitakin rajoituksia ja mahdollisia haittoja, jotka tulee ottaa huomioon niitä käytettäessä. Kaiken kaikkiaan AutoML-kirjastot tarjoavat lupaavan ratkaisun koneoppimisprosessin automatisoinnissa ja niillä on potentiaalia parantaa koneoppimisen tehokkuutta merkittävästi.

3.2 Tree-based Optimization Tool (TPOT)

Tässä tutkimuksessa valittiin yksi automatisoitukoneoppimistyökalu empiriaosuuden ennustusmallin luomiseen. Työkaluksi valittiin TPOT perustuen sen käyttäjäystävälliseen käyttöliittymään (Das & Cakmak 2018) sekä sen tasapainoon korkean suorituskyvyn ja mallin yksinkertaisuuden välillä (Romano et al. 2021).

Tree-based Pipeline Optimization Tool (lyh. TPOT) on avoimen lähdekoodin AutoML algoritmityökalu, joka on rakennettu Pythonin scikit-learn-kirjaston päälle. TPOT käyttää geneettistä ohjelmointia suunnitellakseen ja optimoidakseen automaattisesti sarjan

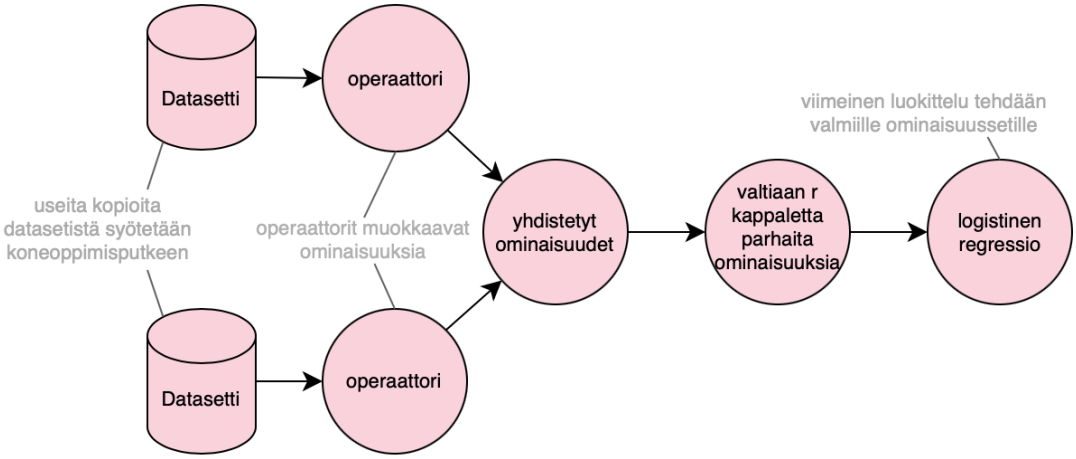
datamuunnoksia ja koneoppimismalleja, jotka pyrkivät maksimoimaan luokittelutarkkuuden annetulle ohjatulle datajoukolle. (Olson et al. 2016) Sen tarkoituksena on hoitaa koneoppimisen vaiheista ominaissuunnittelu, mallin valinta ja hyperparametrioptimointi käsittelyssä olevalle ongelmalle (Waring, Lindvall, Umeton 2020). TPOT:a voidaan käyttää sekä luokittelu- että regressiotehtäviin (TPOT 2023). TPOT:n osuutta koneoppimisprosessin vaiheissa on hahmoteltu kuviossa 2.



Kuvio 2: TPOT koneoppimisprosessissa (mukaillen Olson & Moore 2016)

TPOT:ssa jokainen koneoppimisputken operaattori vastaa koneoppimisalgoritmia, joita ovat ohjatut luokitteluoperaattorit (engl. Supervised Classification Operators), ominaisuuksien esikäsittelyoperaattorit (engl. Feature Preprocessing Operators) ja ominaisuuksien valintaoperaattorit (engl. Feature Selection Operators). Ohjatut luokitteluoperaattorit varastoivat luokittelijan ennustukset uusina ominaisuuksina. Esikäsittelyoperaattorit muokkaavat datasettiä ja palauttavat sen muokattuna. Valintaoperaattorit vähentävät ominaisuuksien määrää datassa käyttäen tiettyjä kriteereitä. Yhdistämällä nämä operaattorit koneoppimisputkeksi, rakennetaan niistä geneettisen ohjelmoinnin avulla puita. (Olson & Moore 2016) Puissa jokainen solmu (engl. Node) edustaa operaattoria, joka suorittaa valitut toiminnot. (Romano et al. 2021)

Kuviossa 3 esitellään tarkemmin puiden avulla luotu koneoppimisputki. Kuviossa kaksi samaa datasettiä toimitetaan putkeen, joita jokainen operaattori muokkaa, yhdistetään yhdeksi datasetiksi ja joita lopulta käytetään luokitteluihin. Operaattorit vastaanottavat datasetit syötteinä ja palauttavat ulostulona muokatun datasetin, joka mahdollistaa mielivaltaisten koneoppimisputkien rakentamisen. (Olson & Moore 2016)



Kuvio 3: Esimerkki koneoppimisputken luomisesta TPOT:lla (Olson & Moore 2016)

Harjoitteluprosessin aikana, TPOT etsii suuresta määrästä mahdollisia koneoppimisputkia sen koneoppimisputken, joka tuottaa parhaat tulokset annetulle datajoukolle. Jokainen TPOT:n valitsema koneoppimisputki on esitetty puuna, ja TPOT kehittää puita useiden sukupolvien ajan valitsemalla parhaiten suoriutuvia yksilöitä ja yhdistämällä niitä. (Romano et al. 2021) Jokaisessa sukupolvessa algoritmi valitsee tietyn määrän koneoppimisputkia maksimoidakseen luokittelutarkkuuden datasetissä samalla minimoiden operaattorien määrää (Olson & Moore 2016). Tuloksena koneoppimisputket koostuvat siis suhteellisen pienistä määristä operaattoreita, jotka voivat silti täyttää ja ylittää kilpailevien koneoppimislähestymistapojen suorituskyvyn (Romano et al. 2021).

TPOT:n geneettisen ohjelmoinnin algoritmi luo tietyn määrän satunnaisia koneoppimisputkia ja arvioi niiden tasapainotettua ristiinvalidointitarkkuutta. (engl. Cross-validation accuracy). Ristiinvalidoinnilla arvioidaan kyseisien koneoppimisputkien suoriutumista. (Olson & Moore 2016) Koneoppimisputken suoriutumista voidaan mitata erilaisilla arviointimittareilla. Luokittelualgoritmi käyttää oletusarviointimittarina tarkkuutta, kun taas oletuksena regressiossa on keskimääräinen neliövirhe (engl. Mean Squared Error, MSE) (TPOT 2023).

K-ositus ristiinvalidointi (engl. K-fold cross-validation) sisältää harjoitusjoukon jakamisen satunnaisesti k taitokseen (engl. folds), jotka ovat suunnilleen samankokoisia. Ristiinvalidoinnissa taitosta käsitellään validointijoukkona, ja muuta harjoitusdataa käytetään koneoppimisputken harjoittamiseen, jonka jälkeen muulla harjoitusdatalla tehdään ennusteita validointitaitoksesta. Tämä toistetaan k kertaa, ja jokaisella kerralla eri taitosta käsitellään validointijoukkona. Lopuksi arviointimittareiden keskiarvoista lasketaan k-ositus ristiinvalidointitulos, joka määritetään kaavan 1 avulla. (James et al. 2021)

$$CV_{(k)} = \frac{1}{k}\sum_{i=1}^{k} M_i \quad (1)$$

, missä

$CV_{(k)}$ = ristiinvalidointipiste

k = taitoksien määrä

$M_{(i)}$ = arviointimittari

TPOT on tehokas AutoML-työkalu, jonka kyky etsiä laajasta valikoimasta malleja tekee siitä houkuttelevan erityisesti, kun halutaan säästää aikaa ja parantaa koneoppimistuloksia. Työkalun potentiaalista huolimatta, sitä tulee kuitenkin käyttää harkiten ongelma-alue ja datan ominaisuudet huomioiden.

3.3 Kuvaus datasetistä

Tämä kandidaatintutkielma on kvantitatiivinen tutkimus, jossa tutkimusaineistona käytetään Pythonin FastF1-kirjastoa. FastF1 on rakennettu Pythonin Pandas- ja Numpy-kirjastojen päälle. FastF1 saa datansa kahdesta eri lähteestä: F1 virallisesta datastreamista ja Ergast web API:sta. Kirjasto tarjoaa kattavan kuvan kilpailujen aikana tuotetuista telemetriadatasta ja yksityiskohtaisemman kuvan auton suorituskyvystä. Se sisältää laajan valikoiman erilaisia muuttuja liittyen Formuloihin. (theOehrly 2023)

Tutkimuksessa FastF1-aineiston pohjalta on muodostettu kahdeksan uutta datasettiä. Tutkimuksessa käytetyt datasetit ovat rakennettu siten, että ne ottavat huomioon vain samojen osakilpailujen datat vuosilta 2018–2022. Datasetit ovat rakennettu Pythonin Pandas-kirjaston tarjoamaan datarakenteeseen, DataFrameen. Kaikki kahdeksan datasettiä sisältää samat muuttujat. Datasetteihin on kerätty dataa eri vuosilta lisäämällä ne allekkain DataFrameen siten, että vuoden 2022 datasetti on päällimmäisenä ja vuoden 2018 datasetti taas alimmaisena.

DataFramen jokainen rivi vastaa yhdestä kuljettajasta kerättyä dataa per osakilpailu. Jokaisena vuonna vuosien 2018–2022 aikana, Formula 1:ssä on ollut 20 kuljettajaa. Tutkimukseen kerätyt datasetit ovat rajattu siten, että ne ottavat huomioon ainoastaan ne osakilpailut, jotka ovat ajettu jokaisena vuonna 2018–2022 välillä. Toistuvien osakilpailuiden määrään

suurin vaikuttaja on Covid-19, jonka vaikutukset näkyivät erityisesti vuoden 2020 Formula 1 -osakilpailuissa, joista osa peruttiin kokonaan. Siten tutkimuksessa käytetty osakilpailuiden määrä rajautuu kahdeksaan.

Koska data tulee suoraan F1-datastreamista ei sitä pysty muokkaamaan ulkopuoliset. FastF1-datasettiä voidaan siis pitää yleisesti luotettavana. On kuitenkin tärkeää lähestyä dataa kriittisesti ja huomioida mahdolliset esiintyvät virheet. FastF1 on avoimen lähdekoodin kirjasto (theOerly 2023), joka tarkoittaa, että kuka tahansa voi päästä käsiksi dataan. Kuitenkin käyttäjien tekemät muokkaukset tapahtuvat vain heidän lataamassa datassaan.

Vaikka FastF1-kirjasto olisi tarjonnut säädatan kaikista tapahtumista, rajattiin se tutkimuksen ulkopuolelle. Tutkimuksessa DataFramen rivit ovat rakennettu siten, että jokainen rivi edustaa yhtä kuljettajaa tietyssä osakilpailussa. Mikäli säädata olisi lisätty DataFrameen, olisi muuttujan arvo ollut sama kaikille kuljettajille samassa osakilpailussa. Tästä syystä datan vaihtelu olisi ollut hyvin pientä, ja siten se ei luultavasti toisi lisäarvoa tutkimukselle.

4 Tutkimuksen toteutus

Tämä tutkimus toteutettiin kvantitatiivisena tutkimuksena. Tutkimuksessa TPOT:n tekemiä ennusteita Formula 1 -osakilpailusijoituksia vertailtiin toteutuneisiin sijoituksiin. Tässä tutkimuksessa käytetään TPOT:n luokittelutoimintoa osakilpailusijoitusten ennustamiseen. TPOT:n suoriutumista mitattiin luokittelualgoritmin oletusarviointimittarin eli ennustetarkkuuden avulla. Tarkoituksena on luokitella ensimmäisen 10 kuljettajan sijoitus osakilpailussa. Toisin sanoen siis ennustaa kuljettajan sijoitus ennalta määritetyistä luokista.

4.1 Tutkimusaineiston esikäsittely

Pythonin AutoML-kirjasto, TPOT, sisältää luokittelualgoritmin, joka automatisoi koneoppimisprosessin vaiheista ominaisuuksien valinnan sekä koneoppimismallin rakentamisen. (Olson et al. 2016) Tutkimusaineiston esikäsittely ei sisälly TPOT:n koneoppimisputken rakennusalgoritmiin, joten se täytyi tehdä itse.

Tutkimuksessa rakennettiin yhteensä kahdeksan datasettiä kahdeksasta eri osakilpailuista vuosilta 2018–2022. Tutkimusaineisto esikäsiteltiin siten, että se sopii DataFramen muotoon. Lopullinen DataFrame rakennettiin siten, että jokaisen vuoden muuttujat kerättiin ensin yhteen DataFrameen yhdistämällä ne toisiinsa kuljettajanumeron avulla. Sen jälkeen kaikkien viiden vuoden DataFramet yhdistettiin laittamalla ne allekkain. Tutkimuksessa dataa eri tapahtumista haettiin kahdesta eri FastF1-kirjaston sisäänrakennetusta DataFramesta: results ja laps. Results-DataFramen rivit olivat indeksoitu kuljettajanumeron mukaan. Laps-DataFramessa rivit olivat järjestetty siten, että ne sisälsivät dataa kaikkien kuljettajien kaikista ajetuista kierroksista.

Suurin osa muuttujista oli mahdollista lisätä suoraan tutkimksessa käytettyihin DataFrameihin FastF1:n results-DataFramesta, ilman erillistä esikäsittelyä. Osa laps-DataFramesta otetuista muuttujista oli myös mahdollista ottaa laskemalla vain rivien määrä per kuljettaja. Talli-muuttuja oli ainoa merkkijono, joten tallien nimet muutettiin vastaamaan numeroita. TPOT pystyy käsittelemään myös merkkijonoja siltä osin, kun ne ovat kategorisoituja

muuttujia (Olson et al. 2016). Tutkimusaineistossa samojen tallien nimet saattoivat kuitenkin vaihdella vuosittain, joten ne muokattiin vastaamaan numeroita.

Kaikkien kolmen harjoituksen nopeusansoista (engl. SpeedTrap) kerättiin kuljettajien nopeudet ja laskettiin niille keskiarvo. Nopeusansoja oli yhteensä neljä kappaletta per harjoitus eli keskinopeuksia oli yhteensä 12 kappaletta. Harjoituksien kierrosaikojen muuttujatyyppi oli timedelta. Jokaisen harjoituksen nopein kierrosaika muutettiin timedeltasta kokonaisluvuksi, muuttamalla se sekunneiksi. Osa datasta oli siirretty suoraan FastF1 paketista tutkimusaineistoon, joten lopputuloksena osa muuttujista oli muuttujatyypiltään kokonais- ja desimaalilukuja ja osa objekteja. Jotta muuttujat olisivat yhtenäisessä muodossa, muuttujatyypit muutettiin joko kokonais- tai desimaaliluvuksi.

Tutkimuksessa käytetty data sisältää jonkin verran puutteita. Joistakin osakilpailun tapahtumista puuttui tietoja esimerkiksi osasta harjoituksista tai aika-ajoista kaikkea dataa ei ollut saatavilla. Datan puutteellisuuden vuoksi jouduttiin osaa arvoista korjaamaan numerolla nolla, jotta niiden lisääminen DataFrameen yksinkertaistuisi. Datasetin rakentamisessa puuttuvia arvoja jouduttiin korvata nollalla, joten tutkimuksen yhtenäisyyden kannalta valittiin, että kaikki puuttuvat arvot (engl. NaN, Not a Number) korvataan arvolla nolla. Myös osa datasta puuttui, koska kuljettaja ei ollut osallistunut tapahtumaan. Tästä syystä on myös perusteltua korvata puuttuvat arvot nollalla. Pranayn (2022) mukaan puuttuvien arvojen korvaaminen nollalla voi olla joissakin tapauksissa järkevä strategia, erityisesti jos ne ovat satunnaisia eikä niissä ole selkeitä kaavoja. Kuitenkin toisissa tapauksissa arvojen korvaaminen nollalla voi aiheuttaa harhaa tai johtaa epätarkkoihin ennusteisiin.

Datasetteihin otetut muuttujat valikoituivat siten, että jokaisesta osakilpailuun liittyvästä tapahtumasta kerättiin dataa. Kaikki kahdeksan datasettiä ovat suhteellisen pieniä ja sisältävät 100 riviä dataa. Datan esikäsittelyn ja DataFramen rakentamisen jälkeen sarakkeiden määrä vaihteli välillä 27–30. Sarakkeiden määrän vaihtelu johtuu erilasista rengastyypeistä, joita käytetään vaihtelevasti riippuen osakilpailusta. Valitut muuttujat ja tapahtumat, joista ominaisuudet ovat kerätty on näkyvissä taulukossa 2.

Taulukko 2: Datasettien muuttujat

	GP	Aika-Ajot	FP1	FP2	FP3
Kuljettajanumero	x				
Lähtösijoitus	x				
Talli	x				
Ajetut kierrokset/rengastyyppi	x				
Ajetut kierrokset		x	x	x	x
Keskinopeudet / SpeedTrap x4			x	x	x
Nopein kierros sekunneissa			x	x	x
Varikkopysähdykset	x				

4.2 Mallin sovitus

TPOT jakaa datan harjoitus- ja testidataksi oletuksena 75/25 suhteessa (Olson et al. 2016). Tutkimuksessa ennustusmallin tarkoituksena on ennustaa vuoden 2022 osakilpailuiden tarkkoja sijoituksia 1–10. Tästä syystä datan jakaminen harjoitus- ja testidataan tehdään manuaalisesti. Testidata muodostuu siten vuoden 2022 sijoituksista 1–10 eli sen suuruus on 10 %. Harjoitusdatan suuruudeksi jää tällöin loput 90 %.

TPOT:n luokittelutoimintoon kuuluu sukupolvien määrän, populaatiokoon sekä ristiinvalidointiarvon valinta. Tässä tutkimuksessa käytetään vakiomuotoista sukupolvien määrää eli viittä sukupolvea. Populaatiokoko määrittää koneoppimisputkien määrän, joka yhdessä sukupolvessa muodostetaan. Populaatiokoko on tutkimuksessa määritetty 100:an, joka on myös oletusasetuksena TPOT:ssa. Suuri sukupolvien määrä ja populaatio koko voi parantaa tutkimuksen tulosta, mutta samalla myös pitkittää datan harjoitteluun käytettyä aikaa. (TPOT 2023) Koska tutkimuksessa käytetyt datasetit ovat suhteellisen pieniä, ei sukupolvien määrää tai populaatio kokoa ole tarvetta kasvattaa. Jokaisessa ennustusmallissa käytettiin ristiinvalidointiarvona numeroa viisi. Isompaa arvoa ei ollut mahdollista käyttää ristiinvalidointialgoritmin takia, joka vaatii isomman datasetin isompaan arvoon.

Sovitusfunktio (engl. Fit function) alustaa geneettisen ohjelmoinnin löytääkseen suurimman pistemäärän saaneen koneoppimisputken perustuen keskimääräiseen K-ositus ristiinvalidointiin (engl. K-fold cross-validation). (Romano et al. 2021) Kun kaikki viisi sukupolvea on käyty läpi, antaa TPOT parhaimman ristiinvalidointituloksen sekä parhaimman

koneoppimisputken mallille. Lisäksi tutkimuksessa TPOT listaa kaikki ominaisuudet sekä ominaisuuksien selitysasteet.

Tutkimuksessa ennustusmallia testattiin yhteensä kahdeksan eri osakilpailun datasettiin. Kaikissa tapauksissa datasetti sovitettiin malliin samalla tavalla ja pyrittiin ennustamaan samaa muuttujaa eli vuoden 2022 osakilpailusijoituksia 1–10.

5 Tutkimustulokset

Tässä luvussa käydään läpi tutkimuksen tulokset. Tulosten käsittely aloitetaan tarkastelemalla mallin antamia ristiinvalidointipisteitä eri osakilpailuista. Sen jälkeen siirrytään tutkimaan tärkeimpiä TPOT:n valitsemia koneoppimismallin ominaisuuksia ja muuttujia. Lopuksi tarkastellaan ennustusmallien ennustustarkkuutta ja sijoituksia, jotka malli on ennustanut oikein.

Taulukko 3: Ristiinvalidointipisteet (CV) osakilpailuille

Osakilpailu	**CV-piste**
Belgian Grand Prix	0,156
Espanjan Grand Prix	0,178
Bahrainin Grand Prix	0,178
Abu Dhabin Grand Prix	0,229
Itävallan Grand Prix	0,122
Unkarin Grand Prix	0,091
Britannian Grand Prix	0,189
Italian Grand Prix	0,189

Taulukossa 3 esitetään ristiinvalidointipisteet tutkituilta osakilpailuilta tuhannesosan tarkkuudella. Taulukosta voidaan havaita, että kaikkien osakilpailuiden tulokset ovat alle 0,25. Parhaimman ristiinvalidointipisteen on saanut Abu Dhabin osakilpailu, joka on myös ainoa piste yli 0,20. Muiden osakilpailuiden ristiinvalidointipisteet ovat suunnilleen samansuuruisia paitsi Unkarin osakilpailu. Unkarin osakilpailun ristiinvalidointipiste on selkeästi kaikista pienin ja sen arvo on alle 0,10. Taulukosta voidaan havaita, että pisteet vaihtelevat välillä 0,91–0,229, mikä osoittaa, että jotkut mallit suoriutuvat paremmin kuin toiset. Kuitenkaan pisteiden erot eivät ole kovin suuria. Suurin ero on Abu Dhabin ja Itävallan osakilpailun välillä, joiden ristiinvalidointipisteet eroavat vain 0,107, joka taas viittaa siihen, että mallit voivat toimia samalla tavalla eri kilpailuissa.

Ristiinvalidointia suorittaessa pyritään määrittämään, kuinka hyvin mallin voidaan olettaa suoriutuvan uudella datalla (James et al. 2021). Tässä tutkimuksessa mallin alhainen ristiinvalidointipiste kertoo siitä, ettei sen voida olettaa suoriutuvan kovinkaan merkittävästi uudessa datassa. Keertishin ja Preethin (2023) tekemässä tutkimuksessa analysoitiin Formula

1 -osakilpailuja eri koneoppimismallien avulla. Tutkimuksessa ristiinvalidointipisteiden tarkkuus vaihteli 0,90–0,95 välillä. Kun tämän tutkimuksen ristiinvalidointipisteitä verrataan Keertishin ja Preethin (2023) tutkimukseen, voidaan todeta, etteivät tulokset ole merkittäviä.

Ominaisuuksien tärkeys (engl. Feature importance) auttaa tunnistamaan tärkeimmät muuttujat, jotka vaikuttavat ennustustulokseen (Brownlee 2020). Tutkimuksessa muuttujien määrät vaihtelivat välillä 27–30 dataseteissä erilaisten rengastyyppien takia. Tutkimuksessa käydään läpi jokaisen mallin kolme korkeimmalla selitysasteella varusteltua ominaisuutta. Muuttujat ja niiden selitysasteet ovat järjestetty osakilpailuittain selitysasteiden suuruuden perusteella taulukossa 4.

Taulukko 4: Ennustusmallien tärkeimmät ominaisuudet

Osakilpailu	**Ominaisuus 1**	**Ominaisuus 2**	**Ominaisuus 3**
Belgian GP	GridPosition (0,136)	DriverNumner (0,114)	FP1SpeedI2 (0,080)
Espanjan GP	DriverNumber (0,569)	GridPosition (0,431)	
Bahrainin GP	DriverNumber (0,452)	GridPosition (0,306)	TeamName (0,147)
Abu Dhabin GP	GridPosition (0,156)	DriverNumber (0,051)	TeamName (0,037)
Itävallan GP	GridPosition (0,204)	Soft (0,074)	TeamName (0,062)
Unkarin GP	TeamName (0,115)	GridPosition (0,082)	FP1SPeedI1 (0,067)
Britannian GP	GridPosition (0,341)	Medium (0,140)	Hard (0,134)
Italian GP	GridPosition (0,261)	TeamName (0,234)	DriverNumber (0,137)

Taulukosta 4 voidaan havaita, että ominaisuus GridPosition eli lähtösijoitus oli tärkeä ominaisuus kaikissa osakilpailuissa. Se oli myös tärkein ominaisuus kuudessa osakilpailuissa kahdeksasta. Tämä viittaa siihen, että kuljettajan lähtösijoituksella on merkittävät vaikutus hänen suoritukseensa kilpailussa. Muehbauer (2017) on aiemmin osoittanut tutkimuksessaan lähtösijoituksen olevan merkittävä tekijä osakilpailuissa menestymisessä, joka tukee tämän tutkimuksen tuloksia.

Tuloksien perusteella toinen tärkeä ominaisuus on DriverNumber eli kuljettajanumero. Kuljettajanumero on kolmen merkittävimmän ominaisuuksien joukossa viidessä osakilpailussa kahdeksasta. Kuljettajanumeron tärkeys indikoi sitä, että kuljettajan henkilöllisyydellä on rooli kilpailun tulosten ennustamisessa.

Kolmas tärkeä ominaisuus tuloksien perusteella on TeamName eli talli, jossa kyseinen kuljettaja ajaa. Talli on yhteensä viidessä osakilpailussa kahdeksasta kolmen merkittävimmän ominaisuuden joukossa, mikä viittaa siihen, että myös tallilla, jossa kuljettaja ajaa on merkitys osakilpailun lopputulokseen. Sicoie (2022) sekä van Kesteren ja Bergkamp (2022) ovat osoittaneet tutkimuksissaan tallin merkityksen osakilpailusijoituksissa.

Taulukosta voidaan myös havaita, että Espanjan osakilpailussa oli ainoastaan kaksi merkittävää ominaisuutta. Kaikkein muiden ominaisuuksien selitysasteet olivat nolla, ja siten ennustusmallin ennustus osakilpailusijoituksista perustui vain näihin kahteen ominaisuuteen.

Tutkimustulosten perusteella muut merkittävät ominaisuudet FP1SpeedI2, Medium, Hard ja FP1SpeedI1 vaikuttavat olevan tärkeitä tietyissä osakilpailuissa. Tämä osoittaa, että renkaiden valinta tai harjoituksissa ajettujen kierrosten keskinopeus voivat vaikuttaa kilpailun lopputulokseen.

Ominaisuuksien selitysasteita tulkittaessa tulee myös huomioida se, että ennustusmalli saattaa valita tiettyjä ominaisuuksia sillä perusteella, ettei muita järkeviä muuttujia ole. Tästä syystä ennustusmalli saattaa yrittää selittää osakilpailujen sijoituksia esimerkiksi kuljettajanumerolla. Siten ei ole mahdollista todeta, että johtuuko ennustusmallin tärkeimmät ominaisuudet esimerkiksi kuljettajan aikaisemmasta menestyksestä vai pyrkimyksestä selittää ennustuksia eniten todennäköisimmällä ominaisuudella.

TPOT:n luokittelu toiminto mittaa mallin suorituskykyä ennustetarkkuudella (engl. Accuracy). (TPOT 2023) Taulukossa 5 käydään läpi jokaisen ennustusmallin ennustustarkkuudet.

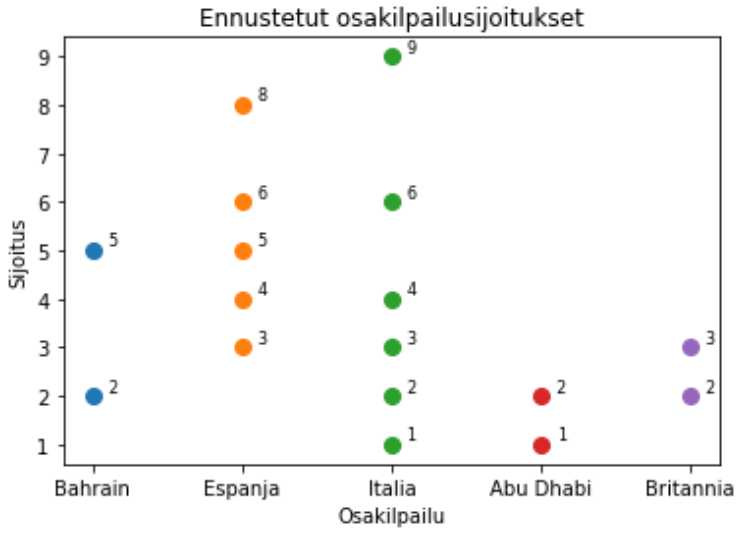
Taulukko 5: Ennustusmallien tarkkuudet

Osakilpailu	**Accuracy**
Belgian GP	0 %
Espanjan GP	50 %
Bahrainin GP	20 %
Abu Dhabin GP	20 %
Itävallan GP	0 %
Unkarin GP	0 %
Britannian GP	20 %
Italian GP	60 %

Taulukosta voidaan havaita, että ennustusmalli tarkkuus Belgian, Itävallan sekä Unkarin osakilpailuille on 0 %. Toisin sanoen ennustusmalli ei osannut ennustaa yhtäkään sijoitusta oikein näissä osakilpailuissa. Malli toimii hieman paremmin Bahrainin, Abu Dhabin sekä Britannian osakilpailuille, joiden kaikkien tarkkuudet ovat 20 %. Siten ennustusmalli on osannut ennustaa kaksi sijoitusta oikein näistä osakilpailusta. Korkeimmat ennustustarkkuudet ovat Espanjan ja Italian osakilpailuilla. Espanjan osakilpailusta malli on osannut ennustaa viisi sijoitusta oikein ja Italian osakilpailusta paikkansa piti kuusi sijoitusta.

Ennustusmallin ennustetut todennäköisyydet ovat suhteellisen vaihtelevia, eikä selkeästi yhtenäisiä. Kun otetaan huomioon kaikki osakilpailut ennustusmalli osasi keskimäärin ennustaa Formula 1 osakilpailusijoituksia oikein 21,25 % tarkkuudella. Ennustusmallin todennäköisyys ennustaa sijoituksia oikein on siten suhteellisen matala. Myös ennusteiden tarkkuuden vaihtelevuus heikentää ennustusmallin suorituskyvyn luotettavuutta.

Tässä tutkimuksessa ennustusmalli yritti ennustaa kymmenen ensimmäisen kuljettajan tarkkoja sijoituksia. Kuviossa 4 tarkastellaan tarkemmin niiden viiden osakilpailun sijoituksia kuljettajille, jotka ennustusmalli oli ennustanut oikein.



Kuvio 4: Ennustetut sijoitukset

Kuviosta 4 voidaan havaita, että oikein ennustetut sijat ovat jokseenkin satunnaisia. Kuitenkin yhteensä neljässä osakilpailussa viidestä, malli on ennustanut sijan kaksi oikein. Sijan kolme on malli osannut ennustaa oikein kolmessa kilpailussa. Sijat yksi, neljä, viisi ja kuusi on malli osannut ennustaa oikein vain kahdessa kilpailussa ja sijoitukset kahdeksan ja

yhdeksän vain yhdessä. Kuviosta voidaan myös havaita, että neljässä osakilpailussa lähes kaikki ennustetut sijoitukset ovat peräkkäisiä.

Kuviosta voidaan tulkita, että ennustukset ovat tarkempia, kun kyseessä on osakilpailusijoitusten kärkisijoitukset. Jotta ennustusmallia voitaisiin tulkita tarkemmin ja sen luotettavuutta arvioida, tulisi sitä testata laajemmalle datasetille. Tulosten perusteella malli vaikuttaa toimivan osittain ja pystymään ennustamaan suhteellisen satunnaisia tuloksia. Mallin luotettavuudesta ei kuitenkaan voida olla varmoja ennustustarkkuuden vaihtelevuudesta johtuen. Koska tutkimusaineisto on pieni, ei automatisoitu koneoppimistyökalu toimi välttämättä täysin oikein, sillä se on suunniteltu käsittelemään suurempia datasettejä. Täten se voi vaikuttaa ominaisuuksien selitysasteisiin ja yrittää ennustaa tuloksia satunnaisesti muuttujilla.

Liitteissä 1, 2, 3, 4 ja 5 kyseisten viiden osakilpailun ennusteet kaikille sijoituksille on esitetty viivadiagrammien avulla. Jokaisesta kuviosta voidaan huomata, että niissä toistuu lähes samanlainen kaava. Alkupään ennustetut sijoitukset pysyvät suhteellisen lähellä toteutuneita sijoituksia. Sijoituksien puolessa välissä, hieman osakilpailun mukaan, ennustetut sijoitukset hyppäävät sijoituksille 14–17. Tämä voi johtua siitä, että malli on liian sovitettu harjoitusdataan eikä se pysty esimerkiksi yleistämään opittua täysin uuteen dataan. Toinen mahdollisuus on, että mallissa käytetyt ominaisuudet eivät edusta tarpeeksi todellista taustalla olevaa suhdetta muun datan ja kohdemuuttujan välillä.

Tutkimustuloksista voidaan päätellä, että ennustusmalli suoriutui kohtalaisesti. Ennustusmalli nosti esille aikaisemmissa tutkimuksissa osoitetut merkittävät ominaisuudet ja ennustusmalli onnistui ennustamaan osan sijoituksista oikein. Ennustusmallin avulla saatujen tuloksien merkittävyyttä alentaa kuitenkin sen alhaiset ristiinvalidointitulokset sekä ennustettujen sijoituksen hajanaisuus sekä ennustetarkkuuden alhaisuus. Suoriutuminen Formula 1:ssä määräytyy monien eri tekijöiden yhdistelmästä, mikä tekee tulosten ennustamisesta erityisen vaikeaa (van Kesteren & Bergkamp 2023).

6 Yhteenveto ja johtopäätökset

Tämän tutkimuksen tavoitteena oli tutkia erilaisia automatisoituja koneoppimismalleja ja sitä, kuinka tarkasti ne pystyvät ennustamaan Formula 1 -osakilpailujen sijoituksia datasettien avulla. Tutkimusaineistona käytettiin FastF1-Python kirjaston pohjalta rakennettuja kahdeksaa datasettiä kahdeksasta eri Formula 1 -osakilpailusta. Aineistot sisälsivät dataa kisoista, aika-ajoista sekä harjoituksista vuosilta 2018–2022. Tutkimuksessa käytetty AutoML-työkalu, TPOT, rakensi ennustusmallin muodostettujen tutkimusaineistojen pohjalta. Mallin rakentaminen tapahtui Python-ohjelmointikielen avulla. Ennustusmalli yritti ennustaa osakilpailujen tarkkoja sijoituksia 1–10 vuodelta 2022. Ennustusmallin suoriutumista arvioitiin ristiinvalidointipisteiden ja ennustustarkkuuksien avulla. Lisäksi tutkimuksessa tarkasteltiin ennustusmallin valitsemia tärkeimpiä ominaisuuksia, jotka vaikuttivat mallin ennustamiin sijoituksiin.

6.1 Vastaukset tutkimuskysymyksiin

Tutkimuksessa tutkimusongelmaan pyrittiin vastaamaan kahden tutkimuskysymyksen avulla. Ensimmäinen tutkimuskysymys on: *Mitä on AutoML ja millaisia sovelluksia sillä on kirjallisuuden mukaan?* Tutkimuksessa selvisi, että AutoML on nopeasti kehittyvä ala, joka on saanut paljon huomiota koneoppimistehtävien automatisoinnin, mallien suorituskyvyn parantamisen sekä ihmisen väliintulon vähentämisen avulla. AutoML mahdollistaa monimutkaisten koneoppimisalgoritmien soveltamisen ongelmien ratkaisuun ilman syvällistä tuntemusta koneoppimisesta. Tutkimuksesta selvisi myös, että AutoML:n ensisijainen tehtävä on vähentää koneoppimisvaiheiden manuaalista työtä ja siten nopeuttaa koko prosessin kulkua.

Tutkimuksessa selvisi, että koneoppimisprosessin automatisoinnissa voidaan käyttää useita erilaisia sovelluksia AutoML:stä. Erilaiset AutoML-sovellukset keskittyivät automatisoimaan hieman eri vaiheita koneoppimisprosessista ja niiden vahvuudet vaihtelivat riippuen käsitteillä olevasta ongelmasta. Tutkimuksen mukaan AutoML-menetelmien käytetyt sovellukset olivat: ruudukkohaku, bayesilainen optimointi, metaoppiminen sekä geneettinen ohjelmointi.

Toinen tutkimuskysymys on: *Mitkä ovat Formula 1 osakilpailusijoitusten ennustamiseen sopivat AutoML-algoritmit ja millaisia tuloksia niiden avulla on saavutettavissa?* Tutkimuksessa AutoML-algoritmeista osakilpailusijoitusten ennustamiseen valikoitui TPOT. TPOT on helppokäyttöinen koneoppimistyökalu, jonka avulla voi rakentaa ennustusmalleja ilman aikaisempaa tuntemusta koneoppimisalgoritmeista. TPOT on hyödyllinen työkalu, joka nopeuttaa koneoppimisprosessia, mutta se ei kuitenkaan korvaa inhimillistä asiantuntemusta. Tästä syystä on tietojen esikäsittely ja mallin tuloksien kriittinen tulkinta ja validointi on perusteellista. Lisäksi TPOT:n kanssa työskentely voi olla aikaa vievää erityisesti, kun kyseessä on suuret datasetit. Tutkimuksessa käytetyt aineistot olivat kuitenkin pieniä, joten ne sopivat hyvin TPOT:lle. Tutkimuksessa TPOT osoittautui käyttäjäystävälliseksi koneoppimistyökaluksi.

TPOT:n rakentaman ennustusmallin tutkimustuloksista selviää, että mallin antamat ristiinvalidointipisteet olivat alhaiset. Keertishin ja Preethin (2023) tutkivat menestystä Formula 1 -sarjassa käyttäen ristiinluokiteltuja monitasomalleja. Tutkimuksessa käytettyjen mallien ristiinvalidointipisteet olivat kaikki yli 0,90. Kun tämän tutkimuksen ristiinvalidointipisteitä verrataan aikaisempaan tutkimukseen, voidaan todeta, että ennustusmallien ristiinvalidointipisteet eivät ole merkittäviä.

Tutkimuksen ennustemalleilla tärkeimmät ominaisuudet olivat lähtösijoitus ja kuljettajanumero. Myös talli, johon kuljettaja kuului, oli merkittävä ominaisuus. Ominaisuuksien kannalta osa ominaisuuksista oli johdonmukaisia aikaisempien tutkimusten kanssa. Muehbauerin (2017) tutkimus on aiemmin osoittanut, että loppusijoitukseen vaikuttaa merkittävästi kuljettajan lähtösijoitus. Sicoien (2022) sekä van Kesterenin ja Bergkampin (2022) tutkimuksien mukaan tallin tärkeys loppusijoituksien kannalta on merkittävä. Aikaisempiin tutkimuksiin perustuvat hypoteesit, H1 ja H2, osoittavat, että lähtösijoitus sekä tiettyyn talliin kuuluminen vaikuttaisivat lopulliseen sijoitukseen. Myös tässä tutkielmassa löydettyjen vaikuttavien ominaisuuksien perusteella on perusteltua sanoa, että nämä kaksi muuttujaa vaikuttavat osakilpailusijoitukseen.

Vaikka samat ominaisuudet toistuvat tärkeinä lähes kaikissa osakilpailuissa, ei toisaalta tutkimuksen perusteella pystytä suoraan toteamaan, että juuri nämä ominaisuudet olisivat merkittäviä. Ennustusmallilla on mahdollisuus pyrkiä selittämään tuloksia eniten järkevimmällä ominaisuudella eikä pelkästään esimerkiksi kuljettajan tai tallin aikaisemman menestyksen perusteella. Myös ennustetarkkuuden perusteella ei ole syytä luottaa tärkeimpiin

ominaisuuksiin, sillä vaikka mallin tärkeimmät ominaisuudet olisivat olleet lähtösijoitus, kuljettajanumero ja talli saattoi sen ennustetarkkuus olla nolla prosenttia.

Ennustetarkkuuksia tutkittaessa havaittiin, että tarkkuudet olivat vaihtelevia. Osa tarkkuuksista osoittautuivat suhteellisen korkeiksi, mutta osa jäi lähemmäs nollaa. Mallin oikein ennustamista osakilpailusijoista pystyi havaitsemaan peräkkäisyyden neljästä kerrasta viidessä. Myös suurin osa oikein ennustetuista sijoituksista oli kärkipään sijoituksia ja vain satunnaisia sijoituksia taaemmilta sijoilta. Mallien kaikissa ennustetuissa sijoituksissa toistui myös tietynlainen kaava. Alkupään ennusteet pysyivät lähellä toteutuneita, mutta keskivaiheilla ennustukset hyppäsivät korkeille sijoituksille. Kaiken kaikkiaan voidaan siis todeta ennustetarkkuuksien eri osakilpailuille olleen kohtalaisen hyviä.

6.2 Tutkimuksen luotettavuus ja jatkotutkimus

Tutkimuksen teoreettisessa viitekehyksessä käytetty teoria on peräisin vertaisarvioiduista julkaisuista sekä kirjallisuudesta, jota on hyödynnetty aikaisemmissa tutkimuksissa. Koneoppiminen alana on vakiintunut ja siihen liittyy huomattavasti tutkimusta (Biamonte et al. 2017). Koneoppiminen ja automatisoitu koneoppiminen kehittyy jatkuvasti, minkä takia alan asiantuntijoiden on pysyttävä ajan tasalla viimeisimmistä edistysaskeleista varmistaakseen omien tutkimuksien ajankohtaisuus. Tästä syystä tutkimuksen teoriapohjaa voidaan pitää luotettavana.

Tutkimuksen luotettavuuden arvioinnissa tutkimuksessa käytettyä aineistoa voidaan pitää suhteellisen luotettavana. Aineisto on kerätty virallisesta Formula 1 -datastreamista, joita ulkopuoliset eivät pääse muokkaamaan. Aineiston luotettavuutta alentaa sen sisältämät puuttuvat arvot, joita esiintyi osassa datasta. Tutkimustulosten luotettavuuden arvioinnissa tulee huomioida ensisijaisesti otoskoko. Tutkimuksessa ennustusmalleja oli vain kahdeksan, joten otos oli suhteellisen pieni. Pieni otoskoko voi vaikuttaa tulosten luotettavuuteen ja yleistettävyyteen heikentämällä niitä. Datasettien koot olivat myös pieniä, koska dataa pystyttiin keräämään vain viiden vuoden ajalta eri ohjelmointirajapinnan takia. Vaikka aikaväli on suhteellisen lyhyt, antoi se silti hyvän yleiskuvan ennustusmallin toimivuudesta. Koska tutkimuksen empiriaosuus on toteutettu Python-ohjelmointikielen avulla, mahdollisuus inhimillisiin virheisiin on olemassa, mikä voi vaikuttaa tutkimuksen tuloksiin.

Jatkotutkimuksen näkökulmasta aihetta tulisi lähestyä useamman eri automatisoidun koneoppimistyökalun kanssa. Tutkiminen muilla AutoML-sovelluksilla mahdollistaisi algoritmien suorituskykyjen vertailemisen ja siten kartuttaisi sopivimman työkalun tähän aihealueeseen. Lisäksi datan määrän kasvattaminen olisi tärkeää ottamalla huomioon muuttujia, jotka tässä tutkimuksessa rajattiin pois kuten sääolosuhteet. Myös aiempien vuosien huomioiminen tutkimuksessa voisi lisätä sekä luotettavuutta että ennustetuloksien tarkkuutta.

Tässä tutkimuksessa pyrittiin ennustamaan kuljettajien tarkkoja sijoituksia 1–10. Tuloksien tarkkuutta voitaisiin parantaa tutkimalla sitä osaako AutoML-työkalu esimerkiksi ennustaa kymmenen parhaan joukossa olevat kuljettajat eikä tarkkoja sijoituksia. Luokitteluasetelman käyttäminen mahdollisesti nostaisi ennustustarkkuutta. Lisäksi eräs näkökulma jatkotutkimuksen kannalta voisi olla ennustuskohteen muuttaminen. Formula 1 -osakilpailuissa on mahdollisuus ennustaa lukuisa muitakin tekijöitä kuten varikkopysähdyksien määrää tai nopeinta kierrosaikaa käyttäen automatisoituja koneoppimismenetelmiä.

Tämä tutkimus osoitti, että ennustusmalli toimii osittain viiden vuoden tutkitulla aikavälillä. Henderssonin ja Kirranen (2018) mukaan ennustusmallien tulokset voivat parantua, kun painotetaan aikaisempia tuloksia. Tästä syystä olisi tärkeää tutkia, miten ennustusmalli suoriutuu pidemmällä aikavälillä. Lisäksi voidaan olettaa erilaisten AutoML-algoritmien kehittyvän yhä tehokkaammiksi koneoppimisen nopean leviämisen ja datan määrän räjähdysmäisen kasvun ansioista. Siten tutkimuksen tuloksia voidaan pitää potentiaalisina jatkotutkimuksen valossa, sillä tämä tutkimus edistää AutoML:n ennustusmahdollisuuksia moottoriurheilun saralla.

Lähteet

AWS (2018) Formula 1 Case Study [Verkkodokumentti]. [Viitattu 13.2.2023]. Saatavilla: https://aws.amazon.com/solutions/case-studies/formula-one/?nc1=h_ls

Badithe, S. (2022) Introducing Featuretools SQL-Automatically Import Relational Data into Featuretools [Verkkodokumentti]. [Viitattu 8.4.2023]. Saatavilla: https://innovation.alteryx.com/featuretools-sql/

Biamonte, J. et al. (2017) Quantum machine learning. *Nature (London).* Vol. 549, No. 7671, pp. 195-202

Brownlee, J. (2020) How to Calculate Feature Importance With Python. [Verkkodokumnetti]. [Viitattu 23.4.2023]. Saatavilla: https://machinelearningmastery.com/calculate-feature-importance-with-python/

Celik, B., Singh, P. & Vanschoren, J. (2022) Online AutoML: An adaptive AutoML framework for online learning. *Special Issue on Automating Data Science*

Das, S. & Cakmak, U. (2018) *Hands-On Automated Machine Learning*. 1st edition. Birmingham, England: Packt.

Feurer, M., Klein, A., Eggensperger, K., Springenberg, J. T., Blum, M., & Hutter, F. (2015) Efficient and robust automated machine learning. *In Advances in Neural Information Processing Systems.* Vol. 28, pp. 2962-2970

FIA (2023a) Cost Cap Administration [Verkkodokumentti]. [Viitattu 20.5.2023]. Saatavilla: https://www.fia.com/events/fia-formula-one-world-championship/season-2022/cost-cap-administration

FIA (2023b) 2023 FIA FORMULA ONE WORLD CHAMPIONSHIP [Verkkodokumentti]. [Viitattu 13.2.2023]. Saatavilla: https://www.fia.com/events/fia-formula-one-world-championship/season-2023/2023-fia-formula-one-world-championship

FIA (2023c) Organisation [Verkkodokumentti]. [Viitattu 11.3.2023]. Saatavilla: https://www.fia.com/organisation

FIA (2023d) Regulations [Verkkodokumentti]. [Viitattu 11.3.2023]. Saatavilla: https://www.fia.com/regulation/category/110

Gutiérrez, E. & Lozano, S. (2020) Benchmarking Formula One auto racing circuits: a two stage DEA approach. *Operational research*. Vol. 20, No. 4, pp. 2059-2083

He, X. et al. (2021) AutoML: A survey of the state-of-the-art. *Knowledge-based-systems.* Vol. 212, pp. 106622

Henderson, D. & Kirrane, L. (2018) A Comparison of Truncated and Time-Weighted Plackett-Luce Models for Probabilistic Forecasting of Formula One Results. *Bayesian Anal.* Vol. 13, No. 2, pp. 335-358

Hutter, F. et al. (2019) *Automated Machine Learning Methods, Systems, Challenges*. 1st ed. Cham, Switzerland: Springe.

IBM (2023a) What is supervised learning? [Verkkodokumentti]. [Viitattu 7.3.2023]. Saatavilla: https://www.ibm.com/topics/supervised-learning

IBM (2023b) What is unsupervised learning? [Verkkodokumentti]. [Viitattu 7.3.2023]. Saatavilla: https://www.ibm.com/topics/unsupervised-learning

James, G., Witten, D., Hastie, T. & Tibshirani, R. (2021) *An introduction to statistical learning: with applications in R. 2nd edition.* New York, New York: Springer

Jenkins, M. (2010) Technological Discontinuities and Competitive Advantage: A Historical Perspective on Formula 1 Motor Racing 1950-2006. *Journal of management studies*. Vol. 47, No. 5, pp. 884–910

Jo, T. (2021) *Machine learning foundations: supervised, unsupervised and advanced learning*. 1st edition. Cham, Switzerland: Springer.

Karmaker, S. K. et al. (2022) AutoML to Date and Beyond: Challenges and Opportunities. *ACM computing surveys.* Vol 54, No. 8, pp.1-36

Keertish K. M. & Preethi, N. (2023) Formula One Race Analysis Using Machine Learning. *Proceedings of 3rd International Conference on Recent Trends in Machine Learning.* Singapore: Springer Nature Singapore.

Louridas, P. & Ebert, C. (2016) Machine Learning. *IEEE software*. Vol. 33, No. 5, pp. 110–115

MathWorks (2023a) Supervised Learning [Verkkodokumentti]. [Viitattu 9.4.2023]. Saatavilla: https://au.mathworks.com/discovery/supervised-learning.html

MathWorks (2023b) Unsupervised Learning [Verkkodokumentti]. [Viitattu 9.4.2023]. Saatavilla: https://au.mathworks.com/discovery/unsupervised-learning.html

MathWorks (2023c) What Is Machine Learning? [Verkkodokumentti]. [Viitattu 13.2.2023]. Saatavilla: https://www.mathworks.com/discovery/machine-learning.html

Muehlbauer, T. (2010) Relationship between Starting and Finishing Positions in Formula One Car Races. *International journal performance analysis in sport.* Vol. 10, No. 2, pp. 98-102

Olson, R. et al. (2016) Evaluation of a Tree-based Pipeline Optimization Tool for Automating Data Sciene. *GECCO '16: Proceedings of the Genetic and Evolutionary Computation Conference 2016*. Pp. 485-492

Olson, R. & Moore, J. (2016) TPOT: A Tree-based Pipeline Optimization Tool for Automating Machine Learning. *Proceedings of the Workshop on Automatic Machine Learning, PMLR.* Vol. 64, pp. 66-74

Patil, A. et al. (2023) A Data-Driven Analysis of Formula 1 Car Races Outcome. *Artificial Intelligence and Cognitive Science: 30th Irish Conference, AICS 2022, Munster, Ireland, December 8–9, 2022, Revised Selected Papers*. Cham, Swizerland: Springer

Pranay, D. (2022) How to Ace the missing value replacement problem [Verkkodokumentti]. [Viitattu 17.5.2023]. Saatavilla: https://towardsdatascience.com/how-to-ace-the-missing-value-replacement-problem-408bd9506b11

Priyadarshini, I. & Cotton, C. (2021) A novel LSTM-CNN-gris search-based deep neural network for sentiment analysis. *The Journal of supercomputing.* Vol. 77, No. 12, pp. 13911-13932

Rafatirad, S., Homayoun, H., Chen, Z. & Dinakarrao, S. (2022) *Machine Learning for Computer Scientists and Data Analysts.* Cham, Switzerland: Springer.

Raschka, S. & Mirajalili, V. (2017) *Python machine learning machine learning and deep learning with Python, scikit-learn, and TensorFlow*. 2nd edition, fully revised and updated. Birmingham, England: Packt.

Romano, J. D. et al. (2021) TPOT-NN: augmenting tree-based automated machine learning with neural network estimators. *Genetic programming and evolvable machines*. Vol. 22, No. 2, pp. 207–227

Schäfer, D. & Hüllermeier, E. (2018) Dyad ranking using Plackett-Luce models based on joint feature representations. *Machine learning.* Vol. 107, No. 5, pp. 903-941

Sicoie, H. (2022) *Machine Learning Framework for Formula 1 Race Winner and Championship Standigs Predictor* [Bachelor's Thesis – Tilburg University] [Viitattu 10.4.2023]. Saatavissa: http://arno.uvt.nl/show.cgi?fid=157635

theOehrly (2023) FastF1 documentation [Verkkodokumentti]. [Viitattu 10.4.2023]. Saatavilla: https://theoehrly.github.io/Fast-F1/index.html

TPOT (2023) Using TPOT [Verkkodokumentti]. [Viitattu 13.4.2023]. Saatavilla: http://epistasislab.github.io/tpot/using/

van Kesteren, E. J., & Bergkamp, T. (2022). Bayesian Analysis of Formula One Race Results: Disentangling Driver Skill and Constructor Advantage. *arXiv e-prints*, arXiv-2203.

Wan, J., Yu, X. & Guo, Q. (2019) LPI Radar Waveform Recognition Based on CNN and TPOT. Symmetry (Basel). Vol. 11, No. 5, pp. 725.

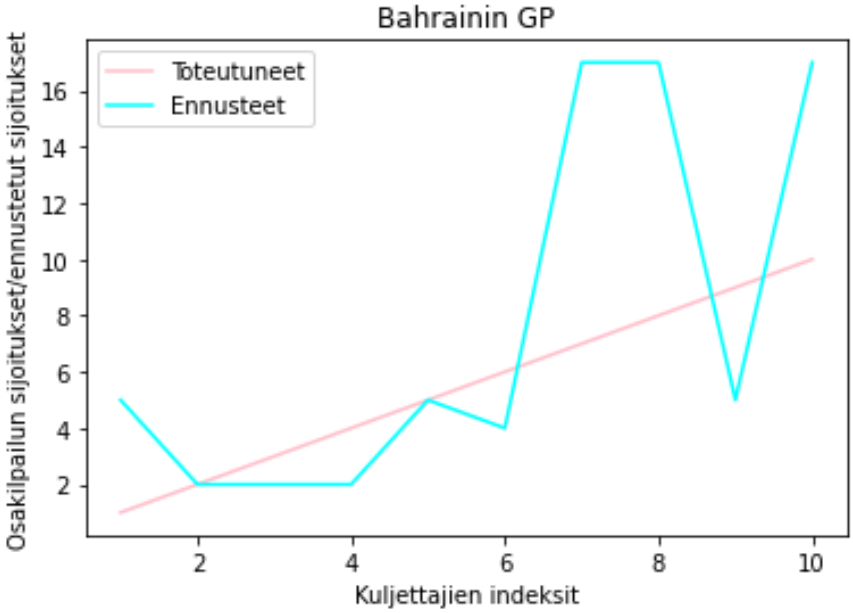
Waring, J., Lindvall, C., & Umeton, R. (2020) Automated machine learning: Review of the state-of-art and opportunities for healthcare. *Artificial Intelligence in Medicine*. Vol. 104, pp. 101822

Yakovlev, A. et al. (2020) Oracle AutoML: a fast and predictive AutoML pipeline. *Proceedings of the VLDB Endowment.* Vol. 13, No. 12, pp. 3166-3180

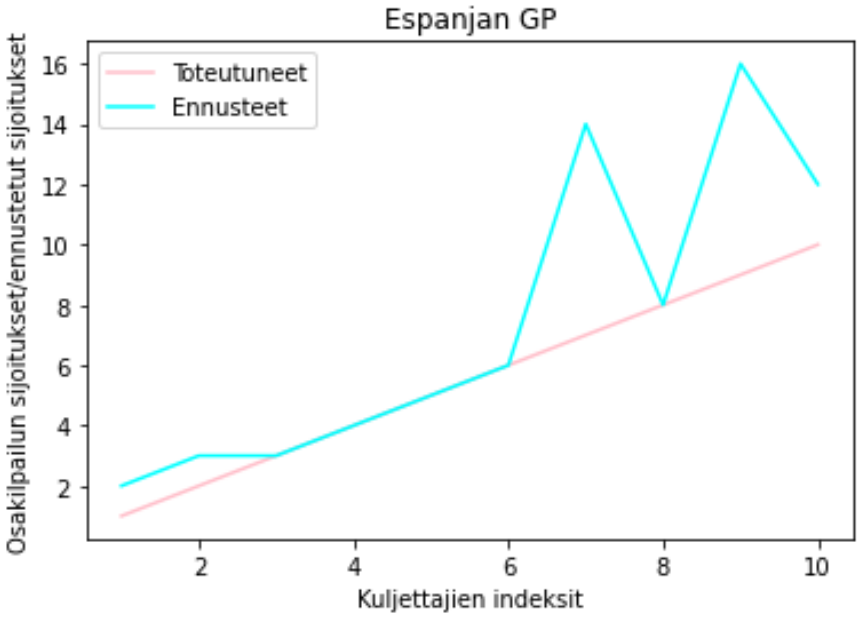
Zhang, P. et al. (2018) Machine Learning Topological Invariants with Neural Networks. *Physical review letters*. Vol. 120, No. 6, pp. 066401–066401

Liitteet

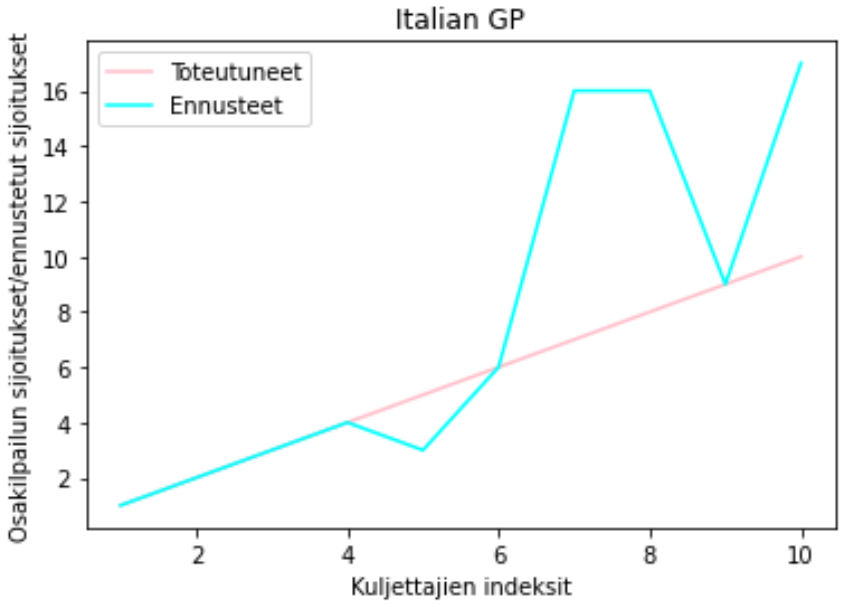
Liite 1. Bahrainin osakilpailun ennustetut sijoitukset



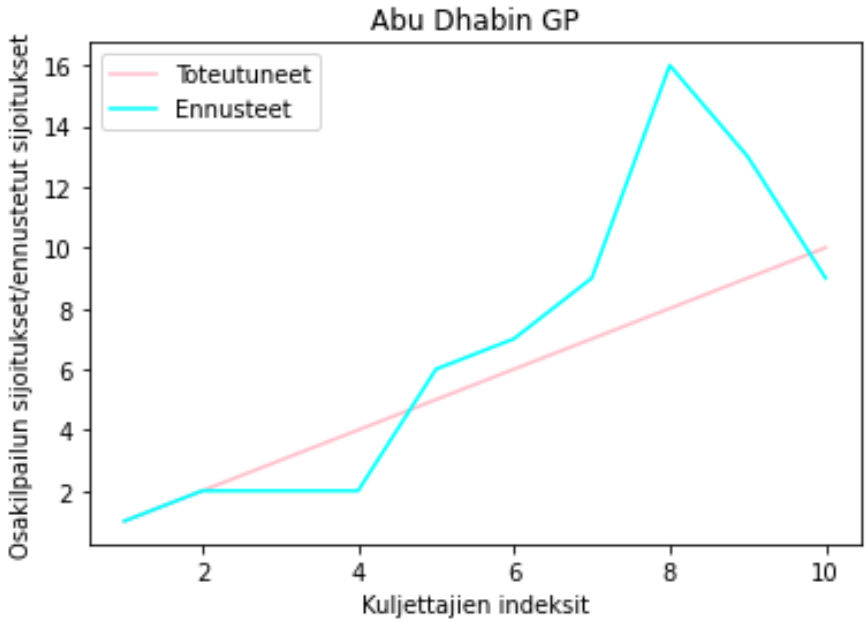
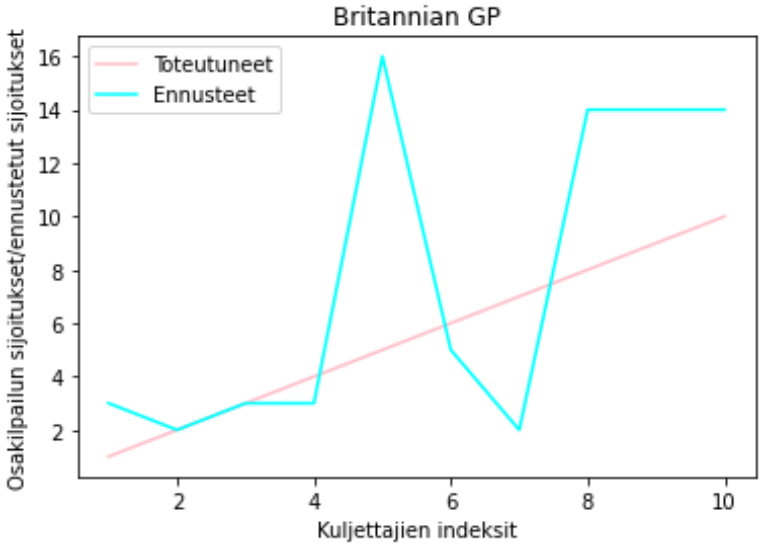
Liite 2. Espanjan osakilpailun ennustetut sijoitukset



Liite 3. Italian osakilpailun ennustetut sijoitukset



Liite 4. Abu Dhabin osakilpailun ennustetut sijoitukset



Liite 5. Britannian osakilpailun ennustetut sijoitukset