Koneoppimisen soveltaminen urheilutulosten ennustamisessa

Turun yliopisto
Tietotekniikan laitos
TkK-tutkielma
Tietotekniikka
Kesäkuu 2025
Veli-Matti Rissanen

TURUN YLIOPISTO
Tietotekniikan laitos

Veli-Matti Rissanen: Koneoppimisen soveltaminen urheilutulosten ennustamisessa

TkK-tutkielma, 20 s.
Tietotekniikka
Kesäkuu 2025

Tässä tutkielmassa selvitetään, kuinka koneoppimismalleja voidaan soveltaa urheilutulosten ennustamiseen joukkueurheiluissa. Tavoitteena on selvittää, kykenevätkö koneoppimismallit tuottamaan ennusteita, joita voisi käytännössä hyödyntää. Tutkielmassa tarkastellaan päätöspuupohjaisia malleja (esim. satunnaismetsä ja XGBoost), logistista regressiota, neuroverkkoja ja tukivektorikonetta. Malleja arvioidaan yleisesti käytetyillä mittareilla: tarkkuus, täsmällisyys, herkkyys, F1-arvo ja neliöllinen keskimääräinen virhe (RMSE). Tutkielman lopussa myös vertaillaan mallien suoriutumista ja selvitetään, mitkä mallit ovat antaneet tarkimmat ennustukset. Tulosten perusteella koneoppimismallien avulla voidaan ennustaa tarkasti. Ennustusten tarkkuus rippuu kuitenkin vahvasti aineistosta ja käytettävistä muuttujista, mikä korostaa näiden huolellisen valinnan tärkeyttä.

Asiasanat: koneoppiminen, ennustaminen, päätöspuut, logistinen regressio, neuroverkko, jalkapallo, koripallo

Sisällys

1 Johdanto

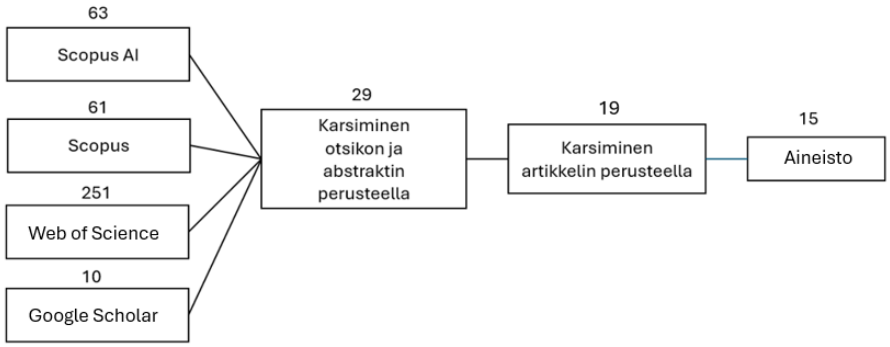
Urheilutulosten ennustamista on tutkittu jo pitkään, sillä tarkkojen ennusteiden avulla olisi mahdollista tienata suuria rahamääriä vedonlyönnillä, parantaa joukkuetta datan perusteella ja jopa keksiä parhaita mahdollisia taktiikoita peleihin. Tutkielmassa käsitellään, kuinka koneoppimista voidaan hyödyntää urheilutuloksien ennustamisessa. Tutkielmassa tarkastellaan useampaa urheilulajia, koska samoja koneoppimismalleja voidaan soveltaa samalla tavalla lähes jokaiseen eri lajiin. Tutkimuksessa selvitetään myös, millä muuttujilla malleja on käytetty tulosten saamiseksi ja miten tarkkoja tulokset ovat olleet. Tutkielmassa pyritään vastaamaan seuraaviin tutkimuskysymyksiin:

- **T**K1: Voiko koneoppimista hyödyntää urheilutulosten ennustamisessa?
- **T**K2: Onko tietty koneoppimismalli tehokkain ennustamiseen?

1.1 Tiedonhaku

Tietoa on haettu kolmesta eri tietokannasta: Scopus, Web of Science ja Google Scholar. Näistä saadut hakutulokset on rajattu ensimmäiseen sivuun. Scopus AI:n hakulauseet oli: ”Receiver Operating Characteristic”, ”how does random forest work?”, ”how does XGboost work?” ja ”how does logistic regression work?”. Scopuksen hakulause: ”machine learning” AND ”sports” AND ”predict”. Web of Sciencen hakulauseet: ”Extreme learning machine: Theory and applications” (highly cited papers),

”ROC curve” AND ”classification evaluation”, ”machine learning” and ”sports” AND ”outcome” AND ”random forest”, ”machine learning” and ”sports” AND ”algorithm” AND ”model” ja ”machine learning” and ”sports” and ”prediction”. Google Scholarin hakulause: ”data manipulation” AND ”pandas” AND ”Numpy” AND ”Scipy”. Kuvassa 1.1 on esitelty tulosten määrät ja kuinka niitä on karsittu. Kuvan luvut kertovat, kuinka paljon on jäänyt jäljelle karsinnan jälkeen.



Kuva 1.1: Tiedonhaku

1.2 Tutkielman rakenne

Tutkielma jakautuu kahteen pääosaan: koneoppimismalleihin ja niiden arviointiin sekä mallien ennustustarkkuuteen ja vertailuun. Ensimmäisessä osassa (luku 2) syvennymme siihen, kuinka mallien algoritmit toimivat ja kuinka niitä on käytetty. Toisessa osassa (luku 3) käsitellään tutkimuksissa käytettyjen mallien arviointeja, vertaillaan niitä keskenään sekä tarkastellaan yhtä kattavampaa tutkimusta, jossa mallien suoritusta on arvioitu laajemmin. Lisäksi tutkielman lopussa (luku 4) esitetään yhteenveto ja pohdintaa aiheesta.

2 Koneoppimismallit ja niiden arviointi

Luvussa tarkastellaan eri malleja, joita on käytetty tulosten ennustamisessa ja niiden pääpiirteitä. Nämä tutkimukset ovat tarkastelun kohteena läpi tutkielman. Luvussa tarkastellaan myös, kuinka malleja tullaan arvioimaan myöhemmin tutkielmassa.

2.1 Tarkasteltavat tutkimukset ja niiden mallit

Wong et al. [1] tekemässä tutkimuksessa ennustusten tekemiseen käytettiin viittä perusalgoritmia ja yhtä neuroverkkoa: logistista regressioanalyysia (LR, engl. Logistic regression), satunnaismetsiä (RF, engl. Random forest), tukivektorikonetta (SVM, engl. Support vector machine), äärimmäistä gradienttivahvistusmallia (XGB, engl. XGBoost) ja kevyttä gradienttivahvistusmallia (LGBM, engl. Light gradient boosting machine), sekä yhtä neuroverkkoa (CNN, engl. Convolutional neural network) [1].

Baratela et al. [2] tutkimuksessa käytettiin k-means-klusterointia yhdistämään samanlaiset näytteet toisiinsa tarkkuuden parantamiseksi. Tutkimuksessa myös käytettiin kolmea ohjatun oppimisen algoritmiä: LR:ää, RF:ää ja XGB:tä. Mallit on optimoitu hyperoptia käyttäen, joka on tarkoitettu hyperparametrien optimointiin ja kouluttamiseen käytetään 10-kertaista ristivalidointia. [2].

Xun [3] tekemässä tutkimuksessa on käytetty päätöspuupohjaista C4.5 ja RF al-

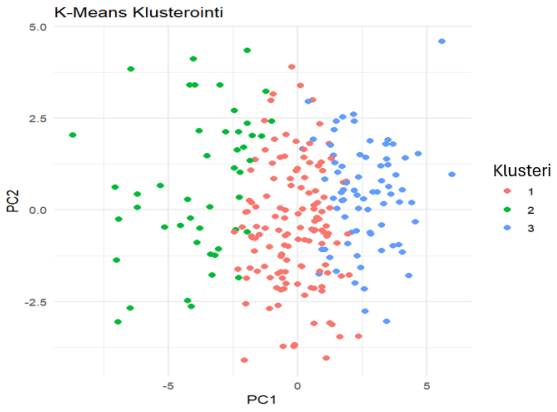
goritmeja. Näiden kanssa on käytetty niin sanottua ”bagging”-menetelmää mallien vakauttamiseksi ja tarkentamiseksi. Bagging-sana tulee englannin sanoista: Bootstrap aggregating, eli suomeksi bootstrap-agregaatti ja bagging on yhdistelmämalli, jonka tarkoituksena on kouluttaa yksittäisiä malleja käyttämällä satunnaisesti poimittuja otoksia opetusaineistosta [3]. Tämä menetelmä vähentää mallin varianssia [4].

Lu et al. [5] tutkimuksessa käytettiin päätöspuupohjaista luokittelu- ja regressiopuita (CART, engl. Classification and regression trees), RF:ää, stokastista gradienttitehostusta (SGB, engl. Stochastic gradient boosting), XGB:tä ja äärimmäistä oppimismallia (ELM, engl. Extreme learning machine) [5].

Monessa eri tutkimuksessa on ollut ainakin yksi päätöspuupohjainen malli, todennäköisesti sen takia, koska nämä mallit ovat vähemmän herkkiä ylisovitukselle ja koska urheilutulosten ennustamisessa voi tulla suuria määriä hyvin erilaista dataa, voi mallin saada helposti ylisovitettua.

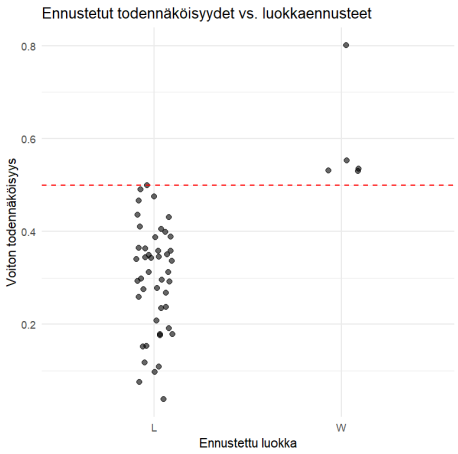
2.2 Mallien pääpiirteet

K-means-klusterointi ei varsinaisesti ole oma koneoppimismallinsa, mutta sitä käytetään usein näiden yhteydessä yksilöimällä havaintoja omiin "klustereihin", alueisiin. Siinä etsitään tarkin klusterimäärä kokeilun ja PCA-esikäsittelymenetelmän avulla, PCA-menetelmä on kuitenkin oma menetelmänsä, mutta sitä käytetään usein k-means-klusteroinnin kanssa. PCA-menetelmä yrittää vähentää datan ulottuvuuksia säilyttäen silti tärkeintä tietoa käyttämällä pääkomponentteja [2]. Kuvassa 2.1 näkyvssä klusteriryppäässä ei ole suoraan näkyvillä, että aineistossa olevat havainnot olisivat klusteroituneet selvästi, mutta siitä saa kuitenkin idean algoritmin toiminnasta.



Kuva 2.1: K-means-klusterointi aineistosta, joka koostuu 251 pelistä englannin valioliigasta. Klusterien määrä k=3 on asetettu ilman tarkempaa tutkimusta optimaalisesta klusterimäärästä.

LR algoritmi on lineaarinen malli ja se etsii yhteyksiä selittävien ja selitettävien muuttujien välillä, muuttaen näiden tuloksen todennäköisyydeksi [2]. Mallin vakio, eli perustaso, saadaan laskemalla lähtökohtaiset todennäköisyydet tietyn tapahtuman esiintymiselle ja puuttumiselle ilman selittäviä muuttujia. Kun tämä vaihe on tehty, lisätään riippumattomat muuttujat malliin ja lasketaan jokaiselle muuttujalle regressiokerroin, sekä p-arvo, joka kertoo muuttujan tilastollisen merkityksen [6]. Kuvassa 2.2 on visualisoitu, kuinka LR, jossa on käytetty useampaa muuttujaa ennustaisi onko jalkapallopeli häviö vai voitto. Mallin mukaan ennustukset, jotka sijoittuvat lähelle punaista katkoviivaa, ovat epävarmoja. Aineistossa oli käytetty vain 251 peliä ja tämän luvun kasvattaminen todennäköisesti vähentäisi mallin mukaan epävarmoja ennustuksia.



Kuva 2.2: Ennustukset 251 englannin valioliigan pelistä koostuvasta aineistosta, mallina käytetty LR.

ELM on neuroverkkoihin pohjautuva algoritmi, joka on kehitetty korjaamaan hitaasti koulutettavia neuroverkkoja. Gradientti-pohjaiset algoritmit, jota käytetään neuroverkkojen kouluttamiseen, sekä parametrien säätelyyn verkoissa, ovat hitaita. ELM on tehty ratkaisemaan tätä ongelmaa valitsemalla ennalta valittuja painoja ja piilokerroksen harhoja satunnaisesti, tämän tarkoituksena on eliminoida hitaita vaiheita optimoinnissa. Tällä keinolla ELM on jopa tuhansia kertoja nopeampi kuin perinteisemmät algoritmit [7].

RF on algoritmi, joka luo useita päätöspuita, joiden tuloksista se luo lopullisen ennustuksen [2]. RF ottaa näytteitä koulutusdatasta ja jokaiseen näytteeseen rakennetaan päätöspuita, jota ei ole vielä tässä kohtaa karsittu, joten kaikki puut ovat vielä tässä kohtaa maksimaalisen kokoisia [8]. Kun metsä on rakennettu, puut antavat oman keskiarvon ennustukseksi ja kun kaikki ovat antaneet ennustuksen, puiden enemmistöarvausta käytetään keskiarvona. Satunnaismetsä on myös vakaampi, sekä tarkempi kuin yksittäiset puut, sillä RF perustuu yhdistelmämenetelmään, joka korjaa epävakautta, jota pienet muutokset voivat aiheuttaa.

XGB on algoritmi, joka myös hyödyntää näitä päätöspuita, mutta hyödyntää erilaisia tehostusmenetelemiä (boosting) luodakseen painotettuja puita ennus-

tukseen [2]. XGB on paranneltu versio gradienttivahvistetusta päätöspuumallista (GBDT, engl. Gradient boosting decision tree), ja GBDT koostuu useista päätöspuista, jotka rakennetaan peräkkäin käyttäen gradienttilaskua optimointimenetelmänä. [9]. Kun puut on muodostettu, jokaisen puun avulla optimoidaan mallia minimoimalla tappiofunktiota. XGB eroaa tästä siten, että se käyttää automaattisesti tehokkaammin tietokoneen prosessorin säikeitä rinnakkaislaskennassa ja käyttämällä Taylorin toisen asteen approksimaatiota tappiofunktion laskemisessa [9].

LGBM on algoritmi, joka kapsuloi eri datatyyppejä säästääkseen muistinkäyttöä verrattuna tietorakenteisiin kuten Numpy tai Pandas [9]. NumPy ja pandas ovat kumpikin python-kirjastoja lukujen ja datan käsittelyyn. NumPy käyttää tietorakenteenaan ndarray, joka on on 50 kertaa nopeampi kuin tavallinen python lista ja Pandas käyttää tietorakenteenaan taulukkomuotoista tietotyyppiä nimeltä Dataframe [10]. LGBM käyttää tietorakenteenaan diskreettiä histogrammia, tämä histogrammi uhraa osan tarkkuudesta kouluttamisen nopeuttamiseksi ja muistin säästämiseksi. Tämän ansiosta LGBM pystyy käsittelemään suuria tietoaineistoja tehokkaammin kuin NumPy-pohjaiset mallit, kuten XGBoost. [9].

C4.5 algoritmi toimii siten, että algoritmi laskee tiedon hankintanopeutta kaikista testiatribuuteista data-aineistossa ja käyttää sitä atribuuttia jakomuuttujana, jonka hankintanopeus on suurin [11]. Päätöspuun rakentuminen onnistuu sitten iteroimalla tätä prosessia valmistumiseen asti [11]. Päätöspuun rakentamisen algoritmin kaava on seuraavanlainen:

$$Hankintasuhde(A) = \frac{Informaatio(D) - Informaatio_A(D)}{JakautumisInformaatio_A(D)} \tag{2.1}$$

jossa $Informaatio(D)$ on aineiston D entropia ja $Informaatio_A(D)$ on entropia datasetille D, kun se jaetaan attribuutin A arvojen mukaan.

$JakautumisInformaatio_A(D)$ lasketaan kaavalla $\sum_{j=1}^{y} p_i log_2 p_i$, jossa p_i tarkoittaa sitä osuutta datasta, jossa A arvo on j, tämä voidaan laskea kaavalla d_j/d [11].

Painokertoimet ovat Lu et al. tekemässä tutkimuksessa [5] laskettu käyttäen tutkimuksen omaa adaptiivista painomenetelmää, joka perustuu käänteisen etäisyyden menetelmään (IDW, engl. Inverse distance weighing). Tämä menetelmä painottaa siis ajallisesti lähempänä olevia pelejä enemmän [5]. Esimerkkinä tästä olisi 3 pelin viiveellä parametri $d = 2$ painottaisi viimeistä peliä eniten. Painokertoimien laskemiseen käytetty kaava on seuraava:

$$AW_{n,d}^{l} = \frac{(l - n + 1)^d}{\sum_1^l n^d} \tag{2.2}$$

jossa l on viiveiden määrä, n on pelin järjestysnumero ja d on painotussäädin

Kaavasta 2.2 saatua painokerrointa $AW_{n,d}^{l}$ käytetään uuden piirteen muodostamiseen seuraavan kaavan mukaisesti:

$$\overline{X}_{i,t}^{l,d} = \sum_{n=1}^{l} AW_{n,d}^{l} * X'_{i,t-n} \tag{2.3}$$

Missä $\overline{X}_{i,t}^{l,d}$ on painoitettu piirre, joka muodostuu joukkueen aiempien pelien piirteiden yhdistelmänä. Painotettua piirrettä, joka on laskettu kaavasta 2.3, käytetään mallin kouluttamiseen syötteenä, jolloin malli hyödyntää paremmin ajallisesti merkityksellistä pelihistoriaa. Painotettuja piirteitä ei kuitenkaan optimoida mallin koulutuksen aikana.

2.3 Mallien arviointi

Tyyppillisesti mallien luotettavuutta mitataan tarkkuudella (engl. accuracy), herkkyydellä (engl. recall), täsmällisyydellä (engl. precision), F1-arvolla ja AUC-arvolla (engl. Area under the curve). Wong ja Meng et al.[1][12] tekemissä tutkimuksissa on kerrottu, kuinka tarkkuus, herkkyys, täsmällisyys, F1-arvo ja AUC-arvo lasketaan.

Näiden arvojen laskemiseksi tarvitaan oikeiden positiivisten (OP), väärien positiivisten (VP), oikeiden negatiivisten (ON) ja väärien negatiivisten (VN) määrät.

- OP = Kotivoitto ennustettu oikein kotivoitoksi.
- VP = Vierasvoitto ennustettu virheellisesti kotivoitoksi.
- ON = Vierasvoitto ennustettu oikein vierasvoitoksi.
- VN = Kotivoitto ennustettu virheellisesti vierasvoitoksi.

AUC tarkoittaa vastaanotinoperaation ominaisuuskäyrän (ROC, engl. Receiver Operating Characteristic) alle jäävää aluetta. ROC-käyrä esittää kuvaajalla OP osuutta VP osuuteen eri kynnyksilä ja AUC-arvo on tämän mittari. AUC-arvo vaihtelee 0-1 välillä ja se arvioi mallin suorituskykyä [12].

Kun muuttujat ovat selvillä, voidaan laskea mittareiden arvot.

$$\text{Tarkkuus} = \frac{OP + ON}{OP + ON + VP + VN} \tag{2.4}$$

$$\text{täsmällisyys} = \frac{OP}{OP + VP} \tag{2.5}$$

$$\text{herkkyys} = \frac{OP}{OP + VN} \tag{2.6}$$

$$\text{F1-arvo} = \frac{2 * \text{täsmällisyys} * \text{herkkyys}}{\text{täsmällisyys} + \text{herkkyys}} \tag{2.7}$$

Tarkkuus tarkoittaa kuinka hyvin malli on onnistunut ennustamaan ottelun voittajan, eli oikeiden luokiteltujen tapausten osuutta kaikista tapauksista. Täsmällisyys

tarkoittaa kuinka hyvin malli on ennustanut halutun joukkueen voiton, eli oikeiden positiivisten ennusteiden osuutta kaikista positiivisista ennusteista. Herkkyys tarkoittaa kuinka hyvin malli on löytänyt oikeita voittoja, eli oikeiden positiivisten ennusteiden tapausten osuus kaikista oikeista positiivisista tapauksista. F1-arvo tarkoittaa täsmällisyyden ja herkkyyden yhdistettyä tulosta.

On myös tutkimuksia, joissa käytetään eri tapoja mallien luotettavuuden testaamiseen, kuten Lu et al. tekemässä tutkimuksessa [5], jossa malleja kokeillaan yleisesti käytetyllä menetelmällä: ristiinvalidoinnilla. Ristiinvalidointi toimii erillisenä testausmenetelmänä, joka on riippumaton muista mittareista. Tutkimuksessa käytetään 10-kertaista ristiinvalidointia 100 kertaa ja mallien tekemiä ennustuksia arvioidaan keskineliövirheellä (RMSE, engl. Root mean square deviation). Keskineliövirheen laskemisessa käytetään seuraavaa kaavaa:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n} e_i^2} \tag{2.8}$$

Jokaiselle mallille laskettiin RMSE eri peliviiveillä (3–6 peliä). Näihin peliviiveihin liittyvät painokertoimet määritettiin IDW-menetelmällä ennen mallin syötteiden luomista.

RMSE-arvo kertoo kuinka paljon ennustettu arvo eroaa todellisesta arvosta, joten jos yhden pelin todelliset pisteet olisivat 100 pistettä ja malli ennustaisi 115, se tarkottaisi, että RMSE olisi 15.

Monet tutkimuksissa käytetyt mallit käyttivät tarkkuutta yhtenä mallia arvioivana tekijänä, mutta Chicco ja Jurman [13] tehdyssä tutkimuksessa huomioidaan, että tarkkuus voi olla epäluotettava, sillä jos käytettävä aineisto on epätasapainossa, tarkoittaen sitä, että jos tietyn luokan muuttujia on enemmän, antaa tarkkuus liian optimistisen arvion mallista.

3 Mallien tarkkuus ja vertailu

Luvussa tarkastellaan luvussa 2.1 esiteltyjen tutkimusten ennustuskohteita ja niiden ennustustarkkuuksia hyödyntämällä luvussa 2.2 esiteltyjä arviointimittareita. Kappaleen lopussa myös vertaillaan tutkielmassa käytyjen tutkimusten tuloksia, sekä yhden laajemman tutkimuksen tuloksia.

3.1 Mallien suoriutuminen

3.1.1 Valioliiga-otteluiden lopputulosten ennustaminen

Wong et al. [1] tutkimuksen kohteena oli ennustaa voittajajoukkutta kuudella eri mallilla ja tuloksia arvioitiin neljällä eri mittarilla. Tutkimuksen mallien käyttämässä datasetissä oli englannin valioliigan kaudet 2019-2022 ja malleissa käytetyt muuttujat ovat esiteltynä taulukossa 3.1, jossa pelitilastot-kategorian muuttujat on otettu koti- ja vierasjoukkueilta.

Taulukko 3.1: Muuttujat ja niiden kategoriat

Kategoria	**Muuttujat**
Ottelutiedot	Pelin aloituksen aika, lopputulos
Pelitilastot	Kulmapotkut, virheet, keltaiset kortit, punaiset kortit viime pelit, viime pelien maalit, laukaukset, laukaukset maalia kohti
Sääolosuhteet	Kosteus, sateen määrä, tuulen suunta, tuulen nopeus, sää

Saman tutkimuksen malleissa opetusdatana käytettiin kahta ensimmäistä valioliigan kautta (2019-2021) ja loppuja, kauden 2022 pelejä käytettiin testaamisessa. Tutkimuksessa tutkittiin ennustuksia sekä sääolosuhteiden kanssa että ilman näitä. Taulukossa 3.2 on esitetty mallien arviointi eri mittareiden mukaan.

Taulukko 3.2: Mallien arviointia eri mittareilla, sääolosuhteilla, sekä ilman [1].

Ilman sääolosuhteita				
Malli	Tarkkuus	Täsmällisyys	Herkkyys	F1
LR	60.9	56.0	37.6	45.0
RF	59.7	53.9	36.9	43.8
SVM	62.9	55.4	65.8	60.1
XGB	60.3	57.8	24.8	34.7
LGBM	61.4	56.4	41.6	47.9
CNN	63.7	60.0	44.3	51.0

Sääolosuhteet mukana				
Malli	Tarkkuus	Täsmällisyys	Herkkyys	F1
LR	61.4	56.9	38.9	46.2
RF	62	58.5	36.9	45.3
SVM	63.1	56	62.4	59
XGB	60.3	58.3	23.5	33.5
LGBM	59.7	54	36.2	43.4
CNN	61.1	56.8	36.2	44.3

3.1.2 Tasapelien huomioiminen jalkapalloennusteissa

Baratela et al. [2] aineistona oli käytetty vuoden 2018 FIFA maailmanmestaruuskilpailua, 2016 UEFA Euroopanmestaruuskilpailua, sekä Espanjan, Italian, Saksan, Englannin ja Ranskan liigoista kaudet 2017-2018. Tutkimuksen kohteena oli ennustaa voittajajoukkuetta. Tutkimuksessa ennustettiin voittajajoukkuetta tasapelit

huomioiden, sekä ilman niitä. Tasapeliin päätyneitä pelejä ei käytetty, jos haluttiin ennustaa ilman tasapeliä ottelun voittajaa, jolloin käytettävä aineisto on yhteensä 1470 ottelua. Jos tasapelit otettiin ennustukseen mukaan, aineisto oli yhteensä 1941 ottelua. Malli on opetettu viiden pelin keskiarvolla taulukossa 3.3 esitetyillä muuttujilla (muuttujat otettu kummaltakin joukkueelta).

Taulukko 3.3: Muuttujat ja niiden kategoriat

Kategoria	**Muuttujat**
Ottelutiedot	Maalit, vierasjoukkueen maalit, pallonhallinta
Pelitilastot	Maalivahdin torjunnat, maalivahdin onnistuneet torjunnat, laukaukset, laukaukset maalia kohti, laukausten maalia kohti tarkkuus, syötöt (toiselle pelaajalle syöttö), onnistuneiden syöttöjen määrä, syöttöpisteet (maali tehty syötöstä), keltaiset kortit, punaiset kortit vierasjoukkueen laukaukset

Tutkimuksessa oli käytetty malleja LR, RF ja XGB, ja nämä mallit suoriutuivat kyseisillä muuttujilla taulukon 3.4 mukaisesti:

Taulukko 3.4: Mallien arviointia eri mittareilla, tasapeleillä, sekä ilman

Tasapeleillä				
Malli	Tarkkuus	Täsmällisyys	Herkkyys	F1
LR	49.28	40.91	43.54	38.98
RF	55.03	47.64	46.03	41.05
XGB	55.65	47.93	46.61	41.89
Ilman tasapelejä				
Malli	Tarkkuus	Täsmällisyys	Herkkyys	F1
LR	69.39	68.04	65.37	65.73
RF	71.44	69.01	66.49	66.92
XGB	66.18	71.13	63.47	63.08

3.1.3 NBA-piste-ennusteet aikaisempien pelien perusteella

Lu et al. tekemässä tutkimuksessa [5] mallien opetusdatana käytettiin NBA:n 2018-2019 kausia, jotka koostuivat yhteensä 2460 pelistä. NBA-kaudella 2018–2019 kotijoukkueet tekivät keskimäärin 110.4 pistettä ottelua kohden ja vierasjoukkueet 108.4 pistettä [14].

Tutkimuksessa mallit olivat tehneet ennustuksia neljällä eri painokertoimella, sekä neljällä eri peli-viiveellä. Painokertoimet olivat: d=0, d=1, d=2, d=3 ja viiveet: 3 peliä, 4 peliä, 5 peliä, 6 peliä. Näiden ennustusten RMSE-arvot löytyvät alla olevasta 3.5 taulukosta:

Taulukko 3.5: RMSE-arvot eri peli-viiveillä ja painokertoimilla [5].

3 Peli-viive				
Malli	d=0	d=1	d=2	d=3
CART	12.5396	12.2611	13.0188	12.8924
RF	12.4307	12.2159	12.6226	12.5646
SGB	12.4195	12.1671	12.4670	12.5261
XGB	12.3062	12.2736	12.6042	12.6124
ELM	12.5172	12.4846	13.0403	12.9621

4 Peli-viive				
Malli	d=0	d=1	d=2	d=3
CART	12.3434	11.7564	12.9753	12.4579
RF	11.9571	11.6303	12.6935	12.2696
SGB	12.0481	11.5586	12.6784	12.1120
XGB	12.0491	11.6941	12.6785	12.0791
ELM	12.2817	11.8020	12.9334	12.4511

5 Peli-viive				
Malli	d=0	d=1	d=2	d=3
CART	13.1326	12.4316	12.8013	13.0725
RF	12.6148	12.1525	12.5351	12.8432
SGB	12.6969	12.2448	12.6745	12.9750
XGB	12.7785	12.3145	12.6760	12.8648
ELM	13.0344	12.6748	13.0100	13.0235

6 Peli-viive				
Malli	d=0	d=1	d=2	d=3
CART	12.3738	12.0896	12.9175	12.4925
RF	12.3614	12.0245	12.9398	12.3072
SGB	12.1067	12.0293	12.9223	12.0876
XGB	12.1655	11.9898	12.8547	12.2934(
ELM	12.3923	12.2520	13.0727	12.3416

3.2 Mallien vertailu ja niiden loppukatsaus

3.2.1 Tutkielmassa käydyt tutkimukset

Kahdessa jalkapalloon liittyvässä tutkimuksessa Wong et al. [1] ja Baratela et al. [2] ennustettiin voittajaa jalkapallossa ja yhdessä tutkimuksessa Lu et al. [5] ennustettiin pistemäärää koripallossa. Kaikissa kolmessa tarkastellussa tutkimuksessa tehtiin ennustuksia käyttäen eri malleja. Painokertoimia ja RMSE-arvoa käyttävän tutkimuksen [5] tekemiä ennustusten tarkkuutta on hankala verrata muihin tutkimuksiin, sillä näissä on käytetty eri mittareita ennustusten arvioinnissa.

Koripallosta tehdyn tutkimuksen Lu et al. [5] parhaat tulokset löytyvät taulukosta 3.6.

Taulukko 3.6: Paras RMSE-arvo eri peli-viiveillä ja malleilla).

Viive (peliä)	Paras RMSE	Malli	Painokerroin
3	12.1671	SGB	1
4	11.5586	SGB	1
5	12.1525	RF	1
6	11.9898	XGB	1

Tämän perusteella SGB-malli painokerrointa 1 käyttäen on tehokkain tuottamaan ennustuksia.

Jalkapallosta tehty Wong et al. [1] tutkimus käytti 6 mallia ennustamisessa: LR, RF, SVM, XGB, LGBM ja CNN. Tutkimus Baratela et al. [2] käytti 3 mallia ennustamisessa: LR, RF ja XGB. Kumpikin tutkimus käytti samoja mittareita ennustusten testaamiseen: tarkkuutta, täsmällisyyttä, herkkyyttä ja F1-arvoa, mutta tässä tutkimuksessa oli ennustukset tehty käyttäen tasapelejä ja ilman, joten vertailun vuoksi ilman tasapelejä tehdyt ennustukset jätetään pois. Malleissa käytetty aineiston määrä sekä muuttujat erosivat hieman toisistaan, mutta ennustuksia voidaan

silti mittareiden avulla vertailla toisiinsa. Ennustukset otetaan kaikkien mittareiden arvojen keskiarvona ja paras ennustus oli tutkimuksen Wong et al. [1] käyttämä SVM, jonka mittarien keskiarvoksi tuli 60.125, ja tämä oli saatu käyttäen sääolosuhteita.

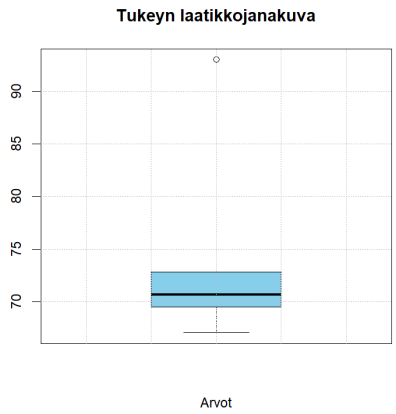
Horvat et al. [15] tehdyssä tutkimuksessa tarkasteltiin laajemmin tutkimuksia ja niiden käyttämiä malleja, muuttujia, tarkkuutta ja mallien testausta. Tutkimuksessa on tarkasteltu 36 eri tutkimusta, joista paras tarkkuus oli 93 prosenttia, joka oli saatu käyttämällä 9 muuttujaa, vuoden 2024 kautta englannin valioliigasta ja mallina LR. Huonoin tarkkuus oli ollut 55.52 prosenttia, joka oli saatu käyttämällä 164 muuttujaa, pohjoisamerikkalaisesta baseball-sarjasta (engl. Major League Baseball, MLB) vuosien 1930-2016 kausia ja mallina oli käytetty XGB:tä.

3.2.2 Kattava vertailututkimus koneoppimismalleista

Horvat et al. [15] käsittelemien tutkimusten tarkkuuden keskiarvo on 72.81 prosenttia, mediaani q(0.5) on 71.5, alakvartiili q(0.25) on 67.04 ja yläkvartaali q(0.75) on 81.37. Kvartiilit on laskettu perinteisellä menetelmällä siten, että ensin määritettiin mediaani ja sen jälkeen aineisto jaetaan kahteen osaan kvartiilien löytämiseksi.

LR osuus aineistosta. Näistä 36 tutkimuksesta 6 tutkimusta käytti LR:ää ja näiden tarkkuuden keskiarvo on 73.95 prosenttia, mediaani 70.65, sekä kvartaalit ovat q(0.25) on 69.5 ja q(0.75) on 72.8. LR keskiarvo menee hieman koko aineiston keskiarvon yli, mutta mallin parasta tarkkuutta (93 prosenttia) voidaan pitää poikkeamana, ainakin jos käytetään tukeyn laatikko-jana kuviota tämän tarkastelemiseen.

Poikkeamat voidaan löytää laskemalla $q(0.25) - 1.5q_r$ ja $q(0.75) + 1.5q_r$ joiden alapuolelle tai yläpuolelle jäävät havainnot luokitellaan poikkeamiksi. Tässä $q_r = q(0.75) - q(0.25)$ on kvartaaliväli. Tuloksista tulee ilmi, että aineiston ainoa selkeä poikkeus on 93 prosenttia.



Kuva 3.1: Tukeyn kaavio LR osuudesta aineistossa.

Bayes-verkon (BN, engl. Bayesian network) osuus aineistosta. Tutkimuksessa Horvat et al. [15] käytyjen 36 tutkimuksen osuudessa vain yksi oli tehty käyttäen BN-mallia. Tämän tarkkuus on 85.28, joka on paljon yli keskiarvon. Mutta koska tämä on ainoa BN, jota oli käytetty, mallin todellisesta keskiarvosta ei voida vielä tehdä varmoja johtopäätöksiä.

SVM osuus aineistosta. Tutkimuksessa Horvat et al. [15] käytyjen 36 tutkimuksen osuudessa neljä oli käyttänyt SVM-mallia. Näiden neljän tutkimuksen keskiarvo tarkkuudessa on 75.52 prosenttia. Tämän mukaan SVM-mallia käyttäen saisi koko aineiston keskiarvoa korkeamaan tarkkuuden. Vaikka 4 tutkimusta onkin melko vähän, se antaa jo selvästi suuntaa siihen, mihin SVM pystyy.

Päätöspuiden osuus aineistosta. Tutkimuksessa Horvat et al. [15] tarkastelemien 36 tutkimuksen osuudessa kuusi oli käyttänyt päätöspuupohjaista mallia. Näiden keskiarvo tarkkuudessa on 67.95, mediaani 66.98, sekä kvartaalit q(0.25) on 56.1 ja q(0.75) on 75.62. Päätöspuiden saama keskiarvo menee alle koko tuktimuksen keskiarvon.

Neuroverkkojen osuus aineistosta. Tutkimuksessa Horvat et al. [15] käytyjen 36 tutkimuksen osuudessa yhdeksän oli käyttänyt neuroverkkopohjaista mallia, ja yksi näistä oli tutkinut kahta urheilulajia, eli otoskooksi tulee yhteensä 10. Näi-

den keskiarvo tarkkuudessa on 72.25, mediaani 71.57m sekä kvartaalit q(0.25) on 68.3 ja q(0.75) on 78.6. Neuroverkot suoriutuvat siis paremmin, kuin tutkimuksen keskiarvo ja ainoa malli, joka suoriutuu paremmin on SVM. Neuroverkon keskiarvo on kuitenkin tarkempi kuin SVM-mallin, sillä niitä on käytetty paljon useammin ennusteita tehdessä.

Samassa tutkimuksessa Horvat et al. [15] havaitaan, että tutkimustulosten mediaani osoittaa jatkuvaa kasvua. Tämä voi johtua siitä, että aiheesta tehdään enemmän tutkimusta, sekä uudet tutkimukset käyttävät aiemmin tehtyjen tutkimusten tekemiä havaintoja omissaan, joka odotetusti kasvattaa tarkkuuksia.

Mikä on myös mielenkiintoista katsoessa samaa tutkimusta on se, että mitä enemmän kausia on käytetty mallejen kouluttamisessa, sitä huonompia ennustuksia malli on antanut. Tästä on vaikea keksiä suoria syitä huonolle menestykselle tietämättä tarkempia tietoja käytetyistä menetelmistä sekä malleista ja näitä ei tutkimuksessa käsitellä kovin tarkasti.

4 Pohdinta ja yhteenveto

Tässä työssä tarkasteltiin erilaisten koneoppimismallejen toimintaa ja suoriutumista urheilutulosten ennustamisessa eri kokoisilla aineistoilla ja erilaisilla muuttujilla. Työssä myös selvitettiin voiko koneoppimista hyödyntää urheilutulosten ennustamisessa (TK1), ja onko jokin tiety koneoppimismalli tehokkain ennustamiseen (TK2).

Tutkimuksessa tarkasteltiin kolmea aiempaa tutkimusta, joissa selvitettiin näissä käytettyjä menetelmiä ja malleja sekä niiden käyttötapoja. Lisäksi tutkittiin, millä mittareilla mallien ennustuksia arvioitiin, ja kuinka ne toimivat käytännössä. Näistä tutkimuksista koneoppimismalli, joka oli suoriutunut parhaiten oli SVM-malli, jonka mittareiden antamien arvioiden keskiarvona oli ollut 60.125 ja kattavammassa vertailututkimuksessa tarkkuuden keskiarvo oli ollut 75.52.

Tutkimuksessa tarkasteltiin myös laajemmin tehtyä tutkimusta, jossa oli tarkasteltu 36 eri tutkimusta aiheesta, josta kävi kävi ilmi, että malleista voi saada hyvinkin paljon tarkempia, kuin mitä tutkielmassa käydyt kolme tutkimusta olivat onnistuneet saamaan. SVM:n tarkkuuden keskiarvo oli ollut 73.95, joka on paljon korkeampi, kuin mitä tutkielman läpikäydyissä tutkimuksissa oli saatu. Tähän voi olla myös monta syytä, kuten aineisto, muuttujat, laji ja mallin optimointi.

Tutkielma osoitti sen, että koneoppimista voidaan hyödyntää urheilutulosten ennustamisessa, riippumatta lajista, sekä, että SVM on osoittautunut tarkimmaksi malliksi. Voi myös esittää, että BN olisi tarkempi, mutta pienestä otoskoosta johtuen, tällaista arviota ei voida pitää tarpeeksi luotettavana.

Tulevaisuudessa tullaan todennäköisesti näkemään entistä tarkempia ja luotettavampia ennustuksia koneoppimisalgoritmien kehittyessä, laskentatehon kasvaessa ja tehtyjen tutkimusten määrän kasvaessa.

Vaikka malleja ei voi täysin vertailla toisiinsa tietämättä, miten ne on koulutettu, tutkimus antaa silti esimerkin niiden toiminnasta ja odotetuista ennustuksista, mutta esitetyt tulokset ovat luonteeltaan suuntaa antavia. Tutkimuksessa käytyjen asioiden pohjalta olisi siis kohtalaisen helppo lähteä kokeilemaan itse ennustusten toistamista, sekä näiden kehittämistä.

Luvussa 3.1 esitetyistä mallien enusteista käy ilmi, kuinka eri muuttujat vaikuttavat mallien suoriutumiseen, kuten ilman sääolosuhteita RF-malli oli saanut tarkkudessa vain 59.7, mutta kun sääolosuhteet otettiin mukaan, nousi tarkkuus 2.2 prosenttiyksikköä, kun taas CNN-mallin tarkkuus laski 2.6 prosenttiyksikköä, kun otettiin sääolosuhteet mukaan.

Uutta tutkimusta tulee jatkuvasti lisää ja näiden ennustusten laatu paranee vuosittain uusien algoritmien ja niiden optimoinnin kehityksen myötä.

Lähdeluettelo

[1] A. Wong, E. Li, H. Le, G. Bhangu ja S. Bhatia, "A predictive analytics framework for forecasting soccer match outcomes using machine learning models", *Decision Analytics Journal*, vol. 14, s. 100 537, maaliskuu 2025, ISSN: 2772-6622. DOI: 10.1016/J.DAJOUR.2024.100537.

[2] E. A. Baratela, F. J. Xavier, T. Peron, P. Ribeiro Villas-Boas ja F. A. Rodrigues, "Predicting soccer matches with complex networks and machine learning", en, *J. Complex Netw.*, vol. 12, nro 6, lokakuu 2024.

[3] H. Xu, "Prediction on Bundesliga Games Based on Decision Tree Algorithm", teoksessa *2021 IEEE 2nd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering, ICBAIE 2021*, Institute of Electrical ja Electronics Engineers Inc., maaliskuu 2021, s. 234–238, ISBN: 9780738131221. DOI: 10.1109/ICBAIE52039.2021.9389986.

[4] G. Ngo, R. Beard ja R. Chandra, "Evolutionary bagging for ensemble learning", *Neurocomputing*, vol. 510, s. 1–14, lokakuu 2022, ISSN: 18728286. DOI: 10.1016/j.neucom.2022.08.055.

[5] C. J. Lu, T. S. Lee, C. C. Wang ja W. J. Chen, "Improving sports outcome prediction process using integrating adaptive weighted features and machine learning techniques", *Processes*, vol. 9, 9 syyskuu 2021, ISSN: 22279717. DOI: 10.3390/pr9091563.

[6] P. Ranganathan, C. Pramesh ja R. Aggarwal, "Common pitfalls in statistical analysis: Logistic regression", *Perspectives in Clinical Research*, vol. 8, s. 148–151, 3 heinäkuu 2017, ISSN: 22295488. DOI: 10.4103/picr.PICR_87_17.

[7] G. B. Huang, Q. Y. Zhu ja C. K. Siew, "Extreme learning machine: Theory and applications", *Neurocomputing*, vol. 70, s. 489–501, 1-3 joulukuu 2006, ISSN: 09252312. DOI: 10.1016/j.neucom.2005.12.126.

[8] Y. Akhiat, Y. Manzali, M. Chahhou ja A. Zinedine, "A New Noisy Random Forest Based Method for Feature Selection", *Cybernetics and Information Technologies*, vol. 21, s. 10–28, 2 kesäkuu 2021, ISSN: 13144081. DOI: 10.2478/cait-2021-0016.

[9] Y. Liang, J. Wu, W. Wang et al., "Product marketing prediction based on XGboost and LightGBM algorithm", teoksessa *ACM International Conference Proceeding Series*, Association for Computing Machinery, elokuu 2019, s. 150–153, ISBN: 9781450372299. DOI: 10.1145/3357254.3357290.

[10] A. Sapre ja S. Vartak, "Scientific Computing and Data Analysis using NumPy and Pandas", *International Research Journal of Engineering and Technology*, 2020, ISSN: 2395-0056. url: www.irjet.net.

[11] Z. Yin ja W. Cui, "Outlier data mining model for sports data analysis", *Journal of Intelligent and Fuzzy Systems*, vol. 40, s. 2733–2742, 2 2021, ISSN: 18758967. DOI: 10.3233/JIFS-189315.

[12] X. Meng, "Soccer match outcome prediction with random forest and gradient boosting models", *Applied and Computational Engineering*, vol. 40, s. 99–107, 1 helmikuu 2024, ISSN: 2755-2721. DOI: 10.54254/2755-2721/40/20230634.

[13] D. Chicco ja G. Jurman, "The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation",

BMC Genomics, vol. 21, 1 tammikuu 2020, ISSN: 14712164. DOI: 10.1186/s12864-019-6413-7.

[14] Basketball-Reference.com. "Basketball Statistics and History". (2025), url: https://www.basketball-reference.com/leagues/NBA_2019.html (viitattu 11. 06. 2025).

[15] T. Horvat ja J. Job, "The use of machine learning in sport outcome prediction: A review", *WIREs Data Mining and Knowledge Discovery*, vol. 10, nro 5, kesäkuu 2020, ISSN: 1942-4795. DOI: 10.1002/widm.1380.