



# **Koneoppivan mallin rakentaminen tuulivoimaloiden jäätämisen ennustamiseen**

Ammattikorkeakoulututkinnon opinnäytetyö

Tietojenkäsittelyn koulutus

Syksy 2024

Paula Tyrväinen

---

Opinnäytetyön tavoitteena oli rakentaa koneoppiva malli tuulivoimaloiden jäätämisen ennustamiseen sekä käsitellä koneoppivan mallin rakennusprosessia. Jäätämisen tarkoittaa jään kertymistä esineen pinnoille ja se aiheuttaa tuulivoimaloille erilaisia haasteita. Työn toimeksiantaja pysyy anonymina, jotta asiakkuudet ja niihin liittyvä data pysyvät myös paremmin anonymina. Opinnäytetyö on luonteeltaan toiminnallinen.

Opinnäytetyön teoriaosiossa perehdytään aluksi Suomen tuulivoimaan yleisesti sekä jäätämiseen ja sen vaikutuksiin tuulivoimaloissa. Työssä esitellään myös aiempia tutkimuksia jäätämisen ennustamiseen liittyen. Teoriaosan pääpaino on kuitenkin koneoppimisessa ja mallin rakentamisen eri vaiheissa. Koneoppimiseen liittyen aiheina käsitellään neuroverkkoja, tukivektorikonetta, logistista regressiota sekä ARIMAA. Teoriaosuus on koottu mahdollisimman tuoreista kirjallisuuslähteistä ja luotettavilta verkkosivustoilta.

Työn toiminnallisessa osiossa analysoitiin ensimmäiseksi tuulipuistoista saatua aineistoa Jupyter Notebookissa. Tutkittavia tuulipuistoja oli kuusi ja näistä jokaisesta tehtiin oma tiedosto. Aineistoa esikäsiteltiin ja luotiin tarvittavia lisämuuttujia. Aineistoon tuotiin lisäksi säähavaintoja Ilmatieteen laitoksen havaintoasemilta, jotka sijaittivat lähimpänä kutakin tuulipuistoa. Malleja rakennettiin ja testitiin käyttäen neuroverkkoja, tukivektorikonetta, logistinen regressio ja ARIMA.

Tuloksien perusteella ARIMA vaikutti olevan malleista toimivin lyhyen aikavälin ennustamiseen, ja neuroverkot taas soveltuivat parhaiten sääennusteen pohjalta pitkän aikavälin ennustamiseen. Yhtä kaikille testiaineiston kohteille sopivaa sisärakennetta ei löytynyt ja tuloksien pohjalta jokaiselle puistolle tulee optimoida oma sisäinen rakenne ARIMAAan ja neuroverkkohiin. Työn lopussa on suunniteltu ohjelmiston rakenne ennustamiseen.

Johtopäätöksenä voidaan havaita, että jäätämisen ennustaminen on haastavaa varsinkin pitkälle tulevaisuuteen, lyhyen aikavälin ennusteet saadaan toimimaan luotettavammin.

The aim of the thesis was to build a machine learning model for predicting the icing of wind turbines. Icing means the accumulation of ice on the surfaces of an object, and it causes different challenges for wind turbines. The commissioner of the work remains anonymous, therefore that their customers and their related data also remain more anonymous. The thesis is functional in nature.

In the theory section of the thesis, wind power in Finland will first be discussed on a general level, along with icing and its effects on wind turbines. The work also presents previous studies related to the prediction of icing. However, the focus of the theory part is on machine learning and the different stages of building a model. Topics related to machine learning include neural networks, support vector machine, logistic regression and ARIMA. The theory part has been compiled from the most recent literary sources and reliable websites.

In the functional part of the work, the material obtained from the wind farms was first analyzed in Jupyter Notebook. There were six wind farms to be investigated and a separate file was created for each of them. The data was preprocessed, and necessary additional variables were created. In addition, weather observations from the Finnish Meteorological Institute's observation stations, which were located closest to each wind farm, were added to the data. Models were built and tested using neural networks, support vector machine, logistic regression and ARIMA.

Based on the results, ARIMA seemed to be the most effective of the models for near-term forecasting, and neural networks were best suited for forecasting based on the weather forecast. One suitable internal structure for all test data objects was not found, and based on the results, each park must optimize its own internal structure for ARIMA and neural networks. At the end of the work, the structure of the software for forecasting is planned.

As a conclusion, it can be observed that forecasting icing is challenging, especially far into the future, near-term forecasts can be made to work more reliably.

Keywords Icing, wind power, machine learning, neural networks

Pages 35 pages and appendices 1 page

## Sanasto

Aikasarja	Ajallisesti järjestettyjen havaintojen joukko.
CRON	Ajastus ohjelmisto, jota käytetään Linuxissa.
Jupyter Notebook	Avoimen lähdekoodin verkkosovellus, jonka tiedostotyyppi on IPYNB.
Klusterointi	Datan ryhmittely samankaltaisten ominaisuuksien perusteella.
Koneoppiminen	Tekoälyn muoto, jossa kone oppii.
Neuroverkot	Laskennallisia malleja.
NumPy	Python-kirjasto, joka tukee suuria taulukoita ja matriiseja.
Matplotlib	Python-kirjasto, graafien ja kuvioiden piirtämiseen.
Pandas	Python-kirjasto, jota käytetään datasettien käsittelyyn.
Python	Yleisesti käytetty ohjelmointikieli.
Regressio	Prosessi, jolla arvioidaan syöte- ja tulosmuuttujien välistä suhdetta.
REMIT	Euroopan parlamentin ja neuvoston asetus (EU) N:o 1227/2011, energian tukkumarkkinoiden eheydestä ja tarkasteltavuudesta.
(S)ARIMA	(Seasonal) Autoregressive Integrated Moving Average = (Kausittainen) autoregressiivinen integroitu liukuva keskiarvo.
Scikit-learn	Avoimen lähdekoodin koneoppimiskirjasto Python-ohjelmointikielelle.
Seaborn	Visualisointikirjasto, joka perustuu Matplotlibiin.
Tukivektorikone	Koneoppimismenetelmä, jota käytetään luokitteluun ja regressioihin.
Visual Studio Code	Ohjelmointiin tarkoitettu koodieditori.

# Sisällys

1	Johdanto .....	1
2	Tutkimusasetelma .....	2
3	Tuulivoima ja jäätäminen .....	3
4	Koneoppiminen .....	5
4.1	Logistinen regressio .....	6
4.2	Tukivektorikone .....	7
4.3	Neuroverkot .....	8
4.4	ARIMA-aikasarjamalli .....	10
5	Mallin rakentamisen vaiheet .....	11
5.1	Ongelman rajaus ja datan hankkiminen .....	11
5.2	Tutustuminen dataan .....	12
5.3	Aineiston esikäsittely .....	12
5.4	Mallien tutkiminen ja parhaimpien valinta .....	13
5.5	Järjestelmän hienosäätäminen .....	13
6	Jäätämisen ennustaminen .....	14
6.1	Menetelmät .....	14
6.2	Mallin määrittely ja tavoitteet .....	14
6.3	Kuvaus ympäristöstä ja käytetystä aineistosta .....	15
6.4	Aineiston esikäsittely .....	16
6.5	Havainnot esikäsittelyn jälkeen .....	18
6.6	Menetelmien testaus .....	20
6.6.1	Neuroverkot .....	22
6.6.2	Tukivektorikone .....	23
6.6.3	Logistinen regressio .....	25
6.6.4	ARIMA .....	26
6.7	Mallin valinta .....	28
6.8	Parametrien optimointi ja tulokset .....	29
6.9	Ohjelmiston rakennesuunnitelma .....	30
7	Johtopäätökset ja pohdinta .....	32
	Lähteet .....	34

## Kuvat, taulukot ja kaavat

Kuva 1. Ajatuskartta.....	2
Kuva 2. Koneoppimisen jako.....	5
Kuva 3. Logistinen regressio.....	7
Kuva 4. Tukivektorikone.....	8
Kuva 5. Neuroverkot.....	9
Kuva 6. Löydetyt jäätämistapahtumat tuulipuistoittain (1 tapahtuma = 15 min) .....	18
Kuva 7. Kosteuden ja lämpötilan vaikutus tehohäviöihin .....	19
Kuva 8. Esimerkki rullaavista arvoista.....	21
Kuva 9. Neuroverkkojen parhaimmat mallit tuulipuistoittain lyhyen aikavälin ennustamiseen .....	23
Kuva 10. Parhaimmat tukivektorikonemallit tuulipuistoittain lyhyen aikavälin ennustamiseen .....	24
Kuva 11. Parhaimmat logistisen regression mallit tuulipuistoittain lyhyen aikavälin ennustamiseen .....	25
Kuva 12. Parhaimmat ARIMA-mallit tuulipuistoittain lyhyen aikavälin ennustamiseen.	27
Kuva 13. Tuulipuistojen jäätämisen ennustamisen osumatarkkuudet malleittain lyhyelle aikavälille .....	28
Kuva 14. Ennusteohjelmiston rakenne.....	31
Taulukko 1. Testitapaukset .....	21

Taulukko 2. Tuulipuistojen parhaimmat jäätämisenustemallit, parametrit ja osumatarkkuus lyhyelle aikavälille ..... 29

Kaava 1. Menetetyn tehon laskenta ..... 16

Kaava 2. Tuulivoiman fysikaalinen malli..... 16

Kaava 3. Käytettävissä olevan tehon arvio..... 17

## **Liitteet**

Liite 1. Aineistonhallintasuunnitelma

# 1 Johdanto

Tuulivoima on noussut merkittäväksi energianlähteeksi Suomessa. Jäätäminen tarkoittaa jään kertymistä jonkin esineen pinnalle kuten tuulivoimaloiden tapauksissa niiden lapoihin. Se aiheuttaa monia haasteita tuulivoimaloille. Jäätäminen voi esimerkiksi pudottaa voimalaitoksen tuotantokyvyn pahimmillaan kokonaan. Tällaista jäätämisen aiheuttamaa tuotantokykyyn vaikuttavaa tietoa pidetään sisäpiiritietona, jonka väärinkäyttö on kielletty. Jotta vältetään vahingossa väärinkäyttämästä jäätämistietoa, on REMIT-asetuksen vuoksi jäätämisen oikea aikainen tunnistaminen ja ennustaminen tärkeää. Ennustaminen ja tunnistaminen mahdollistavat markkinailmoitusten (UMM) tekemisen hyvissä ajoin, jolloin sisäpiiritiedon väärinkäytön mahdollisuus poistuu.

Tässä opinnäytetyössä tätä aihetta lähestytään koneoppimisen näkökulmasta. Opinnäytetyön tavoitteena on rakentaa toimeksiantajan käyttökeskukseen malli, jolla voidaan ennustaa annettujen sääparametrien pohjalta yksinkertaisesti onko olosuhteissa jäätäminen todennäköistä vai ei. Työssä ei niinkään keskitytä jäätämisen fysiikkaan eikä myöskään tuulivoimaloiden toimintaan. Pääosassa on koneoppiminen ja mallin rakentaminen.

Työn tietoperusta, joka on koottu pääosin kirjallisuuslähteistä, muodostuu seuraavista luvuista: tuulivoima ja jäätäminen, koneoppiminen ja mallin rakentamisen vaiheet. Työssä käsitellään koneoppimista yleisellä tasolla ja esitellään koneoppivista menetelmistä logistinen regressio, tukivektorikone, neuroverkot ja ARIMA. Opinnäytetyön käytännön osa eli mallin rakentaminen alkaa kuudennesta luvusta. Lopussa on johtopäätökset ja pohdinta, jossa esitetään muun muassa jatkotutkimus mahdollisuuksia.

Seuraavassa luvussa esitellään tarkemmin opinnäytetyön tutkimusasetelma. Työn päätutkimuskysymys on:

- Miten rakennetaan malli jäätämisen ennustamiseen?

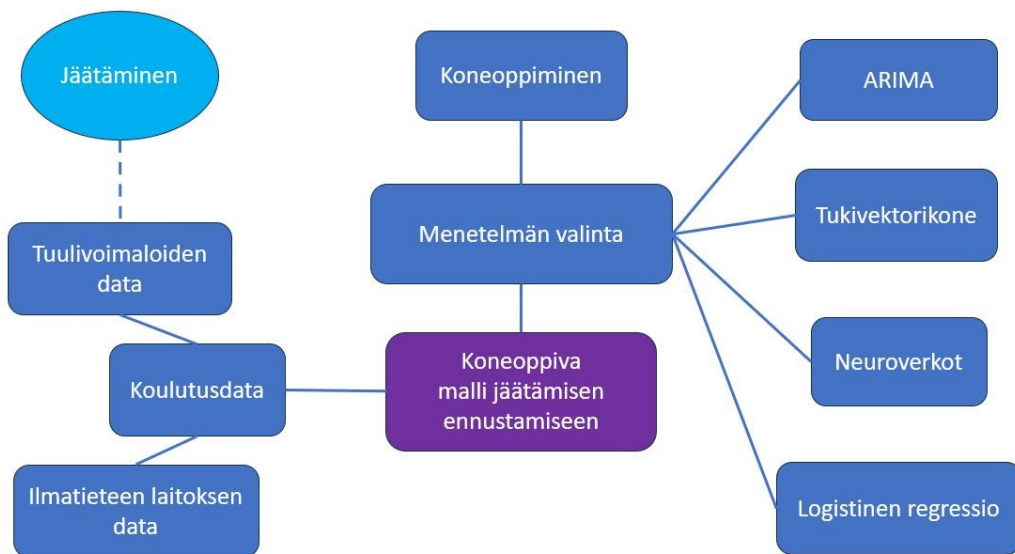
Muut tutkimuskysymykset:

- Mitkä koneoppivat metodit soveltuvat ennustamiseen?
- Mitä vaiheita kuuluu mallin rakentamiseen?
- Mitä jäätäminen on ja miten se vaikuttaa tuulivoimaloihin?

## 2 Tutkimusasetelma

Tämä opinnäytetyö on luonteeltaan toiminnallinen, koska sen päätavoitteena on rakentaa koneoppiva malli, jolla ennustetaan jäätämistä. Työn sisällön selkiyttämiseksi luotiin ajatuskartta, joka näkyy kuvassa 1. Työssä perehdytään koneoppimiseen yleisellä tasolla ja käydään läpi seuraavia menetelmiä: tukivektorikone, logistinen regressio, neuroverkot ja aikasarjamalleista ARIMA.

Kuva 1. Ajatuskartta



Koneoppivan mallin rakentamiseen tarvitaan koulutusdataa, jota saadaan toimeksiantajan käyttämistä tuulivoimaloista sekä Ilmatieteen laitoksen säähavaintoasemilta. Ilmatieteen laitoksen säähavaintoja voi ladata kuka tahansa havaintojen lataus -palvelusta. Jäätämistapahtumat analysoidaan itse tuulipuistojen datasta, sillä niitä ei ole erikseen sinne merkitty. Analysoinnin tueksi perehdytään aiempiin tutkimuksiin tuulipuistojen jäätämisestä, joita on julkaistu paljon viime vuosina. Nämä työt ovat pääosin energiatekniikan näkökulmasta. Tutkimuksia käsitellään tarkemmin seuraavan luvun loppupuolella.

Opinnäytetyön projektinhallinnassa käytetään Kanban-menetelmää, joka perustuu lean-ajatteluun. Kanban on visuaalinen työn ohjauksen menetelmä, jossa työtehtävät esitetään kanban-aulussa tehtäväkortteina. Yksinkertaisimmillaan kanban-taulu sisältää kolme saraketta. Nämä sarakkeet on usein otsikoitu seuraavasti: tehtävät (to-do), kesken (work in progress) ja valmis (done). (Laukkonen & von Kügelgen, 2020, ss. 203–204)

### 3 Tuulivoima ja jäätäminen

Suomessa tuulivoimaloita oli vuoden 2023 lopussa yhteensä 1601 kappaletta ja niiden yhteinen tuotantokapasiteetti oli lähes 7000 megawattia. Tuulipuisto koostuu tyypillisesti noin 6–20 voimalasta, mutta suurimmilla alueilla voi olla jopa yli 100 voimalaa. Suurin osa Suomen tuulivoimaloista sijaitsee länsirannikolla. (Motiva, 2024) Tuulisähköä syntyy, kun tuulisuus on noin 3–25 metriä sekunnissa (Suomen Tuulivoimayhdistys, n.d.-b).

Suomen Tuulivoimayhdistyksen (n.d.-c) mukaan talvisin on enemmän tuotantoa kuin kesällä, koska talvella tuulee enemmän. Talvisää luo kuitenkin jäätämislle otolliset olosuhteet. Jäätäminen tarkoittaa jään kertymistä esineen pinnalle. Tuulivoimaloissa jäätä kertyy erityisesti lapoihin. Jäätäminen voidaan jakaa kahteen eri tyyppiin: pilvijäätämiseen ja jäätävään sateeseen. Yhteistä niille on se, että molemmat jäätämisprosessit tapahtuvat lämpötilan ollessa pakkasen puolella. Pilvijäätämistä ilmenee Suomessa hieman enemmän kuin jäätävää sadetta. Lumisade sen sijaan ei aiheuta jään kertymistä lapoihin. Vaikka voimaloiden materiaalit ja komponentit on suunniteltu kestäväksi kylmiä olosuhteita, niin jäätäminen aiheuttaa silti haasteita. Näihin haasteisiin kuuluvat esimerkiksi tehohäviöt ja voimalan rakenteiden rasittuminen sekä tuulivoimalan alueella liikkuvien turvallisuus, sillä lavat saattavat heittää jäätä. (Suomen Tuulivoimayhdistys, n.d.-a; Brink, 2024)

Jäätämisen ennustaminen on tärkeää koko sähköjärjestelmän tasapainon kannalta, koska jäätäminen voi aiheuttaa suuriakin tehovajeita (Brink, 2024). REMIT-asetus määrää Euroopan laajuiset pelisäännöt energiamarkkinoille. REMIT-asetuksen mukaan voimalaitoksen merkittävän kapasiteetin puuttuminen tulee julkaista markkinoille ennen kuin tietoa hyödynnetään sähkökaupankäynnissä. (Energiavirasto, n.d.)

Ilmatieteen laitos on julkaissut tuuliatlaksen, joka sisältää dynaamisen karttaliittymän. Tuuliatlaksen tarkoitus on kuvata Suomen tuuliolosuhteita ja olla apuvälineenä tuulivoimaloiden rakentajille ja kaavoittajille. Tuuliatlakseen on liitetty myös jäätämisatlas. (Motiva, 2022) Jäätämisatlaksen tuottamisesta on kirjoitettu artikkeli, jossa kerrotaan, että atlaksen laatimisessa on käytetty AROME-sääennustemallia ja standardia ISO12494 (Atmospheric Icing of Structures). Sen mallintamiseen on hyödynnetty samaa meteorologista aineistoa kuin tuuliatlaksessa. Tutkittaviksi parametreiksi oli valittu lämpötila ja kosteus 850 hPa:n painetasolla. Näillä parametreilla oli tarkoitus määrittää suotuisat olosuhteet jäätämislle, mutta artikkelissa huomautetaan, ettei nämä parametrit välttämättä tavoita kaikkia mahdollisia jäätämisiä, jotka liittyvät esimerkiksi matalalla oleviin pilviin. (Hämäläinen & Niemelä, 2016, 2 Representativeness of the simulation period)

Tuulivoimaan ja jäätämiseen liittyen on tehty paljon tutkimuksia ja esimerkiksi vuonna 2022 Okka on tehnyt kandidaatintyön aiheesta Tuulivoimaloiden jäätäminen, joka on erittäin kattava kirjallisuuskatsaus. Siinä on käytetty lähteenä mm. edellä mainittua ISO12494 -standardia. Okan työssä kerrotaan laajemmin jäästä ja jäätämisestä sekä sen vaikutuksista tuulivoimaloiden toimintaan. Työssä on perehdytty myös Suomen tuulivoimaloiden jäätämiseen. Johtopäätöksissä nostetaan esille seuraavia tärkeitä seikkoja: jäätämistä voi tapahtua monenlaisissa olosuhteissa ja Suomessa sitä tapahtuu lähes puolet vuodesta. Lisäksi mainitaan, että jään eri muodot vaikuttavat tuulivoimaloiden toimintaan hieman eritavoin ja lopuksi korostetaan jäätämistapahtumien ennustamisen tärkeyttä, jotta tuulivoimaloiden operointi olisi tehokkaampaa jäätävissä olosuhteissa. (Okka, 2022, ss. 27–28)

Uusitalon (2022, s. 4) tutkielmassa “Analysis and Modeling for Predicting Icing Losses in Finnish Wind Power Production” on nimensäkin mukaisesti analysoitu tuotantodataa, etsitty jäätämistapauksia ja rakennettu jäätämishäviöiden ennustemalli. Ennustemalli oli rakennettu kolmen eri tuulivoimalan tuotantodatasta ja säädatasta, jossa oli mukana tuulennopeus, lämpötila, suhteellinen kosteus, ilman tiheys, pilvisuus ja sademäärä. Uusitalon työssä jäätämistapaukset oli jaettu kolmeen eri luokkaan eli huurrejähän, kirkkaaseen jähän ja lumeen. Työn ennustemallin jähän ja säähän liittyvät arvot oli määritelty jokaiselle tuulipuistolle erikseen. Työn merkittävimpiin löydöksiin kuului kirkkaan jään irtoamishetken ennustaminen.

Kreutz ja kumppanit (2019) ovat tutkineet jäätämisen ennustamista koneoppimisen avulla. Työssä käytetty data on saatu monien eri tuulipuistojen SCADA-järjestelmästä ja dataa on esikäsitelty mm. poistamalla poikkeamia ja virheellisiä jäätämistapahtumia. Mallin menetelmäksi he olivat valinneet neuroverkot. Mallin tavoitteena oli ennustaa jäätämistilanne yhden tunnin päähän tulevaisuuteen. Työn lopussa kerrotaan, että mallin tarkkuudeksi saatiin 84 prosenttia ja tätä olisi mahdollisesti pystytty parantamaan ottamalla sääolosuhteista huomioon myös kosteus tai nestemäinen vesipitoisuus. (Kreutz ym., ss. 426–427)

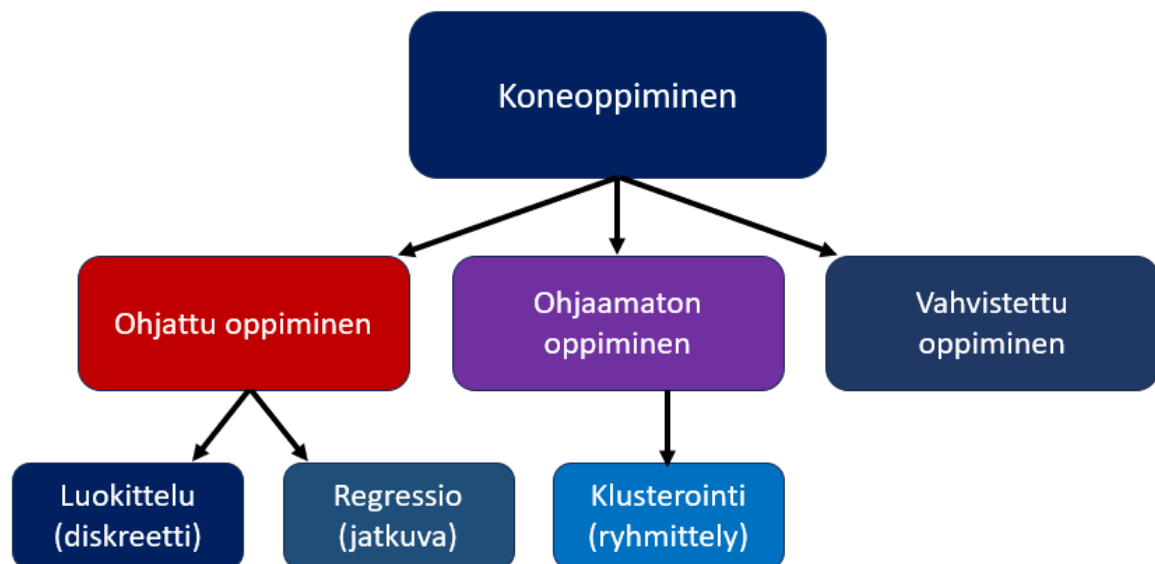
Yhteenvetona voidaan todeta, että eri lähteissä korostettiin tuulivoimaloiden jäätämisen ennustamisen tärkeyttä. Sekä Brinkin (2024) artikkelissa että Kreutzin ja kumppaneiden (2019, 427) tutkimuksessa todetaan jäätämisen ennustamisen olevan haastavaa. Seuraavassa luvussa tutustutaan koneoppimiseen ja muutamiin koneoppiviin menetelmiin.

## 4 Koneoppiminen

Koneoppiminen on tekoälyn muoto, jossa kone oppii joko ohjatusti tai itsenäisesti uusia asioita annetun datan avulla. Koneoppimismallin tavoitteena voi olla esimerkiksi jonkin tietyn asian tunnistaminen, luokittelu tai ennustaminen. Oppimistyylin perusteella koneoppiminen voidaan jakaa seuraaviin luokkiin: ohjattu oppiminen (supervised learning), ohjaamaton oppiminen (unsupervised learning) ja vahvistettu oppiminen (reinforcement learning).

Kuvassa 2 on esitetty koneoppimisen tyylit visuaalisessa muodossa jaon selkiyttämiseksi. Kuvasta nähdään myös, että ohjattu oppiminen voidaan jakaa datan tyyppin mukaan luokitteluun ja regressioon. Ohjaamattoman oppimisen data voidaan klusteroida eli ryhmitellä ominaisuuksiensa perusteella. (Niinimäki ym., 2019, s. 12)

Kuva 2. Koneoppimisen jako



Ohjatussa oppimisessa on käytössä koulutusdataa, josta selviää, millainen lopputulos on toivottu, kun algoritmille annetaan tietty syöte. Algoritmin tehtävänä on löytää datasta piirteitä ja sääntöjä, joiden avulla se voi toistaa koulutusdatassa olleita toivottuja tuloksia. Ohjatun opetusprosessin tuloksena syntynyt malli testataan datalla, jossa ei ole mukana oikeita vastauksia. Kun malli on koulutettu hyvin, niin sen tulisi pystyä päättämään oikeita vastauksia uudesta datasta. (Kolari & Kallio, 2023, s. 129)

Teoksessaan Meedeniya (2023, s. 21) selittää luokittelun ja regression eron siten, että regression avulla voidaan ennustaa jatkuvia arvoja kuten lämpötila tai ikä, kun taas luokittelussa käytetään diskreettejä arvoja kuten kissa tai koira, kylmä tai kuuma, terve tai epäterve.

Ohjaamattoman oppimisen käytössä ei ole valmiiksi luokiteltua dataa ja kone etsiikin itse säännönmukaisuuksia. Kone pyrkii siis ryhmittelemään eli klusteroimaan dataa sen piirteiden pohjalta samankaltaisiin ryhmiin. (Kolari & Kallio, 2023, s. 130) Kämäräinen (2023, s. 96) käyttää teoksessaan klusteroinnista sanaa ryvästys ja huomauttaa, ettei ohjaamaton oppiminen ole yhtä kuin ryvästys vaan paljon laajempi kokonaisuus.

Vahvistetussa oppimisessa kone oppii saamalla palautetta. Palaute voi olla positiivista eli toisin sanoen palkitsevaa tai negatiivista. Kone pyrkii tekemään valintoja siten, että sen positiivisen palautteen määrä kasvaa. (Niinimäki ym., 2019, s. 14)

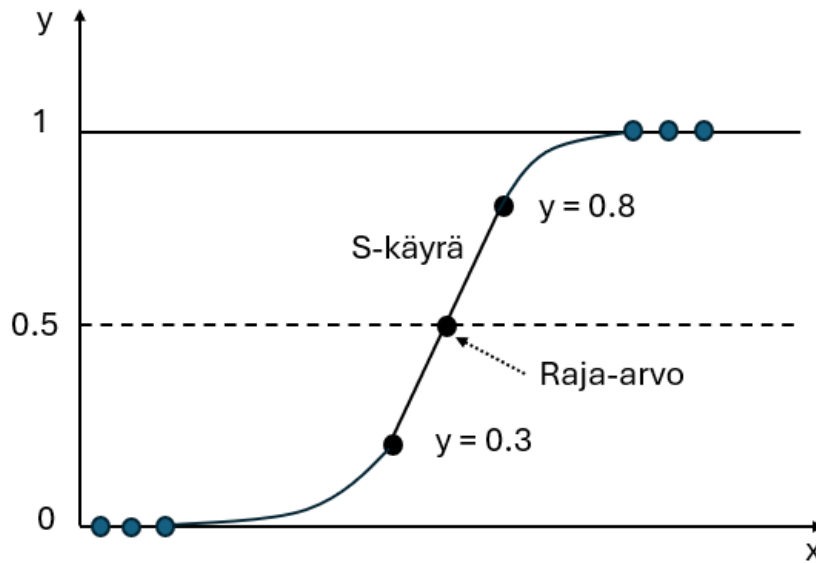
Seuraavissa alaluvuissa on esitelty lyhyesti malleja, joiden katsottiin lähteiden perusteella sopivan todennäköisyyksien ennustamiseen. Opinnäytetyön tavoitteen eli jäätämisenennustemallin tulisi ennen kaikkea pystyä arvioimaan jäätämisen todennäköisyyttä.

## 4.1 Logistinen regressio

Logistinen regressio (logistic regression) sopii todennäköisyyksien ennustamiseen ja binääriseen luokitteluun. Sitä voidaan käyttää suurempienkin aineistojen käsittelyssä ja kun riippumattomien piirteiden välillä ei ole korrelaatiota. Logistisen regression toiminta perustuu sigmoid-funktioon. (Meedeniya, 2023, s. 20)

Kuvassa 3 on esitetty logistisen regression toiminta. Logistinen regressio muuttaa sigmoid-funktion avulla sisääntulomuuttujat todennäköisyydeksi 0 ja 1 välille, jolloin raja-arvona tässä toimii 0.5.

Kuva 3. Logistinen regressio



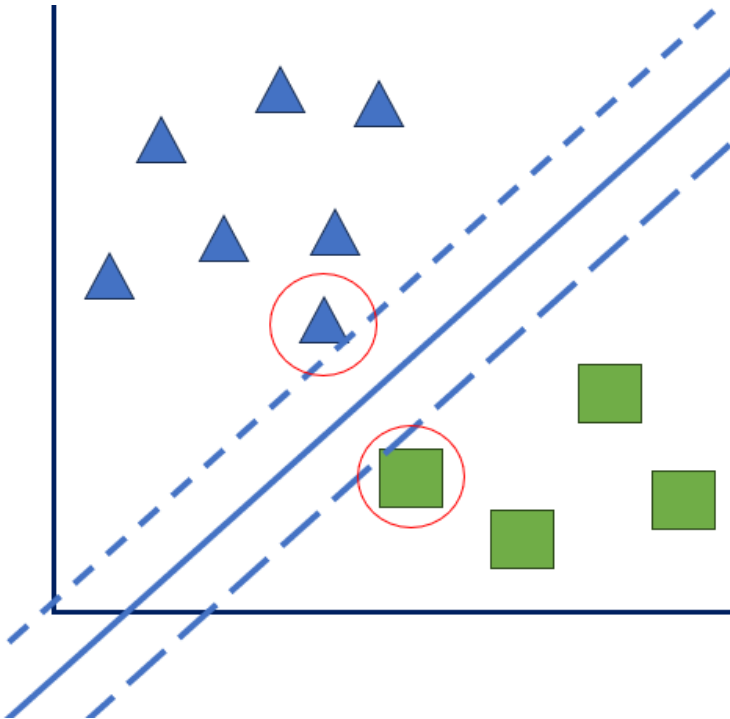
Logistista regressiota voidaan käyttää myös moniluokkaiseen luokitteluun, mutta silloin se hyödyntää softmax-aktivointifunktiota. Silloin siitä voidaan käyttää myös nimitystä softmax-regressio. (Meedeniya, 2023, s. 20) Logistista regressiota käytetään esimerkiksi finanssiriskien ennakoimiseen ja lääketieteellisissä tutkimuksissa (Elements of AI, n.d.-a).

## 4.2 Tukivektorikone

Tukivektorikonetta (support vector machine / SVM) käytetään luokittelu- ja regressio-ongelmien ratkaisemiseen (Kananen & Puolitaival, 2019, s. 120). Sitä voidaan soveltaa myös ennustamiseen. Tukivektorikoneen vahvuuksiin luetaan tarkkuus ja se toimii hyvin pienemmillä dataseteillä, joissa on vähemmän ns. kohinaa. Esimerkkejä sen käyttökohteista ovat roskapostisuodattimet ja tekstien luokittelu. (Kananen & Puolitaival, 2019, s. 123)

Tukivektorikoneen selittämiseen käytetään usein kuvassa 4 näkyvää esimerkkiä, jonka tarkoituksena olisi yksinkertaistaa asiaa mahdollisimman paljon. Kuvan siniset kolmiot ja vihreät neliöt voidaan erotella lineaarisesti. Keskimäinen viiva edustaa parasta hypertasoa eli päätöspintaa, joka erottelee kuviot toisistaan suurimmalla marginaalilla. Punaisella ympyröidyt kuviot ovat lähimpänä tätä hypertasoa, joten nämä ympyröidyt kuviot ovat tässä tapauksessa tukivektoreita. (Niinimäki ym., 2019, ss. 19–20; Bell, 2020, s. 149)

Kuva 4. Tukivektorikone

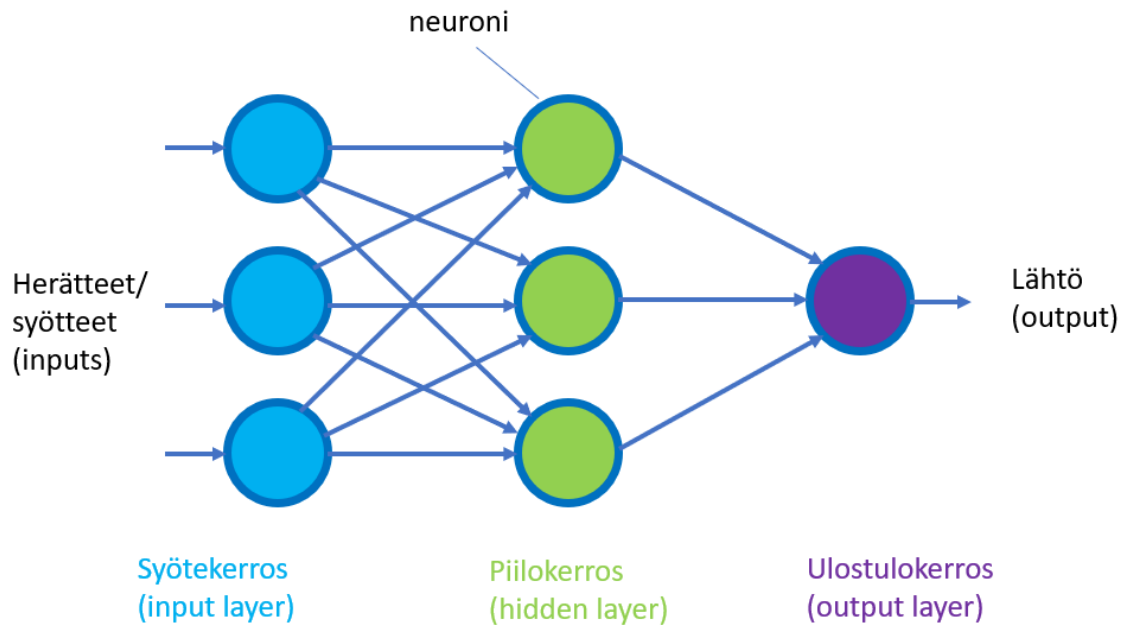


Kananen ja Puolitaival (2019, s. 121) kuitenkin huomauttavat, ettei data useimmitenkaan ole niin puhdasta, että sen voisi erottaa suoralla janalla. Tällaista sekalaista dataa voidaan kutsua epälineaarisesti separoituvaksi. Geronin (2022, s.175) mukaan tukivektorikone ei välttämättä sovellu kaikista parhaiten erittäin suurten tietojoukkojen käsittelyyn.

### 4.3 Neuroverkot

Kolarin ja Kallion (2023, s. 23) mukaan neuroverkot ovat laskennallisia malleja, joiden pyrkimyksenä on matkia ihmisaivojen hermoverkkojen toimintaa. Neuroverkkoja voidaan hyödyntää mm. todennäköisyyksien ennustamiseen (Kananen & Puolitaival, 2019, s. 140). Kuva 5 esittää yksinkertaista neuroverkkoa, josta voidaan nähdä millaisista osista ja kerroksista neuroverkot muodostuvat. Aluksi on syötekerros, seuraavaksi piilokerros ja lopuksi ulostulokerros.

Kuva 5. Neuroverkot



Keskellä olevia piilokerroksia voi olla useita sekä sen sisältämiä neuroneita. Neuronit ovat tiedonkäsittely-yksiköitä (Kolari & Kallio, 2019, s. 23). Neuroverkoilla voi siis olla aluksi useita eri syötteitä (inputs), mutta lopulta niillä on vain yksi lähtöarvo (output) (Kananen & Puolitaival, 2019, s. 129).

Neuroverkkojen erityispiirteisiin kuuluvat esimerkiksi seuraavat: jokainen neuroni voi prosessoida tietoa toisista neuroneista riippumatta ja näin ollen neuroverkot pystyvät prosessoida valtavia tietomääriä samanaikaisesti, tämän lisäksi ne myös tallentavat tietoa (Elements of AI, n.d.-b, I. Neuroverkkojen periaatteet).

Neuroni sisältää painoja, jotka ovat mukautuvia parametreja. Painoja käytetään syötearvojen kertoimina ja kertolaskujen tulokset lasketaan lopuksi yhteen ja tätä yhteenlaskun tulosta kutsutaan syötteiden lineaarikombinaatioksi. Lineaarikombinaation laskennan jälkeen neuroni soveltaa saatuun lukuarvoon aktivointifunktiota. Useimmiten käytettyjä funktioita ovat identiteetti-, porras- ja sigmoidifunktio. (Elements of AI, n.d.-b, II. Miten neuroverkkoja rakennetaan?)

## 4.4 ARIMA-aikasarjamalli

Aikasarja kuvaa havaintoja, jotka on järjestetty ajan mukaan. Aikasarjoja voidaan hyödyntää ennustamiseen, luokitteluun sekä syy-seuraussuhteiden tulkintaan. (Joseph, 2022, 1 Introducing Time Series) Aikasarjojen ennustaminen eroaa regressioista siten, että aikasarjoilla on aina tietty järjestys ja sitä järjestystä ei voi muuttaa. Lisäksi aikasarjamalleilla ennustaminen on mahdollista hyvin vähäisillä ominaisuuksilla. (Peixeiro, 2022, 1.3 How time series forecasting is different from other regression tasks)

ARIMA on lyhenne, joka koostuu seuraavista osista: Autoregressive (AR), Integrated (I) ja Moving Average (MA). AR eli autoregressiivinen osa perustuu menneiden ja nykyisten arvojen autokorrelaatioihin. I tarkoittaa aikasarjan automaattista integrointia, jos se ei ole stationaarinen. MA eli liukuvan keskiarvon osa perustuu menneiden virheiden ja nykyisten arvojen korrelaatioihin. ARIMA-mallilla on kolme hyperparametria:  $p$ ,  $q$  ja  $I$  (tai  $d$ ). AR prosessin järjestys merkitään  $p$ :llä, MA prosessi  $q$ :lla ja integraatio  $I$ :llä tai joissakin tapauksissa  $d$ :llä. (Korstanje, 2021, The ARIMA Model)

Aikasarjamalleista on monia eri variaatioita ja esimerkiksi SARIMA koostuu samoista osista kuten edellä käsitelty ARIMA, mutta siihen tuodaan myös kausivaihtelut (seasonal). Korstanjen (2021, The SARIMA Model) mukaan tämä ei ole vain yksi lisäosa ARIMA-malliin vaan jokaiseen edellä läpikäytyyn osaan (AR, I, MA) lisätään oma kausivaihtelu. Lisäksi Korstanje painottaa, että S-parametri ei ole hyperparametri ja se tulee valita logiikan perusteella, minkä tarkoituksena on ilmaista halutun kausivaihtelun jaksollisuutta. Esimerkiksi kuukausittaisen datan kanssa S-parametri olisi yhtä kuin 12 ja viikoittaisen datan kanssa se olisi yhtä kuin 52.

Purwar (2023) on listannut artikkelissaan ARIMA-mallin etuja sekä huonoja puolia. Etuihin kuuluu ARIMAn kyky havaita lyhyen aikavälin trendejä autoregressiivistä osaa (AR) hyödyntämällä ja pitkän aikavälin trendejä liukuvalla keskiarvolla (MA). Lisäksi etuihin kuuluu Purwarin mukaan se, ettei ARIMA tee oletuksia datan jakaumasta, joten se toimii hyvin minkä tahansa jakauman kanssa sekä parametreja on vähän, joten aikaa ei kulu liiaksi parhaiden parametrien etsimisessä. Huonoihin puoliin Purwar on listannut, ettei ARIMA suoriudu hyvin monimutkaisissa datakuvioissa ja se on herkkä poikkeamille sekä äärimmäisille arvoille, mistä johtuen ennusteet voivat olla epätarkkoja.

## 5 Mallin rakentamisen vaiheet

Tässä luvussa käydään mallin rakentamisprosessia läpi teoriassa ja pyritään löytämään vastaus kysymykseen, miten rakennetaan toimiva malli. Géron (2022, s. 779) on luonut tarkistuslistan koneoppimisprojektin tärkeimmistä vaiheista ja näihin vaiheisiin kuuluvat seuraavat:

1. Ongelman rajaaminen.
2. Datan hankkiminen.
3. Tutustu dataan.
4. Esikäsitle data.
5. Tutki erilaisia malleja ja listaa parhaimmat.
6. Hienosäädä järjestelmääsi.
7. Esitle ratkaisusi.
8. Käynnistä, valvo ja ylläpidä järjestelmääsi.

Géronin listan jokaista vaihetta käsitellään vielä erikseen omilla listoilla, joista löytää hyviä vinkkejä kuhunkin vaiheeseen. Tässä opinnäytetyössä ei käsitellä koneoppimisprojektin vaiheita kuitenkaan täysin edellä esitetyn listan mukaisesti vaan hieman mukailten ja prosessiin on tuotu näkökulmia myös Niinimäen ja kumppaneiden (2019) teoksesta. Listan vaiheita 1–6, joita tässä opinnäytetyössä sovelletaan, käsitellään seuraavissa alaluvuissa tarkemmin.

### 5.1 Ongelman rajaaminen ja datan hankkiminen

Alkuun tulee määrittellä ongelma, joka halutaan mallilla ratkaista sekä tutustua aineistoon ja sen muuttujiin. Ongelman määrittelyyn Géron (2022, ss. 779–780) on listannut esimerkiksi seuraavia apukysymyksiä:

- Miten tai mihin ratkaisua käytetään?
- Voisiko ongelmaa lähestyä ohjatusti vai ohjaamattomasti?
- Miten suoritus mitataan?
- Onko vastaavia ongelmia? Voisiko kokemusta hyödyntää tai siihen liittyviä työkaluja?

Ongelman määrittelyn jälkeen hankitaan tarvittava data. Ennen tätä olisi tärkeä miettiä mitä dataa tarvitaan ja kuinka paljon. Olisi hyvä myös ennakkoon tarkistaa kuinka paljon tilaa data

vie ja luoda sille sopivan kokoiset puitteet. Kun data on saatu, niin se tulisi muuttaa helposti käsiteltävään muotoon ja anonymisoida arkaluontoiset tiedot. (Géron, 2022, s. 780)

## 5.2 Tutustuminen dataan

Tarvittavan datan hankkimisen jälkeen aineistoon tulisi perehtyä tarkemmin ja tutustua sen eri muuttujiin. Muuttujia on erityyppisiä ja ne voidaan jakaa diskreetteihin (discrete variable) ja jatkuviin (continuous variable) muuttujiin. Muuttujan tyyppin tunnistaminen on tärkeää ennen aineiston esikäsittelyä (data preprocessing), koska käsittelyn toimenpiteet riippuvat muuttujan luonteesta, eikä siis jokaista esikäsittelyvaihetta voida soveltaa kaikkiin tapauksiin. Diskreetin muuttujan arvot ilmaistaan joko numeerisesti tai tekstimuodossa ja muuttuja voi olla nominaalinen tai ordinaalinen. Ordinaalisen muuttujan arvoja on mahdollista järjestää merkityksellisellä tavalla, mutta nominaalisilla arvoilla ei voida muodostaa merkityksellistä järjestystä. Jatkuva muuttuja ilmaistaan reaalityyppillä. (Niinimäki ym., 2019, s. 15)

## 5.3 Aineiston esikäsittely

Muuttujiin tutustumisen ja tunnistamisen jälkeen voidaan alkaa esikäsittämään aineistoa. Esikäsittely aloitetaan datan siivoamisella, jonka tarkoituksena on korjata mahdollisia puutteita sekä todennäköisesti virheellisiä tietoja. Siivouksen jälkeen on aineiston yhdistämisvaihe, jossa eri lähteissä olevat aineistot yhdistetään samassa muodossa olevaksi kokonaisuudeksi.

Analysoinnin nopeuttamiseksi aineistoa voidaan myös vähentää seuraavilla menetelmillä: ominaisuuksien valinta (feature selection), piirreirrotus (feature extraction) ja näytteenotto (sampling). Ominaisuuksien valinnassa aineistosta karsitaan epäolennaiset ominaisuudet, mikä parantaa aineiston ymmärrettävyyttä ja esitettävyyttä. Piirreirrotuksen tavoite on kuvata alkuperäistä aineistoa pienemmällä määrällä ulottuvuuksia ja sen käytetyin menetelmä on pääkomponenttianalyysi, jossa luodaan uusi joukko muuttujia alkuperäisen aineiston olennaisimmista piirteistä. Näytteenotto datan vähentämisen menetelmänä tarkoittaa sitä, että aineistosta valitaan jokin osajoukko analyysiä varten. Muuttujat valmistellaan sopivaan muotoon analyysiä varten erilaisilla aineiston muunnosmenetelmillä, joihin kuuluvat normalisointi (normalisation), diskretisointi (discretisation) ja ominaisuuksien luonti (feature construction)/feature generation). Normalisoinnin tarkoituksena on skaalata muuttujien arvot vertailukelpoiselle vaihteluvälille. Diskretisoinnin tarkoituksena on muuntaa jatkuvien

muuttujien arvot diskreeteiksi, jotta aineisto saataisiin helpommin ymmärrettävään muotoon. (Niinimäki ym., 2019, ss. 15–17)

#### **5.4 Mallien tutkiminen ja parhaimpien valinta**

Eri menetelmiä on useita ja parhaimman menetelmän löytääkin testaamalla. Datasta on suositeltavaa ottaa pienempi testijoukko, mikäli dataa on paljon, jotta testivaiheesta päästäisiin nopeammin eteenpäin. Kun malleja on saatu koulutettua, niin niiden suoriutumista vertaillaan. Jokaisen mallin suoritus voidaan tarkistaa N-fold cross-validation -menetelmää hyödyntäen. Mallien tärkeimpiä muuttujia analysoidaan ja tutkitaan millaisia virheitä mallit mahdollisesti tekevät. Analysoinnin jälkeen voidaan suorittaa muutamia iteraatio kertoja ja parannella malleja. (Géron, 2022, s. 782)

#### **5.5 Järjestelmän hienosäätäminen**

Tässä vaiheessa käytetään niin paljon dataa kuin mahdollista. Hyperparametreja voidaan hienosäätää ristiinvalidointi (cross-validation) -menetelmällä. Yhdistelemällä eri malleja on mahdollista saada aikaan parempia tuloksia tehokkaammin. Lopullisen mallin suorituskyky voidaan mitata testiaineistolla, josta saadaan arvio yleistysvirheestä (generalization error). (Géron, 2022, ss. 782–783).

Mallin parametrien ylisovittamista tulisi varoa. Müller ja Guido (2016, 5. Model Evaluation and Improvement) toteavat, että se miten malli suoriutuu uuden datan kanssa, on kiinnostavampaa kuin se, miten malli selviytyy testidatasta. Mikäli testidataa käytetään mallin tai parametrien valintaan, voi testidata kulua niin sanotusti loppuun ja tämä voi johtaa liian optimistisiin arvioihin mallin suoriutumiskyvystä. Tästä syystä data tulisi jakaa näihin kolmeen osaan: harjoitusdataan mallin rakentamista varten, validointidataan mallin ja parametrien valintaa varten sekä vielä testidataan mallin arvioimista varten.

## 6 Jäätämisen ennustaminen

Ennen mallin rakentamista oli tärkeä selvittää millaisissa olosuhteissa jäätämistä tapahtuu. Tutkimuksia aiheesta löytyi paljon, joten näihin tutustuminen oli tärkeää ennen oman mallin luomista. Merkittävimmät parametrit sääolosuhteiden kannalta olivat lämpötila ja kosteus. Jäätämistä tapahtui tutkittujen lähteiden perusteella lämpötilan ollessa pakkasen puolella sekä kosteuden ollessa korkealla. Seuraavissa alaluvuissa on kuvattu mallin rakentamisprosessia alkaen kertomalla menetelmistä sekä mallin määrittelystä ja tavoitteista.

### 6.1 Menetelmät

Työn toteutuksessa hyödynnettiin Kanban-menetelmää projektin eteenpäin viemiseksi. Kanban-menetelmää on avattu lyhyesti Tutkimusasetelma -luvun lopussa. Aluksi toimeksiantajan edustajan kanssa tehtiin määrittelyä siitä, mitä mallilta halutaan sekä sen tavoitteita, joista kerrotaan seuraavassa alaluvussa tarkemmin. Seuraavaksi nämä tavoitteet jaettiin pienempiin osiin ja tehtäviin. Lisätehtäviä muodostui koko työn prosessin aikana. Kanban menetelmänä oli sopiva tähän työhön, koska ei ollut varmuutta esimerkiksi mallien laskenta-ajoista tai työn lopullisesta kokonaisuudesta alkuvaiheessa.

### 6.2 Mallin määrittely ja tavoitteet

Mallin toivottiin olevan mahdollisimman yksinkertainen. Sen ei ollut tarkoitus laskea jäätämisen kestoa tai paljonko jäätäminen aiheuttaa tuotannollisia tappioita. Tarkoituksena oli, että malli osaisi ennustaa onko jäätäminen todennäköistä vai ei. Ulostulona haluttiin mallilta lista tunneista, joilla jäätämistä voi tapahtua.

Työn toimeksiantaja määritteli mallille kaksi päätavoitetta. Ensisijainen tavoite oli luoda malli, jolla pystyttäisiin ennustamaan jäätämistilanne noin 40 tuntia eteenpäin ja tätä työssä kutsutaan jatkossa pitkänaikavälin ennusteeksi. Toinen tavoite oli pystyä ennustamaan seuraavan kahdeksan tunnin jäätämistilanne ja tätä kutsutaan työssä lyhyenaikavälin ennusteeksi.

Tuulipuistojen jäätämisen ennusteen tavoiteltuja hyötyjä oli saada tietoa mahdollisista vaikutuksista tuotantoon hyvissä ajoin ja näin ollen auttaa markkinailmoitusten teossa sekä mahdollisuus varoittaa alueella työskenteleviä jäänheiton vaarasta.

### 6.3 Kuvaus ympäristöstä ja käytetystä aineistosta

Työympäristö muodostui Visual Studio Codesta ja Jupyter Notebookista. Työn ohjelmointikielenä käytettiin pythonia ja seuraavia kirjastoja: pandas, numpy, seaborn, matplotlib, statmodels ja sklearn (scikit-learn).

Tuulipuistoihin liittyvä aineisto saatiin ladattua yrityksen datajärvestä eli tietovarastosta. Tuulipuistoja oli useita ja niistä valittiin yhteensä kuusi. Näistä tuulipuistoista osa sijaitsi rannikkoseudulla ja osa sisämaassa sekä näiden joukossa oli eri turbiinin valmistajia. Samalta alueelta pyrittiin valitsemaan kahden eri valmistajan turbiinit.

Jokaisesta tuulipuistosta tehtiin oma Jupyter Notebook -tiedosto, jossa tutkittiin tarkemmin tuulipuistojen dataa. Tuulipuistojen datassa oli seuraavat muuttujat:

- mitattu teho (active power)
- saatavilla oleva pätöteho (available power)
- tuulen nopeus (wind speed)
- tuulen suunta (wind direction)
- ilman lämpötila (air temperature)
- tehorajoitus (psetpoint).

Lähtöaineiston tukimuuttujat sääennusteista 100 metrin korkeudessa:

- kastepiste ennuste (forecast dew point temperature (YR))
- tuulen nopeuden ennuste (forecast wind speed (YR))
- suhteellinen kosteus ennuste (forecast relativehumidity (YR))
- ilman lämpötilan ennuste (forecast air temperature (YR))
- ilmanpaine ennuste (forecast airpressure (YR)).

Tuulipuiston datan lisäksi tiedostoon tuotiin Ilmatieteen laitoksen tuulipuistoa lähimmältä havaintoasemalta säähavaintoja. Työssä käytetyt Ilmatieteen laitoksen säähavaintomuuttujat:

- suhteellinen kosteus
- lämpötila
- pilvisuus

- kastepiste
- ilmanpaine.

Ilmatieteen laitoksen havaintodataa tuotiin työhön, koska kaikilla tuulipuistoilla ei ollut tarvittavia tukimuuttujia valitulta ajanjaksolta. Huomioitavaa on kuitenkin se, että Ilmatieteen laitoksen havainnot eivät ole turbiinin napakorkeudesta (eli 100 metrin korkeudessa) kuten tuulipuistodatan tukimuuttujat.

## 6.4 Aineiston esikäsittely

Tuulipuistoista häivytettiin ensimmäiseksi kaikki tunnistetiedot, kaikki tehomittaukset muutettiin 'per unit' (pu) -muotoon ja puistot nimettiin aakkosittain seuraavasti: Tuulipuisto A, Tuulipuisto B jne. Per unit -muodossa puiston tehomittaukset jaetaan puiston nimellisteholla. Nimellisteho tarkoittaa suurinta tehoa, jonka puisto pystyy tuottamaan. Per unit -käsittelyllä saadaan eri tehoiset tuulipuistot myös vertailukelpoisiksi keskenään. Yksinkertaistettuna 1.0 pu vastaa 100 % nimellistehoa.

Aineistoon lisättiin kolme apumuuttujaa. Ensimmäinen näistä oli LOST\_PRODUCTION -muuttuja, joka kuvaa menetettyä tuotantoa suhteessa käytettävissä olevaan tuotantokykyyn. Menetetty tuotanto ( $P_{loss}$ ) katsottiin olevan käytettävissä olevan tehon ( $P_{available}$ ) ja mitatun pätehtehon ( $P_{measured}$ ) erotus huomioiden mahdolliset tehorajoitukset ( $P_{limit}$ ) (kaava 1).

Kaava 1. Menetetyn tehon laskenta

$$P_{loss} = \min(P_{available}; P_{limit}) - P_{measured}$$

Tilanteita, joissa tuulipuisto pysähtyy kokonaan, korjattiin  $P_{available}$  -muuttuja yksinkertaisella fysikaalisella mallilla (kaava 2). Tuulivoiman tuottama teho on verrannollinen tuulen nopeuden kolmanteen potenssiin ja massavirtaan (Kokkonen, 2018, s. 8).

Kaava 2. Tuulivoiman fysikaalinen malli

$$P = \frac{1}{2} \rho A v^3$$

Kaavaan 2 sijoittamalla arvot siten, että nimellisteho (1.0 pu) saavutetaan 10 m/s tuulella (toimeksiantajalta saadut parametrit), saadaan yksinkertaistettu arvio käytettävissä olevasta tehosta (kaava 3).

Kaava 3. Käytettävissä olevan tehon arvio

$$P_{avail\ estimated} = \frac{2 v^3}{1000 \frac{1}{pu}}$$

Tämän avulla saatiin korjattua datasta tilanteet, jotka ovat pysäyttäneet tuulipuiston kokonaan huomioiden kuitenkin voimaloiden käynnistysnopeus 3.5 m/s. Toimeksiantajan edustaja auttoi datan korjaamisessa ja epäselvien tilanteiden suodattamisessa.

Lähtöaineistossa ei ollut valmiiksi luokiteltu jäätämistapahtumia, joten niille tuli luoda oma muuttuja IS\_ICING. Jäätämiseksi määriteltiin seuraava tilanne:

- lämpötila alle 0 °C
- tuulen nopeus 3.5 ... 20.0 m/s
- menetetty tuotanto yli 0.15 pu (per unit) eli 15 % nimellistehosta.

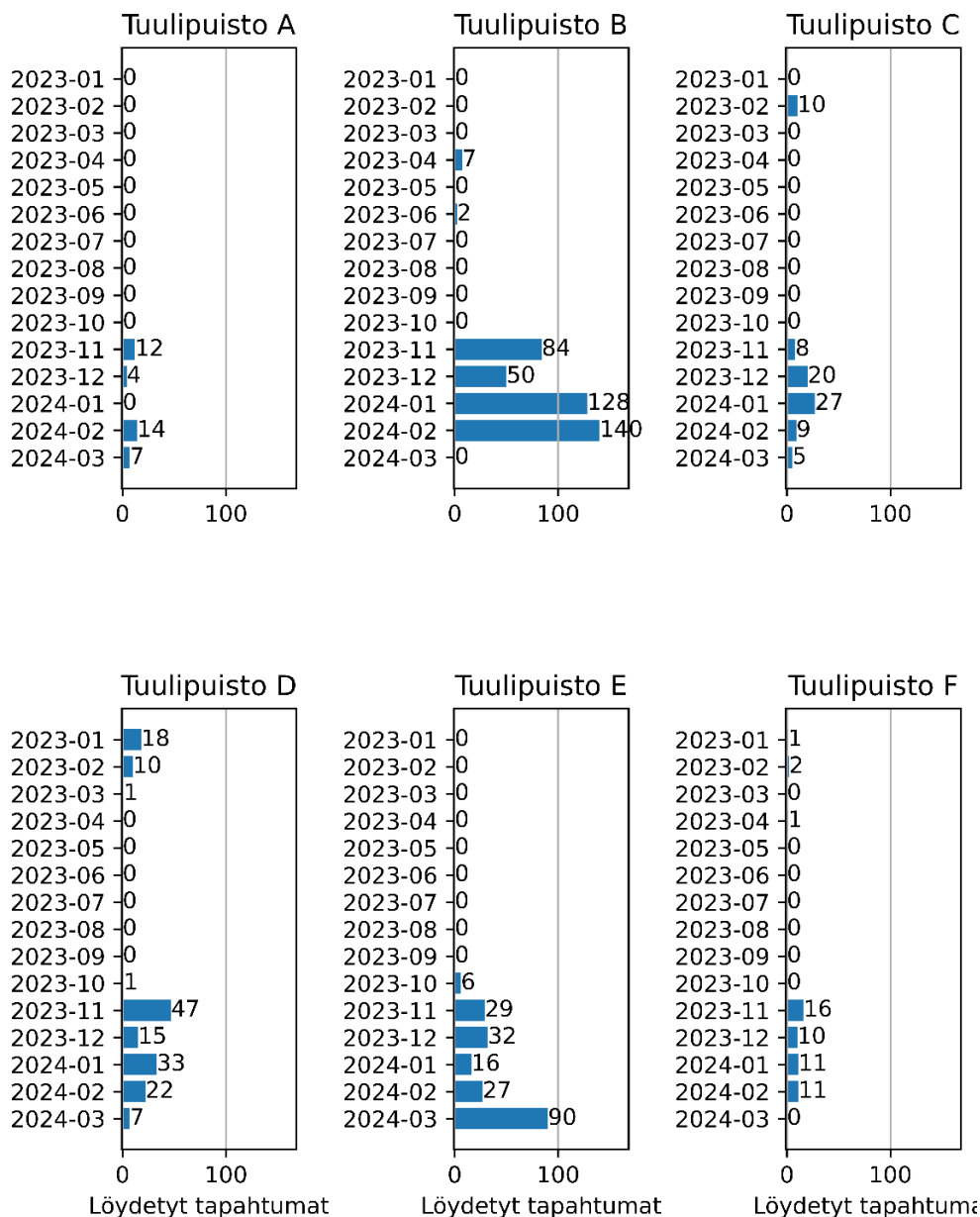
Kun tuulen nopeus on alle 3.5 m/s niin tuuli on alle käynnistysnopeuden ja voimalat eivät tuota. Jos tuulen nopeus on yli 20.0 m/s, myrskysuojaus tekee käytettävissä olevasta tehomuuttujasta epäluotettavan. Menetetyn tuotannon toleranssiksi valittiin 0.15 pu koulutus- ja testivaiheessa, koska aineiston perusteella se vaikutti olevan kohinatason yläpuolella eli virheellisten havaintojen määrä olisi pienempi. Tuotanto käytössä tämän herkkyyttä todennäköisesti tulee säätää vastaamaan tarvetta tarkemmin. Käytetyt voimalan käynnistys, nimellistehon ja myrskysuojauksen tuulen nopeudet, joita tässä on käytetty ovat toimeksiantajan määrittelemiä yksinkertaistuksia eli niistä on häivytetty tuulipuisto kohtaisia tietoja. Jokaisessa tuulivoimala mallissa nämä ovat eri tasoilla, mutta osuvat lähelle yksinkertaistuksen arvoja.

Kolmanneksi lasketuksi muuttujaksi luotiin mitatun tuulen nopeuden kolmas potenssi (WIND\_SPEED\_POW3). Tämän muuttujan oli tarkoitus säästää mallien laskennasta fysiikan oppimista. Esikäsittelyn jälkeen, kun oli lisätty tarvittavat apumuuttujat (LOST\_PRODUCTION, WIND\_SPEED\_POW3, IS\_ICING), oli mahdollista tutkia enemmän muuttujien välisiä riippuvuuksia. Seuraavassa alaluvussa esitellään esikäsittelyn jälkeisiä havaintoja.

## 6.5 Havainnot esikäsittelyn jälkeen

Kuvassa 6 on esitetty kaikki löydettyt jäätämistapahtumat tuulipuistoittain. Yksi jäätämistapahtuma tarkoittaa 15 minuutin ajanjaksoa, jolloin tuulipuisto menettää tuotantoa. Tästä voidaan havaita, että tapahtumia esiintyy pääosin marraskuusta maaliskuuhun, mutta vaihtelua on paljon eri tuulipuistojen välillä. Tuulipuistot A–D sijaitsevat lähempänä rannikkoa kuin tuulipuistot E ja F. Tuulipuistot A ja B sijaitsevat hyvin lähellä toisiaan. Eniten jäätämistapahtumia on tuulipuistoissa B ja E, kun taas vähiten tapahtumia on ollut tuulipuistoissa A ja F.

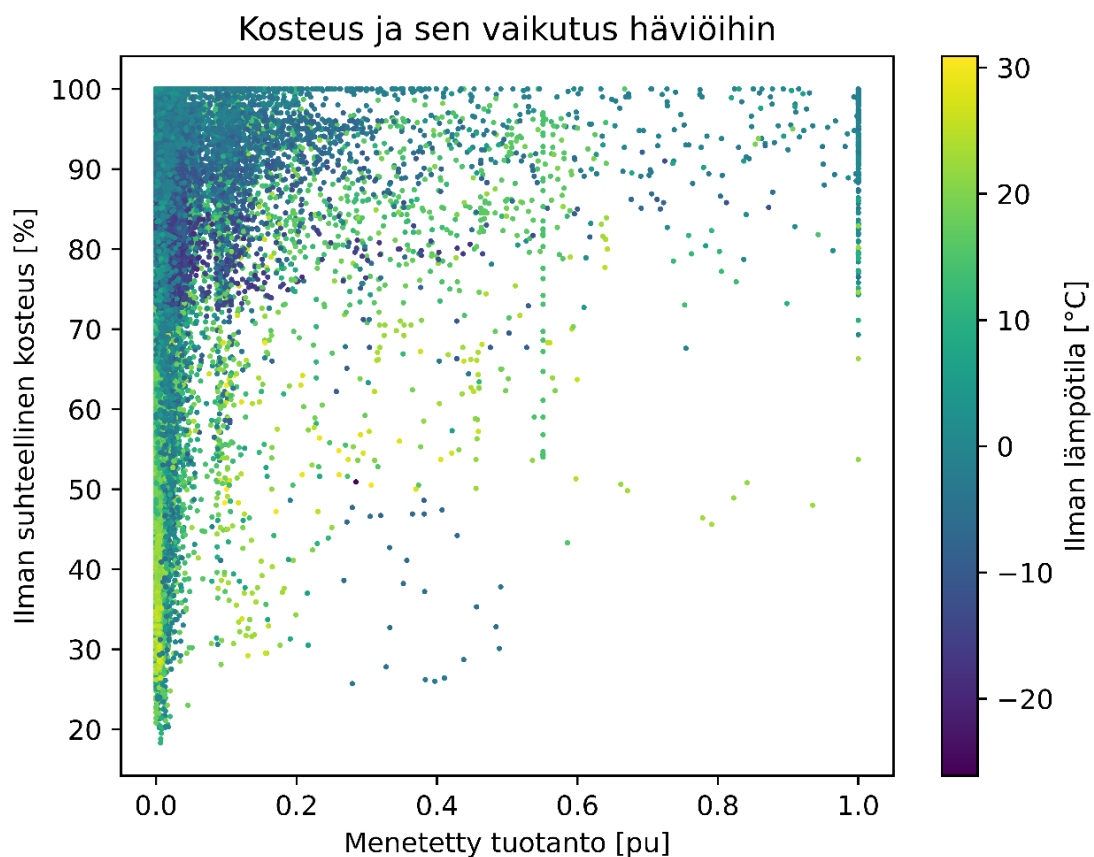
Kuva 6. Löydettyjä jäätämistapahtumia tuulipuistoittain (1 tapahtuma = 15 min)



Kiinnostava havainto näistä on se, että samoissa olosuhteissa vierekkäisten tuulipuistojen herkkyys jäätämislle on hyvin erilainen. Tämän pohjalta vaikuttaisi siltä, että mallit olisi mahdollisesti koulutettava tuulipuisto kohtaisesti, jotta saataisiin parempia tuloksia.

Kuvassa 7 on esitetty kosteuden vaikutus tehohäviöihin yhdistäen dataa kaikista tutkituista tuulipuistoista. Tehohäviöpisteiden väri on koodattu lämpötilan mukaan. Kuvasta voidaan havaita noin 0.05–0.1 pu kohinataso, joka esiintyy kaikissa kosteus ja lämpötila olosuhteissa. Nähdään myös, että suurin osa merkittävistä tehohäviöistä (yli 0.2 pu) tapahtuu, kun kosteus on yli 70 % ja lämpötila on -10–0 °C välillä.

Kuva 7. Kosteuden ja lämpötilan vaikutus tehohäviöihin



Kuvasta 7 voidaan myös todeta, että lähtöaineistossa on paljon kohinaa vasemmassa reunassa kauttaaltaan ja tämä voi olla haaste malleja kouluttaessa. Lähtöaineistosta on haastava määrittellä kohinan syytä, jolloin sen suodattaminen pois on vaikeaa. Seuraavassa aluvuussa kerrotaan menetelmien testausvaiheesta.

## 6.6 Menetelmien testaus

Parhaan menetelmän löytämiseksi menetelmiä testattiin ja vertailtiin niiden keskinäistä suorituskykyä. Suorituskyky mittarina käytettiin sitä, kuinka hyvin malli löytää jäätämistapahtumia. Testattuja menetelmiä olivat logistinen regressio, tukivektorikone, neuroverkot ja ARIMA.

Mallin syöttömuuttujiksi valittiin ilmansuhteellinen kosteus, kastepiste, lämpötila, tuulen nopeuden kolmas potenssi, tuulen nopeus ja ilmanpaine. Kosteus, kastepiste ja lämpötila valittiin esikäsitellyssä havaitun riippuvuuden pohjalta. Tuulen nopeuden kolmas potenssi, tuulen nopeus ja ilmanpaine valittiin auttamaan mallia tunnistamaan jäätämisiä, sillä näillä muuttujilla oli vahva korrelaatio tuotetun tehon kanssa.

Malleja koulutettiin IS\_ICING -muuttujaa vasten. Jokaisen mallin testauksessa datasetti jaettiin sattumanvaraisesti koulutus- ja testidataan käyttäen jakosuhdetta 75 % (koulutus) ja 25 % (testi). Testaus toteutettiin ensin Tuulipuistoilla A ja B, joilla rajattiin parametrit muiden puistojen testeihin. Parametrit iteroitiin brute force -menetelmällä laskemalla monenlaisia kombinaatioita.

Logistisessa regressiossa, tukivektorikoneessa sekä neuroverkoissa käytettiin sisääntulomuuttujina seuraavia: ilman suhteellinen kosteus, kastepiste lämpötila, ilman lämpötila, tuulen nopeus, tuulen nopeuden kolmas potenssi sekä ilmanpaine. Näihin muuttujiin viitataan taulukossa 1 suorina muuttujina.

Työssä käytetty ARIMA-toteutus (python statmodels -kirjasto) tuki ainoastaan sisääntulomuuttujana samaa muuttujaa kuin se ennustaessa antaa ulos. Tämän vuoksi ARIMA -mallille voitiin syöttää vain IS\_ICING -muuttuja. Testauksessa, pois lukien ARIMA, käytettiin erilaisia testitapauksia, joissa oli eri määrä muuttujia taulukon 1 mukaisesti. Kreutzin ja kumppanien (2019) tutkimuksessa oli käytetty aritmeettisiä funktioita (arithmetic function) kuten rullaavia arvoja (keskiarvo, minimi ja maksimi) ja nämä otettiin mukaan myös testeihin. Eri testitapauksilla haluttiin löytää optimaaliset rullausajat.

Taulukko 1. Testitapaukset

Testitapaus (test case)	Suorat muuttujat	Rullaava keskiarvo	Rullaava minimi	Rullaava maksimi
Case 1	Kyllä	Ei rullaavia	Ei rullaavia	Ei rullaavia
Case 2	Kyllä	2 edellistä	2 edellistä	2 edellistä
Case 3	Kyllä	4 edellistä	4 edellistä	4 edellistä
Case 4	Kyllä	8 edellistä	8 edellistä	8 edellistä
Case 5	Kyllä	12 edellistä	12 edellistä	12 edellistä
Case 6	Kyllä	16 edellistä	16 edellistä	16 edellistä
Case 7	Kyllä	2,4,8,12 ja 16 edellistä	2,4,8,12 ja 16 edellistä	2,4,8,12 ja 16 edellistä

Testitapauksia oli yhteensä seitsemän ja niissä oli mukana suorat muuttujat eli testiajankohdan arvot sekä vaihteleva määrä takaisinpäin laskettuja keskiarvoja, minimejä ja maksimeja. Kuvassa 8 on esitetty rullaavien arvojen toimintaa. Kuvan esimerkissä havainnollistetaan, miten neljä edellistä rullaavaa arvoa koostuu edellisestä neljästä 15 minuutista mittauspisteestä taaksepäin katsoen. Eli toisin sanoen kyseiset arvot edustavat mennyttä aikaa.

Kuva 8. Esimerkki rullaavista arvoista



Rullaavat 4 edellistä mittauspistettä

= "4 edellistä  
rullaavaa arvoa"

Seuraavissa alaluvuissa käsitellään eri menetelmien testauksia ja niiden tuloksia lyhyelle aikavälille. Lyhyen aikavälin ennustetta käsitellään tässä opinnäytetyössä, mutta pitkän aikavälin ennusteiden tulokset toimitetaan vain toimeksiantajan käyttöön.

### 6.6.1 Neuroverkot

Neuroverkkojen koulutukseen käytettiin sklearn -kirjaston MLPClassifier -metodia. Metodi valittiin sklearnin dokumentaation pohjalta luokitteluongelmaan. Neuroverkon eri muotoja ja alpha toleranssia iteroitiin ja testattiin erilaisilla muuttujamäärillä (rullaavat keskiarvot, minimiit ja maksimit). Yhteensä malleja testattiin lähes 5000 kappaletta.

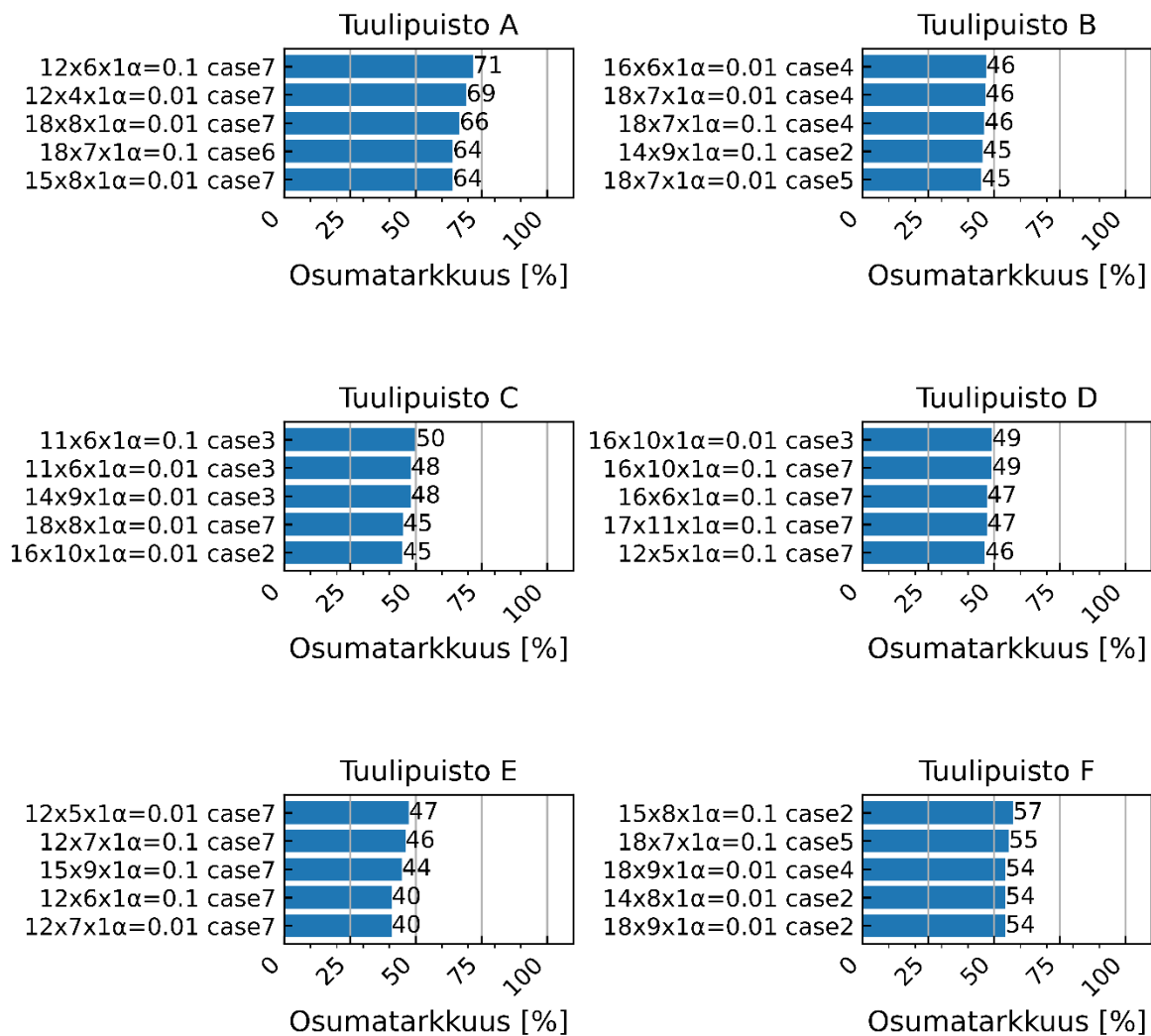
Neuroverkon rakennetta testattiin iteroimalla eri rakenteita ja käyttäytymisen rajaamisen jälkeen päädyttiin seuraavaan testiin. Mallin sisäinen rakenne testattiin kolmella sisäisellä kerroksella. Ensimmäinen kerros oli välillä yhdestä neuronista aina sisääntulomuuttujien määrään, toinen kerros vähintään 1/3 ensimmäisestä kerroksesta ja korkeintaan 2/3 ensimmäisestä sisäisestä kerroksesta ja kolmas kerros korkeintaan puolet toisesta kerroksesta. Näihin päädyttiin testaamisen perusteella, toisen muotoiset mallit eivät tuottaneet parempia tuloksia, joten laskenta-ajan minimoimiseksi tehtiin näin.

Kuvassa 9 on esitetty jokaisen tuulipuiston parhaimmat neuroverkkomallit. Kuvassa neuroverkon rakenne kuvataan kertomalla jokaisen kerroksen neuronien määrä ja erottamalla ne x -merkillä. Esimerkiksi  $12 \times 6 \times 1 \alpha = 0.1$  tarkoitetaan neuroverkkoa, jossa on kolme kerrosta. Ensimmäinen kerros on 12 neuronia leveä, toinen kuusi ja kolmas yhden. Lisäksi toleranssiksi määritelty alpha ( $\alpha$ ) termi on 0.1.

Sisääntulomuuttujina käytettiin taulukon 1 (sivulla 21) mukaisia arvoja. Mallien osumatarkkuus vaihteli 40–71 % välillä, mikä tarkoittaa, etteivät mallit suoriutuneet erityisen hyvin ja olisivat melko epäluotettavia. Havaittiin, että ilman rullaavia keskiarvoja (case 1) mallit eivät löytäneet jäätämistapahtumia ja eivät näin ollen päässeet minkään tuulipuiston parhaimpien mallien joukkoon.

Testejä tehtiin sekä lähiajan että pitkän ajan ennusteiden testaamiseen. Testien perusteella neuroverkko soveltuisi näihin molempiin.

Kuva 9. Neuroverkkojen parhaimmat mallit tuulipuistoittain lyhyen aikavälin ennustamiseen



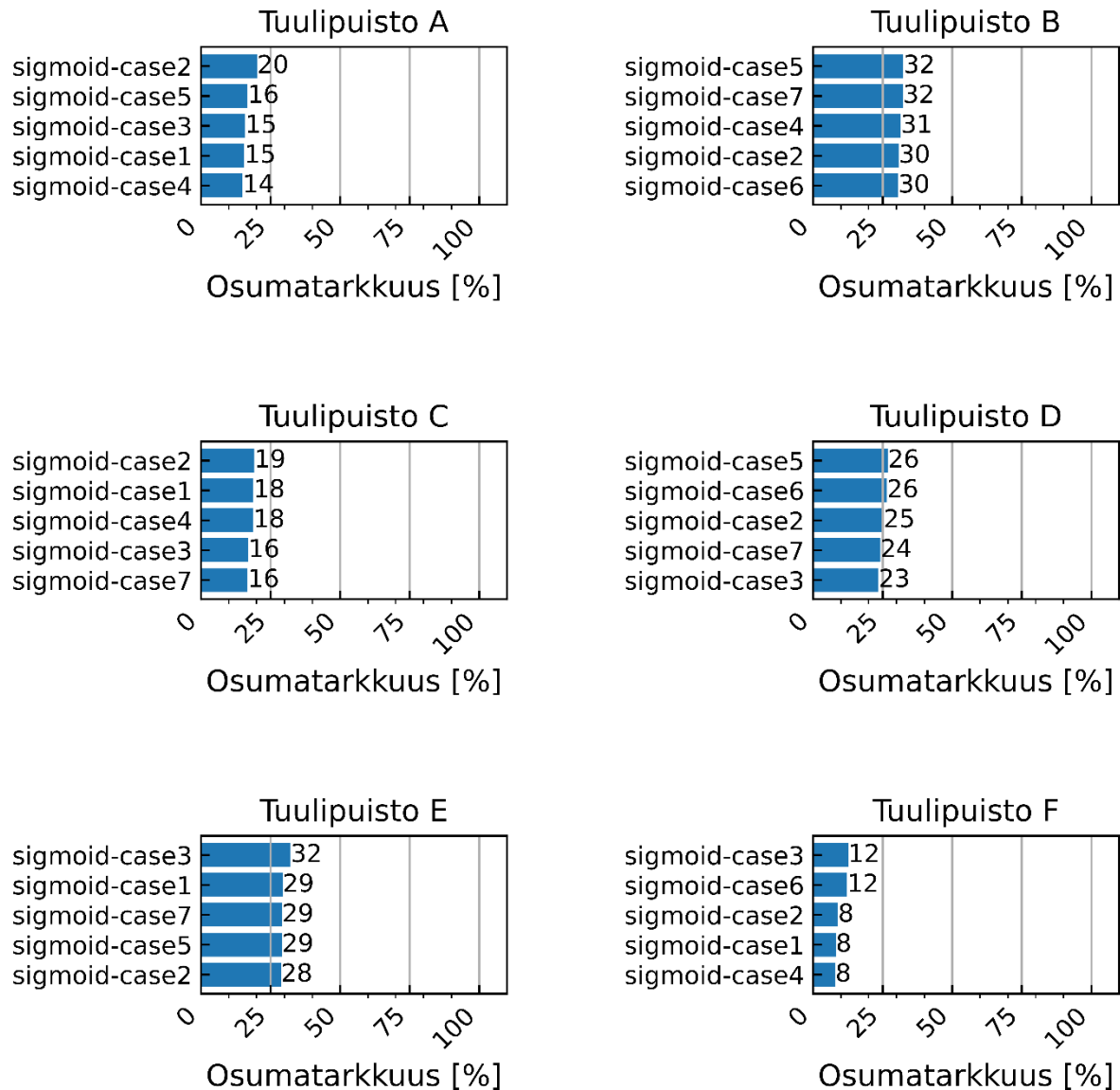
Mallien tuloksista selviää edellä mainittujen seikkojen lisäksi se, että tuulipuistot A ja F, joilla oli verrattain vähemmän havaittuja jäätämistapahtumia, niin mallit olivat hieman tarkempia ja eniten jäätävillä tuulipuistoilla B ja E mallien osumatarkkuus ei ylittänyt 47 %. Seuraavaksi käsitellään tukivektorikoneen testausta ja tuloksia.

## 6.6.2 Tukivektorikone

Tukivektorikoneen koulutukseen käytettiin sklearn -kirjaston SVC -menetelmää, joka on tukivektorikone luokitteluongelmiin. Tukivektorikoneen eri aktivointifunktioita testattiin iteroimalla. Testatut funktiot olivat linear, poly, rbf ja sigmoid. Sisääntulomuuttujina käytettiin taulukon 1 (sivulla 21) mukaisia testitapauksia.

Kuvassa 10 on esitetty parhaiten suoriutuneet tukivektorikonemallit. Osumatarkkuus vaihteli 8–32 % välillä, mistä päätellen mallit olivat erittäin epätarkkoja. Kuvasta voidaan kuitenkin todeta, että testatuista funktioista sigmoid oli ylivoimaisesti paras, kun mukana oli rullaavia arvoja. Muut testatut funktiot eivät päässeet viiden kärkeen missään tuulipuistossa.

Kuva 10. Parhaimmat tukivektorikonemallit tuulipuistoittain lyhyen aikavälin ennustamiseen

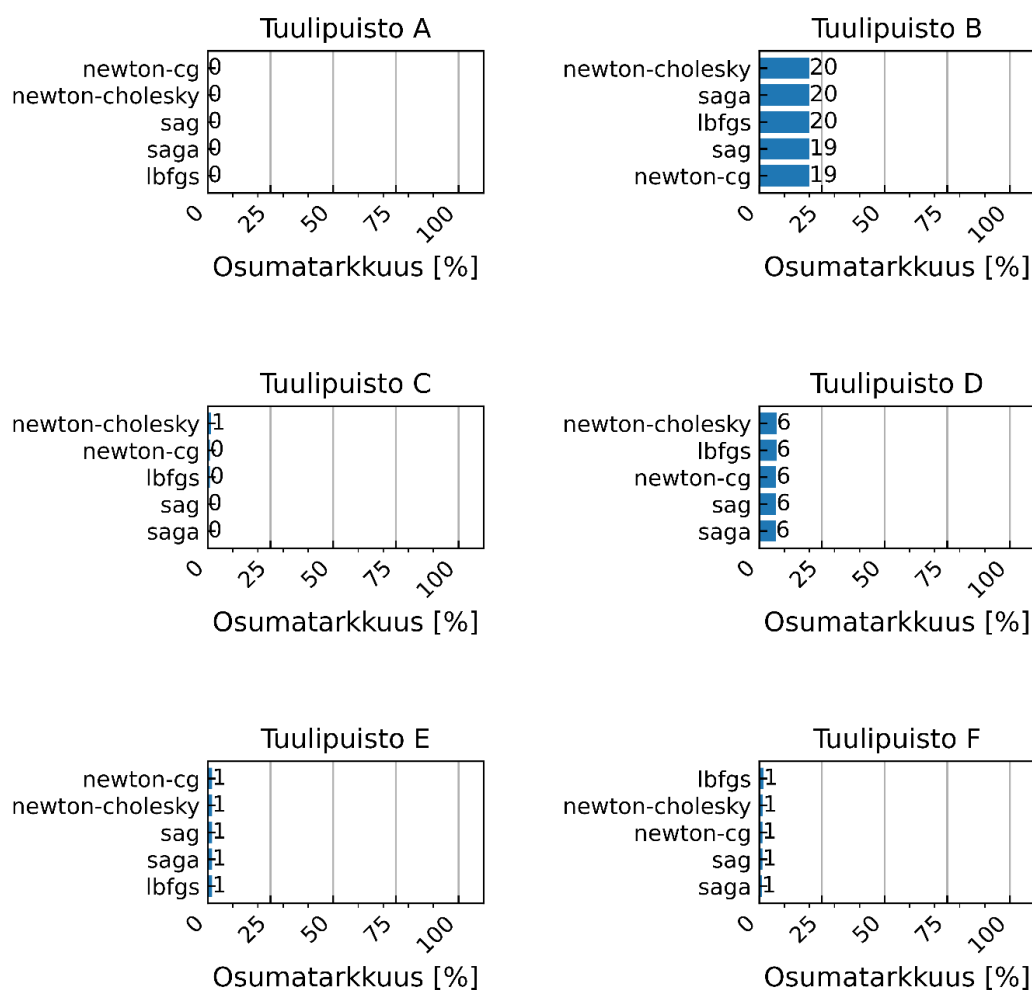


Tukivektorikonemallit eivät siis olleet kovin hyviä osumatarkkuudeltaan. Osumatarkkuus oli parempi tuulipuistoissa B ja E, joissa havaittuja jäätämistapahtumia oli verrattain enemmän. Testejä tehtiin sekä lyhyen että pitkän aikavälin ennusteiden testaamiseen. Testien perusteella tukivektorikone ei välttämättä ole hyvä kumpaakaan käyttöön. Seuraavassa alaluvussa käydään läpi logistisen regression soveltuvuutta.

### 6.6.3 Logistinen regressio

Logistisen regression koulutukseen käytettiin ainoastaan suoria muuttujia ja sklearn - kirjaston eri sovittajia (solver) eli algoritmeja tai menetelmiä. Kuvassa 11 on esitelty parhaimmat mallit, mistä voidaan todeta, ettei logistinen regressio suoriutunut kovin hyvin. Osumatarkkuus oli pyöreä nolla tuulipuistolla A ja 1 % tarkkuus saavutettiin tuulipuistoilla C, E ja F.

Kuva 11. Parhaimmat logistisen regression mallit tuulipuistoittain lyhyen aikavälin ennustamiseen



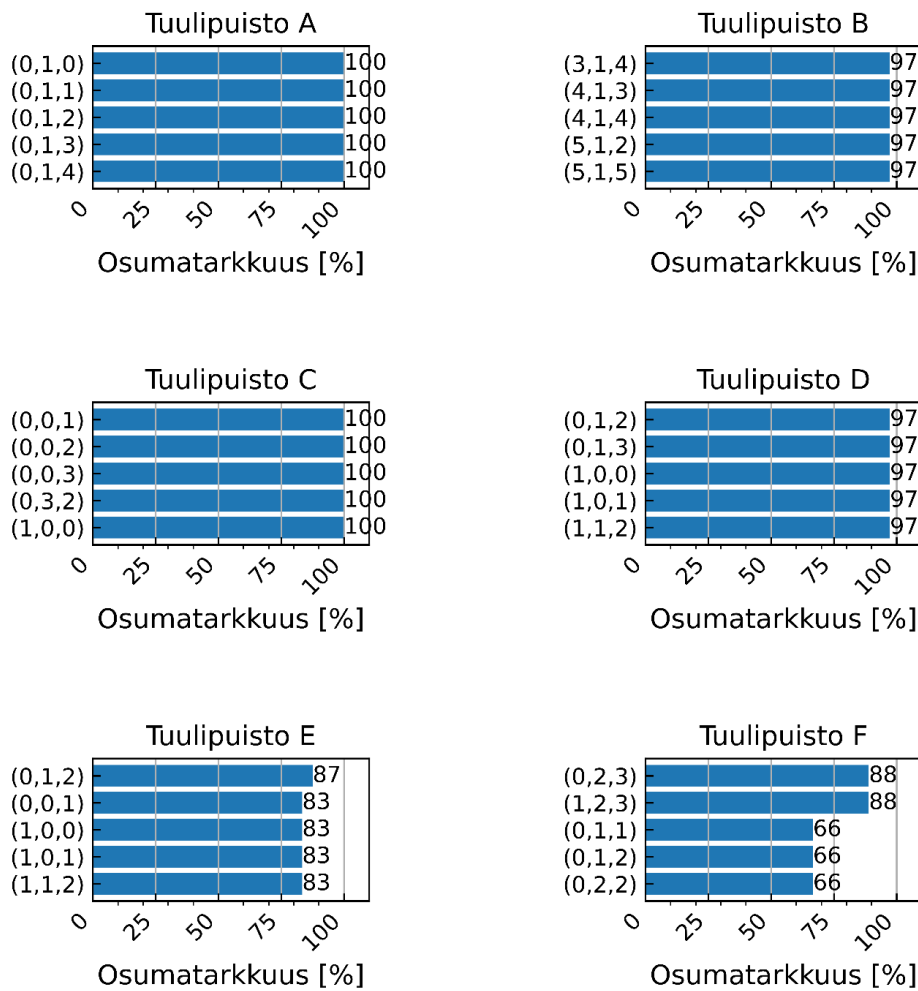
Logistisen regression parhaimmat osumatarkkuudet saatiin tuulipuistoista B ja D. Tuulipuisto B:n tapauksessa logistinen regressio pääsi parhaimmillaan 20 %:n osumatarkkuuteen Newton-Cholesky, SAGA ja L-BFGS (Limited-memory Broyden-Fletcher-Goldfarb-Shanno) -sovittajilla. Testejä tehtiin sekä lyhyen että pitkän aikavälin ennusteiden testaamiseen.

Testien perusteella logistinen regressio ei välttämättä ole hyvä kumpaakaan käyttöön. Seuraavaksi tarkastellaan ARIMA-mallin suoriutumista.

#### 6.6.4 ARIMA

ARIMA-mallina käytettiin statmodels-kirjaston ARIMA-toteutusta. Koska ARIMA-malli ei ottanut sisäänsä kuin ennustettavan parametrin, niin syötettiin muista malleista poiketen ARIMA-mallille ainoastaan IS\_ICING -arvo aikasarjana. Data jaettiin viikon mittaisiin osiin, joista 66 % syötettiin koulutusdatana ja 33 % testinä. Data esikäsiteltiin 1 tunnin keskiarvoiksi, jotta saatiin seuraavaa tuntia ennustava malli. Esikäsitely tehtiin muodostamalla tunnin keskiarvo IS\_ICING -parametrissa. Koska ARIMA ei ole luokitteleva malli, niin keskiarvo muuttaa luokittelun numeraaliseksi. ARIMA-mallin sisäisiä parametreja haettiin brute force -menetelmällä iteroimalla p- (autoregressive), q- (differences) ja e- (moving average) parametreja 0–5 välillä kutakin testaten yhteensä 210 erilaista kombinaatiota. Kuvasta 12 voidaan havaita, että ARIMA pystyi löytämään merkittävän osan tapahtumista. Muodot  $(p=3,q=1,e=4)$ ,  $(p=4,q=1,e=3)$ ,  $(p=4,q=1,e=4)$  ja  $(p=5,q=1,e=5)$  löysivät tapahtumia parhaiten.

Kuva 12. Parhaimmat ARIMA-mallit tuulipuistoittain lyhyen aikavälin ennustamiseen



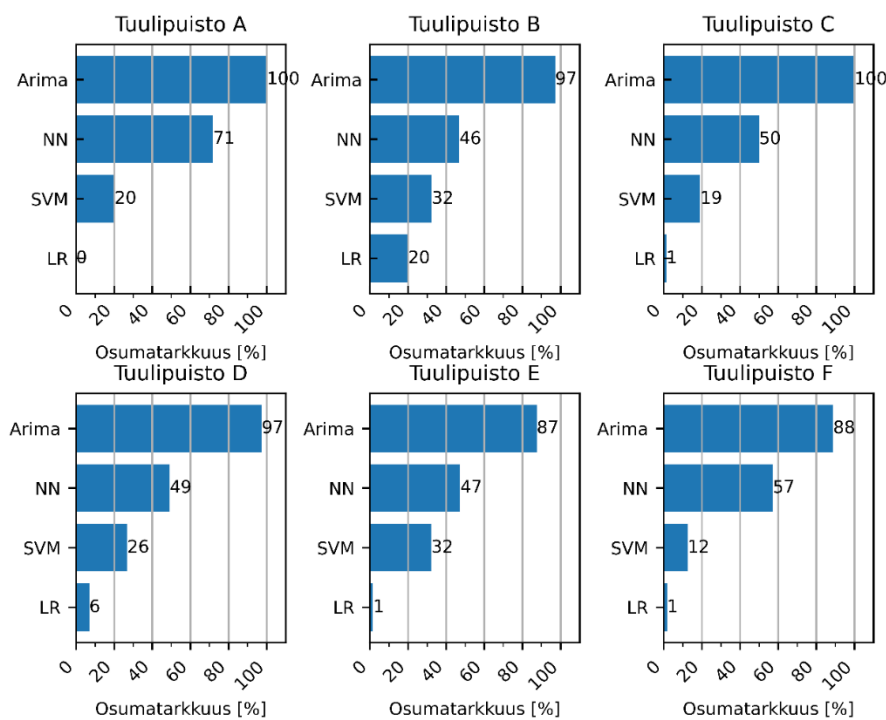
ARIMA löysi tuulipuistojen A ja C kaikki jäätämistapahtumat hieman eri muotoisilla malleilla. Tuulipuistojen E ja F osalta ARIMA ei yltänyt yli 90 %, mutta oli joka tapauksessa osumatarkkuudeltaan silti hyvin tarkka ja parempi kuin muut testatut mallit.

ARIMA vaikuttaisi soveltuvan hyvin varsinkin lyhyen aikavälin ennustamiseen. Pitkän aikavälin ennustamisessa havaittiin testissä haaste sään muuttumisen havainnoissa. Koska ARIMA saa sisääntuloksi vain IS\_ICING -muuttujan, se pystyy ainoastaan havainnoimaan ilman sääennustetta nykypisteestä eteenpäin. Mikäli sää muuttuu, menee ennustaminen täysin pieleen. Seuraavaksi siirrytään mallin valintaan.

## 6.7 Mallin valinta

Testatuista malleista ARIMA oli osumatarkkuudeltaan paras, mutta se soveltuu enimmäkseen lähitulevaisuuden ennustamiseen eli tässä tapauksessa täyttää mallin toisen tavoitteen, jonka tarkoituksena oli ennustaa seuraavan kahdeksan tunnin jäätämistilanne. Mallin ensimmäinen tavoite oli kuitenkin löytää seuraavaa vuorokautta ennustava malli ja siihen käyttöön neuroverkko vaikuttaisi parhaalta valinnalta (kuva 13). Niille voidaan syöttää sääennustetta, jolloin olosuhteita voidaan aidosti katsoa eteenpäin. Sisääntulomuuttujissa pitää olla historiaa eli rullaavia arvoja mukana, jotta saadaan parempia tuloksia.

Kuva 13. Tuulipuistojen jäätämisen ennustamisen osumatarkkuudet malleittain lyhyelle aikavälille



Logistinen regressio ei testien perusteella vaikuttanut sopivan tähän ongelmaan. Logistisen regression osumatarkkuus oli heikko ja se ei löytänyt esimerkiksi Tuulipuistosta A lainkaan jäätämistapahtumia. Tukivektorikoneen (SVM) osumatarkkuus vaihteli tuulipuistoittain 12–32 %:n välillä.

Testien pohjalta paras lopputulos voitaisiin saavuttaa käyttämällä neuroverkkoa ennustamaan sääennusteiden mukaisesti lähipäiviä. Neuroverkoilla tehtyä ennustetta voidaan tarkentaa lähitunneille ARIMA-mallilla.

## 6.8 Parametrien optimointi ja tulokset

Malleja kokeiltiin eri parametreilla ja siten pyrittiin löytämään parhaiten toimivat parametrit ARIMA- ja neuroverkkomalleille. Tuloksia haettiin tuulipuisto kohtaisesti. ARIMA-mallissa testattiin erilaisia p-, q-, e-termejä 0–5 välillä. Neuroverkoissa testaaminen suoritettiin sekä säätämällä sisäistä rakennetta, että alpha toleranssia. Toleranssia säädettiin 0.01–1.0 välillä. Taulukossa 2 on koottuna tuulipuisto kohtaiset parhaiten toimivat ARIMA- ja neuroverkkomallit lyhyen aikavälin jäätämisen ennustamiseen parametreineen ja niiden osumatarkkuudet.

Taulukko 2. Tuulipuistojen parhaimmat jäätämisenennustemallit, parametrit ja osumatarkkuus lyhyelle aikavälille

Tuulipuisto	ARIMA	Neuroverkot
A	(0,1,0) 100%	12x6x1 x alpha=0.1 case7 71.79 %
B	(3,1,4) 97.14 %	16x6x1 x alpha=0.01 case4 46.9 %
C	(0,0,1) 100.0 %	11x6x1 x alpha=0.1 case3 50.0 %
D	(0,1,2) 97.3 %	16x10x1 x alpha=0.01 case3 49.12 %
E	(0,1,2) 87.5 %	12x5x1 x alpha=0.01 case7 47.37 %
F	(0,2,3) 88.89 %	15x8x1 x alpha=0.1 case2 57.14 %

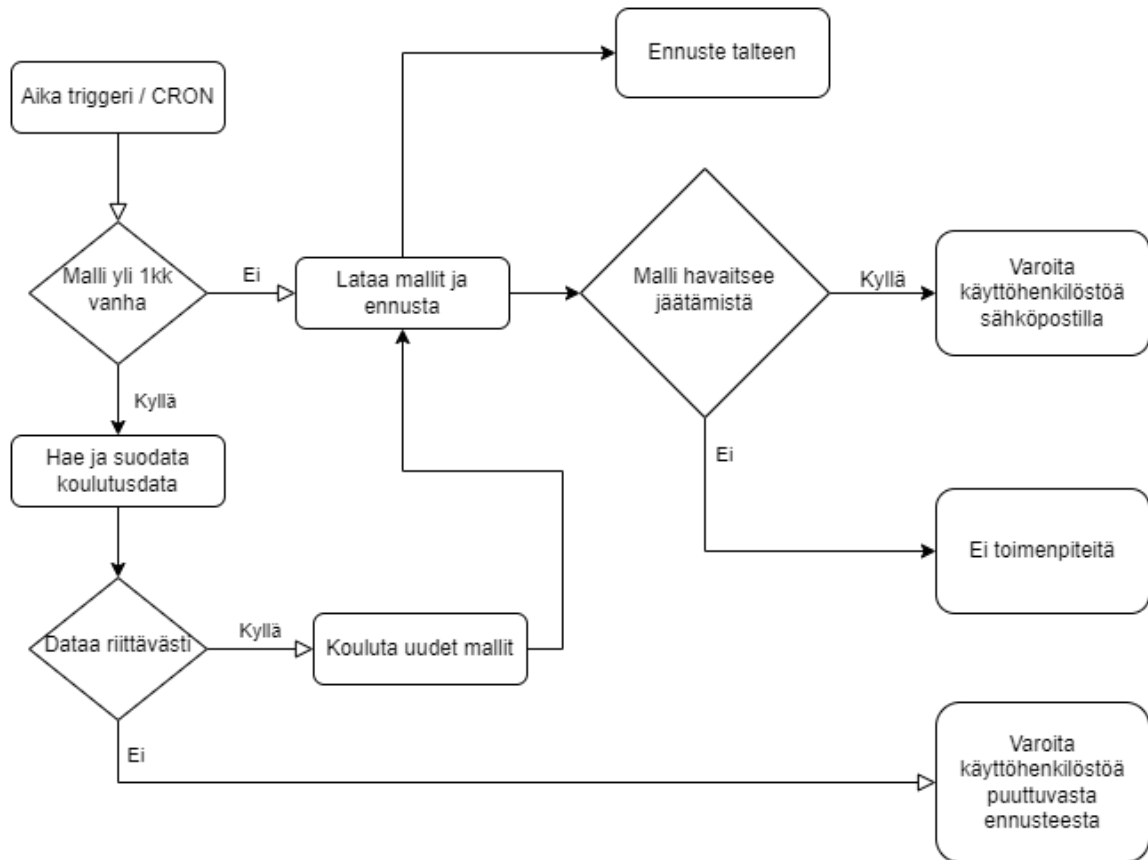
Taulukon 2 tuloksista voidaan havaita, että yhtenäistä kaikille tuulipuistoille sopivaa mallia ei löytynyt. Tästä syystä jokaiselle tuulipuistolle olisi hyvä kouluttaa oma malli ja eri mallirakenteita tulisi testata. Seuraavassa alaluvussa esitellään ohjelmiston rakennesuunnitelma.

## 6.9 Ohjelmiston rakennesuunnitelma

Ohjelman tavoitteena oli pystyä ilmoittamaan käyttöhenkilöille jäätämisen mahdollisuudesta lähiaikoina. Ohjelmisto olisi hyvä rakentaa kahteen osaan taulukon 2 tuloksien mukaisesti. Toinen osuus, joka neuroverkkoja käyttäen ennustaisi sääennusteen pohjalta seuraavan päivän jäätämistä ja ARIMA-mallipohjainen, joka ennustaisi seuraavien käyttötuntien jäätämiskä. Toimiakseen ARIMA-toteutus tarvitsee lähes reaaliaikaisia mittauksia ennustettavista kohteista. Toisaalta sääennusteen pohjalta toimiva neuroverkkototeutus vaatisi kohdedataa ainoastaan koulutusvaiheessa.

Kuvassa 14 on esitetty ohjelmistolle runkorakennetta ottamatta kantaa tietokantaratkaisuihin. Ohjelmiston suoritus käynnistyisi aika tapahtumana (CRON) ja riippuen käytetäänkö ARIMA-toteutusta vai neuroverkkoa, käynnistys olisi kerran vuorokaudessa tai tunnin välein. Tämän jälkeen ohjelmisto tarkistaa mallin koulutuksen iän. Mikäli malli on yli kuukauden vanha, koulutettaisiin uusi malli. Koulutus määrääjain varmistaa, että malli oppii mahdolliset muutokset tuulipuiston käyttäytymisessä. Mallin ollessa riittävän tuore, mallin python pickle ladataan ja sille syötetään mallin vaatima data. Aikasarjassa viimeaikaiset havainnot IS\_ICING -muuttujasta, neuroverkon tapauksessa sääennusteparametrit. Mikäli malli havaitsee jäätämistä, ilmoittaa se siitä sähköpostitse käyttöhenkilöstöä.

Kuva 14. Ennusteohjelmiston rakenne



Mallin koulutus voitaisiin toteuttaa aiemmin läpikäytyjen testien mukaisesti. Data ladataan datajärvestä, suodatetaan ja lasketaan muutama hyväksi havaittu malli. Näistä valitaan parhaiten suoriutuva malli ja se tallennetaan python pickle -muodossa. Mikäli dataa koulutukseen ei ole riittävästi tai sen laatu on huonoa, siitä varoitetaan käyttökäyttöä, jotta ei tule vääriä oletuksia ei jäätävistä olosuhteista.

Ohjelmiston lopullinen toteutus vaatisi kuitenkin erillisen projektin sekä lisää testausta. Testauksia olisi hyvä suorittaa varsinkin marraskuun ja maaliskuun välisenä aikana, sillä tuona aikana jäätämistä tapahtui eniten. Lisäksi ohjelmiston tulisi integroitua vahvasti yrityksen tuotantojärjestelmiin.

## 7 Johtopäätökset ja pohdinta

Työn tavoitteena oli rakentaa toimeksiantajalle koneoppivamalli ennustamaan onko jäätämisen todennäköistä vai ei. Työn lopputuloksena syntyi ohjelmakoodit mallien kouluttamiseen sekä pohjatietoa mallien luotettavuudesta. Lisäksi toteutettiin ohjelmiston runkosuunnitelma tuotanto-ohjelmiston toteutukseen. Tuotanto kelpoisen ohjelmiston toteutus jäi tämän työn ulkopuolelle, sillä työtä ei pystytty testaamaan aidolla ennustetulla. Työtä tehtiin huhtikuusta heinäkuuhun, jolloin jäätämisolosuhteita ei yleensä ilmene.

Opinnäytetyön päätutkimuskysymykseen eli miten rakennetaan malli jäätämisen ennustamiseen, oli melko haastavaa vastata. Työssä käytiin koneoppivan mallin rakentamisen vaiheita teoriatasolla sekä tehtiin testejä eri malleista. Testiin valikoitui tukivektorikone, neuroverkot, logistinen regressio ja ARIMA. Kiinnostusta olisi ollut testata muitakin malleja, mutta rajallisen ajan puitteissa oli valittava vain muutama eri malli. Mallien laskenta vei yllättävän paljon aikaa ja ymmärrys siitä, miksi supertietokoneita tällaisiin tapauksiin yleensä käytetään tuli hyvin selväksi.

Työn haastavuuden johdosta työ oli erittäin opettavainen. Lopputulos ei ollut täysin valmis kokonaisuus, mutta työssä tehtiin kiinnostavia havaintoja. Testien perusteella näytti siltä, että jokaiselle tuulipuistolle olisi hyvä kouluttaa oma malli, sillä tuulipuistojen jäätämisherkkyys vaikutti vaihtelevan paljon. Työn muita tutkimuskysymyksiä olivat seuraavat: mitkä koneoppivat menetelmät soveltuvat ennustamiseen, mitä vaiheita kuuluu mallin rakentamiseen ja mitä jäätämisen on ja miten se vaikuttaa tuulivoimaloihin. Ennustamiseen soveltuvia malleja löydettiin lähteisiin perehtymällä ja näistä valittiin opinnäytetyöhön eniten käytetyt menetelmät eli neuroverkot, tukivektorikone ja logistinen regressio. Opinnäytetyön ohjaajan suosituksesta perehdyttiin edellä mainittujen lisäksi ARIMA-malliin. Tulosten pohjalta testatuista metodeista ARIMA ja neuroverkot vaikuttivat sopivan parhaiten jäätämisen ennustamiseen. Mallin rakentamisvaiheisiin kuuluivat ongelman rajaaminen, datan hankinta, datan esikäsittely, eri mallien tutkiminen ja niistä parhaimpien listaaminen, järjestelmän hienosäätö, ratkaisun esittely ja ylläpito. Jäätämisen tarkoittaa jään kertymistä esineen pinnalle jäätävissä olosuhteissa. Tuulivoimaloissa jäätämisen aiheuttaa tuotannon menetyksiä ja turvallisuusriskin alueella liikkuville.

Jatkokehityksenä olisi mallin testaaminen käytännössä ja sen perusteella mahdollisten parannusten tekeminen. Työssä testaukset tehtiin Ilmatieteen havaintotietoja vasten, mutta tuotanto käytössä mallit pitäisi kouluttaa ennustus muuttujia vasten. Tästä johtuen on mahdollista, että ennusteisiin voi tulla lisävirheitä. Mallin luotettava toiminta olisi hyvä

varmentaa oikeissa olosuhteissa (eli loppusyksyn ja talven välillä) ennen varsinaiseen tuotantokäyttöön ottamista. Tällä hetkellä malli ei ole tuotanto käyttöön soveltuva.

Lisätutkimuksia voisi toteuttaa aikasarjojen ja neuroverkkojen yhdistämisestä, mikä voisi tuottaa tarkempia tuloksia.

## Lähteet

Bell, J. (2020). *Support Vector Machines*. <https://doi.org/10.1002/9781119642183.ch8>

Brink, P. (21.2.2024). Tuulivoiman jäätämisenusteet ylläpitävät sähköjärjestelmän tasapainoa. *Fingrid-lehti*. <https://www.fingridlehti.fi/tuulivoiman-jaatamisennusteet-yllapitavat-sahkojarjestelman-tasapainoa/>

Elements of AI. (n.d.-a). Koneoppiminen: III. Regressio. <https://course.elementsofai.com/fi/4/3>

Elements of AI. (n.d.-b). Neuroverkot. <https://course.elementsofai.com/fi/5>

Energiavirasto. (n.d.). *Markkina- ja energiavirasto*. <https://energiavirasto.fi/markkina- ja energiavirasto>

Géron, A. (2022). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems* (Third edition.). O'Reilly Media, Inc.

Hämäläinen, K. & Niemelä, S. (2016). *Production of a Numerical Icing Atlas for Finland*. *Wind Energy*. <https://doi.org/10.1002/we.1998>

Joseph, M. (2022). *Modern Time Series Forecasting with Python: Explore Industry-Ready Time Series Forecasting Using Modern Machine Learning and Deep Learning*. Packt Publishing.

Kananen, H. & Puolitaival, H. (2019). *Tekoäly: Bisneksen uudet työkalut*. Alma Talent Oy.

Kokkonen, E. (2018). *Tuulivoimatuotannon fyysikaalinen malli* [kandidaatintyö, Lappeenrannan teknillinen yliopisto]. <http://www.urn.fi/URN:NBN:fi-fe201804046336>

Kolari, J. & Kallio, A. (2023). *Tekoäly 123: Matkaopas tulevaisuuteen*. Docendo.

Korstanje, J. (2021). *Advanced Forecasting with Python: With State-of-the-Art-Models Including LSTMs, Facebook's Prophet, and Amazon's DeepAR*. Apress.

- Kreutz, M., Ait-Alla, A., Varasteh, K., Oelker, S., Greulich, A., Freitag, M. & Thoben, K. (2019). Machine learning-based icing prediction on wind turbines. Teoksessa: P. Butala, E. Govekar & R. Vrabič (toim.), 52nd CIRP Conference on Manufacturing Systems (CMS), Ljubljana, Slovenia, June 12-14, 2019. 423–428.  
<https://doi.org/10.1016/j.procir.2019.03.073>
- Kämäräinen, J. (2023). *Koneoppimisen perusteet*. Otatieto.
- Laukkonen, V. & von Kügelgen, M. (2020). *Kaikki koodaa: Päivitä itsesi - käytännön opas ajankohtaisiin digitaitoihin*. Into.
- Meedeniya, D. (2023). *Deep Learning: A Beginners' Guide*. CRC Press LLC.  
<https://doi.org/10.1201/9781003390824>
- Motiva. (3.2.2022). *Tuuliatlas - tuulisuustiedot kartalle*.  
[https://www.motiva.fi/ratkaisut/uusiutuva\\_energia/tuulivoima/tuuliatlas\\_tuulisuustiedot\\_kartalle](https://www.motiva.fi/ratkaisut/uusiutuva_energia/tuulivoima/tuuliatlas_tuulisuustiedot_kartalle)
- Motiva. (16.2.2024). *Tuulivoima Suomessa*.  
[https://www.motiva.fi/ratkaisut/uusiutuva\\_energia/tuulivoima/tuulivoima\\_suomessa](https://www.motiva.fi/ratkaisut/uusiutuva_energia/tuulivoima/tuulivoima_suomessa)
- Müller, A. & Guido, S. (2016). *Introduction to Machine Learning with Python*. O'Reilly Media, Inc.
- Niinimäki, E., Pölönen, I., Rautiainen, I., Tuominen, H. & Äyrämö, S. (2019). 3. Luokittelu- ja ennustemenetelmät. Teoksessa: H. Tuominen & P. Neittaanmäki (toim.), *Tekoälyn perusteita ja sovelluksia*. Jyväskylän yliopisto. <http://urn.fi/URN:ISBN:978-951-39-7796-2>
- Peixeiro, M. (2022). *Time Series Forecasting in Python*. Simon & Schuster.
- Purwar, S. (23.9.2023). ARIMA Model - Understanding ARIMA Model. *Medium*.  
<https://siddp6.medium.com/arima-model-424a1b0f0970>
- Suomen Tuulivoimayhdistys. (n.d.-a). *Jäätäminen*. <https://tuulivoimayhdistys.fi/tietoa-tuulivoimasta-2/tietoa-tuulivoimasta/tuulivoimatuotanto/jaataminen>

Suomen Tuulivoimayhdistys. (n.d.-b). *Tuulivoimatuotannon vaihtelevuus.*

<https://tuulivoimayhdistys.fi/tietoa-tuulivoimasta-2/tietoa-tuulivoimasta/tuulivoimatuotanto/tuulivoimatuotannon-vaihtelevuus>

Suomen Tuulivoimayhdistys. (n.d.-c). *Tuulivoimatuotanto talvella.*

<https://www.tuulivoimayhdistys.fi/media/tuulivoimatuotanto-talvella.pdf>

Uusitalo, H. (2022). *Analysis and Modeling for Predicting Icing Losses in Finnish Wind Power*

*Production* [opinnäytetyö, Aalto-yliopisto]. <https://urn.fi/URN:NBN:fi:aalto-202209045324>

## **Liite 1. Aineistonhallintasuunnitelma**

Opinnäytetyö ja siihen liittyvä aineisto säilytetään tekijän oman tietokoneen C-aseamalla. Varmuuskopiot tulevat erilliselle kovalevylle. Vain tekijällä on pääsy aineistoon.

Toimeksiantaja pidetään anonyymina. Tuulipuistojen aineisto kuuluu toimeksiantajalle. Alkuperäinen aineisto sisältää jonkin verran tuulipuistojen arkaluontoisia tietoja, joten aineiston muuttujia anonymisoidaan siten, ettei yksittäistä tuulipuistoa ole mahdollista tunnistaa.

Opinnäytetyötä ja sen aineistoa säilytetään ainakin vuoden verran työn valmistumisesta tekijän tietokoneella ja erillisellä kovalevyllä.