



TEKNILLINEN TIEDEKUNTA

**Neuroverkkomallin valinta Barkhausen-kohinasta
jäännösjännityksen ennustamiseksi**

Olli Pesonen

PROSESSITEKNIIKAN TUTKINTO-OHJELMA

Kandidaatintyö

Syyskuu 2020

TIIVISTELMÄ

Neuroverkkomallin valinta Barkhausen-kohinasta jäännösjännityksen ennustamiseksi

Olli Pesonen

Oulun yliopisto, Prosessitekniikan tutkinto-ohjelma

Kandidaatintyö 2020, 42 s.

Työn ohjaajat yliopistolla: DI Tero Vuolio, TkT Aki Sorsa

Barkhausenin kohinaan perustuvaa mittausta voidaan hyödyntää ferromagneettisille materiaaleille ja mittauksen avulla voidaan mallintaa materiaalin ominaisuuksia esimerkiksi jäännösjännityksen sekä kovuuden osalta. Barkhausen-mittaus on niin kutsuttu rikkomaton aineenkoetuksen menetelmä, joka mahdollistaa sen hyödyntämisen esimerkiksi teollisuuden kunnossapitokohteissa. Barkhausen kohina syntyy asetettaessa tutkittava ferromagneettinen kappale muuttuvaan magneettikenttään. Muuttuva magneettikenttä aiheuttaa materiaalissa magneettisten alkeisalueiden satunnaisia muutoksia ja liikkeitä, jolloin mittalaitteiden avulla syntyvästä kohinasignaalista on mahdollista irrottaa mallinnettavia piirteitä.

Tämän kandidaatintyön tarkoituksena on selvittää, kuinka hyvin keinotekoinen neuroverkko kykenee ennustamaan hyvinkin stokastisen Barkhausen-ilmiön pohjalta teräskappaleen jäännösjännitystä. Kiinnostuksen kohteena on myös kahden neuroverkkomallin muodostamiseen yhdistetyn deterministisen eteenpäin-valinta-algoritmin suorituskyky. Algoritmien tehtävänä on valita neuroverkkomalliin paras mahdollinen muuttujajoukko ja neuroverkon piilokerroksen neuronien lukumäärä annetuin koeparametrein. Mallien hyvyttä arvioidaan mallin selitysasteen ja absoluuttisen keskivirheen avulla.

Mallinnuksen vaiheita ovat mittausdatan keruu ja sen esikäsittely, muuttujien valinta sekä mallin validointi. Mittausdatana käytetään aiemmassa Barkhausen-tutkimuksessa mitattua ja esikäsiteltyä dataa. Mittausdata jaetaan opetusaineistoon sekä sisäiseen ja ulkoiseen validointiaineistoon. Muuttujien valinta ja neuroverkon piilokerroksen neuronien lukumäärä valitaan eteenpäin-valinta-algoritmeilla. Muuttujien valinta-

algoritmi käyttää opetusaineistoa sekä sisäistä validointiaineistoa. Lopuksi muodostettu neuroverkkomalli validoidaan staattisesti valitulla ulkoisella validointiaineistolla. Työssä tutkitaan myös valmiin neuroverkkomallin Bayes-regularisoinnin vaikutusta mallin suorituskyvylle.

Keinotekoiset neuroverkot kykenevät ennustamaan hyvinkin monimutkaisia ilmiöitä, mutta lisätessä tarkkuutta myös laskenta vaatii enemmän resursseja. Siksi keinotekoisien neuroverkkomallien muodostaminen on tehtävä huolella optimoiden, jotta esimerkiksi raskas eteenpäin-valinta-algoritmi on samalla tehokas sekä muodostetun mallin hyvyyden kannalta että laskennallisesti. Saaduista tuloksista voidaan kuitenkin nähdä, että keinotekoiset neuroverkot ovat hyödyllisiä teräksen jäännösjännitystilojen ennustamisessa Barkhausen-kohinasignaalia.

Asiasanat: Barkhausenin kohina, keinotekoiset neuroverkot, matemaattinen mallinnus, muuttujien valinta

SISÄLLYSLUETTELO

1 Johdanto	5
2 Barkhausenin kohina.....	7
2.1 Materiaaliominaisuuksien ennustaminen Barkhausen-kohinasignaalista	7
3 Matemaattinen mallinnus	9
3.1 Datan esikäsittely	9
3.2 Mallirakenteen valinta.....	10
3.2.1 Lineaarinen monimuuttujaregressio.....	10
3.2.2 Keinotekoiset neuroverkot.....	11
3.3 Muuttujien valinta	13
3.3.1 Suodatus	14
3.3.2 Wrapper	14
3.3.3 Eteenpäin-valinta ja taaksepäin-eliminointi	14
3.3.4 Geneettiset algoritmit	17
3.4 Mallin validointi.....	18
4 Koesuunnittelu.....	21
4.1 Keinotekoinen neuroverkkomalli.....	21
4.2 Kokeissa käytetty data.....	23
4.3 Koeparametrit ja niiden muuttaminen.....	24
5 Tulokset ja pohdinta.....	27
5.1 Mallin hyvyys	27
5.2 Koeparametrien vaikutus mallin hyvyyteen	30
5.3 Bayes-regularisoinnin merkitys mallin hyvyyteen	34
5.4 Malleissa esiintyvät muuttujat	35
6 Johtopäätökset ja yhteenveto.....	38
Lähdeluettelo	41

1 JOHDANTO

Rikkomaton aineenkoetus (engl. *non-destructive testing*) on tärkeä tutkimuskohde nyt ja tulevaisuudessa. Alati kehittyvät mittausmenetelmät ja uudenlaiset mittalaitteet tuottavat suuren määrän entistä tarkempaa dataa, jota on mahdollista hyödyntää niin prosessien ohjauksessa, tuotteiden laatutarkastuksissa kuin prosessilaitteiden kunnossapidossakin. Mitattava ilmiö ei aina anna mahdollisuutta tehdä johtopäätöstä ilmiön tilasta pelkän mittaussuureen perusteella, vaan avuksi voidaan tarvita malli, jonka avulla ilmiötä pyritään parhaan mukaan selittämään. Englantilaisen tilastotieteilijän George E.P. Boxin vapaasti käännetty sanonta ”*kaikki mallit ovat väärinä, mutta jotkut niistä ovat hyödyllisiä*” (1976) luo pohjan kriittiselle ajattelulle myös Barkhausen-ilmiön mallintamisessa.

Barkhausen-ilmiötäkään ei voida mitata suoraan, vaan ennustettavat mekaaniset ominaisuudet ennustetaan Barkhausen-signaalin avulla, joka on riippuva mitattavan materiaalin magneettisista ominaisuuksista. Useat Barkhausen-tutkimukset ovat osoittaneet menetelmän käyttökelpoisuuden ferromagneettisten materiaalien ominaisuuksien tarkastelussa. Kun haluttuun ilmiöön tai suureella kuvattavaan ominaisuuteen päästään käsiksi usean vaiheen kautta, myös virhe on monen tekijän summa. Mittalaite tai -menetelmä voi tuottaa systemaattista virhettä ja kun tähän yhdistetään mallinnusvaiheen oletusarvoinen satunnainen virheellisyys, tulee tulosta tarkastella hyvinkin kriittisesti. Siitä huolimatta mallinnus on yhä tärkeämmässä osassa muun muassa prosessiteollisuuden sovelluskohteissa. Laskentakapasiteetin ja mittausaineiston lisääntyminen mahdollistaa mallintamisen yhä tarkemmalla tasolla monimutkaisempia malleja käyttäen. Mittausaineiston lisääntyminen ei kuitenkaan ole keskiössä, vaan mitatun aineiston laatu. Mikäli mittausaineisto on laadultaan huono, on siitä luotu mallikin huono, vaikka käytettäisiin monimutkaisiakin menetelmiä.

Barkhausenin kohinaa on mallinnettu keinotekkoisten neuroverkkojen lisäksi myös lineaaristen monimuuttujaregressioiden sekä pääkomponenttiregressioiden avulla. Muuttujien valinnassa on hyödynnetty eteenpäin-valinta-algoritmien lisäksi muun muassa geneettisiä algoritmeja, taaksepäin-valintaa sekä simuloitua jäähtymistä. (Sorsa, 2013).

Työssä on tarkoituksena selvittää keinotekoisien neuroverkkomallin hyödynnettävyyttä teräksen jäännösjännityksen ennustamisessa Barkhausenin kohinasignaalista. Tavoitteena työn kokeellisessa osassa on saada irti neuroverkkomallin hyvyttä kuvaavista luvuista perustelua neuroverkon soveltuvuudelle ennustaa jäännösjännitystä Barkhausenin kohinasignaalin perusteella.

2 BARKHAUSENIN KOHINA

Barkhausen-kohina on ferromagneettisilla aineilla esiintyvä ilmiö, joka syntyy asetettaessa kappale ulkoiseen magneettikenttään. Kappale magnetisoituu ulkoisen magneettikentän vaikutuksesta ja magneettisuudessa havaitaan muutoksia ulkoisen magneettikentän vahvistuessa ja heikentyessä. Barkhausen-kohinaa voidaan hyödyntää kappaleiden mikrorakenteen tutkimisessa, sillä kohinan luonteella on selvä yhteys materiaalin ominaisuuksiin. Barkhausen-kohinaa mittaamalla voidaan ennustaa esimerkiksi kappaleen mikrorakennetta, jännitystiloja ja kovuutta. (Jiles, 2000) Barkhausen-kohinamittaus on niin kutsuttu rikkomattoman aineenkoetuksen menetelmä.

Barkhausen-kohinan synty johtuu magneettisiin alkeisalueisiin liittyvistä muutoksista, joita ulkoinen muuttuva magneettikenttä aiheuttaa. Magneettisilla alkeisalueilla magneettiset dipolimomentit ovat samansuuntaiset ja yksittäisten alkeisalueiden dipolimomenttien suunnat ovat satunnaisia, mikäli kappale ei ole ulkoisessa magneettikentässä. Suunnat asettuvat siten, että kappale ei itsessään ole magneettinen. Kohina johtuu pääosin magneettisten alkeisalueiden seinämien liikkeestä ulkoisen magneettikentän muuttuessa vahvistuen tai heikentyen. Materiaalin rakenteessa olevat poikkeamat rajoittavat rajapintojen liikkeitä ja rajapinnoilla tapahtuvat äkilliset muutokset aiheuttavat muutoksia kappaleen magneettisuudessa. Magneettisuuden muutokset voidaan mitata ja näin saadaan aikaan Barkhausen-kohinasignaali. (Jiles, 2000)

2.1 Materiaaliominaisuuksien ennustaminen Barkhausen-kohinasignaalista

Sorsa et al. (2012) tutki pintakarkaistun teräksen jäännösjännitystilojen ja kovuuden ennustettavuutta Barkhausen-kohinamittauksen pohjalta. Mallin rakennus koostui signaalin esikäsittelystä, ennustemuuttujien valinnasta, mallin identifioinnista sekä mallin validoinnista. Esikäsittelyvaiheessa signaali suodatettiin liukuvan neliöllisen keskiarvon (RMS) ja entropian suodatuksilla ja datasta poistettiin epäolennaisia osia, kuten vajaita magnetointikiertoja. Signaalista voidaan irrottaa erilaisin matemaattisin menetelmin piirteitä, joita Sorsan et al. (2012) työssä olivat esimerkiksi Barkhausen-profiilin huipun korkeus sekä leveys, remanenssi ja koersiivisyys. Barkhausen-profiileista lasketuista

piirteistä sekä suoraan mittausvälineistön tuottamista muuttujista aikaansaatiin 72 muuttujan aineisto. Muodostetun lineaarisen ennustemallin muuttujat valittiin deterministisellä eteenpäin-valinta-menetelmällä. Mallin validointi tehtiin käyttämällä leave-one-out-ristiinvalidointia. Mallin suorituskykyä pidettiin hyvänä Barkhausen-kohinan stokastistisen luonteen aiheuttamat haasteet huomioon ottaen. Mallin korrelaatiokerroin opetusaineistolle jäännösjännitystä ennustaessa oli 0,88 ja kovuutta ennustaessa 0,96. Vastaavat korrelaatiokertoimien arvot validointiaineistolle olivat 0,94 ja 0,76. Tutkimuksessa ehdotettiin mallintamisen muuttujanhaun suorittamista stokastisten menetelmien kuten geneettisten algoritmien avulla, jotta muuttujanvalinnassa päädyttäisiin suuremmalla todennäköisyydellä globaaliin optimiin paikallisen optimin sijasta. (Sorsa et al. 2012)

Sorsa et al. (2011) vertaili Barkhausen-kohinan mallinnuksen muuttujanvalintamenetelmiä. Stokastisista hakumenetelmistä käytössä olivat geneettiset algoritmit sekä simuloitu jäähtyminen (simulated annealing) ja deterministisistä hakumenetelmistä eteenpäin-valinta sekä taaksepäin-eliminointi. Tutkimuksessa todettiin, että stokastiset muuttujanvalintamenetelmät suoriutuvat ennustemuuttujien valinnasta merkittävästi paremmin kuin deterministiset menetelmät. Kaikki tutkitut hakumenetelmät todettiin hyödyllisiksi muuttujanvalinnassa taaksepäin-eliminointia lukuun ottamatta.

Neuroverkkomallin soveltamista Barkhausenin kohinasignaalille teräksen jännityksen ennustamiseksi on tutkinut muun muassa Wang et al. (2013). Keinotekoisien neuroverkon painokertoimien optimointi perustui back propagation -algoritmiin. Neuroverkon tulokerroksen tulomuuttujia oli kuusi ja niitä olivat muun muassa signaalin neliöllinen keskiarvo, lämpötila ja signaalin huipun arvo. Keinotekoisien neuroverkon lähtömuuttujana oli ennustettava jännitys. Olennainen kysymys keinotekoista neuroverkkoa rakentaessa on piilokerroksen neuronien lukumäärä. Liian vähäinen neuronien määrä aiheuttaa mallin ennustuskyvyn heikkenemistä ja liian suuri määrä taas voi aiheuttaa ylisovitusta syötetylle datalle. Tutkimuksessa neuronien lukumäärä valittiin aiempien empiiristen tutkimusten perusteella. Sekä ensimmäiselle että toiselle piilokerrokselle tuli valituksi 12 neuronia. Optimaalinen malli saavutettiin 405 opetuskerran jälkeen ja tuloksia pidettiin yleisesti tarkkoina ja mallin yleistyskykyä hyvänä. Mallin ennusteen suhteellisen virheen arvo oli korkeimmillaan 7,4369 % ja suhteellisen virheen keskiarvo oli 1,0618 %. (Wang et al., 2013)

3 MATEMAATTINEN MALLINNUS

Mallinnuksen voidaan ajatella koostuvan neljästä vaiheesta: mittausaineiston esikäsitteystä, mallirakenteen valinnasta, mallin parametrien identifioinnista, sekä mallin validoinnista ja hyvyyden arvioinnista. Jotta mallin suorituskykyä voidaan arvioida mahdollisimman realistisesti, mittausaineisto jaetaan tyypillisesti opetus- ja testausaineistoihin, millä halutaan varmistaa mallin riittävä yleistyskyky. Mallin rakenne voi olla esimerkiksi lineaarinen regressio yhdellä tai usealla muuttujalla, osittainen pienimmän neliösumman regressio (PLSR), pääkomponenttiregressio (PCR), tai keinotekoinen neuroverkko. Mallin testauksella varmistetaan mallin toimivuus eli yleistyskyky opetusaineiston ulkopuolisen testausaineiston avulla. (Baumann, 2003)

3.1 Datan esikäsitteily

Mittausdatan esikäsitteily on erittäin tärkeä muuttujan valintaa ja mallin identifiointia edeltävä vaihe. Epätarkkaa ja käsittelemätöntä dataa käytettäessä on riski tehdä vääriä johtopäätöksiä datasta luodun mallin avulla. Datan joukossa voi olla esimerkiksi perusjoukosta huomattavastikin poikkeavia havaintoja (engl. *outlier*) mittaustilanteessa tapahtuneen virheen takia. Esikäsitteilyssä dataa voidaan esimerkiksi suodattaa, täydentää, standardoida ja korjata. Tyypillinen suodatustapa on tehdä raakasignaalin liukuvan keskiarvon suodatus, jossa lasketaan n perättäisen mitta-arvon keskiarvo. (Runkler, 2012) Raakasignaalia suodattamalla saadaan yleensä kasvatettua signaali-kohina-suhdetta, ja näin ollen vähennettyä signaalista laskettujen piirteiden satunnaisvirhettä.

Sorsa (2013) käytti materiaalien ominaisuuksien ennustettavuutta tutkineessa väitöstutkimuksessaan Barkhausenin kohinasignaalin esimerkiksi datapisteiden poistoa vajaille magnetointikiertoille ja liukuvan keskiarvon suodatusta raakasignaalin. Datan esikäsitteily on erityisen tärkeää Barkhausen-kohinatutkimisessa, sillä mittaus tapahtuu epäsuorasti. Barkhausen-kohinalla pyritään ennustamaan materiaalin mekaanisia ominaisuuksia, joihin päästään käsiksi Barkhausen-signaalin avulla. Barkhausen-signaaliin itsessään vaikuttavat materiaalin magneettiset ominaisuudet. (Sorsa 2013)

3.2 Mallirakenteen valinta

Mallin rakennetta valittaessa on hyvä tuntea, millainen tutkittava ilmiö on ja tarkastella millaisia riippuvuuksia selitemuuttujien ja ennustettavien muuttujien välillä vallitsee. Mallintamisen tavoitteena on saada mahdollisimman hyvä matemaattinen kuvaus mitatusta ilmiöstä ja täten mahdollisesti hyödyllinen tapa ennustaa sitä. (Guyon & Elisseeff, 2003)

3.2.1 Lineaarinen monimuuttujaregressio

Linearisella regressiomallilla voidaan kuvata hyvin tilannetta, jossa selitettävä muuttuja on lineaarisesti riippuvainen selittävästä muuttujista. Selitettävä muuttuja (y) on siis mallissa vastemuuttuja ja sen vaihtelua pyritään selvittämään selittävien muuttujien (x_i) avulla. Lineaarinen monimuuttujaregressiomalli voidaan kirjoittaa seuraavasti:

$$y = b_0 + b_1x_1 + b_2x_2 + \dots + b_nx_n = b_0 + \sum_{i=1}^n b_ix_i, \quad (1)$$

missä y on selitettävä muuttuja, x_i on selittävä muuttuja ja b_i on selittävää muuttujaa vastaava regressiokerroin, jonka arvo kertoo estimoidun vaikutuksen suuruuden ja suunnan. Yhden muuttujan lineaarisessa regressiossa selittäviä muuttujia on yksi, kun taas monimuuttujamallissa selittäviä muuttujia on useita. Linearisessa regressiomallissa esiintyy myös regressioparametrejä (b_n), joista b_0 on vakiotermi ja muut regressiokertoimia. Regressiokertoimet ovat selittävien muuttujien painokertoimia eli ne kuvaavat, kuinka paljon selitettävä muuttuja muuttuu, kun selittävä muuttuja muuttuu yhden yksikön verran. (Gujarati, 2018)

Selittävät muuttujat voivat olla jatkuva-arvoisia, kuten massa ja lämpötila tai kategorisia, kuten sukupuoli ja kansalaisuus. Kategorisia muuttujia kuvataan tyypillisesti kokonaisluvuilla. Kun halutaan ennustaa jatkuva-arvoisten muuttujien vaihtelua, on kyseessä regressio-ongelma. Mikäli ennustettava muuttuja on kokonaisluku, on kyseessä regressio-ongelman sijaan luokitteluongelma. On kuitenkin huomattava, että myös jatkuva-arvoisista muuttujista voidaan tehdä kategorisia diskretoimalla. Kategoristen muuttujien erikoistapauksia ovat niin kutsutut dummy-muuttujat. Dummy-muuttujien arvot määritellään olevan joko 0 tai 1. Muuttujan arvo on tällöin 0, jos kyseinen

ominaisuus ei ole voimassa ja 1, mikäli ominaisuus on voimassa. On huomattava, että dummy-muuttujien koostamiseen vaaditaan useampi sarake ($K - 1$ saraketta, jossa K on kategorioiden määrä), kuin alkuperäisen kategorisen muuttujan esittäminen vaatii. (Gujarati 2018)

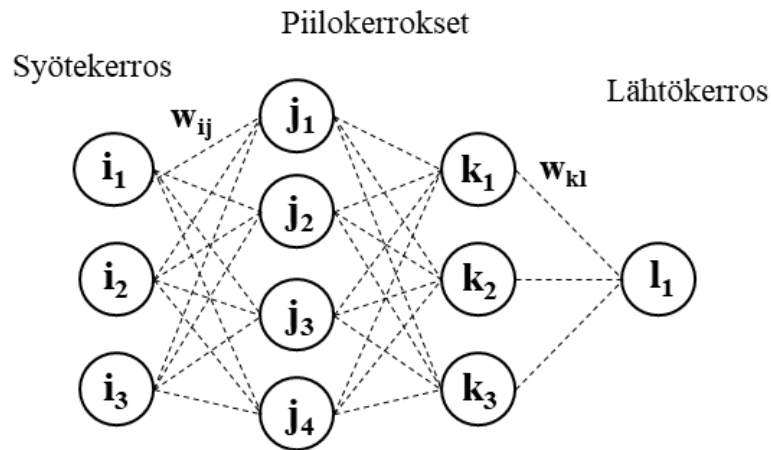
Regressiomallin hyvyttä voidaan arvioida R^2 -arvolla eli mallin selitysasteella. Mallin selitysaste kuvaa, kuinka paljon testausaineiston selitettävän muuttujan varianssia pystytään selittämään valitulla mallilla. Selitysaste voi yleensä saada arvoja väliltä $[0,1]$ ja esimerkiksi mikäli $R^2 = 0,54$ voidaan 54 % selitettävän muuttujan havainnoista selittää mallin avulla. Mallin selitysaste saadaan kaavalla:

$$R^2 = 1 - \frac{SS_R}{SS_T} = \frac{SS_E}{SS_T}, \quad (2)$$

missä SS_T on kokonaisneliösumma, SS_E on residuaalineliosumma ja SS_R on regressioneliösumma. Kokonaisneliösumma kuvaa y-havaintojen kokonaisvaihtelua eli kuinka paljon mitatut arvot poikkeavat mitattujen arvojen keskiarvosta. Residuaalineliosumma taas kuvaa sitä, kuinka paljon ennustetut arvot poikkeavat mitatuista arvoista. Regressioneliösummalla kuvataan kuinka paljon ennustetut arvot poikkeavat mitattujen arvojen keskiarvosta. (Gujarati, 2018)

3.2.2 Keinotekoiset neuroverkot

Keinotekoinen neuroverkko on tapa mallintaa ilmiöitä matkien biologisten hermoverkkojen toimintaa. Keinotekoinen neuroverkko koostuu syöte-, piilo- ja lähtökerroksista. Kuvassa 1 on esitetty rakennekaavio keinotekoiselle monikerrosneuroverkolle.



Kuva 1. Kaaviokuva keinotekoisesta monikerrosneuroverkosta (Mukaiillen Lecun et al. 2015)

Keinotekoisessa neuroverkossa informaation voidaan ajatella virtaavan kerrosrakenteen läpi. Täysin kytketyssä keinotekoisessa neuroverkossa syötekerroksen neuronit (kuvassa neuronit i) ovat yhdistetty jokaiseen piilokerroksen neuroniin (j ja k), joiden lukumäärä on valittavissa neuroverkkoa luodessa. Jokainen piilokerroksen neuronille saapuva syöte on painotettu painokertoimella w_i ja nämä painotetut syötteet summataan. Piilokerroksen neuronin lähtönä on aktivaatiofunktion saama arvo, kun aktivaatiofunktion tulona on painotettujen syötteiden summa. Piilokerros voi itsessään koostua useista kerroksista eli piilokerroksen neuronin lähtö ohjataan joko seuraavalle piilokerrokselle kaikkiin neuroneihin tai neuroverkon lähtökerrokseen. (Lecun et al. 2015)

Edelleen lähtökerroksella jokaisen neuronin (l) aktivaatiofunktioon syötetään tulona piilokerrokselta saatu lähtöarvo. Aktivaatiofunktion tarkoitus on siis saada aikaan neuronille saapuneiden tulojen painotetusta summasta yksittäinen lähtöarvo, joka voidaan syöttää edelleen seuraavan kerroksen neuronille. Regressio-ongelmien ratkaisussa neuroverkon aktivaatiofunktiona voi toimia esimerkiksi jokin sigmoidinen käyrä, kuten logistinen funktio tai hyperbolinen tangentti. Neuroverkon opetus perustuu neuronien lähtö- ja tavoitearvojen välisen erotuksen minimoimiseen. Tavoitteena on siis löytää pienin mahdollinen opetusvirhe. Hyvin tyypillisesti neuroverkon painokertoimia ei kyetä määrittämään neuroverkkoa luodessa ja ne joudutaan määrittämään satunnaisesti. Neuronien lähtöarvojen virheet johtuvat neuronille saapuneiden tulojen painokertoimissa olevista poikkeamista. Tätä varten keinotekoisesta neuroverkkoa on opetettava eli on

löydettävä painokertoimet, jotka minimoivat virheen neliösumman ja mahdollistavat neuroverkon hyvän yleistyskyvyn. Opetusta varten muodostetaan mallinnusvirheeseen perustuva kustannusfunktio. Tyypillisesti optimointiongelman ratkaisu pyritään löytämään siten, että opetusvirheen minimoinnin lisäksi neuroverkko yleistyy hyvin uudelle datalle. Eräs tällainen ratkaisu voi löytyä kustannusfunktion globaalista minimistä etenkin, jos kustannusfunktio on regularisoitu. Kustannusfunktion globaali minimi saavutetaan siis silloin, kun ei ole mahdollista löytää painokerroinvektoria, jolla kustannusfunktion arvo olisi pienempi kuin jo löydetty. Painokertoimien optimoinnissa hyödynnetään usein osittaisderivointiin perustuvaa back propagation -algoritmia, jolla saatujen tulosten perusteella painokertoimia muutetaan. (Lecun et al., 2015)

3.3 Muuttujien valinta

Mittausteknologian kehittyessä saatavilla olevan mittausdatan määrä kasvaa. Suuret määrät dataa sisältävät usein myös suuria määriä mallinnuksen näkökulmasta epäolennaisia muuttujia. Epäolennaisten muuttujien käyttö voi johtaa tilanteesta kehitetyn mallin suorituskyvyn heikentymiseen, mallin vaikeampaan tulkintaan ja mallin opettamiseen kuluneen ajan kasvamiseen. Muuttujien valinnan merkitys korostuu ja toisaalta ylisovituksen riski on suurempi, kun tarkastellaan mittausaineistoja, joissa havaintojen määrä on pieni. Näin ollen sopivan ennustemallin valinnassa on keskeistä pystyä valitsemaan olennaisimmat muuttujat koko datajoukosta. Osa muuttujista voi olla esimerkiksi kollineaarisia toisten suhteen. Toisaalta muuttujat voivat olla yksinään hyödyttömiä, mutta hyödyllisiä yhdessä muiden muuttujien kanssa, mikä aiheuttaa riskin hyödyllisen muuttujan hylkäykselle. Eräs keskeinen riski epäoleellisten muuttujien käytössä on myös mahdollisuus ylisovittamiseen, jolloin luotu malli kuvastaa opetusdataa liian hyvin, mutta yleistyy huonosti opetusdatan ulkopuoliseen aineistoon. Tällöin data, jolla malli on opetettu voi tulla ennustetuksi täydellisesti, mutta uudet havainnot eivät selity opetusaineiston pohjalta luodulla mallilla ollenkaan tai ainoastaan heikosti. (Guyon & Elisseeff, 2003)

Muuttujien valinta voidaan jakaa kolmeen menetelmään, suodatus- (filter), wrapper- sekä sulautettuihin (embedded) menetelmiin. Suodatuksessa muuttujat valitaan ainoastaan käytössä olevan datan perusteella siten, että muodostetaan muuttujille järjestys jonkin kriteerin perusteella. Wrapper-menetelmässä muuttujat valitaan opetusaineiston avulla oppimistehtävää hyödyntäen. Embedded- eli yhdistelmämenetelmissä yhdistetään

parametrien regularisointi ja muuttujienvälyntä. Esimerkki tällaisesta menetelmästä on Lasso-regressio. (Guyon & Elisseeff, 2003)

3.3.1 Suodatus

Suodatuksessa muuttujien järjestyksen aikaansaamiseksi tehtävä muuttujien arvotus voidaan tehdä esimerkiksi korrelaation perusteella. Muuttujia valitaan tai eliminoidaan järjestyksen perusteella yksi kerrallaan niin kauan, kun mallin suorituskyky paranee. Suodatusmenetelmän etu on sen laskennallinen keveys ja tehokkuus, mutta löydetty muuttujajoukko on harvoin kuitenkaan optimaalinen. (Guyon & Elisseeff, 2003)

Toisaalta suodatus voidaan mieltää myös deterministiseksi menetelmäksi, sillä suodatuksella voidaan yrittää tehostaa deterministisiä eteenpäin-valinta-algoritmeja, johonkin suodatusheuristiikkaan perustuen.

3.3.2 Wrapper

Wrapper-menetelmät voidaan jakaa edelleen deterministisiin ja stokastisiin hakumenetelmiin. Deterministiset hakumenetelmät kuten eteenpäin-valinta ja taaksepäin-eliminointi ovat vaiheittaisia menetelmiä. Eräs deterministisen haun ongelmista on haun juuttuminen paikalliseen optimiin. Deterministiset hakutavat ovat laskennallisesti raskaampia kuin suodatusmenetelmät, mutta tyypillisesti kevyempiä kuin stokastiset menetelmät. (Benoudjit et al. 2004; Guyon & Elisseeff, 2003)

3.3.3 Eteenpäin-valinta ja taaksepäin-eliminointi

Eteenpäin-valinta etenee siten, että aluksi identifioidaan n kappaletta malleja, jossa n on muuttujien lukumäärä. Mallien suorituskyky testataan ja parhaan mallin muuttuja tulee valituksi alussa tyhjänä olleeseen malliin. Seuraavaksi identifioidaan $n - 1$ kappaletta malleja ja edelleen suorituskyvyn testauksen jälkeen paras kahden muuttujan malli tulee valituksi. Tämä toteutus eteenpäin valinnasta lisää siis muuttujia malliin ahneesti (engl. *greedy search*). Menettelytapaa jatketaan niin kauan kuin aikaan saadun mallin suorituskyky paranee, tai on valittu ennalta määriteltä määrä muuttujia. Taaksepäin-eliminointi etenee päinvastaiseen suuntaan eli se aloitetaan täydestä mallista, johon on sisällytetty kaikki muuttujat. Mallista eliminoidaan yksi muuttuja kerrallaan ja menettelyä jatketaan niin kauan, kuin uuden mallin suorituskyky on edellistä parempi.

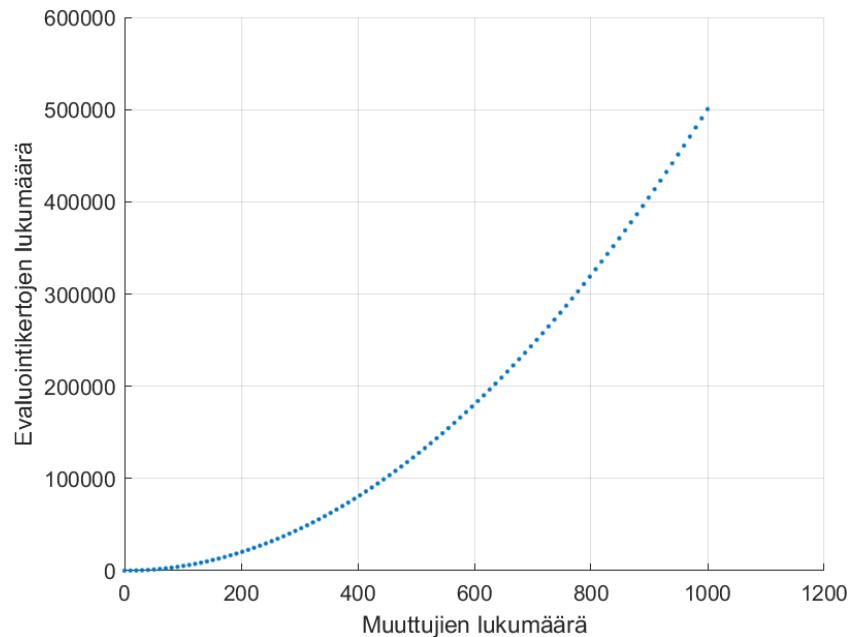
Eteenpäin-valintaa pidetään laskennallisesti tehokkaampana menetelmänä kuin taaksepäin-eliminointia. Taaksepäin-eliminoinnin etuna voidaan kuitenkin pitää sitä, että se testaa paremmin yhdessä toisten muuttujien kanssa toimivia muuttujia, mutta toisaalta malleja, jotka sisältävät paljon turhia muuttujia. Eteenpäin-valinnassa muuttujan merkitystä arvioidessa ei huomioida muiden muuttujien osalta kuin jo malliin valitut muuttujat. Täten eteenpäin-valinta voi löytää osaltaan heikomman muuttujajoukon. (Benoudjit et al., 2004; Guyon & Elisseeff, 2003)

Algoritmin kompleksisuutta voidaan edellä mainituilla algoritmeilla arvioida huonoimman mahdollisen tapauksen perusteella. Tällainen tapaus ilmenee eteenpäin-valinnalla silloin, kun muuttujajoukosta tulee malliin valituksi kaikki muuttujat ja taaksepäin-eliminoinnilla silloin, kun muuttujajoukosta lopulliseen malliin valitaan vain yksi muuttuja. Hakualgoritmin tehokkuuteen vaikuttaa siten, onko todellisen mallin dimensio huomattavasti suurempi tai pienempi kuin datan dimensio. Kyseisissä huonoimmista mahdollisista tapauksista olisi tällöin ollut laskennallisesti tehokkaampaa käyttää vastakkaista menetelmää. Huonoimman mahdollisen tapauksen perusteella arvioituna eteenpäin-valinta ja taaksepäin-eliminointi ovat täten yhtä kompleksisia. Mikäli mallin dimensio on vain hieman pienempi kuin datan dimensio, on taaksepäin-eliminointi ehdottomasti laskennallisesti tehokkaampi muuttujanvalintamenetelmä. Vastaavasti, jos mallin dimensio on huomattavasti pienempi kuin datan dimensio, on eteenpäin-valinta tehokkaampi menetelmä. Eteenpäin-valinta- ja taaksepäin-eliminointi-algoritmien aikakompleksisuutta voidaan arvioida evaluointikerroilla mitattuna seuraavasti:

$$n + n - 1 + n - 2 + \dots + 1 = \frac{n(n+1)}{2} = \frac{n^2 + n}{2} = O(n^2), \quad (3)$$

missä n on muuttujien lukumäärä.

Huonoimman mahdollisen tapauksen evaluointikertojen lukumäärä eteenpäin-valinnalle ja taaksepäin-eliminoinnille muuttujajoukon koon funktiona on esitetty kuvassa 2.



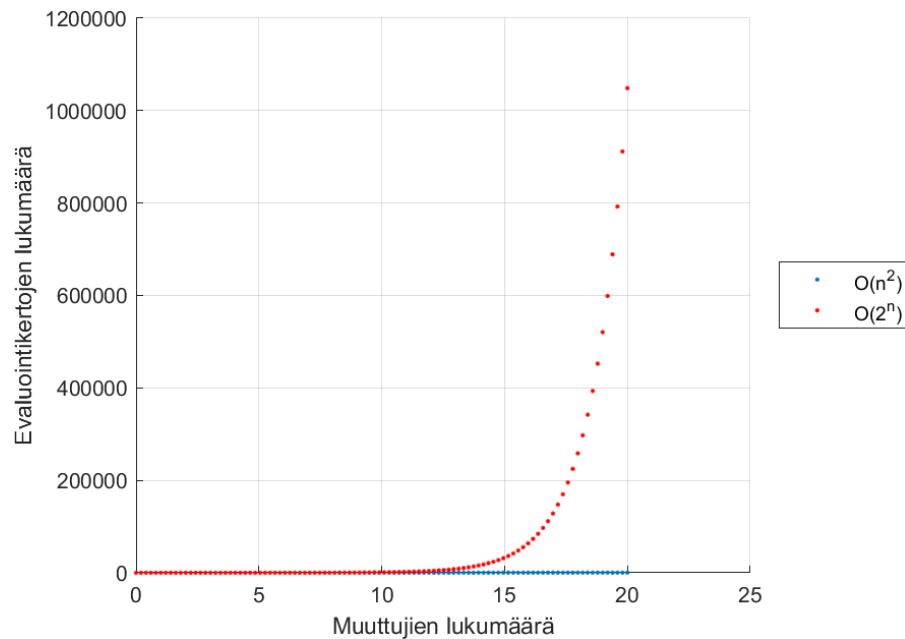
Kuva 2. Evaluointikertojen lukumäärä muuttujajoukon koon funktiona huonoimmalle mahdollisimmalle tapaukselle

Vastaavasti niin kutsutulle uuvuttavalle hakualgoritmille, jossa testataan jokainen mahdollinen muuttujakombinaatio, saadaan kompleksisuudeksi:

$$2^{n-1} = O(2^n), \quad (4)$$

missä n on muuttujien lukumäärä. Kompleksisuus tulee siis tarjolla olevien muuttujakombinaatioiden määrästä. On kuitenkin huomattava, että uuvuttava haku löytää varmuudella globaalisti optimin muuttujajoukon, mutta suurilla datoilla sen käyttäminen on epäkäytännöllistä.

Edellä mainittujen kompleksisuuksien vertailu on havainnollistettu kuvassa 3. Havaitaan, että deterministinen haku on ehdottomasti tehokkaampi menetelmä muuttujajoukon koon kasvaessa uuvuttavan haun kompleksisuuden eksponentiaalisen luonteen takia.



Kuva 3. Deterministisen haun ja uuvuttavan haun vertailu

3.3.4 Geneettiset algoritmit

Biologiaan ja edelleen evoluutioon pohjautuvat geneettiset algoritmit ovat stokastisiin menetelmiin kuuluva optimointimenetelmä. Muuttujavalintaongelman ratkaisuun soveltuvassa geneettisessä algoritmissa kromosomit ovat binäärisiä eli yksittäiset geenit ovat ykkösiä tai nollia. Geneettisen algoritmin idea on, että yksilöistä koostuvan populaation geenipooli sisältää mahdollisesti ratkaisun ongelmaan. Ratkaisu ei itsessään ole aktiivinen, vaan se on jakautuneena useisiin populaation yksilöihin. Ratkaisuja voi olla useita ja geneettisten algoritmien avulla pyritäänkin tarkastelemaan useita ratkaisuja yhtäaikaaisesti. Hyödyntämällä populaation sisältämää informaatiota pyritään löytämään uusia entistä parempia ratkaisuja ongelmaan. Ratkaisuun voidaan päästä lisääntymisen kautta tapahtuvan rekombinaation avulla tai geenistön mutaatioiden avulla. Rekombinaatiossa valituista vanhemmista muodostetaan yksi tai useampia jälkeäisiä. Uusi genotyyppi syntyy rekombinaatiossa tapahtuvan tekijäinvaihdunnan aikana, jolloin kromosomit vaihtavat kromatidien osia päikseen. Mikäli lisääntyvät yksilöt kantavat hyviä ominaisuuksia genotyypissään, on mahdollista, että ne periytyvät luoden entistä paremman genomien jälkeläiselle. Mutaatioissa yksilön genomiin tulee muutoksia joko pistemäisesti tai osasta genomia. Luonnollisessa evoluutiossa mutaatioilla on harvoin suurta merkitystä yksilön fenotyypistä ajatellen, mutta geneettisillä algoritmeilla mutaatioiden aiheuttamat satunnaiset muutokset ovat olennainen osa hakua. Binäärisesti

koodattujen algoritmien rekombinaatiossa ainoastaan yhdistetään populaatiossa olevaa tietoa, kun taas mutaatiot muuttavat olemassa olevaa tietoa. Tällöin haku voi edetä myös ratkaisuihin, joita alkupopulaatio ei sisällä. (Sivanandam & Deepa, 2008)

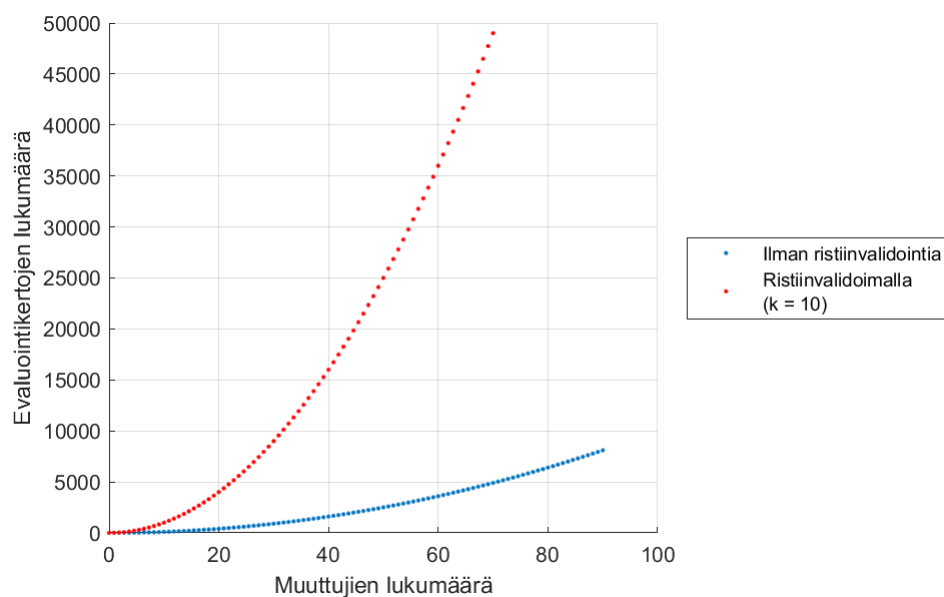
3.4 Mallin validointi

Data-aineiston jakaminen testaus- ja validointiaineistoihin on olennaista, jotta mallin validointiin käytetään opetuksesta riippumatonta dataa. Aineisto jaetaan lähes aina satunnaisesti ja täten opetus- ja validointiaineistoihin voi tulla valituksi tarkoituksen kannalta epäedullisia datapisteitä. Esimerkiksi luokitteluongelmissa aineistot voivat olla epätasapainoisia, jolloin jotkut luokat ovat yliedustettuja ja toiset aliedustettuja. Tämän takia aineiston jako ja mallin validointi tulisi toistaa useita kertoja, jotta satunnaisuuden aiheuttamat poikkeamat pystytään minimoimaan. (Guyon & Elisseeff, 2003) Tällaisessa tilanteessa kannattaa myös käyttää luokkakohtaisia otoksia.

Mikäli mittausaineiston havaintojen määrä on pieni, voidaan sen käyttöä mallin identifioinnissa ja testauksessa tehostaa ristiinvalidointimenetelmillä. Leave-one-out-menetelmässä (LOO) jätetään yksi mittausaineiston datapiste (pisteitä n kappaletta) validointiin ja loput aineiston datapisteet ($n - 1$ kappaletta) käytetään mallin opetukseen. Tällöin kaikki datapisteet tulevat käytetyksi sekä opetuksessa että validoinnissa. Menetelmää toistetaan n kertaa, jolloin jokaista datapistettä on käytetty validoinnissa kerran ja opetusaineistona $n - 1$ kertaa. Leave-multiple-out-menetelmässä (LMO) mittausaineistosta jätetään useita pisteitä mallin validointiin. Täten LMO-menetelmällä mallin rakennuksen kannalta opetusaineisto on niukempi, mutta toisaalta testausaineisto on laajempi. LMO-menetelmän etuna voidaan pitää pienempää riskiä ylisovittamiseen verrattuna LOO-menetelmään, mikäli menetelmää toistetaan riittävän monta kertaa. LMO-menetelmä on kuitenkin toistettava useita kertoja, sillä aineiston jakamisen ollessa satunnaista voidaan pienillä toistomäärillä päätyä väriin johtopäätöksiin validoinnin tulosten suhteen. K-fold-menetelmässä mittausaineisto jaetaan k yhtä suureen osaan, joista kerrallaan yhtä käytetään mallin validointiin ja muita $k - 1$ osaa mallin opetukseen. Tämä toistetaan k kertaa, jolloin jokainen osa-aineisto tulee käytetyksi kerran validointiaineistona ja $k - 1$ kertaa opetusaineistona. Näin ollen, jos k on datapisteiden lukumäärä, on kyseessä itseasiassa LOO-menetelmä. K-fold-menetelmässä aineiston jako on satunnainen kuten LMO-menetelmässäkin, joten validointi olisi hyvä suorittaa useita

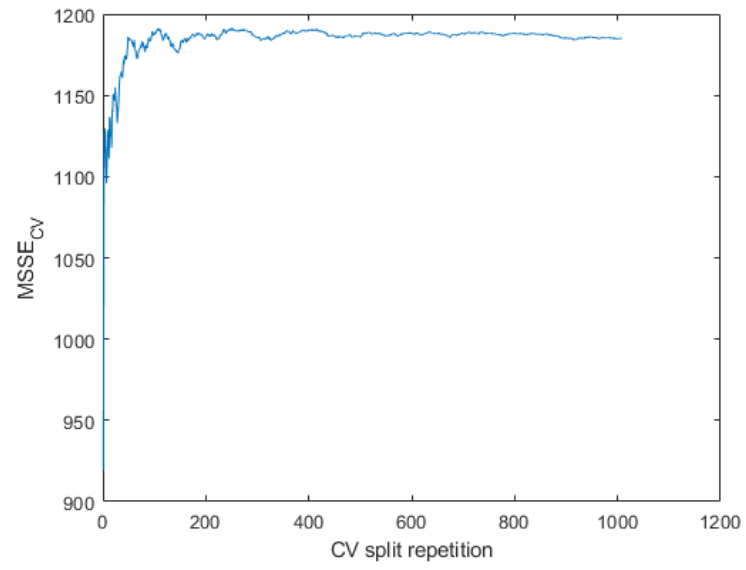
kertoja, jotta datan satunnaisen jaon aiheuttamilta vääriä johtopäätöksiltä vältytään. (Baumann, 2003)

Ristiinvalidoinnin käyttäminen muuttujanvalinnassa kuitenkin lisää merkittävästi haun aikakompleksisuutta, sillä sen sijaan että jokainen malli evaluoitaisiin kerran, se evaluoidaan k kertaa, jossa k on ristiinvalidointijakojen lukumäärä. Kuvassa 4 on vertailtuna evaluointikertojen lukumäärää muuttujajoukon koon funktiona ristiinvalidoinnilla ja ilman. Kuten kuvaajasta nähdään, ristiinvalidoinnilla on suuri merkitys mallin validoinnissa toistomäärän kasvattamiseksi.



Kuva 4. Ristiinvalidoinnin vaikutus evaluointikertojen määrään

Kuvasta 5 nähdään erään testiaineiston estimointivirheen suuruus ristiinvalidoinnissa aineiston jakamiskertojen funktiona. Kuten huomataan, on estimointivirheen suuruudessa pienillä toistomäärillä huomattavaa oskillointia ja lopullinen virhe asettuu huomattavasti suuremman arvon lähetyville, kuin mitä pienet toistomäärät antavat olettaa eikä ristiinvalidoinnin merkitystä pidä tämän perusteella vähätellä. Esimerkin tapauksessa olisimmekin siis arvioineet mallin suorituskyvyn optimistisesti, jos olisimme käyttäneet niin kutsuttua hold-out menetelmää, eli toistaneet ristiinvalidointia vain kerran.



Kuva 5. Residuaalineliosummien keskiarvo ristiinvalidointikertojen funktiona

4 KOESUUNNITTELU

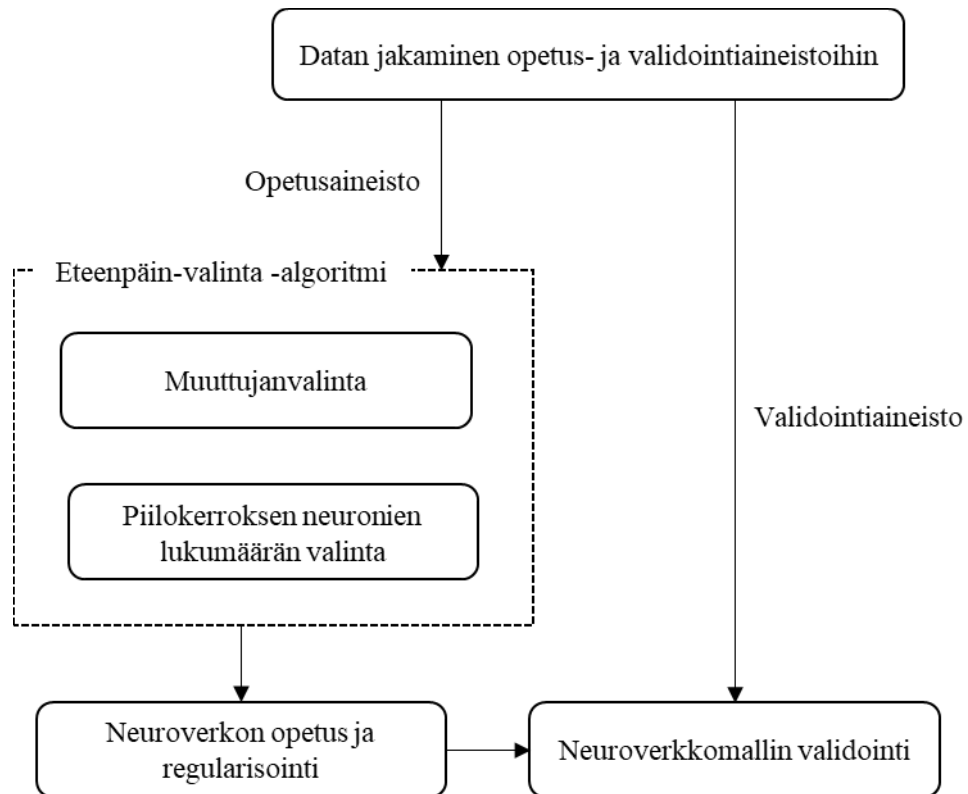
Tutkielman kokeellisessa osassa tarkastellaan keinotekoisien neuroverkkomallin ennustuskykyä Barkhausenin kohinasignaaliin. Kokeellisessa osiossa käytetään Matlab-laskentaohjelmistoa.

4.1 Keinotekoinen neuroverkkomalli

Kokeellisessa osiossa käytetyn keinotekoisien neuroverkon opetusalgoritmina käytetään *Extreme Learning Machinea* (ELM), joka soveltuu yhden piilokerroksen eteenpäin syöttäville neuroverkoille (engl. *single hidden layer feedforward neural network, SLFN*). ELM:n avulla voidaan saavuttaa erittäin hyvin yleistäviä neuroverkkomalleja verrattain lyhyessä opetusajassa. ELM-algoritmissa syöte- sekä piilokerroksen välisten neuroneiden painokertoimet määritetään satunnaisesti ja piilo- sekä lähtökerrosten väliset painokertoimet määritetään pienimmän neliösumman menetelmällä. Menetelmän käyttö opetusalgoritmina voi mahdollistaa jopa tuhansia kertoja nopeamman opetusajan, kuin perinteisemmät SLFN-neuroverkoille käytetyt gradienttimenetelmiin perustuvat opetusalgoritmit kuten back propagation. Gradienttimenetelmiin perustuvilla opetusalgoritmeilla on myös taipumus jumittua paikallisiin minimikohtiin ja kyseiset algoritmit vaativat huomattavan paljon iteratiivisia opetusvaiheita neuroverkon hyvän oppimisen takaamiseksi. (Huang et al. 2004)

Käytetty keinotekoinen neuroverkko koostuu syöte-, piilo- ja lähtökerroksista. Neuroverkon piilokerroksen aktivaatiofunktio on hyperbolinen tangentti ja lähtökerroksen aktivaatiofunktio on lineaarinen. Neuronien väliset painokertoimet valitaan pienimmän neliösumman (PNS) menetelmällä neuroverkon laskennallisen resurssientarpeen vähentämiseksi. Piilokerroksen neuronien lukumäärä valitaan muuttujanvalinta-algoritmin yhteydessä ja eräänä koeparametrina onkin piilokerroksen

neuronien enimmäislukumäärä (Num_HN_Max). Kuvassa 6 on esitetty neuroverkkomallin identifiointiprosessin vuokaavio.



Kuva 6. Vuokaavio neuroverkkomallin identifiointiprosessista

Kokeellisessa osiossa ohjelma ajetaan kahdella deterministisellä eteenpäin-valintaan perustuvalla muuttujanvalinta-algoritmilla, FS1 ja FS3 (*forward-selection*). FS3-algoritmissa muuttujanvalinta suoritetaan ns. ahneesti eli jokainen muuttujajoukko testataan ja muuttuja tulee lisätyksi, mikäli mallin suorituskyky paranee. Eteenpäin-valinta-algoritmin pseudokoodi muuttujanvalinnalle on esitetty taulukossa 1.

Taulukko 1. Eteenpäin-valinta-algoritmin pseudokoodi muuttujanvalinnalle

```

Eteenpäin valinta


---


While  $J_{k+1} < J_k$ 
    For each  $x_i$  in  $X_k = \{1, 2, \dots, M-k\}$ 
        1) Lisää muuttuja malliin
        2) Laske kustannusfunktion arvo
           mallille
           (toistettu ristiinvalidointi)
    end
    1) Valitse paras malli  $M-k$  mallin joukosta
    2) Jos kustannusfunktion arvo paranee, päivitä malli
       sisältämään  $k$  kappaletta muuttujia
end

```

Selitteet: J = kustannusfunktio, M = muuttujien lukumäärä datassa, k = muuttujien lukumäärä mallissa

FS1-algoritmi on tehostettu Pearsonin tulomomenttikorrelaatiokerroinsuodatuksella. Muuttujat arvotetaan korrelaatiokerrointen perusteella laskevaan järjestykseen ja eteenpäin-valinta suoritetaan tässä järjestyksessä. Suodatuksen tarkoituksena on keventää laskennallista kompleksisuutta verrattuna FS3-algoritmiin.

4.2 Kokeissa käytetty data

Kokeellisessa osiossa käytettiin Santa-aho et al. (2012) suorittamassa Barkhausen-tutkimuksessa mitattua dataa. Mittauksia varten valmistettiin näytteitä niukkaseosteisesta kuumavalssatusta RAEX400-teräksestä. Näytteet pintakarkaistiin, jotta teräksen austeniittisuus voitiin varmistaa mittausten alkaessa. Barkhausenin kohinasignaali on mitattu käyttäen Stresstech Oy:n Rollscan 300 BN -mittauslaitteistoa. Magnetointijännite mittauksessa oli $8 V_{pp}$ (jännite huipusta huippuun) ja magnetointitaajuus oli 125 Hz. Barkhausenin kohinadatan analysointitaajuutena käytettiin 70–200 kHz. Teräksen jäännösjännitykset on mitattu röntgendifraktion avulla käyttäen Stresstech Oy:n XStress 3000 X-ray-jäännösjännitysdifraktiometriä. Difraktiometrin asetukset määritettiin siten, että mittaussyvyydeksi tuli 5–6 μm . Teräsnäytteitä taivutettiin vaiheittain kuormaa lisäten. Barkhausen-signaali ja jäännösjännitys mitattiin ennen jokaista kuormanlisäystä ja kuormanlisäyksen jälkeen. Jotkut kuormitukset aiheuttivat näytteeseen plastisia muodonmuutoksia ja kyseiset pisteet poistettiin datasta ilmiön tullessa havaituksi. (Santa-Aho et al., 2012)

Mittausdata koostui 98 havainnosta. Mittausdatasta erotettiin 47 muuttujaa, joita olivat esimerkiksi signaalin neliöllinen keskiarvo sekä kohinaprofiilin korkeus, leveys ja paikka. Muuttujanvalintaprosessin monimutkaistamiseksi muuttujakandidaattien joukko kaksinkertaistettiin lisäämällä alkuperäiset muuttujat toiseen kertaan, mutta sekoittamalla muuttujien arvot mittauksen kesken. Näin ollen muuttujakandidaattien kokonaislukumäärä oli 94. Ulkoinen validointiaineisto otettiin erille mittausdatasta ja se oli täten staattinen. Jotta testausdata kattaisi koko mittausalueen, mittausdata järjestettiin kasvavaan järjestykseen ja ulkoiseen validointiaineistoon valittiin joka seitsemäs havainto.

4.3 Koeparametrit ja niiden muuttaminen

Kokeissa muutettiin kolme koeparametria: aineiston jakamiseen liittyvää arvoa l , ristiinvalidoinnin toistomäärää ja keinotekoisen neuroverkon piilokerroksen neuronien enimmäislukumäärää. Aineistonjakoparametri l määrää missä suhteessa mittausdata tulee jaetuksi opetus- ja validointiaineistoihin. Ristiinvalidoinnin toistomäärällä säädetään, kuinka monta kertaa mallin identifioinnissa suoritetaan ristiinvalidointi. Neuronien enimmäislukumäärällä säädetään yläraja mallin identifiointialgoritmillemme eli kuinka monta neuroniamme voi maksimissaan tulla valituksi neuroverkon piilokerrokselle. Kokeellisessa osiossa käytettyjen koetasojen arvot parametreittain ovat lueteltuna taulukossa 2. Koetasojen testaamisessa käytettiin täydellistä faktorisuunnitelmaa (engl. *full factorial design*). Kaikkiaan kokeita on 49 kappaletta ja jokaista koetta toistettiin 10 kertaa, jotta satunnaisuuden vaikutusta pystytään vähentämään etenkin aineiston jakamisen osalta. Kokeelle, jossa aineistoa ei jaettu ($l = 0$ ja $N_{\text{rep}} = 0$) ajettiin laskenta vain $N_{\text{HN,max}}$ arvolla 20, sillä kun aineistoa ei jaeta eikä ristiinvalidointia suoriteta, on algoritmin valintatulos riippuvainen ainoastaan neuronien hakualueen laajuudesta, kun oletetaan piilokerroksen painokertoimien satunnaisen alustamisen vaikutuksen olevan pieni.

Taulukko 2. Koesuunnitelman koetasot

l	N_{rep}	$N_{\text{HN,max}}$
0	0	1
0,2	1	5
0,35	10	12
0,5	20	20
	50	

Selitteet: l = aineiston jako, N_{rep} = ristiinvalidointitoistojen lukumäärä, $N_{\text{HN,max}}$ = piilokerroksen neuronien lukumäärä.

Koeparametrien muuttamisella valituin koetasoin pyritään selvittämään muutosten vaikutuksia muuttujanvalinta-algoritmin valitseman keinotekoisen neuroverkkomallin suorituskykyyn, yleistävyyteen ja valittuihin muuttujajoukkoihin. Muuttujien valintaa tehdessä ja mallin suorituskykyä evaluoidessa käytetään tilastollisia tunnuslukuja kuten keskivirheen neliöjuurta (RMSE), residuaalineliosummien keskiarvoa (MSSE), mallin selitysastetta (R^2) ja absoluuttista keskivirhettä (MAE).

Muuttujanvalinnassa tarkastelu pohjautuu valittuihin muuttujiin liittyviin kysymyksiin, kuten mitä muuttujat ovat ja voidaanko niitä pitää tilastollisesti merkittävänä. Useamman toiston koesuunnitelmalla saadaan myös tietoa kuinka useasti yksittäiset muuttujat tulevat valituksi malliin ja tehdä päätelmiä ovatko ne merkittäviä keinotekoista neuroverkkomallia luodessa Barkhausenin kohinasignaaliin.

Mallin valinta-algoritmin suorituskykyä mitataan osittain ajallisesti, joskaan ei suoranaisesti laskenta-ajan suhteen, vaan arvioiden mallien evaluointikertoja käytetyn koetasokombinaation suhteen. Laskenta-aika ei ole välttämättä paras mittari tutkitun mallin arvioinnissa, sillä se on vahvasti riippuvainen laskentaan käytetyn tietokoneen laskentakapasiteetista. Laskentakapasiteettiin voivat vaikuttaa heikentävästi esimerkiksi muut tietokoneella taustalla pyörivät prosessit ja siksi suoritusajan mittaamista ajatellen identtisten koeolosuhteiden saavuttaminen jokaiselle kokeelle on haastavaa.

Keinotekoisien neuroverkon ennustavuuden ja suorituskyvyn kannalta mielenkiinto keskittyy erityisesti siihen, miten neuroverkkomalli kykenee ennustamaan Barkhausen-datasta jäännösjännitystä verrattuna esimerkiksi lineaariseen monimuuttujaregressiomalliin. Neuroverkon rakenteen kannalta olennainen kysymys on

se, miten neuronien määrä piilokerroksella vaikuttaa ennusteeseen. Toisaalta taas on mielenkiintoista, kuinka deterministiset hakualgoritmit suoriutuvat neuronien määrän valinnasta, sillä lokaalien optimien määrä on suurempi, kuin lineaarisen monimuuttujaregression tapauksessa.

5 TULOKSET JA POHDINTA

Yksittäisten mallien suorituskykyä arvoitiin jakamalla mallit huonoihin, hyviin ja erinomaisiin. Jakamisen perusteena käytettiin mallin absoluuttista keskivirhettä (MAE) testausaineistolle. Jaon raja-arvot on esitetty taulukossa 3.

Taulukko 3. Mallien arvioinnin rajat MAE-kriteerillä

MAE [MPa]	Mallin hyvyys
<50	Erinomainen
[50, 80]	Hyvä
>80	Huono

5.1 Mallin hyvyys

MAE-arvoja testausaineistolle vertailtiin sekä ELM-malleille että Bayesin regularisoiduille (BR) malleille. FS1- ja FS3-algoritmeilla ajettujen kokeiden tuottamien mallien jaotellut lukumäärät on esitetty taulukossa 4.

Taulukko 4. Mallien lukumäärät hyvyyden mukaan jaoteltuna

Mallin hyvyys	BR (FS1)	ELM (FS1)	BR (FS3)	ELM (FS3)
Huono	247	334	299	334
Hyvä	182	153	130	139
Erinomainen	61	3	61	17

Huomataan, että Bayesin regularisointi tuottaa huomattavasti suuremman määrän erinomaisia malleja kuin mitä Extreme Learning Machine itsessään tuottaa. Noin 12 % Bayes-regularisoiduista malleista on erinomaisia sekä FS1- että FS3-algoritmeilla, kun taas ELM-malleille vastaavat prosentuaaliset osuudet ovat noin 0,6 % ja 3,5 %.

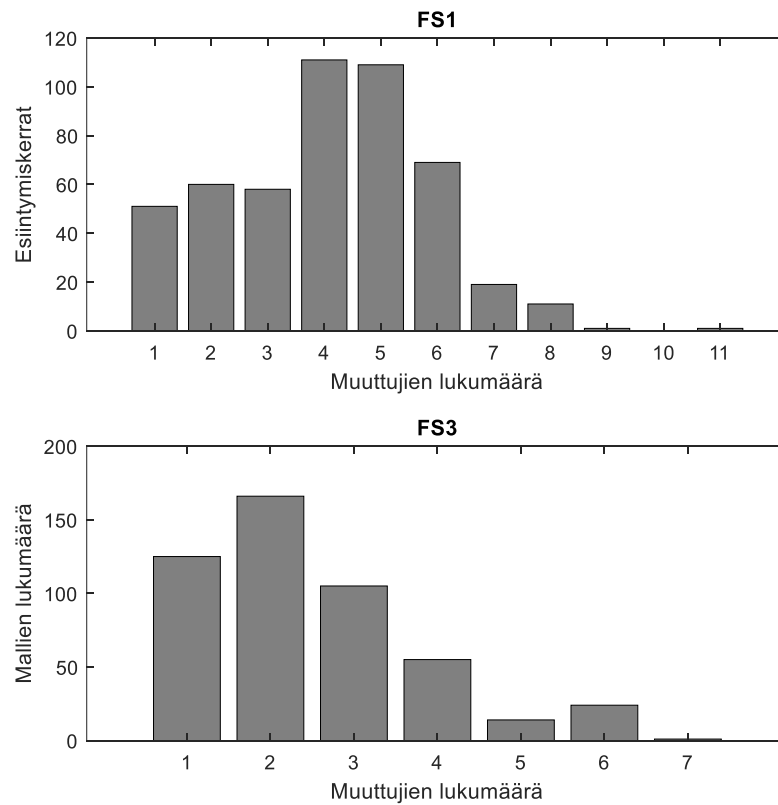
FS3-algoritmillä malleista erinomaisia joko regularisoituna tai ELM-mallina oli 70 kappaletta, mikä vastaa noin 14 % malleista. Malleista erinomaisia sekä regularisoituna, että ELM-mallina oli 8 kappaletta (noin 1,6 % malleista). FS1-algoritmillä malleista erinomaisia joko regularisoituna tai ELM-mallina oli 63 kappaletta (noin 13 % malleista). Ainoastaan yksi malli oli erinomainen sekä regularisoituna että ELM-mallina. Taulukossa 5 on esitetty erinomaisten mallien sisältämien muuttujien lukumäärän keskiarvot ja piilokerroksen neuronien lukumäärien keskiarvot.

Taulukko 5. Erinomaisten mallien keskiarvoja

Keskiarvoluku	(FS1) Malli erinomainen BR tai ELM (n = 63)	(FS1) Malli erinomainen BR ja ELM (n = 1)	(FS3) Malli erinomainen BR tai ELM (n = 70)	(FS3) Malli erinomainen BR ja ELM (n = 8)
Muuttujien lkm.	5,7	5	3,6	3,75
Piilokerroksen neuronien lkm.	8,6	9	10	13

Vaikka tutkimusdata kokonaisuudessaan sisältää runsaasti muuttujia, on erinomaisissa malleissa muuttujien lukumäärät suhteessa tarjolla olevien muuttujien kokonaismäärään (94 kappaletta) maltilliset. Havaitaan, että FS3-algoritmin valitsemisissa malleissa on vähemmän muuttujia kuin FS1-algoritmin valitsemisissa malleissa. FS1-algoritmissa tehtävä Pearsonin tulomomenttikorrelaatiokerroinsuodatus keventää laskennallisuutta runsaasti, mutta toisaalta heikentää ELM-mallien suorituskkyä kuten taulukosta 4 voidaan havaita. Erinomaisten mallien lukumäärä on suurempi FS3-algoritmillä. FS3-algoritmin heikkous verrattuna FS1-algoritmiin on sen laskennallisesti suurempi vaativuus johtuen ahneesta hakutavasta, jossa jokainen muuttujajoukko tulee testatuksi ja paras kombinaatio valituksi. Taulukosta 5 huomataan myös erinomaisten mallien osalta, että kun vähemmän muuttujia tulee valituksi FS3-algoritmin toimesta, vaatii malli enemmän piilokerroksen neuroneita ollakseen erinomainen. Muuttujien määrän pienentyessä vaaditaan enemmän muuttujien piirteitä tarkastelevia neuroneita, jotta mallin tarkkuus ei heikkene.

Kuvassa 7 on esitetty kaikkien mallien sisältämien muuttujien lukumäärän jakaumat FS1- ja FS3-algoritmeille.



Kuva 7. Muuttujien lukumäärä malleissa

FS3-algoritmin valitsemien mallien muuttujien lukumäärän keskiarvot (n_{avg}) ja keski poikkeamat (n_{std}) ristiinvalidointitoistojen lukumäärän funktiona tilanteille, joissa validointiaineiston koot ovat vakioita on esitetty taulukossa 6

Taulukko 6. Muuttujien määrän keskiarvot ja keskipoikkeamat ristiinvalidointikertojen funktiona

$CV_{rep,max}$	$l = 0,35$		$l = 0,5$	
	n_{avg}	n_{std}	n_{avg}	n_{std}
1	2,15	0,83	2,40	0,94
10	2,20	0,80	2,18	0,62
20	2,65	0,95	2,35	0,87
50	2,38	1,03	3,15	0,98

Havaitaan, että muuttujien lukumäärän keskiarvo kasvaa ristiinvalidointitoistojen lukumäärän kasvaessa, kuten myös käy määrän keskipoikkeamalle. Ristiinvalidointitoistojen kasvaessa algoritmi tulee toisin sanoen epävakaammaksi valitsemaan muuttujia lukumäärältään suuremmalta skaalalta.

5.2 Koeparametrien vaikutus mallin hyvyyteen

FS1- ja FS3-algoritmeilla valittujen mallien selitysasteiden (R^2) keskiarvot testausaineistolle on esitetty taulukossa 7.

Taulukko 7. Mallien selitysasteiden keskiarvot testausaineistolle

Muuttujanvalinta-algoritmi	Malli	R^2_{avg}
FS1	ELM	0,82
FS1	BR	0,88
FS3	ELM	0,80
FS3	BR	0,86

Tarkastellessa FS3-algoritmin valitseman mallin selitystasetta ja absoluuttista keskivirhettä ristiinvalidointitoistojen funktiona siten, että validointiaineiston koko sidotaan vakioksi, havaitaan selkeä korrelaatio. Tarkastellaan tilanteita, joissa validointiaineiston koot ovat $l = 0,5$ ja $l = 0,35$. Taulukoissa 8 ja 9 on esitetty mallien

selitysasteiden keskiarvot ja absoluuttisten keskivirheiden keskiarvot edellä mainituissa tilanteissa.

Taulukko 8. Ristiinvalidointitoistojen vaikutus mallin hyvyyteen ($l = 0,5$)

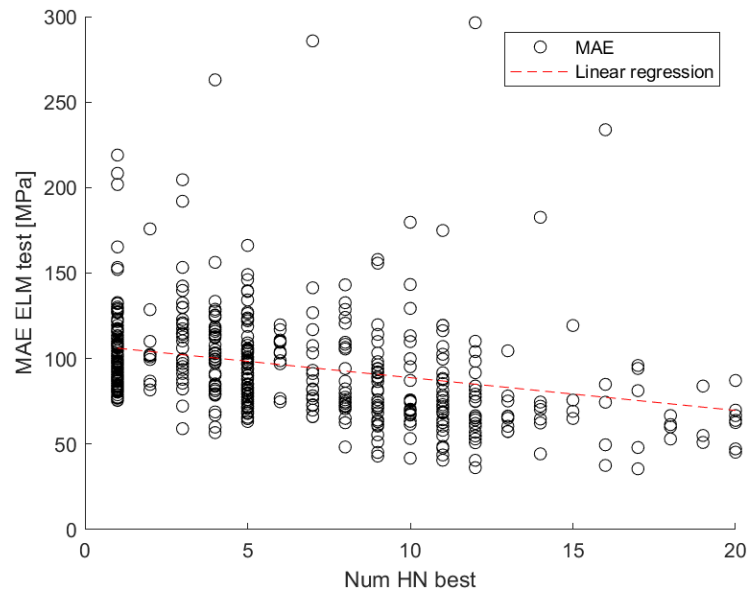
$CV_{rep,max}$	$R2_{ELM,test}$	$R2_{BR,test}$	$MAE_{ELM,test}$	$MAE_{BR,test}$
1	0,79	0,83	97,15	86,52
10	0,84	0,86	86,94	81,07
20	0,83	0,88	90,41	75,83
50	0,85	0,9	88,81	69,8

Taulukko 9. Ristiinvalidointitoistojen vaikutus mallin hyvyyteen ($l = 0,35$)

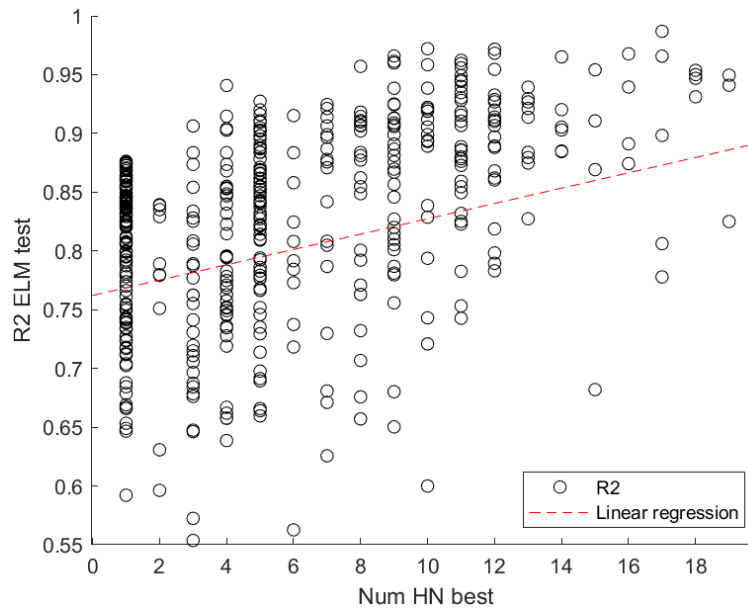
$CV_{rep,max}$	$R2_{ELM,test}$	$R2_{BR,test}$	$MAE_{ELM,test}$	$MAE_{BR,test}$
1	0,75	0,84	104,48	82,62
10	0,82	0,86	91,29	81,77
20	0,78	0,86	108,02	79,77
50	0,84	0,89	90,67	73,43

Havaitaan, että mallien selitysasteet testausaineistolle kasvavat huomattavasti sekä ELM-malleilla että regularisoiduilla malleilla ristiinvalidointitoistokertojen kasvaessa. Esimerkiksi kun aineistonjakoparametri $l = 0,5$, kasvaa mallien selitysasteiden keskiarvo testausaineistolle noin 8 % annetuin koeparametrein. Vastaavasti mallien absoluuttisten keskivirheiden keskiarvot pienenevät roimasti ristiinvalidoinnin toistokertojen kasvaessa. Huomattavin ero tapahtuu regularisoiduilla malleilla, kun aineistonjakoparametri $l = 0,5$. Tällöin mallien absoluuttisten keskivirheiden keskiarvo pienenee yli 19 %. Vastaava keskiarvo pienenee noin 12,5 % aineistonjakoparametrin ollessa $l = 0,35$. Edellä mainittujen perusteella voidaan siis päätellä ensinnäkin, että ristiinvalidointitoistokertojen kasvattaminen parantaa mallin kykyä ennustaa testausaineistoa. Toisaalta mitä suuremmaksi validointiaineiston koko kasvaa, on suotavaa kasvattaa ristiinvalidoinnin toistokertojen lukumäärää, jotta mallin selitysaste kasvaa ja virhe pienenee.

Piilokerroksen neuronien lukumäärästä voidaan nähdä absoluuttisen keskivirheen osalta negatiivinen korrelaatio ja mallin selityksasteen osalta positiivinen korrelaatio. Kuvassa 8 on esitetty FS3-algoritmin muodostamien ELM-mallien absoluuttinen keskivirhe testausaineistolle mallin piilokerroksen neuronien lukumäärän funktiona. Kuvassa 9 on taas vastaavasti esitetty ELM-mallin selityksaste testausaineistolle mallin piilokerroksen neuronien lukumäärän funktiona.



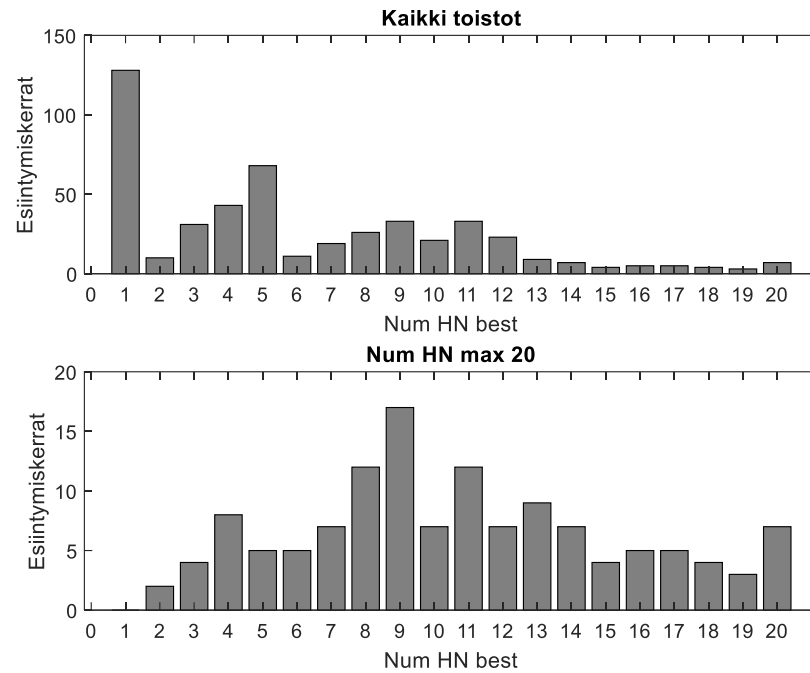
Kuva 8. ELM-mallin absoluuttinen keskivirhe piilokerroksen neuronien lukumäärän funktiona



Kuva 9. ELM-mallin selitysaste testausaineistolle piilokerroksen neuronien lukumäärän funktiona

Piilokerroksen neuronien lukumäärällä on selvä yhteys mallin hyvyyteen, mikä olikin odotettavissa, sillä yleisesti neuronien lisääminen parantaa datan piirteiden erotusta ja parantaa täten mallin suorituskykyä.

Kuvassa 10 on esitetty piilokerroksen neuronien lukumäärän jakauma tilanteessa, jossa on kaikki kokeen 490 toistoa (yllä) sekä tilanteessa, jossa piilokerroksen neuronien lukumäärää säättävä koeparametri $N_{HN,max}$ on asetettu arvoon 20 (alla). Tällöin piilokerroksen neuronien lukumäärää ei käytännössä rajoiteta, vaan nähdään, millaiseksi piilokerroksen neuronien lukumäärän jakauma todellisuudessa rajoittamattomana asettuu. Yleisimmät piilokerroksen neuronien lukumäärät rajoittamattomassa tilanteessa ovat 9, 11 ja 8. Lukumäärät osuvat taulukossa 5 esitettyihin erinomaisten mallien neuronien lukumäärään liittyvien keskiarvolukujen lähetyville. Tästä on pääteltävissä, että FS3-algoritmi kykenee hyvin valitsemaan oikean määrän neuroneita piilokerrokselle erinomaisen mallin muodostamisen takaamiseksi.



Kuva 10. Piilokerroksen neuronien lukumääräjakaumat FS3

5.3 Bayes-regularisoinnin merkitys mallin hyvyyteen

Bayesilaista regularisointia voidaan soveltaa eteenpäin syöttävän keinotekoisen neuroverkon opetuksessa. Bayes-regularisoinnilla pyritään vähentämään keinotekoisen neuroverkkomallin ylisovittamista ja vähentämään varianssia. Regularisoinnissa kustannusfunktioon, jonka minimiä opetuksessa haetaan, lisätään ylimääräinen termi. Tämä ylimääräinen termi huomioi neuroverkon painokertoimien neliösumman ja mahdollistaa valmiin opetetun mallin virheen pienenemisen. (Foresee & Hagan, 1997)

Taulukossa 10 on esitetty kuinka monta kertaa 490 toistosta Bayes-regularisoitu malli tai ELM-malli oli parempi. Havaitaan, että Bayes-regularisointi tuottaa keskimäärin seitsemän kertaa kymmenestä paremman mallin kuin ELM. Barkhausen-kohinan kaltaiselle hyvin stokastiselle ilmiölle mittausdata on erittäin vivahteikasta, mikä altistaa suurella todennäköisyydellä ylisovittamiselle etenkin neuronien lukumäärän ja täten signaalin piirteiden erotustarkkuuden kasvaessa. Sama ilmiö huomattiin myös malleja jaoteltaessa erinomaisiin, hyviin ja huonoihin (taulukko 4).

Taulukko 10. Bayes-regularisoinnin ja ELM-mallin vertailu selitysasteen perusteella

Muuttujanvalinta-algoritmi	BR parempi lkm.	BR parempi [%]	ELM parempi lkm.	ELM parempi [%]
FS1	345	70,4	145	29,6
FS3	337	68,8	153	31,2

Keskimäärin FS3-algoritmillä valittujen mallien selitysaste parani regularisoinnilla 98,2 % ja vastaavasti absoluuttinen keskiarvo pieneni keskimäärin 10,9 MPa. Selitysasteen paranemisen tarkastelu vinoutuu joidenkin yksittäisten mallien kohdalla, joille ELM-mallin selitysaste on täysin onneton, mutta regularisointi saa mallista selitysasteen kannalta jo hyvinkin kohtalaisen. Esimerkiksi erään ELM-mallin selitysaste testausaineistolle on 0,0026, mutta regularisoidun mallin selitysaste on 0,799 ja selitysaste kasvaa täten 30600 %. Suodatettaessa aineistosta nämä edellä mainitut poikkeavat havainnot siten, että mallin selitysaste kasvaa regularisoitaessa korkeintaan 50 % saadaan tulokseksi, että regularisointi paransi mallin selitysastetta testausaineistolle keskimäärin 4,8 % ja pienensi mallin absoluuttista keskivirhettä keskimäärin 9,2 MPa. Edellä mainittujen perusteella voidaan pitää Bayesin regularisointia täysin aiheellisena vaiheena tämän kaltaisessa neuroverkkomallin valinnassa Barkhausenin kohinasignaalille.

5.4 Malleissa esiintyvät muuttujat

FS3-algoritmin valitsemista muuttujista viisi yleisintä malleissa, jotka olivat erinomaisia joko regularisoituna tai ELM-mallina, olivat muuttujat 19 (*Q5*), 17 (*Signal2noise*), 11 (*Stdev*), 14 (*Range*) ja 23 (*Q95*). Taulukossa 11 on esitetty ehdolliset todennäköisyydet sille, millä todennäköisyydellä edellä mainitut muuttujat ovat mukana, kun malli on erinomainen, joko regularisoituna tai ELM-mallina.

Taulukko 11. Yleisimmät muuttujat FS3-algoritmin valitsemisissa erinomaisissa malleissa

Muuttuja	Muuttuja mukana mallissa ja malli erinomainen, lukumäärä	P(muuttuja mallissa malli erinomainen)
17	52	74,3 %
19	28	40 %
11	21	30 %
14	18	25,7 %
23	15	21,4 %

Muuttujat Q_5 ja Q_{95} viittaavat kvantiileihin eli esimerkiksi arvon Q_{95} alapuolella on 95 % havainnoista. Muuttuja *Signal2noise* kuvastaa signaalin sekä kohinan suhdetta. Muuttuja *Stdev* puolestaan kuvastaa signaalin keskihajontaa ja *Range* on Barkhausen-signaalin vaihteluväliä kuvaava muuttuja.

FS1-algoritmin valitsemisissa erinomaisissa malleissa oli neljä erityisen suurella todennäköisyydellä erinomaisessa mallissa esiintyvää muuttujaa: 17 (*Signal2noise*), 31 (*PSD*), 19 (Q_5) ja 30 (*ENT500*). Lisäksi muuttuja 13 (*Kurtosis*) esiintyi noin joka viidennessä erinomaisessa mallissa. Taulukossa 12 on esitetty ehdolliset todennäköisyydet sille, millä todennäköisyydellä edellä mainitut muuttujat ovat mukana, kun malli on erinomainen joko regularisoituna tai ELM-mallina.

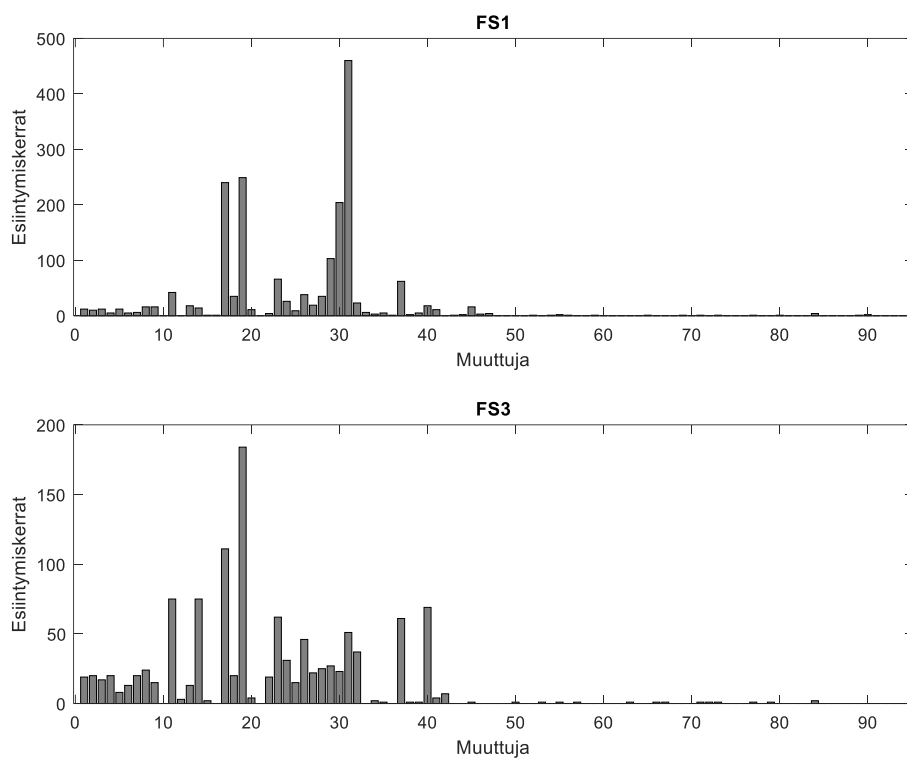
Taulukko 12. Yleisimmät muuttujat FS1-algoritmin valitsemisissa erinomaisissa malleissa

Muuttuja	Muuttuja mukana mallissa ja malli erinomainen, lukumäärä	P(muuttuja mallissa malli erinomainen)
17	56	88,90 %
31	56	88,90 %
19	39	61,90 %
30	35	55,60 %
13	13	20,60 %

Muuttuja *PSD* kuvastaa tehosppektritiheyttä, joka saadaan signaalin tehosppektristä. Tehosppektri saadaan aikaan taajuusalueanalyysin avulla, jossa hyödynnetään Fourier-muunnosta. Tehosppektri esittää signaalin amplitudin taajuuden funktiona. (Sorsa 2013)

Muuttuja *ENT500* on Barkhausen-signaalin entropia ja Muuttuja *Kurtosis* kuvastaa signaalin huipukkuutta eli jakauman muotoa.

Kuvassa 11 on esitetty jakauma, jossa näkyy x-akselilla kaikki datassa esiintyvät muuttujat ja y-akselilla esiintymiskertojen lukumäärät. Kuvasta 11 nähdään, että myös keinotekoiset muuttujat (48–94) tulivat valituksi malliin joiain satunnaisia kertoja, joskin erittäin harvoin. Tämä osoittaa sen, että käytettäessä pientä aineistoa, voidaan huonolla tuurilla huonokin muuttuja saada näyttämään hyvältä, mikäli aineiston jakaminen on tehty satunnaisesti. Tämänkään perusteella toistomäärän kasvattamisen tärkeyttä ei voida kiistää.



Kuva 11. Muuttujien esiintyvyyden malleissa

6 JOHTOPÄÄTÖKSET JA YHTEENVETO

Kandidaatintyön tarkoituksena oli selvittää keinotekoisen neuroverkkomallin soveltuvuutta teräskappaleen jäännösjännitystilojen ennustamiseen Barkhausen-kohinasignaalista. Lisäksi tarkastelun alla oli kahden deterministisen eteenpäin-valinta-algoritmin soveltuvuus neuroverkkomallin identifioinnissa muuttujien sekä piilokerroksen neuronien lukumäärän valinnassa. Muodostetun keinotekoisen neuroverkkomallin kustannusfunktio regularisoitiin ja tarkasteltiin myös regularisoinnin merkitystä mallin ennustavuudelle. Työssä käytettiin Matlab-laskentaohjelmistoa.

Mallin hyvyttä arvioitiin mallin selitysasteen ja absoluuttisen keskivirheen avulla. Mallit jaettiin absoluuttisen keskivirheen perusteella huonoihin, hyviin ja erinomaisiin malleihin. Havaittiin, että Bayes-regularisointi kasvatti erinomaiseksi luokiteltujen mallien lukumäärää huomattavasti, FS1-algoritmilla määrä oli noin 20-kertainen ja FS3-algoritmilla noin kolminkertainen. Bayes-regularisointi tuotti keskimäärin noin seitsemän kertaa kymmenestä paremman mallin kuin mitä regularisoimaton oli. Regularisointi paransi mallin ennustuskykyä huomattavasti, minkä takia on perusteltua käyttää regularisointia keinotekoista neuroverkkomallia identifioidessa Barkhausen-kohinasignaalille, mitä tukee myös havaintojen määrän niukkuus. Tulos oli odotettu neuroverkon opetusalgoritmina käytetyn Extreme Learning Machinen luonteen perusteella. Opetusalgoritmi on tarkoitettu hyvin nopeaksi ja laskennallisesti kevyeksi, mutta silti useissa käytännön tilanteissa tarpeeksi tarkaksi luomaan keinotekoisia neuroverkkomalleja. Joskin on hyvä huomioida käytetyn luokitteluperusteen subjektiivisuus, eikä täten voida sanoa, että jakaminen huonoihin, hyviin ja erinomaisiin malleihin tulee tehdä juuri näillä absoluuttisen keskivirheen raja-arvoilla. Mallintaessa täydellisen ennustavuuden saavuttaminen on varmasti lähes mahdotonta, mutta käytännön sovellutuksissa voidaan saada hyviäkin tuloksia, vaikka saavutettu malli ei olisikaan täydellinen. Saaduista tuloksista voidaan kuitenkin nähdä, millaiset vaikutukset koeparametreilla on keinotekoisesta neuroverkkomallin hyvyyteen ja täten tiedetään millaiset laskentaparametrit todennäköisesti mahdollistavat erinomaisia malleja.

Käytetyt eteenpäin-valinta-algoritmit suoriutuivat mallinnuksesta hyvin. Pearsonin tulomomenttikorrelaatiokerroinsuodatusta hyödyntänyt FS1-algoritmi menestyi hiukan heikommin, kuin ahnetta hakutapaa käyttänyt FS3-algoritmi, mikä oli algoritmien

luonteenkin perusteella odotettavissa. FS1-algoritmin suodatus keventää laskennallista tarvetta huomattavasti, mutta kuten taulukosta 5 havaitaan, se ei hävinnyt FS3-algoritmilta suorituskyvyssä suurestikaan. Kuitenkin odotusten mukaisesti FS3-algoritmi suoriutui muuttujien ja neuronien lukumäärän valinnassa algoritmeista paremmin tuottaen suuremman määrän erinomaisia malleja. FS1-algoritmi valitsi keskimäärin suuremman määrän muuttujia, mutta pienemmän määrän piilokerroksen neuroneita kuin FS3-algoritmi. Täten FS3-algoritmin valitessa uuvuttavan haun avulla tilanteeseen optimaalisen muuttujajoukon, on pääteltävissä, että Barkhausen-signaalin neuroverkkomallin optimaalinen muuttujajoukko on suhteellisen pieni verrattuna datan dimensioon. FS3-algoritmin valitessa pienemmän muuttujajoukon on toki loogista, että neuronien lukumäärä on tällöin suurempi, jotta muuttujien piirteiden erotustarkkuus ei kärsi. Tilanteeseen optimaalisesta muuttujajoukosta ja ennen kaikkea muuttujien lukumäärästä nähdään, että eteenpäin-valinta on oletettavasti laskennallisesti tilanteessa tehokkaampi kuin taaksepäin-eliminointi ja kykenee todennäköisesti tuottamaan yleistyskykyisempiä malleja.

Koeparametrien osalta mielenkiintoisin havainto oli ristiinvalidointitoistojen vaikutuksesta mallin hyvyyteen, kun validointiaineiston koko sidottiin staattiseksi. Ristiinvalidointitoistojen kasvaessa mallien selitysaste kasvoi keskimäärin huomattavasti ja vastaavasti mallien absoluuttinen keskivirhe pieneni keskimäärin runsaasti. Ristiinvalidoinnin merkitystä ei täten voi kiistää neuroverkkomallin identifiointiprosessissa, sillä ristiinvalidointitoistojen lukumäärän kasvattaminen parantaa mallin kykyä ennustaa testausaineistoa. Toisaalta kun verrattiin kahdella eri aineistonjakoparametrin arvolla mallin hyvyyttä ristiinvalidointitoistojen funktiona, havaittiin, että on suotavaa kasvattaa ristiinvalidointikertojen lukumäärää, mikäli ristiinvalidointiaineiston koko kasvaa, jotta saavutetaan paraneva mallin ennustavuus. Piilokerroksen neuronien osalta nähtiin hyvinkin odotettuja muutoksia mallin ennustavuuden osalta. Piilokerroksen neuronien lukumäärän kasvaessa neuroverkkomallin ennustavuus parani, eli selitysaste kasvoi ja absoluuttinen keskivirhe pieneni. Tarkastellessa tilanteita, joissa neuroverkon piilokerroksen neuronien maksimilukumäärän asettava parametri sai suurimman arvon (20), voitiin havaita, että eteenpäin-valinta-algoritmit kykenivät hyvin valitsemaan optimaalisen määrän neuroneita, jotta muodostettu malli olisi erinomainen. Johtopäätöksenä voidaan todeta, että käytettyjen algoritmien laskentaparametreilla ja erityisesti kustannusfunktiolla on suuri vaikutus valintatulokseen.

Malleissa esiintyvistä muuttujista nähdään ennen kaikkea se, että eteenpäin-valinta-algoritmit onnistuivat löytämään suorituskkyisiä malleja datajoukosta, johon oli lisätty myös keinotekoisia muuttujia hakuprosessin monimutkaistamiseksi. Kuten odotettua, eteenpäin-valinta-algoritmit valitsivat tiettyjä hyviä muuttujia toistuvasti ja tämä nähtiin eritellessä ehdollisia todennäköisyyksiä sille, että tietty muuttuja oli mukana mallissa, kun malli oli erinomainen. Esiintymistodennäköisyydet olivat osalla muuttujista huomattavasti suuremmat kuin muilla, eikä useasti esiintyvien muuttujien joukossa ollut keinotekoisia muuttujia.

Saadut tulokset osoittavat keinotekoisien neuroverkkomallin käyttökelpoisuuden teräskappaleen jäännösjännityksen ennustamisessa Barkhausenin kohinasignaalista. Kokeet osoittivat, että käytetty ahneeseen hakuun perustuva deterministinen eteenpäin-valinta-algoritmi kykeni valitsemaan optimaalisia muuttujajoukkoja ja piilokerroksen neuronien lukumääriä neuroverkkomalliin hyvin. Ristiinvalidoinnin merkitys mallin hyvyyteen oli merkittävä ja tarkan mallinnuksen takaamiseksi ristiinvalidoinnin toistomäärä on suotavaa pitää korkeana. Haittapuolena toistomäärän kasvattamisella on laskennallisen keveyden heikkeneminen. Vielä optimaalisempiin muuttujajoukkoihin ja tätä kautta tarkempiin malleihin voidaan mahdollisesti päästä käyttämällä muuttujien valinnassa stokastisia algoritmeja kuten simuloitua jäähtymistä ja geneettisiä algoritmeja. Näiden stokastisten algoritmien ongelmana on, että ne ovat laskennallisesti hieman raskaampia kuin tässä työssä käytetyt eteenpäin-valinta-algoritmit, varsinkin jos lähtötilanteessa ei ole asiantuntijatietoa soveltuvista muuttujista tai niiden määrästä. Barkhausen-mittauksen sovelluskohteita on lukuisia, eikä kaikkea potentiaalia ole varmastikaan vielä valjastettu. Raskasta kuormitusta kokevat teräksiset laitteiden osat kuten tuulivoimalan vaihteet ovat hyvä esimerkki kohteesta, jossa rikkomaton aineenkoetus on ehkä jopa ainoa tapa saada tietoa materiaalin tilasta. Tällaisissa kohteissa Barkhausen-kohinamittauksen hyödyntäminen voi olla järkevää. Tässä tutkimuksessa saatuja tietoja pystytään toivottavasti hyödyntämään tulevaisuuden alan tutkimuksissa ja ne toivottavasti pystyvät antamaan näkökulmaa teollisuuden sovelluskohteisiin.

LÄHDELUETTELO

Baumann, K. (2003). Cross-validation as the objective function for variable-selection techniques. *TrAC - Trends in Analytical Chemistry*, 22(6), 395-406.

Benoudjit, N., Cools, E., Meurens, M., & Verleysen, M. (2004). *Chemometric calibration of infrared spectrometers: Selection and validation of variables by non-linear models*

Box, G. E. P. (1976). Science and statistics. *Journal of the American Statistical Association*, 71(356), pp. 791-799.

Foresee, F.D., & Hagan, M.T. (1997). Gauss-Newton approximation to Bayesian learning. *Proceedings of International Conference on Neural Networks (ICNN'97)*, 3, 1930-1935 vol.3.

Gujarati. (2018). *Linear regression: A mathematical introduction* SAGE Publications, Inc.

Guyon, I., & Elisseeff, A. (2003). *An introduction to variable and feature selection* Microtome Publishing.

Huang, G., Zhu, Q., & Siew, C. (2004). Extreme learning machine: A new learning scheme of feedforward neural networks. Paper presented at the *IEEE International Conference on Neural Networks - Conference Proceedings*, 2 985-990.

Jiles, D. C. (2000). *Dynamics of domain magnetization and the Barkhausen effect*

Lecun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436-444.

Runkler, T. A. (2012). *Data analytics: Models and algorithms for intelligent data analysis*. Wiesbaden; New York: Springer Vieweg.

Santa-Aho, S., Vippola, M., Saarinen, T., Isakov, M., Sorsa, A., Lindgren, M., Lepistö, T. (2012). *Barkhausen noise characterisation during elastic bending and tensile-compression loading of case-hardened and tempered samples*

Sivanandam, S. N., & Deepa, S. N. (2008). Introduction to genetic algorithms. *Introduction to genetic algorithms* (pp. 1-442)

Sorsa, A., & Leiviskä, K. (2011). (2011). Comparison of feature selection methods applied to Barkhausen noise data set. Paper presented at the *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 44(1 PART 1) 14699-14704.

Sorsa, A., Leiviskä, K., Santa-Aho, S., & Lepistö, T. (2012). *Quantitative prediction of residual stress and hardness in case-hardened steel based on the Barkhausen noise measurement*

Sorsa, A. (2013). *Prediction of material properties based on non-destructive Barkhausen noise measurement*