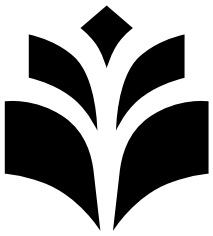
Sosiaali- ja terveydenhuollon asiakkaan segmenttimuutoksen ennustaminen neuroverkkojen avulla

Paavo Ilmari Koivistoinen

Pro gradu -tutkielma



UNIVERSITY OF
EASTERN FINLAND

Tietojenkäsittelytieteen laitos
Tietojenkäsittelytiede
Elokuu 2021

ITÄ-SUOMEN YLIOPISTO, Luonnontieteiden ja metsätieteiden tiedekunta, Joensuu
Tietojenkäsittelytieteen laitos
Tietojenkäsittelytiede

Koivistoinen, Paavo Ilmari: Sosiaali- ja terveydenhuollon asiakkaan segmenttimuutoksen ennustaminen neuroverkkojen avulla
Pro gradu -tutkielma, 69 s.
Ohjaajat: FT Virpi Hotti ja FL Jari Pekkanen
Elokuu 2021

Tiivistelmä: Tutkielma käsittelee sosiaali- ja terveydenhuollon asiakkaan segmenttimuutoksen ennustamista sosiaali- ja terveydenhuollon operatiivisista järjestelmistä kerättyjen tietojen avulla, hyödyntäen neuroverkkoja. Tässä tutkielmassa on toteutettu aihetta tukeva kokeellinen osuus, jossa sosiaali- ja terveydenhuollon yli 70-vuotiaiden asiakkaiden segmenttimuutosta ennustettiin kuuden kuukauden päähän viimeisestä palvelutapahtuman merkinnästä. Kokeellista osuutta varten sosiaali- ja terveydenhuollon asiakkaat segmentoitiin dynaamisen Pärjääjä-mallin mukaisesti. Lisäksi tietoa kerättiin asiakkaiden perustiedoista ja palvelutapahtumien merkinnöistä. Datajoukko kerättiin Keski-Uudenmaan -sote -kuntayhtymän tietovarastosta. Kokeellisen osuuden tarkoituksena oli selvittää, kuinka hyvin neuroverkkoja voidaan hyödyntää segmenttimuutoksen ennustamisessa. Lisäksi tutkielmassa tutkittiin kuinka neuroverkkoarkkitehtuurin valinta vaikuttaa ennustemallin tarkkuuteen ja kuinka Suomen sosiaali- ja terveydenhuollon dataa voidaan hyödyntää koneoppimisen tarkoituksissa. Kokeellisessa osuudessa valittiin käytettäväksi klassisia neuroverkkoja, takaisinkytkeytyviä neuroverkkoja, sekä konvoluutioneuroverkkoja. Neuroverkkojen rakenteet, sekä kouluttamiseen käytetty datajoukko perustuivat muissa vastaavissa tutkimuksissa tehtyihin havaintoihin. Kokeellisessa osuudessa neuroverkoissa valittiin käytettäväksi 36 erilaista parametriyhdistelmää. Paras saavutettu ennustetarkkuus oli 0.853, ja se saavutettiin käyttämällä matalaa takaisinkytkeytyvää neuroverkkoarkkitehtuuria. Tällä arkkitehtuurilla saavutettiin myös keskimäärin parhaimmat ennustetarkkuudet verrattuna muihin neuroverkkoarkkitehtuureihin. Tutkielmassa huomattiin, että neuroverkkoarkkitehtuuri vaikuttaa ennustemallin tehokkuuteen ja arkkitehtuurin valinta voidaan perustella käytetyllä datajoukolla. Sosiaali- ja terveydenhuollon tietojärjestelmien yhdistämisen suurimpia esteitä olivat tietojärjestelmien siiloutuminen ja poikkeavat tietorakenteet. Tiedon hyödyntämisessä tärkeiksi asioiksi nousivat standardit koodistot, käsitemallinnus ja tehokas tietovarastointi.

Avainsanat: sosiaali- ja terveydenhuolto; segmentointi; neuroverkot; sähköiset terveys- ja potilastiedot; ennustemallit

ACM CCS (2012)
•Computing methodologies →Neural networks; •Applied computing →Health care information systems;

UNIVERSITY OF EASTERN FINLAND, Faculty of Science and Forestry, Joensuu
School of Computing
Computer Science

Koivistoinen, Paavo Ilmari: Predicting social and healthcare customer's segment change using neural networks
Master's thesis, 69 p.
Supervisors: PhD Virpi Hotti and PhLic Jari Pekkanen
Elokuu 2021

Abstract: The thesis focuses on the prediction of social and health care customer segment change using data collected from social and health care operating systems, utilizing neural networks. As part of the thesis, an experimental part has been carried out, in which the six-month segment change on a social and health care client over the age of 70 was predicted. For the experimental part, the social and health care client was dynamically segmented according to the Pärjääjä model. In addition, information was collected from patient data and visitation records. The data set was collected from a data warehouse in the Keski-Uudenmaan -sote consortium. The purpose of the experimental part was to find out how effectively neural networks can be utilized in predicting segment change. In addition, the thesis investigated how the choice of neural network architecture affects the accuracy of the prediction model and how social and healthcare operational data can be used to produce machine learning models. In the experimental part, classical neural networks, recurrent neural networks, and convolutional neural networks were selected for use. The structures of neural networks, as well as the data set used for training, were based on findings from other similar studies. For this purpose, 36 different combinations of parameters were chosen for use in neural networks. The best achieved predictive accuracy was 0.853 and it was achieved using a shallow recurrent neural network architecture. This architecture also achieved the best prediction accuracies on average. Neural network architecture caused a notable difference in performance and the used architecture should be selected based on the used dataset. Different data structures and isolation of operational data cause significant problems in data utilization. Standard codes, concept models, and the right data warehousing techniques were possible solutions.

Keywords: social- and healthcare; segment; artificial neural networks; electronic health records; forecast

ACM CCS (2012)
•Computing methodologies →Neural networks; •Applied computing →Health care information systems;

Esipuhe

Tutkielma on tehty Itä-Suomen yliopiston tietojenkäsittelytieteen laitoksella vuonna 2021. Kiitän kaikkia tutkielmani teossa tukeneita ihmisiä.

Joensuussa 16.8.2021

Aliquando insanire iucundum est. -Menandros

Lyhenteet

TP : Oikea positiivinen (True positive)

TN : Oikea negatiivinen (True negative)

FP : Väärä positiivinen (False positive)

FN : Väärä negatiivinen (False negative)

LSTM : Pitkä lyhytkestomuisti (Long short-term memory)

CNN : Konvolutioneuroverkko (Convolution neural network)

SQL : Relaatiotietokantojen kyselykieli (Structured query language)

HNHC : Suuren tarpeen ja suurien kustannuksien potilas (High-need, high-cost)

Notaatiot

x : Opetusdatan syötevektori

y : Opetusdatan tulostevektori

X : Opetusdata $\Big((x_1, y_1), ..., (x_n, y_n)\Big)$.

$\hat{y}$: Neuroverkon tuottama estimaatti syötteelle x

$\phi()$: Aktivointifunktio

θ : Neuroverkon parametrit

$E(\theta, X)$: Virhefunktio

Sisällys

1. Johdanto

Tiedon määrän kasvu, sähköisten tietojärjestelmien edistyminen (Neittaanmäki, Tuominen ym., 2019) ja koneoppimisen mallien kehittyminen (Kai ym., 2013) ovat mahdollistaneet monia käytännön sovelluksia, esimerkiksi oppimisanalytiikkan sovellukset (Mubarak ym., 2021), sosiaalisen median julkaisujen suosiota ennustavat mallit (De ym., 2017) ja henkilön sairastumisen riskiä ennustavat mallit (Muniasamy ym., 2019). Osa koneoppimisen avulla tuotetuista ennustemalleista suoriutuvat tehtävässä paremmin kuin ammattilaiset (Hutson, 2017). Esimerkiksi neuroverkoilla toteutettu ennustemalli pystyi eräässä tutkimuksessa ennakoimaan tulevaa sydänkohtausta tarkemmin kuin lääkärit (Hutson, 2017). Koneoppimisen mallit, kuten neuroverkot, ovat siis soveltuvia ennustemallien toteuttamiseen. Tehokkaiden ennustemallien avulla resursseja voidaan hyödyntää tehokkaammin. Ennustemallien avulla voidaan myös huomata asioita, jotka ovat ihmiselle vaikeasti havaittavia (Albu & Stanciu, 2015). Nämä johtavat monissa toimenpiteissä kustannusten laskuun.

Suomen sosiaali- ja terveydenhuollosta aiheutuvat kustannukset ovat kasvaneet tasaisesti viime vuosina (Neittaanmäki & Lehto, 2018). On ennustettu, että nämä kustannukset tulevat myös nousemaan tulevaisuudessa (Neittaanmäki & Lehto, 2018). Sosiaali- ja terveydenhuollon kustannusten kasvua voidaan pyrkiä hillitsemään tuottamalla palveluita tehokkaammin. Neittaanmäki et al. mainitsevat tekoälyn merkittävänä työkaluna, jonka avulla näitä kustannuksia voisi hillitä (Neittaanmäki & Lehto, 2018). Tekoälyn avulla voidaan tuottaa ennustemalleja, joiden avulla pystytään toteuttamaan esimerkiksi ennaltaehkäiseviä hoitotoimenpiteitä. Suomen sosiaali- ja terveydenhuollon tietojärjestelmien integrointi ja hyödyntäminen ovat tärkeässä roolissa ennustemallien tuottamisessa (Neittaanmäki & Lehto, 2018). Suomessa 2019 säädetty laki sosiaali- ja terveystietojen toissijaisesta käytöstä mahdollistaa terveystietojen tietoturvallisen käyttöä ohjaus-, valvonta-, tutkimus ja tilastotarkoituksessa (Sosiaali- ja terveysministeriö, 2021).

Suomessa on tutkittu ja kehitetty potilaslähtöisiä palvelupolkumalleja (Niemelä & Kivipelto, 2019). Näiden palvelupolkujen tarkoitus on resursoida sosiaali- ja tervey-

denhuollon hoitopolku asiakkaan tarpeiden mukaan (Niemelä & Kivipelto, 2019). Näitä tarpeita voidaan selvittää segmentoimalla sosiaali- ja terveydenhuollon asiakkaita erilaisiin asiakasryhmiin (Mäkinen, 2018). Suomessa kehitetty Suuntima-malli on yksi käytössä olevista segmentointimalleista (Mäkinen, 2018). Suuntima-mallin avulla sosiaali- ja terveydenhuollon asiakkaat voidaan segmentoida neljään erilaiseen segmenttiin hoidon vaativuuden ja potilaan voimavarojen perusteella (Pirkanmaan sairaanhoitopiiri, 2021). Segmentointimalleja voidaan myös hyödyntää ennustemallien toteuttamisessa. Voimme esimerkiksi kouluttaa ennustemallin, jonka avulla voidaan ennustaa sosiaali- ja terveydenhuollon asiakkaan riskiä joutua suurien kustannusten segmenttiin. Tätä ennustetta voitaisiin hyödyntää myös sosiaali- ja terveydenhuollossa ennaltaehkäisevissä palveluissa estämässä riskien realisoitumista. Onnistuneet väliintulot edistävät sosiaali- ja terveydenhuollon asiakkaan terveyttä ja tuottavat säästöjä palveluiden järjestäjille (Neittaanmäki & Lehto, 2018).

Tämän tutkielman toteuttamisen suurimpana motivaationa on oma henkilökohtainen halu olla edistämässä kustannustehokkaampaa ja asiakaslähtöisempää sosiaali- ja terveydenhuoltoa Suomessa. Tutkielmassa käytettäväksi menetelmäksi ennustemallien tuottamiseen valikoitui neuroverkot ja niiden erilaiset arkkitehtuurit. Neuroverkot ovat saaneet lupaavia tuloksia ennustetarkkuuden osalta monissa tutkimuksissa (Shamshirband ym., 2020). Tutkielman tuoman mahdollisuuden myötä oli myös mielenkiintoista lähteä hyödyntämään Suomen sosiaali- ja terveydenhuollon tietojärjestelmien tuottamaa dataa ennustemallien toteutuksessa.

Tämä tutkielma on toteutettu yhteistyössä Keski-Uudenmaan sote -kuntayhtymän (Keusote) tietovarastointihankkeen kanssa. Tutkimuksen tuloksia hyödynnetään Keusoten sosiaali- ja terveydenhuollon asiakkaiden segmenttimuutosten ennustamiseen.

1.1 Tutkimuskysymykset

Tutkielman tutkimuskysymykset ovat muodostuneet kokeellisen osuuden ympärille. Tutkielman kokeellisessa osuudessa tarkoituksena on ennustaa sosiaali- ja terveydenhuollon asiakkaan segmenttimuutosta hyödyntäen neuroverkkoja. Nämä ennustemallit pyrkivät ennustamaan tapahtuuko asiakkaalla segmenttimuutos kuuden kuukauden päästä viimeisestä sosiaali- ja terveydenhuollon käynnistä. Tutkielman tavoitteena on tutkia kuinka tarkasti neuroverkoilla voidaan ennustaa segmenttimuutosta hyödyntäen sosiaali- ja terveystietoa. Tutkielmassa pyritään myös selvittämään, onko erilaisilla neuroverkkoarkkitehtuureilla vaikutusta ennustemallin suorituskykyyn. Lisäksi tarkastellaan kuinka

Suomen sosiaali- ja terveydenhuollon tuottamaa tietoa voidaan hyödyntää ja millaisia haasteita koulutukseen käytettävien datajoukkojen keräyksessä voi olla.

Tutkimuskysymykset ovat seuraavat:

- Kuinka Suomen sosiaali- ja terveydenhuollon dataa voidaan kerätä, jotta sitä voi hyödyntää koneoppimisen ennustemallien kouluttamisessa?
- Kuinka tarkasti neuroverkot pystyvät ennustamaan sosiaali- ja terveydenhuollon asiakkaiden segmenttimuutoksia?
- Miten neuroverkon arkkitehtuuri vaikuttaa sosiaali- ja terveydenhuollon asiakkaan segmenttimuutoksen ennustemallin suorituskykyyn?

1.2 Tutkielman rakenne

Tutkielman toinen luku käsittelee neuroverkkojen teoreettista taustaa. Luvussa perehdytään kokeellisessa osuudessa käytettyihin aktivointifunktioihin, virhefunktioihin ja erilaisiin neuroverkkoarkkitehtuureihin. Luvussa käydään läpi klassisien neuroverkkojen, LSTM-neuroverkkojen ja konvoluutioneuroverkkojen arkkitehtuurien perusteet. Tavoitteena on määritellä kuinka neuroverkot toimivat ja kuinka neuroverkoilla toteutettuja ennustemalleja voidaan kouluttaa ja arvioida. Lisäksi luvussa perehdytään erilaisten neuroverkkoarkkitehtuurien hyödyntämiseen sosiaali- ja terveydenhuollossa.

Tutkielman kolmas luku esittelee sosiaali- ja terveydenhuollon asiakkaan segmenttimuutoksen ennustamisen periaatteet. Luvussa perehdytään sosiaali- ja terveydenhuollon asiakkaan segmentoimiseen. Tutkielman kokeellista osuutta varten luvussa perehdytään Suomessa käytettyihin Suuntima- ja Pärjääjä-malleihin. Luvussa perehdytään myös Suomen sosiaali- ja terveydenhuollossa tuotettuun dataan ja sen hyödyntämiseen koneoppimisen kontekstissa. Tavoitteena on käydä läpi kuinka sosiaali- ja terveydenhuollon operatiivisten järjestelmien tietoa voidaan integroida suureksi hyödynnettäväksi kokonaisuudeksi. Lisäksi luvussa tehdään kirjallisuuskatsaus tutkimuksista joissa sosiaali- ja terveydenhuollon asiakkaiden tai potilaiden segmenttimuutoksia on ennustettu erilaisilla menetelmillä.

Tutkielman neljännessä luvussa kuvataan tutkielman kokeellisen osuuden toteutus. Kokeellisessa osuudessa toteutetaan sosiaali- ja terveydenhuollon asiakkaiden dynaaminen kuukausitason segmentointi. Ennustemalleja varten koulutusdataan kerätään lisäksi

erilaisia palvelutapahtumien merkintöjä kymmenen kuukauden ajalta sosiaali- ja terveydenhuollon asiakkaasta. Tämä data on peräisin Keski-Uudenmaan sote -kuntayhtymän tietovarastosta. Luvussa toteutetaan 36 erilaista neuroverkkomallia käyttäen avoimen lähdekoodin python-kirjastoja. Neuroverkkojen koulutus toteutetaan kokeellisessa osuudessa kerätyn datajoukon avulla. Näiden ennustemallien tarkoitus on ennustaa sosiaali- ja terveydenhuollon asiakkaan riskiä valitulle segmenttimuutokselle kuuden kuukauden päästä viimeisestä palvelutapahtuman merkinnästä. Käytettävät neuroverkkoarkkitehtuurit ovat klassinen neuroverkko, LSTM-neuroverkko ja konvoluutioneuroverkko. Lisäksi luvussa vertaillaan neuroverkkomallien antamia tuloksia hyödyntäen esiteltyjä suorituskyvyn mittareita. Myös erilaisten neuroverkkoarkkitehtuurien vaikutusta saatuihin tuloksiin analysoidaan.

Tutkielman viidennessä ja viimeisessä luvussa käydään läpi muodostuneet johtopäätökset tutkimuskysymyksien osalta. Luvussa pohditaan myös kokeellisen osuuden tuloksien merkitystä Suomen sosiaali- ja terveydenhuollon kannalta. Lisäksi luvussa käsitellään aiheen mahdollisia jatkotutkimuskohteita.

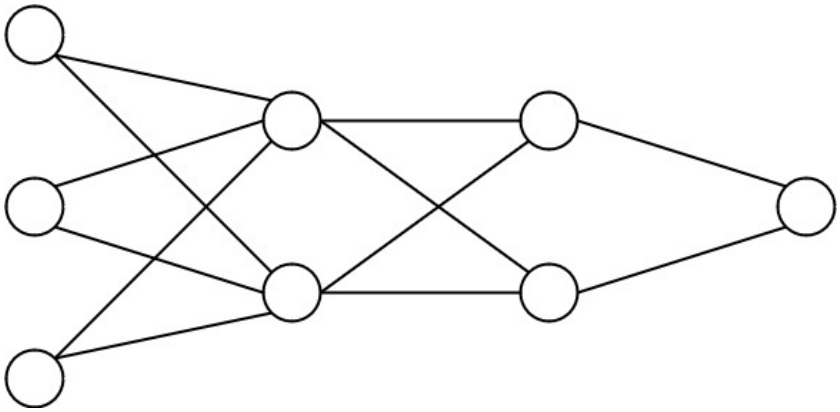
2. Neuroverkot

Keinotekoiset neuroverkot (Artificial neural networks, ANNs) ovat tiedon käsittelyn matemaattisia malleja, jotka ovat saaneet vaikutteita biologisten aivojen neuronien toiminnasta (Bishop, 2006). Neuroverkot luokitellaan osaksi koneoppimista (Goodfellow ym., 2016) ja edustavat siellä omaa alakategoriaansa. Neuroverkkoja - ja varsinkin syviä neuroverkkoja (Deep neural networks, DNNs) - voidaan hyödyntää esimerkiksi ennustemallien tai luokittelijoiden tekemiseen ja niiden avulla on pystytty saavuttamaan todella lupaavia ennustetarkkuuksia esimerkiksi kuvantunnistuksen (Pak & Kim, 2017), puheentunnistuksen (Zhang ym., 2018), sekä sosiaali- ja terveydenhuollon saralta (Miotto ym., 2016). Lisäksi neuroverkkojen kyky löytää syötedatasta monimutkaisia yhteyksiä (Feature extraction) (Bishop, 2006), joka voi olla muissa tekoälyn menetelmissä haastavampaa, on todella kasvattanut neuroverkkojen suosiota.

Luvussa 2.1 käydään läpi neuroverkon rakennetta ja neuronien matemaattinen esitysmuoto. Luvussa 2.2 käydään läpi erilaisia neuroverkoissa käytettyjä aktivointifunktioita ja niiden tarkoitusta. Luvussa 2.3 alustetaan neuroverkon kouluttamista käymällä läpi erilaisia virhefunktioita ja niiden merkitystä neuroverkossa. Luvussa 2.4 käydään läpi erilaisia neuroverkkoarkkitehtuureja, joita käytetään tutkielman kokeellisessa osuudessa. Luvussa 2.5 käydään läpi neuroverkkojen kouluttamiseen liittyvää teoriaa ja perehdytään adam-optimointialgoritmiin. Viimeisessä luvussa 2.6 käydään läpi kuinka binääristen luokittelijamallien tehokkuutta voidaan vertailla erinäisen mittarien avulla.

2.1 Neuroverkkojen rakenne

Neuroverkko koostuu keinotekoisista neuroneista ja näiden välillä olevista linkeistä (Bishop, 2006). Neuronit muodostavat useita kerroksia, joista ensimmäinen kerros on syötekerros (Input layer) ja viimeinen vastekerros (Output layer). Syvissä neuroverkoissa näiden kahden kerroksen välissä on myös vähintään kaksi piilotettua kerrosta (Hidden layer). Kuva (2.1) havainnollistaa tämän kerroksittaisen rakenteen. Syötekerroksen data



Kuva 2.1: Syvä neuroverkko, joka koostuu kolmesta syötekerroksen neuronista, kahdesta ensimmäisen piilokeroksen neuronista, kahdesta toisen piilokerroksen neuronista ja yhdestä vastekerroksen neuronista (Goodfellow ym., 2016).

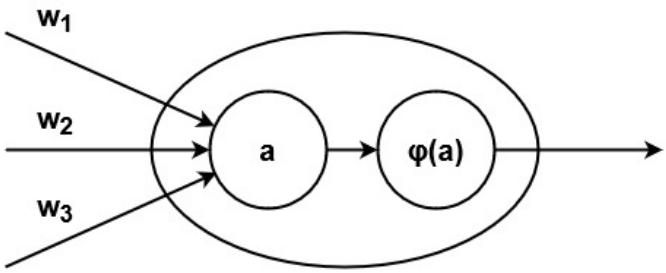
kuljetetaan linkkejä pitkin seuraavaan kerrokseen, josta käsittelyn jälkeen se viedään seuraavaan kerrokseen, kunnes saavutaan vastekerrokseen. Tämän kaltaisia neuroverkkoja kutsutaan eteenpäin kytketyiksi neuroverkoiksi (Feedforward neural networks, FNNs), jotka ovat yleisimpiä neuroverkkorakenteita (Bishop, 2006). Lisäksi on olemassa myös takaisinkytkettyjä neurovekkoja (Recurrent neural networks, RNNs) (Goodfellow ym., 2016), joissa data voi kulkea neuronilta takaisin ylempiin kerroksiin tai takaisin itseensä.

Neuroverkon syötekerrokseen annetaan syötevektori **x**, joka on ilmaistu kaavassa (2.1) (Goodfellow ym., 2016). Syötevektori **x** sisältää erilaiset arvot $x_1 ... x_n$. Jokaista syötevektorin **x** arvoa $x_1 ... x_n$ kohti, neuroverkon syötekerroksessa on yksi syöteneuroni. Nämä syötteet ovat linkitetty seuraavan kerroksen neuroneihin. Linkeillä on omat painokertoimet, jotka ilmaistaan vektorilla **w**. Painokertoimen avulla voidaan säätää parametrin vaikutusta vasteeseen. Painokertoimien joukko on ilmaistu kaavassa (2.1) (Goodfellow ym., 2016) (Goodfellow ym., 2016)

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n. \end{bmatrix} \quad \mathbf{w} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix}. \tag{2.1}$$

Seuraavaksi neuronissa lasketaan aktivointisumma a, joka kuvaa kuinka vahvasti neuronia on aktivoitu linkkien kautta. Tämä on esitetty kuvassa (2.2). Neuroniin vaikuttavat siis kaikki siihen linkitetyt neuronit. Matemaattisesti ilmaisten neuronin aktivaatio on syötevektorin **x** ja linkkien painovektorin **w** tulo, johon lisätään vielä neuronikohtainen vakiotermi (Bias) b. Vakiotermin avulla voidaan säätää neuronin aktivoitumisen

herkkyyttä. Kokonaisuutena neuronin aktivaatiosumma a lasketaan kaavalla (2.2) (Goodfellow ym., 2016)



Kuva 2.2: Kaavio yhden neuronin rakenteesta. Neuroverkkoon saapuvat syötteet kerrotaan linkkien painokertoimilla $w_1 ... w_3$. Nämä tulot summataan yhteen ja summaan lisätään neuronin vakiotermi b. Saatu aktivaatiosumma a syötetään aktivointifunktioon $\phi()$. Tämä arvo lähetetään seuraavan kerroksen neuroneille (Bishop, 2006).

$$a = \mathbf{x}^{\mathrm{T}} \cdot \mathbf{w} + b = \left(\sum_{i=1}^{n} x_i w_i \right) + b. \tag{2.2}$$

Kun neuronin aktivointisumma a on laskettu, syötetään se neuronin aktivointifunktioon $\phi()$. Aktivointifunktio kuvaa kuinka vahvasti neuroni aktivoi siihen linkittyneitä neuroneita. Aktivointifunktion vaste on arvo, joka syötetään seuraavan kerroksen neuroneihin. Aktivointifunktioihin perehdytään tarkemmin luvussa 2.2.

Kun aktivointisumma on laskettu, neuroni vie tämän arvon $\phi(a)$ linkkejä pitkin seuraaville neuroneille, jossa sama edellä kuvattu prosessi toistuu, kunnes saavutaan vastekerrokseen. Vastekerroksessa oleva arvo on $\hat{y}_n$, joka kuvaa neuroverkon estimaattia kyseiselle syötevektorille x_n. Neuroverkkoa voidaan siis käsitellä funktiona, joka saa syötteen x ja palauttaa tuloksen $\hat{y}$. Se, kuinka tehokkaasti neuroverkko pystyy esittämään datajoukon, riippuu neuroverkon neuronien määrästä, kerroksien määrästä, sekä kaikista neuroverkon parametreista. Neuroverkon parametrit ovat kaikkien kerroksien neuronien välisten linkkien kertoimet w, sekä neuronien vakiotermit b.

2.2 Aktivointifunktiot

Aktivointifunktion (Activation function) tarkoituksena on muodostaa neuroverkosta epälineaarinen malli (Goodfellow ym., 2016), jotta neuroverkko oppii tehokkaammin

ratkaisemaan monimutkaisempia tehtäviä (Goodfellow ym., 2016). Aktivointifunktiot tekevät aktivaatiosummalle a epälineaarisia muunnoksia (Nwankpa ym., 2018). Aktivointifunktiota ilmaistaan tässä tutkielmassa merkinnällä $\phi()$.

Aktivointifunktioita on paljon erilaisia (Nwankpa ym., 2018). Niitä sovelletaan erilaisiin neuroverkkoarkkitehtuureihin ja tarkoituksiin (Nwankpa ym., 2018). Monikerroksisissa neuroverkoissa on yleistä käyttää $Tanh$, $Sigmoid$, $ReLU$ ja $Softmax$ funktioita (Nwankpa ym., 2018). Osa aktivointifunktioista on hyvin yleiskäyttöisiä, mutta jotkin ovat kehitetty tietyn tarkoitusperän mukaan (Nwankpa ym., 2018). Aktivointifunktioiden valintaan vaikuttaa myös aktivointifunktion vasteen vaihteluväli (Nwankpa ym., 2018). Neuroverkkojen eri kerroksissa voidaan myös tarvittaessa käyttää eri aktivointifunktioita (Goodfellow ym., 2016).

Yleisiä aktivointifunktioita ovat $Tanh$ ja $Sigmoid$ (Nwankpa ym., 2018), joissa syöte muuttuu a tietylle vaihteluvälille (Nwankpa ym., 2018). Tämän auttaa vasteiden kasvun rajoittamisessa, ja siten helpottaa vertailua (Nwankpa ym., 2018). Nämä aktivointifunktiot ovat esitetty seuraavasti kaavoissa (2.3) ja (2.4) (Nwankpa ym., 2018),

$$sigmoid(x) = (1 + e^{-x})^{-1}, \tag{2.3}$$

$$tanh(x) = \frac{(e^{x} - e^{-x})}{(e^{x} + e^{-x})}. \tag{2.4}$$

Yksi todella yleisesti käytössä oleva aktivointifunktio on $ReLU$-funktio (Rectified linear unit) (Nwankpa ym., 2018). Tämä aktivointifunktio päästää läpi vain positiivisen syötteen (Nwankpa ym., 2018), joka voidaan ajatella neuronien aktivoitumista kuvaavaksi. Neuronit tulostavat siis aina positiivista syötettä, eivätkä negatiiviset vasteet ole mahdollisia. $ReLU$-funktio on kuvattu kaavassa (2.5) (Nwankpa ym., 2018),

$$ReLU(x) = \begin{cases} 0 & \text{jos } x < 0 \\ x & \text{muuten} \end{cases}. \tag{2.5}$$

$LReLU$-funktio (Leaky rectified linear unit) on variaatio $ReLU$-funktiosta. Tähän aktivointifunktioon syötetyt negatiiviset arvot pääsevät hyvin pieninä arvoina funktiosta

läpi. Tämän on huomattu empiirisissä kokeissa nopeuttavan neuroverkkojen oppimista (Maas ym., 2013). $LReLU$-funktio on kuvattu kaavassa (2.6) (Maas ym., 2013),

$$LReLU(x) = \begin{cases} x & \text{jos } x \geq 0 \\ \frac{x}{a_i} & \text{muuten} \end{cases}. \tag{2.6}$$

$Softmax$-funktio on taas muista poikkeava sillä, että sen avulla kaikkien sen aktivoivan kerroksen neuronien syötteistä tulee todennäköisyyttä kuvaavia (Nwankpa ym., 2018). Jokainen arvo on lukuvälillä $[0, 1]$ (Nwankpa ym., 2018), ja kaikkien kerroksen neuronien summaksi tulee 1 (Nwankpa ym., 2018). Tätä aktivointifunktiota käytetään usein luokitteluongelmia (Classification) ratkaisevien neuroverkkojen vastekerroksessa (Nwankpa ym., 2018), jossa jokainen neuroverkon vastekerroksen neuroni kuvaa kuinka todennäköisesti syöte x kuuluu luokkaan l_n (Nwankpa ym., 2018). Tämä on kuvattu kaavassa (2.7) (Nwankpa ym., 2018),

$$softmax(x_i) = \frac{exp(x_i)}{\sum_j exp(x_j)}. \tag{2.7}$$

2.3 Virhefunktiot

Neuroverkkojen kouluttamista varten tarvitaan metodi, jonka avulla kouluttamisen tulosta voidaan mitata. Tarkoitusta varten on kehitetty virhefunktio (Cost function) (Goodfellow ym., 2016), jota merkitään $E(\theta, X)$. Se on oleellinen osa neuroverkon tehokasta kouluttamista (Goodfellow ym., 2016). Virhefunktio laskee, kuinka oikeellisia tai virheellisiä tuloksia nykyiset parametrit θ antavat harjoitusdatalle X (Goodfellow ym., 2016). Virhefunktio myös yksinkertaistaa erityisesti monimutkaisten mallien suorituskyvyn vertailua, koska virhefunktion vaste on aina yksi skalaari (Goodfellow ym., 2016).

Virhefunktion avulla voidaan muuttaa neuroverkon kouluttaminen virhefunktion vasteen optimointiongelmaksi (Goodfellow ym., 2016), jossa tavoitteena on minimoida virhefunktion vaste (Goodfellow ym., 2016). Sen lisäksi virhefunktiota hyödynnetään neuroverkon parametrien θ muutoksen määrittelyssä (Goodfellow ym., 2016).

Virhefunktioita on useita erilaisia (Goodfellow ym., 2016), ja niillä on toisistaan poikkeavia ominaisuuksia, joiden perusteella haluttu funktio otettaisiin käyttöön neu-

roverkossa. Yleensä neuroverkoissa, joissa ratkaistaan regressio-ongelmia, käytetään keskineliövirhettä (Mean squared error, MSE) (Goodfellow ym., 2016). Luokitteluongelmissa käytetään sen sijaan usein risti-entropiaa (Cross-entropy) (Goodfellow ym., 2016).

Keskineliövirheessä lasketaan jokainen neuroverkon vasteen $\hat{y}_n$ ja oikean arvon y_n erotus (Goodfellow ym., 2016). Näistä erotuksista otetaan neliö ja sen jälkeen ne summataan yhteen (Goodfellow ym., 2016). Tämä on esitetty kaavassa (2.8) (Goodfellow ym., 2016),

$$E(\theta, X) = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2. \tag{2.8}$$

Summaamalla erotusten neliöitä, saadaan esimerkiksi se matemaattinen ominaisuus, että funktion vaste on aina positiivinen (Goodfellow ym., 2016). Keskineliövirheen ongelmana on kuitenkin se, että siinä painotetaan suuresti poikkeavia havaintoja enemmän kuin useita pienempiä virheitä, johtuen potenssin ominaisuudesta.

Risti-entropiassa (Cross-Entropy) on toinen virhefunktio (Goodfellow ym., 2016). Risti-entropia rankaisee ennusteita, jotka ovat hyvin varmoja väärästä luokitteluluokasta (Goodfellow ym., 2016). Tämä on kuvattu kaavassa (2.10). Binääriluokittelussa kaava voidaan esittää vielä yksinkertaisemmin. Binääriluokittelun laskentatapa on esitetty kaavassa (2.9) (Goodfellow ym., 2016),

$$E(\theta, X) = -(ylog(p) + (1 - y)log(1 - p)). \tag{2.9}$$

Tässä y merkitsee, kuuluuko oikea tulos luokkaan; y on siis 1, jos oikea arvo kuuluu luokkaan, ja 0, jos ei kuulu luokkaan. p merkitsee sitä, kuinka suurella todennäköisyydellä neuroverkon vaste $\hat{y}$ uskoo sen kuuluvan luokkaan. Tämä on esitetty kaavassa (2.10) (Goodfellow ym., 2016),

$$E(\theta, X) = -\sum_{i}^{C} log(p_o(\hat{y}_i)) y_{o,i}. \tag{2.10}$$

$y_{o,i}$ merkitsee, onko kyseinen oikea tulos y luokkaa i. $y_{o,i}$ on siis 1, jos y_i kuuluu luokkaan

i, ja muuten 0. C:n avulla merkitään kuinka monta eri luokkaa meillä on luokittelevassa neuroverkossa. $p_o(\hat{y}_i)$ avulla merkitään todennäköisyyttä sille, että neuroverkon vaste $\hat{y}$ kuuluu luokkaan i.

2.4 Neuroverkkoarkkitehtuurit

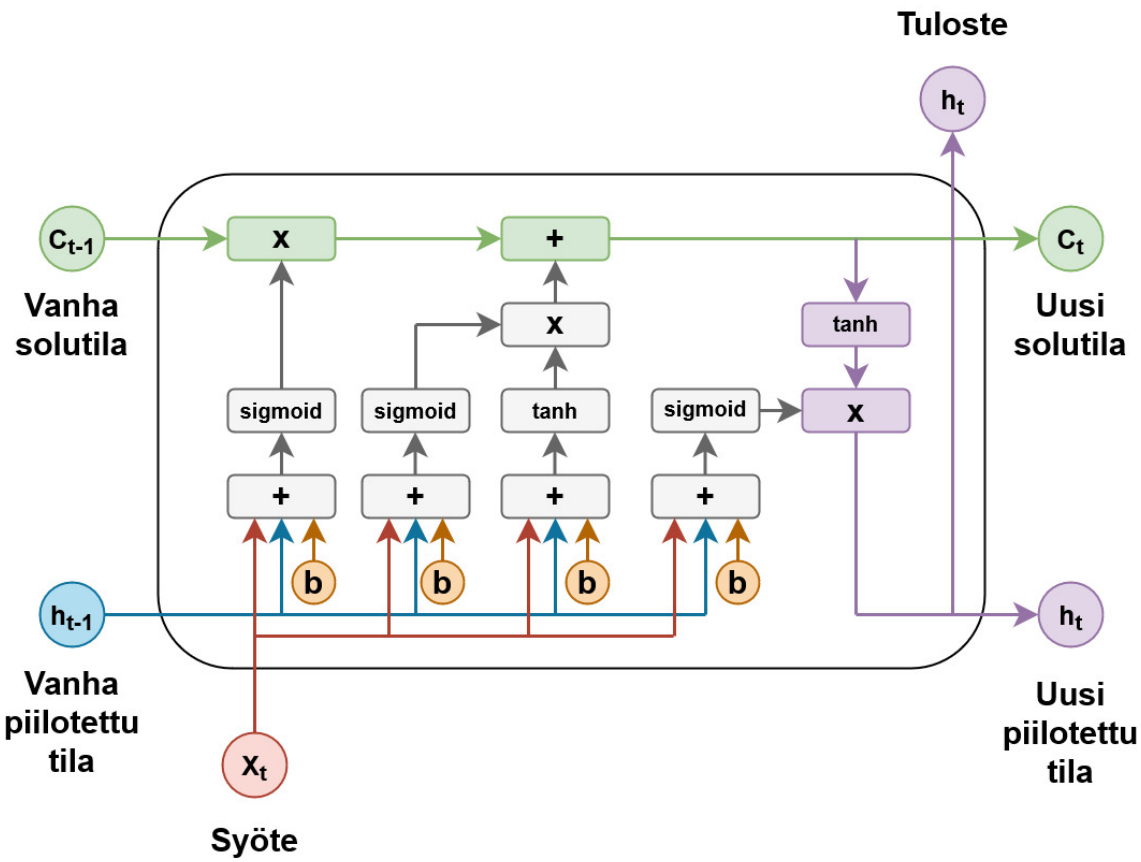
Neuroverkoilla on useita eri käytännön sovelluksia, ja tästä syystä myös neuroverkkoarkkitehtuureja on kehitetty erilaisten asetelmien tarpeisiin. Luvussa 2.1 esitelty eteenpäinkytketty syvä neuroverkko on yleisin arkkitehtuuri, joka on usean erilaisen neuroverkkoarkkitehtuurin perusta. Muut neuroverkkoarkkitehtuurit tyypillisesti lisäävät erinäisiä kokonaisuuksia tai muuttavat verkon rakennetta. Neuroverkkoarkkitehtuureista yleisesti tunnettuja ovat syvä neuroverkko (Deep neural networks), takaisinkytketty neuroverkko kuten pitkä lyhytkestomuisti (Long short-term memory, LSTM), konvoluutioneuroverkko (Convolutional neural network, CNN), autoenkoodaaja (Autoencoder) ja generatiivinen kilpaileva verkosto (Generative adversarial network, GAN) (Bishop, 2006)[1]. Näistä konvoluutioverkkojen arkkitehtuuria ja LSTM-arkkitehtuuria käsitellään tarkemmin tässä luvussa, koska niitä hyödynnetään tutkielman kokeellisessa osuudessa.

Takaisinkytketyt neuroverkot perustuvat neuroverkkoarkkitehtuuriin, jossa sisäistä muistia hyödyntämällä pystytään käsittelemään tulevia syötteitä tehokkaasti (Goodfellow ym., 2016). Esimerkiksi säätilanteen kehittymisen ennustamisessa on oleellista ymmärtää muutaman edellisen päivän kehitys. Takaisinkytketyt neuroverkot pystyvät havaitsemaan näitä yhteyksiä tehokkaasti ja hyödyntämään näitä estimoinnin aikana (Pandey & Janghel, 2019). Yksi takaisinkytketyn neuroverkon alalaji on LSTM-arkkitehtuuri, jonka avulla neuroverkolle voidaan kouluttaa pitkän aikavälin riippuvuussuhteita muita takaisinkytkettyjä neuroverkkoja tehokkaammin (Pandey & Janghel, 2019). LSTM-arkkitehtuurit ovat toimineet hyvin ennustemalleissa, joissa käsitellään aikaan sidottua syötettä, kuten esimerkiksi kielen tulkintaa, jossa konteksti on tärkeä osa kielen ymmärtämistä. Tästä syystä LSTM-arkkitehtuuri on yleistynyt erittäin vahvasti takaisinkytkettyjen neuroverkkojen standardiksi (Le ym., 2019).

Kuvassa (2.3) näkyy yksi LSTM-arkkitehtuurissa esiintyvä muistisolu (Memory block). Solu laskee ja kuljettaa kahta tilaa sisällään joita kutsutaan solutilaksi (Cell state) ja piilotetuksi tilaksi (Hidden state). Solutila merkitään kuvassa C ja sen tarkoitus on säilyttää pääketjun datakulku. Solutila viedään seuraavaan muistisoluun kahden epälineaarisen muunnoksen lävitse. Piilotettu tila h lisätään syötteeseen x, jota hyödynnetään uuden

[1]Termien suomennukset ovat kerätty osoitteesta https://course.elementsofai.com/fi/5/3

solutilan C, sekä uuden piilotetun tilan h laskemiseksi.



Kuva 2.3: LSTM-neuroverkon rakenne kuvattuna kaaviona. Vanhan solutilan C_{t-1} ja vanhan piilotetun tilan h_{t-1} avulla syöte x_t muunnetaan vasteeksi h_t ja päivitetään solutila C_t, sekä piilotettu tila h_t. Kaaviossa b on vakiotermi. $Sigmoid$ ja $Tanh$ ovat aktivointifunktioita ja + ja x operaatiot vastaavat vektorioperaatioita (Le ym., 2019).

Ensimmäisessä vaiheessa lasketaan unohdusportin f_t arvo, jonka avulla LSTM-neuroverkko määrittelee informaation, jota ei hyödynnetä. Tämä voidaan laskea kaavan (2.11) avulla (Pandey & Janghel, 2019)

$$f_t = sigmoid(W_f \cdot [h_{t-1}, x_t] + b_f), \tag{2.11}$$

jossa $sigmoid$ on kaavassa (2.3) esitetty aktivointifunktio, W_f painokertoimien matriisi, x_t syöte, h_{t-1} piilotettu tila ja b_f muistisolun vakiotermi.

Seuraavaksi lasketaan kuinka paljon syöte x_t vaikuttaa nykyiseen solutilaan C_t. Edellinen solutila C_{t-1} kerrotaan kaavalla (2.11) lasketulla unohdusportin arvolla f_t. Tähän

tuloon lisätään laskettu arvo $\tilde{C}_t$, joka kerrotaan $Sigmoid$-funktiolla saadulla arvolla i_t. Uusi solutila C_t lasketaan seuraavia kaavoja (2.12), (2.13) ja (2.14) käyttäen (Pandey & Janghel, 2019)

$$i_t = sigmoid(W_i \cdot [h_{t-1}, x_t] + b_i), \tag{2.12}$$

$$\tilde{C}_t = tanh(W_C \cdot [h_{t-1}, x_t] + b_i), \tag{2.13}$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t, \tag{2.14}$$

jossa C_{t-1} ja C_t ovat edellinen ja nykyinen solutila, W painomatriisi, $Sigmoid$ ja $Tanh$ aktivointifunktioita ja b vakiotermi.

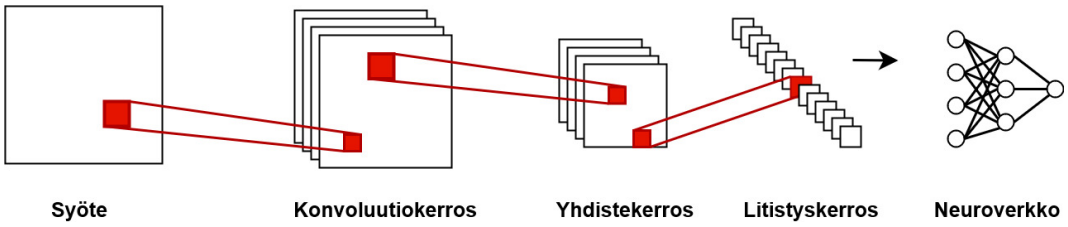
Viimeiseksi lasketaan uuden solutilan C_t avulla LSTM-neuroverkon vasteen, joka on myös uusi piilotettu tila. Tämä lasketaan syötteen kaavan (2.15) avulla, joka suodatetaan lopuksi kertomalla o_t $tanh$-funktion läpi käytetyllä solutilalla. Tämä on esitetty kaavojen (2.15) ja (2.16) avulla (Pandey & Janghel, 2019)

$$o_t = sigmoid(W_o[h_{t-1}, x_t] + b_o), \tag{2.15}$$

$$h_t = o_t * tanh(C_t), \tag{2.16}$$

jossa o_t on väliaikainen vastetila ennen suodatusta ja h_t on lopullinen vaste.

Konvoluutioverkot ovat laajalle yleistynyt syväoppimisen haara, joka on saanut alkunsa visuaalisen aistin tekniikoiden avulla Fukushiman ja Kunihikon 1980-luvulla kehittämästä neocognitron-verkosta (Fukushima ym., 1983). Konvoluutioverkko voi löytää tehokkaan esityksen raakakuvassa esiintyvästä hahmosta, mikä tekee kuvien tunnistamisen mahdolliseksi tehokkaalla tavalla (Dumoulin & Visin, 2016). Konvoluutioverkon arkkitehtuuri ja toimintamalli on havainnollistettu kuvassa (2.4). Konvoluutioverkossa syötteelle toteutetaan konvoluutio-operaatio matriisien kertolaskun sijasta muutamassa kerroksessa, jonka jälkeen se syötetään syvään neuroverkkoon. Konvoluutioverkon hyötyjä ovat harvan datan automaattinen käsittely ja tehokas ominaisuuksien poiminta raakadatasta. Konvoluutioverkko tarvitsee kuitenkin suuren datajoukon toimiakseen tehokkaasti. Myös parametrien asettaminen voi osoittautua haasteelliseksi (Pandey & Janghel, 2019).



Kuva 2.4: Konvoluutioneuroverkon arkkitehtuuri. Syötematriisi yhdistyy konvoluutiokerrokseen, joka muodostaa ominaisuuskarttoja. Ominaisuuskartat syötetään yhdistämiskerrokseen, jonka jälkeen moniulotteiset matriisit litistetään. Tämä litistetty kerros yhdistetään neuroverkkoon syötteenä (Pandey & Janghel, 2019).

Konvoluutioverkot koostuvat yleensä konvoluutiokerroksesta (Convolution layer), yhdistämiskerroksesta (Pooling layer) ja täysin yhdistetystä kerroksesta (Fully connected). Syötematriisista kerätään ominaisuuksia konvoluutiokerroksissa käytetyin parametrin ja muodostetaan ominaisuuskarttoja (Feature map). Tätä toistetaan kunnes kaikki ominaisuuskartat ovat muodostettu. Yhden ominaisuuskartan laskeminen voidaan ilmaista kaavalla (2.17) (Pandey & Janghel, 2019)

$$Y^l_{m,n,i} = w^l_i x^l_{m,n} + b^l_i, \tag{2.17}$$

jossa Y merkitsee i:ttä ominaisuuskartan l:ttä kerrosta, x syöttökorjauksen aluetta, w painokerrointa ja b vakiotermiä.

Lasketuille ominaisuuskartoille tehdään transformaatio, kuten aiemmissa neuroverkoissa. Tämä on ilmaistu kaavassa (2.18) (Pandey & Janghel, 2019)

$$Z^l_{m,n,i} = \phi\left(Y^l_{m,n,i}\right), \tag{2.18}$$

jossa $\phi()$ merkitsee käytettyä epälineaarista aktivointifunktiota ja Z transformoitua ominaisuuskarttaa.

Tyypillisiä aktivointifunktioita konvoluutiossa ovat jo aiemmin kaavoissa esitetyt $ReLU$, $Sigmoid$ ja $Tanh$ (Pandey & Janghel, 2019). Yhdistämiskerroksen tarkoituksena on pienentää muodostuneiden matriisien kokoa. Tavoitteena on pienentää ominaisuuskartto-

jen resoluutiota ilman, että ominaisuuskartan ennustevoima katoaisi. Yhdistämiskerros voidaan ilmaista matemaattisesti kaavalla (2.19) (Pandey & Janghel, 2019)

$$X^l_{m,n,i} = pool(Z^l_{m,n,i}) \forall (q, p) \in R_{m,n} \tag{2.19}$$

jossa $pool()$ merkitsee yhdistämisfunktiota ja X saatua lopputulosta.

Lopuksi konvoluutioverkon vaste muotoillaan yhdeksi parametrijonoksi, joka syötetään täysin yhdistettyyn (Fully connected) neuroverkkoon. Neuroverkossa syötteistä lasketaan vastetta, kuten luvussa 2.1 esitetään.

Pandey ja Janghel (2019) tutkimuksessa todetaan, että neuroverkkoarkkitehtuurilla merkittävä rooli toteutuksen kannalta. Erilaiset arkkitehtuurit sisältävät positiivisia ja negatiivisia vaikutuksia (Pandey & Janghel, 2019). Takaisinkytketyt neuroverkot pystyvät tallentamaan aikasarjatietoa (Pandey & Janghel, 2019). Tämän arkkitehtuurin avulla pystytään käyttämään syötteenä eripituista listaa peräkkäisistä datapisteistä (Pandey & Janghel, 2019). Takaisinkytketyt neuroverkot eivät kuitenkaan ole tehokkaita syvempien neuroverkkojen osana (Pandey & Janghel, 2019). Lisäksi takaisinkytkettyjen neuroverkkojen koulutuksessa voidaan törmätä erilaisiin ongelmiin katoavan gradientin (Vanishing gradient) kanssa (Pandey & Janghel, 2019). Konvoluutioneuroverkkojen avulla voidaan saada hyviä koulutustuloksia vaikka datajoukko olisi harva (Sparse) (Pandey & Janghel, 2019). Konvoluutioneuroverkot tarvitsevat kuitenkin paljon koulutusdataa hyviä tuloksia varten (Pandey & Janghel, 2019).

2.5 Neuroverkkojen kouluttaminen

Neuroverkkojen kouluttaminen voidaan luokitella joko ohjattuun oppimiseen (Supervised learning), ohjaamattomaan oppimiseen (Unsupervised learning) tai vahvistusoppimiseen (Reinforcement learning) (Bishop, 2006). Ohjatussa oppimisessa neuroverkkoa koulutetaan jo tiedossa olevia tuloksia käyttäen (Goodfellow ym., 2016). Tällöin datasta löytyy syötedata x ja sitä vastaava tulosdata y, jolloin neuroverkko pyrkii löytämään funktion, jonka avulla voitaisiin mallintaa kaikki annetut datapisteet (x, y). Ohjaamattomassa oppimisessa koulutuksen yhteydessä ei ole haluttua tulosdataa y, vaan koulutustilanteessa koulutettava malli pyrkii itse muodostamaan mallin lähtödatasta, esimerkiksi löytämällä poikkeavat syötteet tai löytämällä syötteiden yhteisiä tekijöitä (Bishop, 2006).

Vahvistusoppimisessa, neuroverkon koulutuksessa malli saa simuloidulta ympäristöltä palkinnon, jos se tuotti halutun suuntaisia tuloksia (Bishop, 2006). Tällöin algoritmi pyrkii maksimoimaan palkintojen määrän löytäen mahdollisesti halutun, hyvän tai uuden toimintatavan. Tutkielmassa keskitytään ohjatun oppimisen käsittelyyn, sillä kyseinen kouluttamismalli on käytössä tutkielman kokeellisessa osuudessa.

Ohjatussa oppimisessa datapisteet (x, y) voidaan jakaa kolmeen luokkaan: koulutusdataan, validointidataan, sekä testausdataan. Koulutusdatan X_{train} datapisteiden avulla neuroverkkoa koulutetaan ja tähän luokkaan laitetaan noin 50-60% kaikesta saatavilla olevasta datasta (Kattan ym., 2011). Validointidataa X_{valid} käytetään koulutusprosessissa tehtävään validointiin, jonka avulla voidaan estää mallin liiallista koulutusdatan mukailemista (Overfitting), jonka takia malli antaa hyviä tuloksia koulutusdatalle, mutta ei kykene enää ennustamaan luotettavasti, kun uusia datapisteitä tulee koulutusdatan ulkopuolelta. Validointidataan varataan yleensä koko datajoukosta noin 10-20% kaikesta datasta, tai se voidaan myös kerätä datan ulkopuolelta erilliseksi kokonaisuudeksi (Kattan ym., 2011). Koulutuksen jälkeen arvioidaan mallia käyttäen testidataa X_{test}, jota ei ole käytetty mallin kouluttamiseen mitenkään. Testidatan suuruus koko datajoukosta on noin 30% kaikesta datasta (Kattan ym., 2011).

Ohjatussa oppimisessa neuroverkkojen koulutus perustuu takaisinvirtaus-algoritmiin (Bishop, 2006). Algoritmi perustuu neuroverkon virhefunktion $E(\theta, X)$ minimointiin. Käytännössä koulutusdatan syötearvoille x lasketaan vaste $\hat{y}$, joka on neuroverkon estimaatti annetulle syötteelle x. Näitä estimaatteja vertaillaan haluttuihin koulutusdatan kohdearvoihin y. Takaisinvirtaus-algoritmin avulla voidaan laskea derivaatan, jonka perusteella voidaan päivittää neuroverkon painotuksia W ja vakiokertoimia b (Bishop, 2006). Tämän jälkeen lasketaan uudet estimaatit käyttäen päivitettyä neuroverkkoa. Tätä prosessia toistamalla neuroverkko pystyy minimoimaan virhefunktiota $E(\theta, X)$.

Neuroverkkojen kouluttamisessa käytetään erinäisiä optimointialgoritmeja (Kingma & Ba, 2014). Optimointialgoritmin tarkoitus on minimoida käytetyn virhefunktion $E(\theta, X)$ vaste. Tutkielman kokeellisessa osuudessa käytetään Adam-algoritmia, joka on ensimmäisen asteen gradienttipohjainen optimointialgoritmi stokastisille funktioille (Kingma & Ba, 2014). Käytännössä Adam-algoritmilla takaisinvirtaus-algoritmin iteraatioiden määrää voidaan vähentää, jolloin neuroverkon kouluttamisessa säästetään laskentatehoa. Algoritmi on yleisesti vakiintunut varsinkin syvien neuroverkkojen optimointialgoritmina, koska se on osoittanut tehokkuutensa monissa syväoppimisen toteutuksissa (Kingma & Ba, 2014)s. Tutkimuksen kokeellisessa osuudessa ei tarkastella muiden optimointialgoritmien vaikutusta, vaan kaikissa toteutuksissa käytetään Adam-algoritmia.

Kokeellisessa osuudessa käytetään myös aikaisin lopettavaa -metodia (early stopping) (Prechelt, 1998). Jonka avulla voidaan estää neuroverkon koulutuksessa ylioppiminen. Tässä hyödynnetään validointidataa X_{valid}, jolla testataan iteraatioiden jälkeen millaisia tuloksia virhefunktio $E(\theta, X)$ antaa (Prechelt, 1998). Jos virhefunktion antamat vaste kasvaa, vaikka koulutusdatalla virhefunktion vaste pienenee, voidaan päätellä, että neuroverkko muistaa koulutuksessa käytettyä dataa liian hyvin, jolloin ennustemallin ennustekyky laskee reaalidatan kanssa (Prechelt, 1998). Tällöin aikaisin lopettava algoritmi katkaisee koulutuksen iteraatiot, jolloin ennustekyky säilyy mahdollisimman tehokkaan.

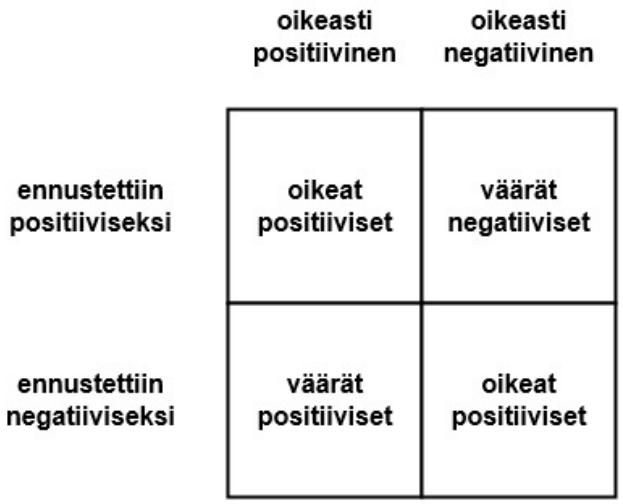
2.6 Koulutettujen neuroverkkojen arviointi

Neuroverkoilla toteutettavia ennustemalleja voidaan vertailla testidatan avulla. Voidaan laskea kuinka paljon oikeita ja vääriä luokitteluja koulutettu ennustemalli tekee. Jotta erilaisia malleja voidaan vertailla keskenään, tarvitaan yhtenäisiä mittareita, joilla voidaan vertailla kvantitatiivisesti mallien tuloksia. Tämän tutkielman kokeellisessa osuudessa koulutettujen neuroverkkojen suorituskyvyn vertailussa hyödynnetään sekaannusmatriisia (Confusion matrix), jonka kentät muodostuvat ennustettujen luokkien ja todellisten luokkien kombinaatioista. Sekaannusmatriisin kenttien avulla saadaan myös laskettua suorituskyvyn mittareita. Näihin mittareihin kuuluvat tarkkuus (Accuracy), täsmällisyys (Precision), herkkyys (Recall), johdonmukaisuus (Spesificity) ja f-arvo (F-score). Tässä luvussa käydään läpi ennustemallin suorituskyvyn mittarit ja niiden merkitys tulosten vertailussa.

Tämän tutkimuksen kokeellisessa osuudessa ennustamisessa käytetään binääristä luokittelua. Tässä luvussa perehdytään binäärisien luokittelualgoritmien arviointiin erilaisien suorituskykymittareiden avulla. Binäärinen luokittelu on yleisin luokitteluongelma (Sokolova & Lapalme, 2009). Luokittelulla voidaan esimerkiksi yrittää jaotella, onko tuotearvio positiivinen vai negatiivinen. Tämän tutkielman kokeellisessa osuudessa ennustetaan valitun segmenttimuutoksen riskiä. Tämä riski pyöristetään, jolloin ennustemallin vaste on aina 1 tai 0. Tällöin binääriluokitteluun usein sovellettuja arviointimalleja ja suorituskykymittareita voidaan hyödyntää tämän tutkimuksen kokeellisessa luvussa 4.

Sekaannusmatriisissa, joka on esitetty kuvassa (2.5), esitellään matriisin nelikenttäjako. Jako toteutetaan kahden akselin avulla; kuvan vaaka-akseli jakaa tulokset todellisen luokan mukaan. Todelliset luokat tutkimuksen kokeellisessa osuudessa ovat

"segmenttimuutos tapahtuu"tai "muutosta ei tapahdu". Kuvan pystyakseli jakaa saadut tulokset ennustetun luokan perusteella. Tällöin toiselle puolelle jäävät ne tulokset, joissa ennustetaan segmenttimuutosta, ja toiselle puolelle ne, joissa segmenttimuutosta ei ennusteta. Sekaannusmatriisin neljää kenttää kutsutaan oikeiksi positiivisiksi (True positive, TP), oikeiksi negatiivisiksi (True negative, TN), vääriksi positiiviksi (False positive, FP) ja vääriksi negatiivisiksi (False negative, FN). Oikea positiivinen kuvaa, kuinka ennustemalli ennusti segmenttimuutoksen tapahtumista, kun muutos tapahtui myös oikeasti. Väärät positiiviset kuvaavat tilannetta, jossa ennustemalli on ennustanut segmenttimuutosta, mutta segmenttimuutosta ei oikeasti tapahtunut. Oikeat negatiiviset ja väärät negatiiviset määritellään vastaavasti, mutta negatiivisille tapahtumille.



Kuva 2.5: Sekaannusmatriisi, jossa nelikenttä kuvaa binäärisen luokittelun kaikkia tuloksia. Kentät muodostuvat ennusteen ja oikean tuloksen perusteella.

Tarkkuus (Accuracy) on suorituskykymittari, jonka avulla mitataan, kuinka usein ennustemallin tekemä ennuste täsmää oikean tuloksen kanssa. Mittari ei kuitenkaan ota huomioon ennustemallin koulutuksessa käytetyn datajoukon arvojen tasapainosta. Esimerkiksi malli, joka ennustaa aina positiivista tulosta kun koulutusdatasta löytyy 90% positiivisia tuloksia, olisi mallin tarkkuus myös 90%. Tästä epäluotettavuudesta johtuen tarvitaan myös muita mittareita ennustemallin arvioinnin tueksi. Tarkkuus voidaan laskea kaavan (2.20) avulla (Sokolova & Lapalme, 2009)

$$\text{tarkkuus} = \frac{tp + tn}{tp + fp + tn + fn}. \tag{2.20}$$

Täsmällisyys (Precision) on mittari, joka kertoo kuinka moni niistä tuloksista, jotka ennustemalli luokitteli positiivisiksi, ovat oikeasti positiivisia tuloksia. Mittari on siis oikeiden positiivisten ja kaikkien positiivisten ennusteiden suhde, joka voidaan laskea kaavalla (2.21) (Sokolova & Lapalme, 2009)

$$\text{täsmällisyys} = \frac{tp}{tp + fp}. \tag{2.21}$$

Herkkyyden (Recall) avulla voidaan selvittää, kuinka monta oikeasti positiivista tulosta luokiteltiin positiiviseksi. Jos ennustemallin on kriittistä onnistua ennustamaan mahdollisimman monta oikeasti positiivista tulosta, on tämä mittari oleellinen. Herkkyys lasketaan oikeiden positiivisten ja kaikkien positiivisten suhteena, kuten kaavassa (2.22) on esitetty (Sokolova & Lapalme, 2009)

$$\text{herkkyys} = \frac{tp}{tp + fn}. \tag{2.22}$$

Johdonmukaisuus (Specificity) on suorituskykymittari, jonka avulla voidaan mitata kuinka moni ennustetuista tuloksista, jotka olivat oikeasti negatiivisia, pystyttiin ennustamaan oikein. Johdonmukaisuus on siis oikean negatiivisen ja kaikkien negatiivisten suhdeluku, joka voidaan laskea kaavaa (2.23) käyttäen (Sokolova & Lapalme, 2009)

$$\text{johdonmukaisuus} = \frac{tn}{fp + tn}. \tag{2.23}$$

F-arvo (F-score) on mittari, jossa yhdistetään edellisiä mittareita. Käytännössä f-arvo mittaa täsmällisyyden ja herkkyyden yhteisvaikutusta, jolloin molemmat tulokset vaikuttavat kokonaisarvoon. F-arvo voidaan laskea kuten kaavoissa (2.24) ja (2.25) on esitetty (Sokolova & Lapalme, 2009)

$$\beta = \frac{\text{täsmällisyys}}{\text{herkkyys}} \tag{2.24}$$

$$\text{f-arvo} = \frac{(\beta^2 + 1)tp}{(\beta^2 + 1) + \beta^2 fn + fp}. \tag{2.25}$$

Näitä mittareita hyödynnetään tutkielman kokeellisessa osuudessa vertailemaan tuotettuja ennustemalleja. Tutkielman painotus on tarkkuuden mittarissa, koska tutkimuskysymyksellä viitataan kuinka tarkasti neuroverkoilla tuotettu ennustemalli pystyy ennustamaan haluttua segmenttimuutosta. Tarkkuuden ominaisuuksien takia hyödynnetään myös muita mittareita, joilla varmistetaan ennustemallien ennustekyvyn tehoa.

3. Sosiaali- ja terveydenhuollon asiakkaan segmenttimuutoksen ennustaminen

Sosiaali- ja terveydenhuollon asiakkaan segmenttimuutoksen ennustamista varten on määriteltävä, millaisiin segmentteihin asiakas voidaan luokitella. Segmentoinnilla tarkoitetaan ihmisryhmien jakamista erilaisiin joukkoihin tiettyjen sääntöjen mukaisesti. Segmenttien avulla ihmisille voidaan kohdentaa soveltuvampia palveluita ja tuotteita. Ihmisten segmentointia on käytetty laajasti markkinoinnissa (Wood ym., 2019). Tämä näkyy erilaisille kuluttajaryhmille kohdennettuina tuotteina ja palveluina. Nykyään segmentointia hyödynnetään myös muilla aloilla kuten sosiaali- ja terveydenhuollossa (Chong ym., 2019). Tällä tavalla sosiaali- ja terveydenhuollon asiakkailla voidaan tarjota entistä kohdennetumpia palveluita, jotka tehostavat asiakkaan tarpeiden täyttymistä. Sosiaali- ja terveydenhuollon asiakkaita voidaan myös segmentoida erinäisiin ryhmiin esimerkiksi riskien tai kustannuksien mukaan. Erilaisia sosiaali- ja terveydenhuollon asiakkaiden segmentoinnin malleja on monia erilaisia (Joynt ym., 2017). Näitä malleja on esimerkiksi suomalainen ikäihmisten segmentointiin kehitetty Suuntima-malli (Pirkanmaan sairaanhoitopiiri, 2021) ja kokonaisvaltaisesti väestöä segmentoiva Bridges to Health -malli (Lynn ym., 2007). Näiden lisäksi on olemassa myös muita kansallisia että kansainvälisiä segmentointimaljeja (Wood ym., 2019).

Sosiaali- ja terveydenhuollon asiakkaan segmenttimuutoksen ennustamiseen käytetään erilaisia kerättyjä tietoja. Esimerkiksi sosiaali- ja terveydenhuollon palvelutapahtumien merkinnät luovat hyvän kuvan nykyisestä tilanteesta. Voisi siis olettaa, että näiden tietojen avulla pystytään myös toteuttamaan erinäisiä ennusteita. Suomen sosiaali- ja terveydenhuollossa on käytössä erinäisiä koodistoja (Neittaanmäki, Tuominen ym., 2019) joita hyödyntämällä ennustemalleja voidaan toteuttaa. Tuotetun tiedon lisäksi on tärkeä ymmärtää kuinka tietoa voidaan hyödyntää. Yhden operatiivisen järjestelmän tieto ei luo kokonaisvaltaista kuvaa sosiaali- ja terveydenhuollon asiakkaan tilanteesta.

Yhdistämällä monia tietojärjestelmiä ja keräämällä tiedot tietovarastoon pystymme muodostamaan laajan ja helposti hyödynnettävän pohjan ennustemallien toteutukselle.

Luvussa 3.1 käsitellään sosiaali- ja terveydenhuollon asiakkaan segmentointia. Luvussa myös käsitellään sen vaikutuksia sosiaali- ja terveydenhuollon toimintaan. Lisäksi perehdytään tutkielman kokeellista osuutta varten Suuntima- ja Pärjääjä-malleihin. Luku 3.2 käsittelee kuinka Suomen sosiaali- ja terveystietoa voidaan hyödyntää ennustemallien toteuttamisessa. Luvussa käsitellään tietoaltaan ja tietovaraston hyödyntämistä koneoppimisen näkökulmasta. Lisäksi luvussa 3.3 toteutetaan kirjallisuuskatsaus, jossa käydään läpi sosiaali- ja terveydenhuollon asiakkaan segmenttimuutoksen ennustamiseen keskittyviä tutkimuksia.

3.1 Segmentointi sosiaali- ja terveydenhuollossa

Sosiaali- ja terveydenhuollossa segmentointia hyödynnetään potilaskeskeisessä hoidossa (Mäkinen, 2018). Potilaskeskeisen hoidon tarkoituksena on parantaa sosiaali- ja terveydenhuollon palveluita mahdollistamalla kohdennetumpia palveluita sosiaali- ja terveydenhuollon asiakkaalle (Mäkinen, 2018). Segmentoinnilla sosiaali- ja terveydenhuollon asiakkaita voidaan jakaa erilaisiin ryhmiin, joiden tarpeet ovat tarpeeksi heterogeenisiä muihin ryhmiin verrattuna. Näiden ryhmien hoitopolkuja voidaan segmentoinnin avulla kohdentaa ryhmän tarpeiden mukaisiksi. Asiakaslähtöisten hoitopolkujen avulla sosiaali- ja terveydenhuollon resursseja voidaan kohdentaa tehokkaammin niille potilaille, jotka niitä tarvitsevat (Mäkinen, 2018). Tämä tekee myös sosiaali- ja terveydenhuollosta tasapuolisempaa, koska hoitoketjut muodostetaan segmentoinnin mukaan. Sosiaali- ja terveydenhuollon asiakkaita voidaan myös segmentoida erilaisilla hierarkian tasoilla (Chong & Matchar, 2017). Makrotasolla segmentointi toteutetaan kansallisella tai globaalilla tasolla. Tällä tasolla potilaiden variaatio on kaikista suurinta. Mesotasolla segmentoidaan esimerkiksi samanlaisia sairauksia sairastavia potilaita. Mikrotasolla segmentointi painottuu yksilön tasolla.

Yksi merkittävimmistä sosiaali- ja terveydenhuollon asiakasryhmistä, johon segmentointi keskittyy, on kalliita palveluita usein tarvitseva potilas (High-need, high-cost, HNHC) (Jean-Baptiste ym., 2017). Nämä HNHC-potilaat voivat aiheuttaa erittäin suuria kulueriä sosiaali- ja terveydenhuollossa. Suomessa on tutkittu, että yhtä tai useampaa kroonista sairautta sairastavat potilaat, joita hoidetaan moniammatillisilla tiimeillä, eri organisaatioissa aiheuttavat noin 75% sosiaali- ja terveysalan kustannuksista (Neittaanmäki, Lehto ym., 2019). Tästä syystä segmentoinnilla ja asiakaslähtöisellä hoidolla

tavoitellaan hoidon tehostamista kaikista kalleimpien potilaiden osalta, jotta saatu hyöty olisi kaikista suurin. Lisäksi HNHC-potilaiden tunnistamisella pyritään estämään sosiaali- ja terveydenhuollon asiakkaan tilan huonontumista.

Suomessa hyvinvointialueen tehtäviin kuuluu asiakaslähtöisten hoito- ja palvelukokonaisuuksien muodostaminen yhtyeensovittamalla sosiaali- ja terveyspalveluita (Mäkinen, 2018). Tämä hyödyttää juurikin niitä potilaita, jotka tarvitsevat useita palveluita. Yhtenäinen kokonaisuus on mahdollista muodostaa asiakassegmentin avulla, jolloin erilaiset palvelut ja toimijat osaavat jo pelkän segmentin avulla järjestää erinäisiä hoitopolkuja tietylle potilassegmentille. Hyvin suunniteltu ja integroitu hoitopolku vähentää tarpeettomia päällekkäisiä palveluja eri organisaatioiden välillä, parantaen kustannustehokkuutta ja luoden mielekkäämpää palvelua potilaalle (Mäkinen, 2018).

Potilaiden segmentoinnissa käytetään kolmenlaisia datalähteitä (Jean-Baptiste ym., 2017). Ensimmäinen vaihtoehto on käyttää kvantitatiivista dataa, kuten henkilön hallinnollista tietoa sairaaloissa käynneistä tai kuluista. Toinen vaihtoehto on käyttää datalähteenä kvalitatiivista dataa, joka voi sisältää esimerkiksi lääkärin päätöksiä tai tekstimuotoisia kirjauksia. Lisäksi on hybridimalli, jossa hyödynnetään sekä kvantitatiivista että kvalitatiivista dataa. Hybridimallin käyttö vaikuttaa myös olevan näistä kolmesta luotettavin datalähde (Jean-Baptiste ym., 2017).

Wood ym. (2019) vertailevat yleisiä sosiaali- ja terveydenhuollon alalla käytettyjä segmentointimalleja. Yleiset kategoriat ovat päätöksiin perustuva segmentointi (Judgemental), määrätty lokerointi (Proscribed binning), päätöspuut (Decision trees) ja ryhmittely (Clustering). Päätöksiin perustuvissa segmentointimalleissa käytetään usein esimerkiksi ikää, sukupuolta, kroonisia sairauksia tai näiden yhdistelmiä, joissa segmentointi tapahtuu tietyn jaon perusteella. Tämä tapa on yksinkertainen ja tehokas, mutta luokat voivat jäädä matalatasoisiksi. Määrätyssä lokeroinnissa sääntöjen määrä kasvaa ja niissä voidaan käyttää moninaista mielivaltaista logiikkaa. Tässä menetelmässä segmenttien erottuvuus heikkenee muihin verrattuna, mutta etuna metodissa on segmenttien helppo ymmärrettävyys (Wood ym., 2019). Määrätyn lokeroinnin segmentointimalleihin kuuluu tutkielman kokeellisessa osuudessa käytetty Suuntima-malli sekä myös mainittu Bridges to Health -malli. Päätöspuut ovat tehokas tapa tuottaa segmentointimalli, jossa segmenttien välinen erottuvuus on hyvä. Ryhmittelyssä huomattiin, että se ei ole kovin tehokas segmentointimallina (Wood ym., 2019).

Luvussa 3.1.1 käydään läpi Suomessa toteutettu segmentointimalli. Lisäksi luvussa perehdytään Suuntima-mallin mukaisiin sosiaali- ja terveydenhuollon asiakkaan segmentteihin. Luvussa 3.1.2 perehdytään Pärjääjä-malliin. Tämä malli on Suuntima-

malliin perustuva dynaaminen segmentointimalli.

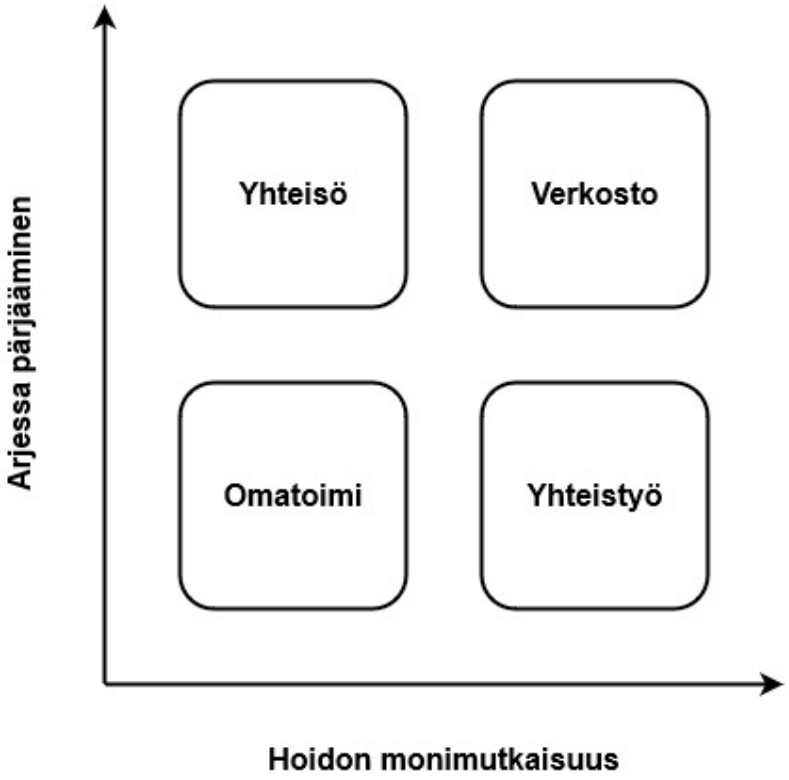
3.1.1 Suuntima-malli

Tämä luku selvittää kuinka Suomessa kehitetty Suuntima-malli toimii ja kuinka sen avulla voidaan segmentoida sosiaali- ja terveydenhuollon asiakkaita hoitotarpeen mukaan. Luvussa käydään myös läpi Suuntima-mallin neljä potilassegmenttiä ja niiden yleiset kuvaukset sekä merkityksen sosiaali- ja terveydenhuollossa. Suuntima-mallin mukainen segmentointi on oleellinen, koska se on tutkimuksen kokeellisessa osuudessa käytettävän Pärjääjä-mallin oleellinen taustoitus. Syväoppimismalli pyrkii ennustamaan sosiaali- ja terveydenhuollon segmenttimuutoksen Suuntima-mallin mukaiseen jaotteluun perustuen.

Suuntima-malli on Kurkiaura-hankkeessa 2011-2015 kehitetty kysymyksiin perustuva työväline, jonka avulla sosiaali- ja terveydenhuollon asiakas ja ammattilainen pystyvät yhdessä vastaanoton yhteydessä luokittelemaan asiakkaan yhteen neljästä eri segmentistä: yhteistyö-, omatoimi-, yhteisö- ja verkostoasiakkuus (Pirkanmaan sairaanhoitopiiri, 2021). Tämä sosiaali- ja terveydenhuollon asiakkuuksien jaottelu on esitetty kuvassa (3.1). Työkalun alkuperäisenä tarkoituksena oli löytää asiakaslähtöisiä palvelumalleja sydänsairastuneille, mutta työkalua on sovellettu myös ikäihmisten luokitteluun (Mattila, 2016). Potilassegmenttien tarkoituksena on tunnistaa erilaisia asiakkaita, ja näin tarjota kohdennetumpaa palvelua, joka ilmenee tehokkaampana resurssien käyttönä ja parempana hoitoketjuna.

Suuntima-mallin mukainen sosiaali- ja terveydenhuollon asiakkaan segmentointi toteutetaan asiakkaan ja ammatilaisen kanssa yhdessä vastaanotolla tai sairaalassa käyttäen työkalun tarjoamia kysymyksiä (Mattila, 2016). Sekä asiakas että ammattilainen vastaavat heille suunnattuihin kysymyksiin, jonka jälkeen työkalu laskee Suuntima-mallin mukaisen potilassegmentin, joka on yksi neljästä segmenttiryhmästä. Asiakkaalle suunnatut kyselyn kysymykset pyrkivät selvittämään asiakkaan mielialaa, kykyä huolehtia itsestään, arjessa pärjäämisen voimavaroja ja läheisten henkilöiden vaikutusta omaan jaksamiseen (Mattila, 2016). Kyselyssä ammattilaiselle suunnatut kysymykset pyrkivät selvittämään millainen asiakkaan sairauksien vaikutus on asiakkaan toimintakykyyn, tarvitseeko asiakas moniammattilaista hoitoa ja ovatko asiakkaan kognitiiviset kyvyt riittävällä tasolla hoidon vaativuuteen nähden (Mattila, 2016).

Suuntima-mallin neljä segmenttiä sijoittuvat kokonaisuutena nelikentäksi kahden akselin muodostamaan kaavioon kuten esitetty kuvassa (3.1). Vaaka-akselin avulla esitetään,



Kuva 3.1: Suuntima-mallin mukainen nelikenttä. Vaaka-akseli kuvaa asiakkaan kuntoutuksen vaativuutta ammattilaisen näkökulmasta ja pystyakseli kuvaa asiakkaan näkemystä hänen omasta arjessa pärjäämisestään (Mattila, 2016).

kuinka vaativa asiakkaan kuntoutusprosessin arvioidaan olevan. Tämä akseli perustuu ammattilaiselta saatuihin vastauksiin työkalun perusteella. Lisäksi pystyakseli kuvaa, kuinka arjessa pärjääminen onnistuu asiakkaalta. Tämän suuntaiseen heiluntaan on eniten vaikutusta asiakkaan omilla vastauksilla Suuntima-mallissa. Kun molempien akselien arvot ovat tiedossa, sijoitetaan piste kuvaajaan. Tämän jälkeen nähdään, mihin nelikentän segmenttiin asiakas kuuluu.

Omatoimi-asiakkuus on Suuntima-mallin asiakassegmentti, jossa hoitoprosessin vaativuus on matala, ja asiakas kokee arjessa pärjäämisen helpoksi (Mäkinen, 2018). Omatoimi-asiakkaiden toimintakyky on siis hyvä, eivätkä he tarvitse apua arjessa pärjäämiseen sairautensa tai terveysongelmansa kanssa. Hoitosuunnitelmat painottuvat akuuttien sairauksien hoitoon. Omatoimi-asiakkaan hoitopolun tavoite on parantaa asiakkaan tilaa niin, että hän pystyy toimimaan taas täysin itsenäisesti. Omatoimi-asiakkaille ehdotetaan palveluita, jotka edistävät toimintakykyä. Esimerkiksi liikuntaryhmät, neu-

volan palvelut, ennaltaehkäisevät palvelut ja matalan kynnyksen palvelut. Omatoimi-asiakkaat pystyvät koordinoimaan hoitojaan itse, ja he eivät tarvitse apua esimerkiksi laskujen maksamiseen tai sähköisten palveluiden käyttöön. Tämä asiakassegmentti ei kuormita terveyden- ja sosiaalihuollon resursseja suureksi, koska hoito ei ole vaativaa, eikä asiakas tarvitse avustajia toimintakyvyn säilyttämiseen. Siksi on tavoiteltavaa, että mahdollisimman moni asiakas pysyisi tässä segmentissä mahdollisimman pitkään.

Yhteistyö-asiakkuudessa asiakkaat pärjäävät arjessaan, mutta hoidon vaativuuden ja monimutkaisuuden vuoksi he tarvitsevat useiden erilaisten ammattilaisten vastaanotto-käyntejä (Mäkinen, 2018). Hoitopolun tavoitteena on ylläpitää nykyistä toimintakykyä ja jaksamista. Hoitosuunnitelman painopiste on toimintakyvyn ylläpidossa sekä yhteisessä hoito- ja palvelusuunnitelmassa. Hoidolla on koordinaattori, ja ammattilaisen kanssa pyritään järjestämään yhteisvastaanottoja. Kun voimavarat ovat hyvät, hoitona voidaan käyttää esimerkiksi kotisairaalaa, kotisairaanhoitoa, kuntoutusta, terapiaa ja muita tarpeellisia hoidon palveluita. Hoidolle ominaista on yhtenäinen suunnitelma, jossa selviää hoidon kokonaisuus. Yhteydenotossa potilas pyrkii hyödyntämään sähköisten välineiden itsenäistä käyttöä yhteydenpidossa. Yhteistyö-asiakkuus on selkeästi kalliimpi asiakassegmentti resurssien näkökulmasta, koska hoitokeinot ovat monimutkaisempia.

Yhteisö-asiakkuuden asiakassegmentissä asiakas kokee omien voimavarojensa heikentyneen merkittävästi, mutta hoidon toteutus on selkeää. Tämän asiakassegmentin hoitopolun tavoite on nostaa asiakkaan toimintakykyä ja voimavaroja (Mäkinen, 2018). Tästä syystä hoidon painopiste on kokonaisuuden arviointi ja uusien keinojen löytäminen, joiden avulla asiakas saa akuutit sairaudet hoidettua, ja voimavaroja mahdollisesti vahvistettua. Asiakasta ajanvarauksessa auttavat omahoitaja tai muu ammattilainen, joka koordinoi hoitoa. Vastaanottovaihtoja voi olla esimerkiksi yksilövastaanotto tai kokonaisarviointi. Hoitopolun mahdollisia keinoja ovat esimerkiksi tukipalvelut arjessa pärjäämiseen, vertaistuki, omaisten tuki, ryhmätoiminta, terveysaseman palvelut ja palvelukeskukset. Hoitoa koordinoi mahdollisesti omainen, terveyskeskuksen hoitaja tai muu tukihenkilö. Varaukset hoidetaan pääsääntöisesti puhelimella.

Verkosto-asiakkuudessa asiakkaan voimavarat eivät välttämättä riitä asioiden hoitamiseen, hoito myös ollessa haasteellista sekä monitahoista (Mäkinen, 2018). Hoitopolun tavoite on taata asiakkaan arjessa pärjääminen ja mahdollistaa kotiin palaaminen. Hoitosuunnitelman painopiste keskittyy hoidon kokonaisuuden hallintaan sekä mahdollisten haittojen välttämiseen. Asiakkaalla on mahdollisesti kotihoitaja tai omaishoitaja, joka järjestää asiakkaan ajanvaraukset. Käytetyt palvelut ovat hoidosta ja sairaudesta riippuen: kotihoito, edunvalvonta, kotihoidon lääkärikäynnit, ateriapalvelut, sekä mielenterveys- ja vammaispalvelut. Hoitoa koordinoi kotihoitaja tai omaishoitaja. Yhteydenpidossa

käytetään pääsääntöisesti puhelinta, ja ammattilaiset tekevät säännöllisesti ennakoivia yhteydenottoja. Mahdollisia sähköisiä palveluita voidaan hyödyntää asiakkaan omaisten kanssa. Verkosto-asiakkuus on kaikista raskain hoitopolku resurssien näkökulmasta, ja se ei myöskään ole mielekästä asiakkaalle itselleen. Tästä syystä ennakoivilla hoitotoimilla tähän segmenttiin siirtymistä voidaan viivyttää.

Suuntima-työkalu auttaa asiakasta ymmärtämään omaa hoidollista tilannettaan ja edistää sosiaali- ja terveyspalveluiden asiakaslähtöisempää toteuttamista. Potilaan segmentoinnista on siis hyötyä sekä asiakkaan että hoitohenkilökunnan näkökulmasta (Mäkinen, 2018). Lisäksi Suuntima-mallin hyötyihin lukeutuvat luvussa 3.1 mainittu kustannustehokkuus, sekä potilaskeskisempi hoitoketju ja integraation edistäminen. Suuntima-malli on kuitenkin toteutettu kyselypohjaisella menetelmällä, mikä johtaa siihen, että segmentti ei päivity tilanteen muuttuessa. Tällöin kysely pitäisi tehdä mahdollisesti uudestaan. Lisäksi kyselyn teetättäminen on resursseja kohtuullisen paljon käyttävä prosessi.

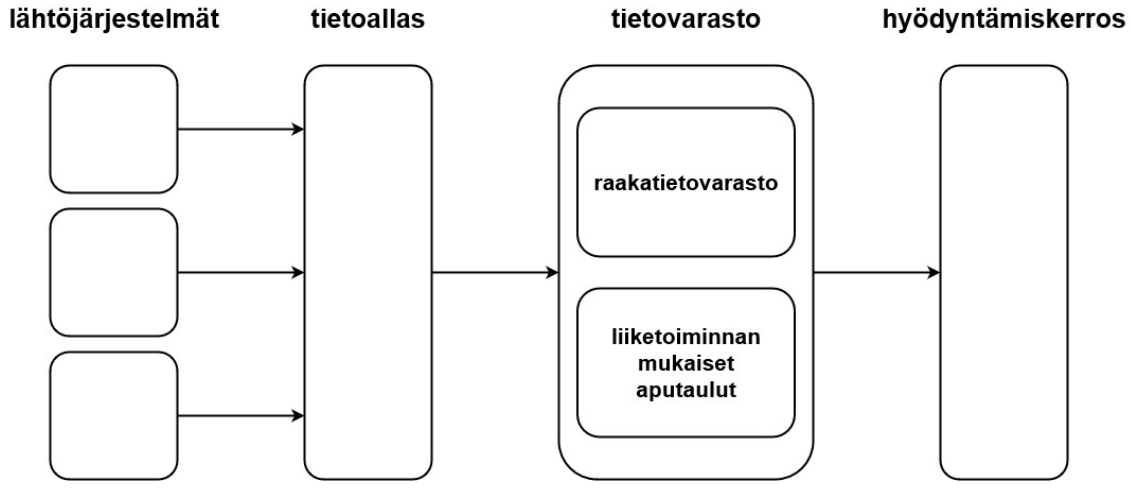
3.1.2 Pärjääjä-malli

Koska Suuntima-mallissa toteutettu segmentointi on staattinen, ei kyseinen luokittelu asiakkaalle muutu muulloin, kun vain tekemällä uuden kyselyn. Asiakkaan segmentti kuitenkin muuttuu ajan saatossa, ja tutkielmassa ei pystytä toteuttamaan kyselyitä kaikille asiakkaille esimerkiksi kuukauden kierrolla, koska tämä olisi resurssien kannalta hyvin kallis ja epäkäytännöllinen prosessi. Lisäksi Suuntima-mallin kyselyitä kohdennetaan asiakkaille, joille ammattilaiset päättävät sen olevan hoidon kannalta hyödyllistä. Tästä syystä Suuntima-mallin mukaista segmentointia ei toteuteta jokaiselle sote-alueella olevalle henkilölle. Segmenttimuutoksen ennustaminen neuroverkoilla kuitenkin tarvitsee mahdollisimman paljon koulutusdataa, sillä tulokset keskimäärin paranevat verkkojen ja datamäärän kasvaessa (Goodfellow ym., 2016). Tästä syystä tutkielman kokeellisessa osuudessa käytetään pärjääjä-mallia, jossa segmentit päätellään dynaamisesti käytettyjen tietojen perusteella. Pärjääjä-malli on Keusotessa käytössä oleva segmentointimalli, jonka segmentit ovat Suuntima-mallin mukaisia. Malli poikkeaa ainoastaan sillä, ettei segmentin päättelyyn käytetä ammattilaisen ja asiakkaan täyttämiä kyselyitä. Dynaamisessa segmentoinnissa hyödynnetään sosiaali- ja terveydenhuollon asiakkaan tietoja, jonka avulla Suuntima-mallin mukaiset segmentit päätellään kuukausitasolle käytettyjen palveluiden perusteella. Pärjääjä-mallin avulla voidaan myös segmentoida kaikki sote-alueen asiakkaat prosessoimalla tietovarastossa olevat tiedot. Tällä tavalla neuroverkkomalleihin saatavan datan määrä kasvaa, mikä edistää potilaan segmenttimuutoksen ennustamista.

Dynaamisessa segmentoinnissa jokaiselle henkilölle voidaan laskea segmentti automaattisesti tietylle ajankohdalle, hyödyntäen esimerkiksi tietovarastossa säilytettyä tietoa henkilöstä. Käytännössä toteutus riippuu siitä, millaista tietoa segmentointiin voidaan käyttää. Esimerkiksi potilaan käyntien lukumäärää tietyn kuukauden aikana voidaan hyödyntää kuukauden segmentin päättelyssä. Lisäksi erilaisia palveluita voidaan käyttää segmentoinnissa apuna. Pärjääjä-mallin mukainen dynaaminen segmentoinnin toteutus käydään läpi luvussa 3.1.2, jossa tietovarastosta saatuja potilastietoja hyödynnetään substanssiosaamisen perusteella siten, että voidaan tuottaa dynaaminen segmentointi.

3.2 Suomen sosiaali- ja terveystiedon hyödyntäminen koneoppimisessa

Jotta Suomen sosiaali- ja terveydenhuollon asiakkaiden segmenttimuutosta voidaan ennustaa neuroverkkojen avulla, tulee ennustemallien koulutuksessa käytettävä datajoukko kerätä. Tässä luvussa käydään läpi kuinka sosiaali- ja terveydenhuollon operatiivisissa järjestelmistä oleva data voidaan kerätä yhtenäiseksi ja hyödynnettäväksi tietovarastoksi, josta koulutusdatan rakentaminen on mahdollista.



Kuva 3.2: Tietovarastoinnin arkkitehtuurikuvaus. Operatiivisten tietojärjestelmien tuottama tieto tuodaan tietoaltaalle. Tietoaltaalta tehdään siirto tietovarastoon Data Vault 2.0 -periaatteen mukaisesti. Tietovarastossa muodostetaan lisäksi liiketoiminnan mukaisia aputauluja. Tietovaraston avulla muodostetaan hyödyntämiskerrokseen tietotarpeiden mukaisia tauluja, sekä datajoukkoja.

Koska Suomen sosiaali- ja terveydenhuollon tietojärjestelmät ovat tuotettu useilla eri

järjestelmillä useissa eri palveluissa, on data hyvin siiloutunutta (Neittaanmäki & Lehto, 2018). Tällä tarkoitetaan tilannetta, jossa sosiaali- ja terveydenhuollon operatiivisessa järjestelmässä olevaa tietoa ei ole suunniteltu hyödynnettäväksi muualla kuin alkuperäisessä järjestelmässä. Tämä esiintyy yleensä poikkeavina tietoarkkitehtuureina, sekä poikkeavina käsitteinä. Sosiaali- ja terveydenhuollon asiakkaan kokonaisvaltaisen palvelutapahtumien ketjun muodostamiseksi tulee monia operatiivisia järjestelmiä integroida yhtenäisen datan tuottamiseksi. Järjestelmien integroimiseen on olemassa monia ratkaisuja (Linstedt & Olschimke, 2015). Tässä tutkielmassa perehdytään useissa Suomen sosiaali- ja terveydenhuollon tietovarastointiprojekteissa käytettävään DigiFinlandin käsitemallinnuksen säännöstöihin (Virta-hanke/ DigiFinland Oy, 2021), sekä Dan Linstedin kehittämään Data Vault 2.0 periaatteeseen (Linstedt & Olschimke, 2015).

Kuvassa (3.2) on esitetty arkkitehtuurikuvaus, jota hyödynnetään sosiaali- ja terveystiedon integroimisessa Keusoten tietovarastointiprojektissa. Arkkitehtuurista näkyy kuinka yksittäiset tietojärjestelmät tuodaan tietoaltaalle, jossa tiedot, joista henkilö voidaan tunnistaa, pseudonymisoidaan. Tällä tarkoitetaan tietojen käsittelyä niin, ettei tietoa voida yhdistää henkilöön ilman lisätietoja. Pseudonymisoitu tieto siirretään tietovarastoon, jossa siitä muodostetaan raakatietovarasto (Raw data vault). Raakatietovarastossa alkuperäinen tuotettu tieto säilyy, eikä eroavia järjestelmiä pyritä vielä integroimaan. Raakatietovaraston avulla kuitenkin tuotetaan liiketoiminnan näkökulmasta lisätauluja (Business vault), joissa eri lähtöjärjestelmien sisältämät samat käsitteet yhdistetään yhtenäiseksi käsitetauluksi. Näitä tauluja, sekä raakatietovarastossa olevia tauluja käytetään hyödyntämiskerroksessa, jossa tuotetaan raportointiin, sekä koneoppimisen käyttötarkoitukseen sopivia tietorakenteita.

Luvussa 3.2.1 perehdytään muutamaan Suomen sosiaali- ja terveydenhuollossa käytettyyn koodistoon. Lisäksi tutkitaan sosiaali- ja terveystiedon määritelmää ja ominaisuuksia. Luvussa 3.2.2 käydään läpi tietoaltaan määritelmä ja sen osuus tietojärjestelmien integroinnissa. Luvussa 3.2.3 käydään läpi DigiFinlandin mukainen käsitemallinnus, sekä Data Vault 2.0 -periaatetta, jonka avulla yhtenäinen tietovarasto voidaan toteuttaa. Luvussa 3.2.4 käsittelee tietovaraston hyödyntämistä ja siihen liittyviä rakenteita joiden avulla ennustemallien kouluttamiseen vaadittava datajoukko voidaan muodostaa.

3.2.1 Sosiaali- ja terveystieto

Sosiaali- ja terveydenhuollon operatiiviset järjestelmät tuottavat paljon erilaista dataa (Häyrinen ym., 2008). Data perustuu useisiin erilaisiin koodistoihin, joiden avulla hoitotapahtumia tai tuloksia voidaan dokumentoida (Häyrinen ym., 2008). Nämä koodistot

ovat usein kansainvälisiä (Häyrinen ym., 2008). Suomessa on käytössä kansallisia ja kansainvälisiä koodistoja (Neittaanmäki & Lehto, 2018). Tässä luvussa käsitellään sosiaali- ja terveystiedon rakennetta ja käydään läpi Suomen sosiaali- ja terveydenhuollon koodistoja.

Häyrinen ym. (2008) tekemässä tutkimuksessa viitattiin kansainvälisen standardisoimisjärjestön (International organization for standardization, ISO) sähköisten terveystietojen (Electrical health recods, EHR) määritelmään. Tämän määritelmän mukaan sähköiset terveystiedot ovat digitaalisesti arkistoituja potilastietoja, jotka ovat tallennettu turvallisesti ja johon pääsevät ovat valtuutettuja käyttäjiä (Häyrinen ym., 2008). Ne sisältävät takautuvaa, ylläpitävää sekä tulevaa tietoa ja niiden ensisijainen tarkoitus on tukea jatkuvaa, tehokasta ja laadukasta terveydenhuoltoa (Häyrinen ym., 2008). Sähköisiä terveystietoja jaotellaan tyypillisesti hoidon asteen avulla ja tyypillinen tapa on jaotella sähköiset terveystiedot perusterveydenhuoltoon, keskiasteen hoitoon ja korkean asteen hoitoon (Häyrinen ym., 2008). Tutkimuksessa huomattiin, ettei monikaan sähköinen terveystietojärjestelmä tarjoa julkisesti tietoa datan tietorakenteesta (Häyrinen ym., 2008). Tästä syystä tarkan ja yleisen määritelmän muodostaminen kansallisen tai kansainvälisen terveystiedon tietorakenteesta on vaikea määrittää. Usein sosiaali- ja terveystieto koostuu sosiaali- ja terveydenhuollon asiakkaan käyntiin liittyvistä merkinnöistä (Häyrinen ym., 2008), joita ovat esimerkiksi diagnoosimerkintä, käyntisyy ja hoidon tarpeen määrittely (Virta-hanke/ DigiFinland Oy, 2021). Sosiaali- ja terveydenhuollon asiakkaista muodostuvat tiedot tallennetaan potilastietojärjestelmiin usein tapahtumatasolla (Neittaanmäki & Lehto, 2018). Dokumentoituja tieto-osia sähköisissä terveystiedoissa ovat esimerkiksi kartoitukset, terveyden arvioinnit, terveys- ja hoitosuunnitelmat, sairaushistoria, fyysiset tutkimukset, diagnoosit, testit, hoidot ja lääkitykset (Häyrinen ym., 2008). Suomessa terveyden- ja sosiaalihuollon tiedot ovat usein erillisissä tietojärjestelmissä (Neittaanmäki & Lehto, 2018). Suomessa sosiaali- ja terveystiedon tietorakenne perustuvat käytettyyn tietojärjestelmään ja tästä syystä tietorakenteet voivat poiketa toisistaan useissa paikoissa (Neittaanmäki & Lehto, 2018). Tätä kuvataan yleensä tietojärjestelmien siiloutumisena, sillä datan hyödyntäminen kansallisella tasolla on haastavaa poikkeavista tietorakenteista johtuen.

Tämän tutkielman kokeellisessa osuudessa sosiaali- ja terveydenhuollon potilasdataa kerätään Keski-Uudenmaan sote -kuntayhtymän tietovarastosta. Suomen sosiaali- ja terveydenhuollossa koodistot ovat terveyden ja hyvinvoinnin laitoksen (THL) ylläpitämiä (Terveyden ja hyvinvoinnin laitos, 2021d). Kokeellisessa osuudessa tullaan hyödyntämään ICD-10, ICPC-2 ja asiointitapa merkintöjä kun sosiaali- ja terveydenhuollon asiakkaiden palvelutapahtumista kerätään tietoa Keski-Uudenmaan sote -kuntayhtymän tietovarastosta.

Taulukko 3.1: Tautiluokitus ICD-10-koodiston pääluokat (Terveyden ja hyvinvoinnin laitos, 2021d)

Pääluokat	Selite
A00–B99	Tartunta- ja loistauteja
C00–D48	Kasvaimet
D50–D89	Veren ja verta muodostavien elinten sairaudet sekä eräät immuunimekanismin häiriöt
E00–E90	Umpierityssairaudet, ravitsemussairaudet ja aineenvaihduntasairaudet
F00–F99	Mielenterveyden ja käyttäytymisen häiriöt
G00–G99	Hermoston sairaudet
H00–H59	Silmän ja sen apuelinten sairaudet
H60–H95	Korvan ja kartiolisäkkeen sairaudet
I00–I99	Verenkiertoelinten sairaudet
J00–J99	Hengityselinten sairaudet
K00–K93	Ruuansulatuselinten
L00–L99	Ihon ja ihonalaiskudoksen sairaudet
M00–M99	Tuki- ja liikuntaelinten sekä sidekudoksen sairaudet
N00–N99	Virtsa- ja sukupuolielinten sairaudet
O00–O99	Raskaus, synnytys ja lapsivuoteus
P00–P96	Eräät perinataaliaikana alkaneet tilat
Q00–Q99	Synnynnäiset epämuodostumat, epämuotoisuudet ja kromosomipoikkeavuudet
R00–R99	Muualla luokittamattomat oireet, sairaudenmerkit sekä poikkeavat kliiniset ja laboratoriolöydökset
S00–T98	Vammat, myrkytykset ja eräät muut ulkoisten syiden seuraukset
V01–Y98	Vammojen, sairauksien ja kuoleman ulkoiset syyt
Z00–ZZB	Tekijöitä, jotka vaikuttavat terveydentilaan ja yhteydenottoihin terveyspalvelujen tuottajiin

ICD-10 (International Statistical Classification of Diseases and Related Health Problems) on kansainvälisesti laajalti käytössä oleva ja Suomessa kansallisena käytetty koodisto (Terveyden ja hyvinvoinnin laitos, 2021d). Koodistoa käytetään kuolinsyy- ja sairastavuustilastotietoja kerättäessä, potilasasiakirjan diagnoosimerkinnöissä, lääkärinlausunnoissa, sekä sosiaalihuollon asiakasasiakirjoissa (Terveyden ja hyvinvoinnin laitos, 2021d). Suomessa ICD-10 otettiin käyttöön 1996, ja sen sisältämien erinäisten kategorioiden määrä on todella laaja. Vuoden 2007 koodistopalvelimen versiossa oli käytössä 14362 erilaista koodia. Koodiston pääluokat on esitelty taulukossa (3.1). Koodisto on hierarkkinen, jonka takia datajoukon keräyksessä on oleellista määritellä kerättävän hierarkian taso koodista.

Taulukko 3.2: ICPC-2-koodiston pääluokat (Terveyden ja hyvinvoinnin laitos, 2021c)

Pääluokat	Selite
A	Yleiset ja epämääräiset
B	Veri, verta muodostavat elimet sekä immuunijärjestelmä
D	Ruuansulatus
F	Silmä
H	Korva
K	Sydän ja verenkierto
L	Tuki- ja liikuntaelimistö
N	Hermosto
P	Mielenterveys
R	Hengityselimet
S	Iho
T	Umpieritys/aineenvaihdunta/ravitsemus
U	Virtsaelimet
W	Raskaus, synnytys ja perhesuunnittelu
X	Naisen sukuelimet
Y	Miehen sukuelimet
Z	Sosiaaliset ongelma

ICPC-2 (International Classification of Primary Care © Wonca 2005-2014) eli kansainvälinen perusterveydenhuollon luokitus, on Suomessa käytössä terveydenhuollon potilastietojärjestelmissä (Terveyden ja hyvinvoinnin laitos, 2021a). Koodiston avulla voidaan määritellä potilaan hoitoon hakeutumisen tulosyitä, toteutettuja hoitotoimintoja ja tehtyjä interventioita (Terveyden ja hyvinvoinnin laitos, 2021a). ICPC-2 koodisto on Suomessa päivitetty viimeksi vuonna 2010 (Terveyden ja hyvinvoinnin laitos, 2021a). ICPC-2 sisältää ICD-10 verrattavia koodeja, mutta ei sovellu yksin kategorisoimaan kaikkia sosiaali- ja terveydenhuollon asiakkaiden diagnooseja (Terveyden ja hyvinvoinnin laitos, 2021c). Koodiston pääluokat on esitelty taulukossa (**??**). Huomioitavia koodiston pääluokkia ovat varsinkin W- ja Z-koodit, sillä sosiaalisten ongelmien merkinnät täydentävät ICD-10 koodistoa tehokkaasti käyntisyiden osalta.

Asiointitapa on terveyden ja hyvinvoinnin laitoksen ylläpitämä koodisto jota käytetään Suomen sosiaali- ja terveydenhuollon tietojärjestelmissä kuvaamaan asiakkaan ja ammattilaisen välistä asiointia (Terveyden ja hyvinvoinnin laitos, 2021b). Koodisto sisältää koodeja erilaisille käynneille ja yhteydenotoille. Tämän tutkielman kokeellisessa osuudessa käytetyt asiointitapakoodit ovat esitettynä taulukossa (3.3).

Taulukko 3.3: Keusoten tietovarastossa käytetyt erilaiset asiointitapakoodit

Koodi	Selite
R10	Asiakkaan käynti vastaanotolla
R20	Ammattihenkilön käynti asiakkaan kotona
R30	Ammattihenkilön käynti asiakkaan työpaikalla
R40	Sairaalakäynti
R41	Ammattihenkilön käynti muualla kuin kotona tai työ
R50	Puhelinyhteys
R51	Sähköinen asiointi
R52	Reaaliaikainen etäasiointi
R55	Kirje
R56	Etäasiointi ilman reaaliaikaista kontaktia
R60	Ammattihenkilöiden välinen konsultaatio
R70	Asiakirjamerkintä ilman asiakaskontaktia
R71	Ammattihenkilöiden välinen neuvottelu
R72	Asiakkaan asian hoito
R80	Vuodeosastohoito
R90	Muu asiointi

3.2.2 Tietoallas

Tässä luvussa käydään läpi tietoaltaan määritelmä sekä sen vaikutus tietojärjestelmien integrointiin. Lisäksi käsitellään Suomen sosiaali- ja terveydenhuoltoon liittyviä muita järjestelmiä joiden avulla tietoa voisi rikastaa.

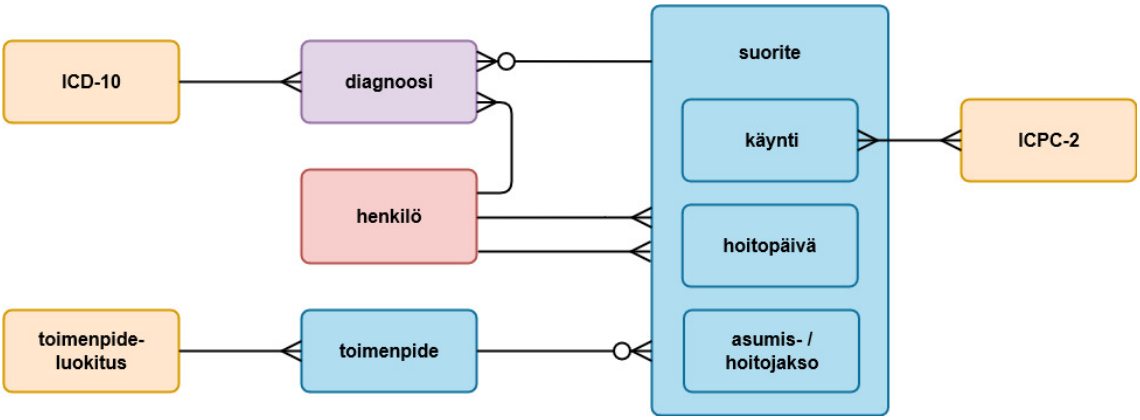
Tietoallas on suurten datamäärien arkisto, joka sisältää sekä jäsenneltyä (Structured data) että jäsentelemätöntä dataa (Unstructured data) (Stein & Morrison, 2014). Tietoaltaan tarkoitus on kerätä dataa useista erilaisista tietojärjestelmistä ja muista tietolähteistä. Tietoallas voi myös tietovarastosta poiketen tallentaa esimerkiksi kuvia, dokumentteja ja sensoridataa (Stein & Morrison, 2014). Integrointiin sisältyy myös riskejä, sillä useat erilliset järjestelmät estävät tehokkaammin suurien arkaluontoisten tietojen, kuten potilasdatan vuotamista epäluotettaville tahoille (Franz ym., 2020). Tästä syystä integraatio tulee toteuttaa standardien mukaisesti noudattaen tietoturvallisen datan käsittelyn periaatteita.

Sosiaali- ja terveydenhuollon tietojärjestelmät tuottavat paljon sosiaali- ja terveysdataa. Siiloutuneista järjestelmistä on kuitenkin vaikea tuottaa kokonaisvaltaista dataa sosiaali- ja terveydenhuollon asiakkaista. Sirpaloitunut ja siiloutunut data on haaste sosiaali- ja terveystietoihin pohjautuvien digitaalisten ratkaisujen kuten koneoppimisen ennustemallien kehittämisessä (Franz ym., 2020). Tehokas sosiaali- ja terveystiedon

hyödyntäminen pystytään saavuttamaan yhtenäisellä tietoaltaalla, johon kerätään kaikkien sote-kuntayhtymän operatiivisten järjestelmien tiedot. Tulevaisuudessa olisi myös mahdollista toteuttaa koko Suomen laajuinen tietoallas, johon kerättäisiin kaikista operatiivisista järjestelmistä potilasdataa, kliinistä dataa, tutkimusdataa, sekä muita sosiaali- ja terveydnhuoltoon liittyviä tietopankkeja, kuten kansaneläkelaitoksen ja biopankin dataa. Tällä integraatiolla olisi mahdollista toteuttaa monia analytiikan sovelluksia kuten interventiosuosituksia, riskiryhmien tunnistamista sekä kansallisen terveydentilan raportointia (Neittaanmäki, Lehto ym., 2019).

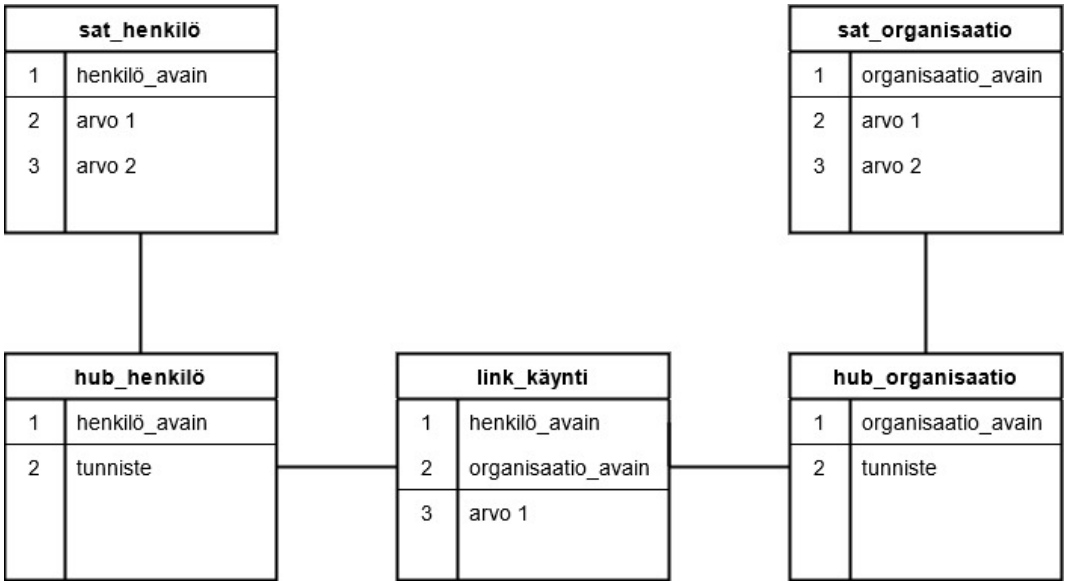
3.2.3 Tietovarasto

Tietoaltaalla datan hyödyntämisen haasteena on usein useiden järjestelmien vahvasti poikkeavat datan tietoarkkitehtuurit (Franz ym., 2020). Esimerkiksi kahden erilaisen terveydenhuollon potilasjärjestelmän tieto samanlaisesta terveydenhuollon tapahtumasta voi poiketa täysin toisistaan. Tietovarastoinnin tarkoituksena on toteuttaa tietomalli, jonka avulla pystytään yhtenäistämään erilaisten tietojärjestelmien käsitteistö ja eriävät data-arkkitehtuurit yhdeksi hyödynnettäväksi kokonaisuudeksi. Tutkielmassa hyödynnetään DigiFinlandin ylläpitämää sosiaali- ja terveydenhuollon käsitemallinnuksen ohjetta (Virta-hanke/ DigiFinland Oy, 2021), sekä Dan Linstedin kehittämää Data Vault 2.0 - periaatetta (Linstedt & Olschimke, 2015). Tässä luvussa käydään läpi tietovarastoinnin sekä sosiaali- ja terveydenhuollon käsitemallinnuksen periaatteet Suomen sote-organisaatioiden näkökulmasta.



Kuva 3.3: DigiFinlandin tuottamien käsitemallinnussääntöjen mukainen rajattu käsitemalli sosiaali- ja terveydenhuollon asiakkaasta. Taulut ovat värikoodattuja käsitetyyppeihin, jossa punainen on master data -käsite, sininen on tapahtuma-käsite, violetti sopimus-käsite ja oranssi referenssikäsite.

Tietovarastoa muodostaessa tulee toteuttaa käsitemallinnus, jonka avulla tietoaltaalla olevat taulut voidaan lisätä tietovarastoon. Käsitemallinnuksen avulla pystytään yhdistämään useiden eri lähdejärjestelmien tauluja yhden yhteisen käsitteen alle. Suomen sosiaali- ja terveydenhuollossa käytettävä DigiFinlandin määrittelemät ja ylläpitämät käsitemallinnussäännöstöt koostuvat neljästä käsitetyypistä, jotka ovat: master data -, sopimus-, tapahtuma-, sekä referenssi-käsite. Master data -käsitteet ylläpitävät tietoa itsenäisistä käsitteistä, jotka eivät tarvitse muita tauluja viittauksiksi. Tällaisia käsitteitä ovat esimerkiksi henkilö, organisaatio ja palvelu (Virta-hanke/ DigiFinland Oy, 2021). Sopimus-käsitteet ovat käsitteitä, jotka eivät ole itsenäisiä vaan viittaavat muihin tauluihin. Lisäksi sopimus-käsitteissä ominaista on alku- sekä loppuaika. Näitä käsitteitä ovat esimerkiksi henkilön palvelusuunnitelma, osoite, sekä pitkäaikainen diagnoosi (Virta-hanke/ DigiFinland Oy, 2021). Tapahtumakäsitteet ovat käsitteitä, joille on merkitty aikaleima siitä milloin tapahtuma on tapahtunut. Esimerkiksi käynti- ja palvelutapahtuma ovat tällaisia käsitteitä (Virta-hanke/ DigiFinland Oy, 2021). Referenssi-käsitteet sisältävät esimerkiksi sosiaali- ja terveydenhuollossa käytettyjä koodistoja sekä niiden selkokielisiä selitteitä. Näitä käsitteitä ovat esimerkiksi ICD-10, ICPC-2 ja käyntisyy (Virta-hanke/ DigiFinland Oy, 2021). Kuvassa (3.3) esitetään kuinka erilaiset käsitteet voivat olla linkitettynä toisiin käsitteisiin muodostaen kokonaisen käsitemallin.



Kuva 3.4: Data Vault 2.0 mukainen arkkitehtuuri, jossa tieto tallennetaan satelliitti-, link- ja hub-tauluihin. Esimerkkiarkkitehtuurissa käsitteistä henkilö ja organisaatio on muodostettu hub-taulut. Nämä käsitteet yhdistyvät käsitteessä käynti, joka muodostaa linkkitaulun. Lisäksi henkilön ja organisaation attribuutit ovat sateliitti-tauluissa yhdistettynä hub-tauluihin (Linstedt & Olschimke, 2015).

Käsitemallinnuksen jälkeen uudet käsitteet tuodaan tietoaltaalta raakatietovarastoon, jossa hyödynnetään Data Vault 2.0 -periaatteita. Raakatietovaraston tarkoitus on säilyttää

tieto ennallaan yhdistäen kuitenkin eri lähtöjärjestelmien samat käsitteet yhtenäisiin rakenteisiin (Linstedt & Olschimke, 2015). Tässä vaiheessa pyritään integroimaan tiedot käsitemallin tasolla huomioiden kuinka hyvin lähtöjärjestelmän tietorakenteet tukevat käsitemallia. Tämän avulla uusia tietojärjestelmiä voidaan lisätä tietovarastoon ilman uudelleen toteutettavaa integroimista tai mallintamistatarvetta (Linstedt & Olschimke, 2015). Data Vault 2.0 -tietomalli muodostuu kolmesta erilaisesta tietorakenteesta: hub-, satelliitti-, sekä linkki-taulu (Linstedt & Olschimke, 2015). Hub-taulu muodostetaan jokaiselle käsitteelle ja näihin tauluihin kerätään kaikki tietoaltaan tietyn käsitteen yksilöivät tunnisteet. Esimerkiksi sosiaali- ja terveydenhuollon asiakas on yksi hub-taulu. Tähän tauluun lisätään silloin jokaisen asiakkaan yksilöivä tunniste. Yksilöivä tunniste on jotain minkä avulla reaalimaailmassa käsite tunnistetaan (Linstedt & Olschimke, 2015). Henkilöt yksilöidään yleensä henkilötunnuksen avulla, koska nimi ei suoranaisesti takaa yksilöintiä. Toinen rakenne tietomallissa on satelliitti-taulu. Tämä taulu linkittyy hub-tauluun yksilöivän avaimen avulla. Satelliitti-taulun tarkoituksena on säilöä kaikki attribuutit yhdelle henkilölle. Esimerkiksi sosiaali- ja terveydenhuollon asiakkaan syntymäaika, kotikunta ja muut tiedot voidaan varastoida satelliitti-tauluun ja linkittää hub-taulussa olevaan henkilöön. Tällöin eri lähtöjärjestelmistä voidaan lisätä uusi satelliitti-taulu, joka kertoo samasta asiakkaasta uusia tietoja. Viimeisenä tietorakenteena on linkki-taulu, jonka avulla kaksi tai useampi käsitettä muodostavat käsitteiden välisen suhteen. Tästä esimerkkinä asiakkaan ja organisaation välinen hoitokäynti. Data Vault 2.0-periaatteen mukainen tietomalli on esitetty kuvassa (3.4).

Raakatietovaraston avulla pystytään muodostamaan liiketoiminnan näkökulmasta lisätauluja, jotka tuovat lisäarvoa tietovarastosta saatavalle tiedolle. Näitä lisätauluja muodostetaan organisaation tarpeisiin kuten analytiikkaan tai raportointiin. Näitä käsitteitä ovat esimerkiksi eri lähtöjärjestelmien satelliitti-taulujen yhdistäminen yhtenäiseksi tietokantatauluksi. Tätä uutta rakennetta pystytään hyödyntämään, kun tietovarastoa halutaan käyttää koko organisaation laajuisesti. Näiden lisätaulujen muodostaminen on toisinaan haastavaa, sillä useat eri sosiaali- ja terveydenhuollon tietojärjestelmät sisällyttävät erilaista tietoa. Lisäksi jotkin järjestelmät eivät tuota yhdistämisessä vaadittavia tietokenttiä. Tästä syystä tietovarastolla olevasta tiedosta muodostetut lisätaulut voivat välillä jäädä vaillinaisiksi.

3.2.4 Hyödyntämiskerros

Tietovaraston hyödyntäminen voi jossain tapauksissa olla resurssien kannalta raskasta. Hyödyntämiskerroksessa pyritään tuottamaan aputauluja, joiden avulla tietovarastoa

hyödynnettäessä laskemiseen varattujen resurssien tarve vähenee. Tässä luvussa käsitellään raakatietovaraston taulujen ja organisaation toiminnan lisätaulujen hyödyntämistä. Lisäksi käsitellään hyödyntämiskerroksessa usein käytetyt tähti-mallin mukaiset fakta- ja dimensiotaulut. Nämä käsitteet ovat oleellisia sillä, tutkielman kokeellisessa osuudessa datajoukko kerätään hyödyntämiskerroksen fakta- ja dimensiotauluista.

Kuva 3.5: Esimerkki tähtimallin arkkitehtuurista. Keskellä olevaan faktatauluun linkittyy kaksi dimensiotaulua.

Hyödyntämiskerroksen tarkoitus on muodostaa aputauluja, joita voidaan hyödyntää tietovaraston ulkopuolisissa toiminnoissa, kuten analytiikassa sekä raportoinnissa. Tietoa ei suoranaisesti muokata, mutta se muotoillaan kohdejärjestelmissä oleviin muotoihin (Linstedt & Olschimke, 2015). Aputauluissa voidaan tehdä esimerkiksi raskaita päättelyitä, joiden tekeminen analytiikan tai raportoinnin sovelluksessa hidastaisi prosessia.

Yksi yleisesti raportoinnissa ja muissa toimissa käytetty malli on tähti-malli (Star schema), joka koostuu fakta- ja dimensiotauluista (Linstedt & Olschimke, 2015). Faktataululla tarkoitetaan taulua, johon kerätään tyypillisesti tapahtumatyyppisiä käsitteitä mahdollisimman tarkalla tasolla. Esimerkiksi erilaiset tapahtuma- tai kustannuskäsitteet on mahdollista muotoilla faktatauluksi. Dimensiotaulut sisältävät faktatauluihin linkittyviä avainkenttiä, joiden avulla faktataulun riviltä voidaan hakea lisätietoa dimensiotauluista esimerkiksi referenssikoodin selkokielisiä selitteitä tai ajan esitystapaa helpottavia kenttiä. Kuvassa (3.5) on esitetty tähti-mallin mukaisesta arkkitehtuurista.

3.3 Segmenttimuutoksen ennustaminen

Sosiaali- ja terveydenhuollon asiakkaiden segmenttimuutoksen ennustamista on tutkittu jossain määrin (Chechulin ym., 2014; Morid ym., 2020; Ng ym., 2020). Monet tutkimukset pohjautuvat terveydenhuollon potilaiden erilaisten segmenttien muutoksien

ennustamiseen. Näitä segmenttejä ovat esimerkiksi suurimpien kustannusten potilaat ja pitkäaikaiset suuren tarpeen potilaat. Monien tutkimuksien tavoitteena on luoda ennustemalleja, joiden avulla ennustettavaa asiaa voitaisiin ennaltaehkäistä. Sosiaali- ja terveydenhuollon asiakkaan riskien tunnistamisen avulla voidaan tuottaa tehokasta ja ennaltaehkäisevää terveydenhuoltoa. Lisäksi useat tutkimukset ovat painottuneet terveydenhuollon asiakkuuksiin (Chechulin ym., 2014; Morid ym., 2020; Ng ym., 2020). Esimerkiksi neuroverkkoja on hyödynnetty ennustemalleissa, joiden tarkoitus on ennustaa yksittäisen sairauden riskiä potilaalla (Shanker, 1996). Nämä tutkimukset ovat lähellä tutkielman aihetta, mutta tässä tutkielmassa painotetaan enemmän kokonaisvaltaista hoitoa tukevien segmenttimuutoksien ennustamista. Aiemmissa tutkimuksissa on käytetty erilaisia menetelmiä tilastotieteestä ja koneoppimisesta. Tutkimuksissa käytetyt tietojärjestelmät ja niistä kerätyt datajoukot vaihtelevat paljon. Tästä syystä malleja ei voi kopioida suoraan tämän tutkielman kokeellisen osuuteen.

Morid ym. (2020) tutkimuksessa potilaan vakuutuskorvaushakemuksista muodostettiin datajoukko, jonka avulla koulutettiin konvoluutioarkkitehtuuriin perustuvia neuroverkkoja. Nämä ennustemallit pyrkivät ennustamaan potilaan tulevaa kustannussegmenttiä. Koulutusdata muodostettiin terveys- ja lääkevakuutuskorvaushakemuksista vuosilta 2013 - 2016. Datajoukon päivätasolla tallennetut tapahtumat muunnettiin kuukausitason ryhmittelyiksi. Yhtä kuukautta kohti asiakkaan tietojoukko koostui 608 parametrista. Näistä parametreista kaksi oli kuukauden terveys- ja apteekkikuluja, 180 toimenpidemerkintää, 83 diagnooseja, 336 lääkereseptejä ja 7 käyntisyy-koodia. Koulutusdata muodostui sosiaali- ja terveydenhuollon asiakkaan 24 kuukauden jaksosta. Ennustettava kuukausi valittiin 12 kuukauden päähän viimeisestä koulutusdatan kuukaudesta. Jokainen datajoukon potilas muodosti yhden 24 kertaa 608 matriisin. Tutkimuksessa moniulotteista koulutusdataa käytettiin vastaavalla tavalla kuin visuaalista dataa konvoluutioverkoissa. Tämän avulla konvoluutioverkon oli mahdollista poimia moniulotteisia tapahtumia, joiden avulla voitiin ennustaa kustannussegmenttiä.

Morid ym. (2020) tutkimuksessa datajoukon koko oli 91 tuhatta potilasta, 6.3 miljoonaa terveysvakuutuskorvausta ja 1.2 miljoonaa lääkevakuutushakemusta. Tästä 70 prosenttia käytettiin neuroverkon kouluttamiseen ja validointiin. Loput 30 prosenttia käytettiin mallin testaukseen. Tutkimuksessa konvoluutioneuroverkot toteutettiin TensorFlow:n Python -kirjaston avulla. Tutkijoiden itse kehittämä neuroverkko käytti $LReLU$-aktivointifunktiota, adam-optimointialgoritmia, sekä keskimääräistä neliövirhettä (Mean square error, MSE) virhefunktiona. Tutkimuksessa rakennettu konvoluutioneuroverkko koostui kolmesta konvoluutiokerroksesta (Convolution layer), sekä jokaisen konvoluutiokerroksen omasta yhdistämiskerroksesta (Pooling layer). Lopuksi viimeinen yhdistämiskerros liitettiin klassiseen neuroverkkoon.

Morid ym. (2020) tutkimuksessa ennustettavat potilaat luokiteltiin viiteen ryhmään kustannusten perusteella. Ryhmä 1 oli halvimpien kustannusten luokka ja ryhmä 5 kalleimpien kustannusten luokka. Potilaan kustannusten määrä laskettiin vakuutuskorvauksien summista viimeisimmän vuoden ajalta. Tutkimuksen tuloksina havaittiin, että tutkijoiden itse kehittämä konvoluutioarkkitehtuuri onnistui saamaan 94.53 prosentin ennustetarkkuuden. Lisäksi ennustemalli onnistui päihittämään ennustetarkkuudessa vertailussa olleet yleisesti tunnetut konvoluutiomallit ResNet-34 (He ym., 2015) ja AlexNet (Krizhevsky ym., 2012).

Chechulin ym. (2014) tutkimuksessa kehitettiin ennustemalli, jonka avulla voidaan ennustaa terveydenhuollon potilaan riskiä joutua suurimpien kustannuksien potilassegmenttiin. Suurimpien kustannusten segmentiksi määriteltiin ne potilaat, joiden terveydenhuollon kustannukset olivat suurimman viiden prosentin joukossa kaikista potilaista. Ennustemallit koulutettiin ennustamaan tähän segmenttiin joutumisen riskiä vuoden päästä viimeisestä datapisteestä. Datajoukko kerättiin aikaväliltä 2008 - 2011. Segmentin ennustamiseen käytettiin kolmen vuoden ajanjaksoa ja ennuste tehtiin viimeisestä datapisteestä vuoden päähän. Datajoukosta karsittiin ne potilaat, jotka olivat kuolleet alle vuoden päästä viimeisestä datapisteestä, koska he eivät voineet enää kuulua kalleimpaan potilasryhmään. Lisäksi datajoukosta poistettiin alle viiden vuoden ikäiset potilaat. Koulutusdataan kerättiin tietoa potilaiden tapahtumista terveydenhuollon palveluissa. Nämä tapahtumat sisälsivät lääkärin tapaamisen, akuutin sairaalahoidon, päiväkirurgian, ensiavun, dialyysin, syöpähoidon, poliklinikkakäynnit, kuntoutuksen, mielenterveyspalvelut, pitkäaikaishoidon ja kotihoidon. Potilaiden aiheuttamat kustannukset pääteltiin laskemalla näistä erilaisista palveluista keskituntihinta ja kertomalla se potilaan palvelunkäyttöjen määrällä. Datajoukkoa muodostaessa potilaista kerättiin myös perustiedot kuten ikä, sukupuoli, sosioekonominen status ja koodistoja kuten ICD-10. Lisäksi käyntien ja sairaalahoitojaksojen määrät laskettiin summatiedoiksi datajoukkoon. Kerätyssä datajoukossa potilaiden määrä oli 10 300 856. Datajoukossa käytettyjen parametrien määrä oli 69. Ennustemallina käytettiin logistista regressiota. Tilastollinen menetelmä antoi tarkkuudeksi 94.2%, herkkyydeksi 42.2% ja johdonmukaisuudeksi 97%. Tuloksissa huomattiin iän olevan hyvin merkittävä tekijä ennusteen tarkkuuden osalta.

Ng ym. (2020) tutkimuksessa kehitettiin ennustemalli, jonka avulla voidaan ennustaa asiakkaan todennäköisyyttä joutua terveydenhuollon pysyvän korkean tarpeen asiakkaaksi (Persistent high utilizers, PHUs), ohimevän korkean tarpeen asiakkaaksi (Transient high utilizers, THUs) tai matalan tarpeen asiakkaaksi (Non-high utilizer, non-HUs). Korkean tarpeen asiakkaiksi (High utilizers, HUs) määriteltiin ne potilaat, joiden tervey-

denhoidon aiheuttamat kulut olivat vuoden aikana kymmenen korkeimman prosentin joukossa tai vähintään 8150 Amerikan dollaria ($). Pysyvän korkean tarpeen asiakkaat olivat niitä henkilöitä, joiden korkea tarve pysyi yhtäjaksoisena vähintään kolme vuotta. Ohimenevän korkean tarpeen asiakas taas määriteltiin siten, että potilas oli vähintään yhden vuoden, mutta enintään kolme peräkkäistä vuotta korkean tarpeen asiakkaana. Datajoukko koostui 152 497 terveydenhuollon asiakkaasta. Datajoukko muodostettiin terveydenhuollon laskuista ja siihen liitetyistä tiedoista. Näitä tietoja olivat määrätyt lääkitykset, laboratorio- ja radiologiatutkimukset, leikkaukset, tarkennetut tutkimukset, lääkärin tapaamiset, terapiat ja muut luokittelemattomat hoitotapahtumat. Ennustemallien kouluttamiseen otettiin 70% datajoukosta. Loput 30% käytettiin mallien testaukseen. Tutkimuksessa vertailtiin kolmea erilaista algoritmia. Nämä algoritmit olivat rangaistuspohjainen regressio (Penalized regression), tukivektorikone (Support vector machine) ja satunnaismetsään (Random forest) perustuva XGBoost-malli (Extreme gradient boosting) (Chen & Guestrin, 2016). Tutkimuksessa parhaiten suoriutunut ennustaja oli XGBoost-malli. Malli sai testidatalla AUC-mittarin arvoiksi 79.8% ja 95%, jossa luottamusväli (Confidence interval, CI) oli 78.8% ja 80.8% välillä.

Kaikissa edellä mainituissa ennusteiden tulokset osoittautuivat vähintäänkin lupaaviksi. Ennustemallit pystyivät ennustamaan sosiaali- ja terveydenhuollon asiakkaiden segmenttimuutoksia parhaimmillaan 94.53 prosentin tarkkuudella. Tutkimuksissa nousseita ennustemalleja olivat logistinen regressio, XGBoost-malli ja konvoluutioneuroverkko. Tutkimukset perustuivat vahvasti terveydenhuollon tuottamaan dataan esimerkiksi vakuutuskorvauksista kerättynä datana. Sosiaalihuollon datalähteitä ei hyödynnetty kuin Chechulin ym. (2014) toteuttamassa tutkimuksessa, jossa sosioekonominen status oli yksi mallin parametreista. Tutkimusten ennustemallien parametrimäärät vaihtelivat 69 ja 608 välillä. On huomattavaa, että neuroverkoilla tehdyissä malleissa oli suurin parametrien määrä. Kaikissa tutkimuksissa käytetyt datajoukot olivat laajoja sosiaali- ja terveydenhuollon asiakasmäärien osalta. Käytetyissä datajoukoissa henkilöiden määrä oli monista kymmenistä tuhansista moneen miljoonaan. Sosiaali- ja terveydenhuollon asiakkaan ennustettavia segmenttejä olivat kustannusluokat ja korkean palvelutarpeen luokat. Nämä ovat lähellä tutkielman kokeellisessa osuudessa ennustettavaa Pärjääjämallin mukaista verkosto-segmenttiä, koska kaikissa segmenteissä kustannukset ja hoidon tarve ovat pääsääntöisesti hyvin korkeita.

Morid ym. (2020) tutkimuksen konvoluutioneuroverkko vaikutti vahvasti tämän tutkielman kokeellisen osuuden toteutukseen. Tutkimuksen lupaavien tulosten perusteella konvoluutioneuroverkko valittiin yhdeksi arkkitehtuuriksi. Tutkimuksessa käytetty $LReLU$-aktivointifunktion soveltuvuutta testataan tämän tutkielman kokeellisen osuudessa. Lisäksi kuukausitason datapisteet vaikuttivat tehokkaalta tavalta säilyttää osa

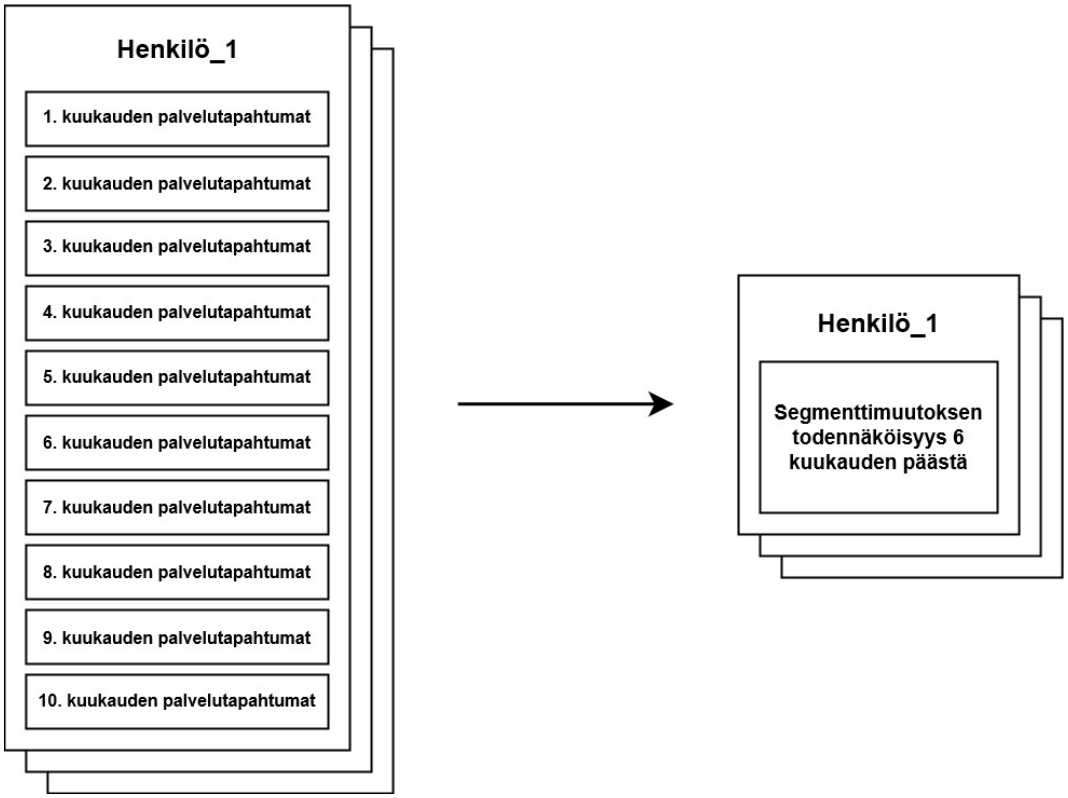
käyntitietojen aikasarjallisuudesta. Yleisesti tutkimuksissa korostuivat sosiaali- ja terveydenhuollon asiakkaiden ikä sekä annetut diagnoosit. Molemmat ovat tutkimusten perusteella vahvoja indikaattoreita kustannusten ja hoidon tarpeen ennustamiseen. Tutkielman kokeellisessa osuudessa datajoukkoon kerätään vastaavia arvoja, sillä niiden ennustevoimasta on selkeitä näyttöjä tutkimusten perusteella.

4. Kokeellisen osuuden toteutus

Tämän tutkielman kokeellisessa osuudessa tutkitaan, kuinka suuri ennustetarkkuus voidaan saavuttaa neuroverkoilla toteutetulla ennustemallilla, joka pyrkii ennustamaan sosiaali- ja terveydenhuollon asiakkaan Pärjääjä-mallin mukaista segmenttimuutosta kuuden kuukauden päähän. Kuvassa (4.1) on esitetty visualisaatio tutkielman kokeellisessa osuudessa toteutettujen ennustemallien koulutuksessa käytössä olevasta datajoukosta ja ennustamisen kohteesta. Lisäksi tutkitaan erilaisten neuroverkkoarkkitehtuurien vaikutusta ennustemallien ennustetarkkuuteen. Koulutettavat mallit koulutetaan ennustamaan, muuttuuko sosiaali- ja terveydenhuollon asiakkaan Pärjääjä-mallin mukainen asiakassegmentti verkosto-asiakkuudeksi kuuden kuukauden päästä viimeisestä havainnosta.

Kokeellisessa osuudessa vertaillaan klassista neuroverkkoarkkitehtuuria, LSTM arkkitehtuuria, sekä konvoluutioneuroverkkoja. Lisäksi vertaillaan virhefunktion, aktivointifunktion ja neuroverkon syvyyden vaikutuksia tuloksiin. Neuroverkkojen kouluttamista varten oleva data kerätään Keski-Uudenmaan sote -kuntayhtymän tietovarastosta. Tietoja hankitaan yli 70 vuotta täyttäneistä asiakkaista ja data on aikaväliltä 1.1.2018 - 31.7.2021. Ikäryhmä valittiin, koska suuri osa yhteistyö-asiakkuuksista alkaa vanhemmalla iällä. Aikajakso valittiin, koska dynaamisesti segmentoituja palveluita oli eniten tällä ajanjaksolla. Tämä johtuu historiassa tuotettujen palveluiden dynaamisen segmentoinnin haastavuudesta. Aikaväli on kuitenkin varsin riittävä, vaikka historiallista dataa karsiutuukin jossain määrin pois.

Luvussa 4.1 käydään läpi kuinka dynaaminen asiakassegmentointi toteutetaan, jotta koulutusdatan tulos-arvoja voidaan kerätä. Luvussa 4.2 puolestaan käsitellään, kuinka aineisto hankitaan ja muodostetaan tietovarastosta. Lisäksi tiedon keräyksen rajauksia kuvataan hakuvaiheessa. Luku 4.3 käsittelee koulutusdatan muodostamista eri arkkitehtuurien syötteitä varten. Lisäksi luvussa kerrotaan kuinka koulutusdata esikäsitellään neuroverkkojen koulutusta varten. Luvussa 4.4 käydään läpi valittujen neuroverkkomallien kouluttamista, parametrien valintaa, sekä muita neuroverkon koulutukseen liittyviä valintoja. Luvussa 4.5 esitetään neuroverkkojen tuloksia suorituskykymittarien avulla.



Kuva 4.1: Visualisaatio kerättävästä tietojoukosta ja ennustemallin vasteesta. Kerättävässä datajoukossa on sosiaali- ja terveydenhuollon asiakkaan kymmenen kuukauden niputetut palvelutapahtumien tiedot. Kerätyn datajoukon avulla ennustetaan asiakkaan segmenttimuutoksen riskiä kuuden kuukauden päästä. Jokainen kuukausi sisältää tietovarastolta summatut koodistomerkinnät, jotka ovat esiintyneet kyseisenä kuukautena.

4.1 Dynaamisen segmentoinnin toteutus

Sosiaali- ja terveydenhuollon asiakkaiden kuukausitason segmenttejä muodostetaan dynaamisesti Keusoten tietovarastossa. Tietovarastoon luodaan apukyselyitä, joiden avulla luodaan asiakkaiden kuukausitason segmenttien taulu. Tämä on pohjana kaikelle koulutukselle, koska ennustemallit koulutetaan ennustamaan, muuttuuko Pärjääjä-mallin mukainen segmentti muista segmenteistä ryhmään 4, joka on verkosto-asiakkuus.

F_palvelukaytto-faktatauluun on muodostettu kokeellisen osuuden kannalta tarpeelliset tiedot; se sisältää potilaan yksilöivän avaimen, aikaleiman, sekä sarakkeen dim_asiakkuusryhma. dim_asiakkuusryhma-sarake yhdistää jokaisen palvelutapahtuman johonkin Pärjääjä-mallin mukaisesta segmentistä tai antaa palvelutapahtumalle

arvon, joka viittaa luokittelemattomaan palvelutapahtumaan. Tämä kenttä muodostetaan hyödyntämällä referenssitaulua, joka on organisaation substanssiosaajien kanssa muodostettu taulu, jossa palvelut luokitellaan Pärjääjä-mallin mukaisiin segmentteihin. Pärjääjä-segmenttiin kuuluvia palveluita ovat esimerkiksi pienen tuen palvelut ja itse hankittavat tukipalvelut. Verkosto-segmenttiin kuuluvat palvelut ovat raskaita palveluita kuten kotihoidon yökäynnit, laitoskuntoutus ja muut kalliit tukihoitomuodot. Osa palveluista voivat myös olla vaikeammin määriteltävissä. Tällöin palvelun segmentointi saattaa tapahtua esimerkiksi asioinnin toistuvuuden perusteella. Jos asiakas tarvitsee samaa palvelua useasti, muuttuu segmentointi. Harvoin käytettynä palvelu voidaan päätellä toiseen segmenttiin.

Seuraavaksi luodaan F_palvelukaytto-faktataulusta uusi aputaulu, jossa segmentoidaan potilaiden kuukaudet Pärjääjä-mallin mukaisesti. Kuukausitason segmentointi on tehokas tapa käsitellä segmenttimuutoksia, koska usein palveluita ei käytetä päivittäin. Sosiaali- ja terveydenhuollon asiakas voi olla yleisellä tasolla raskaita palveluita tarvitseva henkilö, jolle on tehty jo valmiiksi esimerkiksi ruokahuollon tilaus kuukaudeksi eteenpäin. Tämä kuitenkin näkyisi useana päivänä matalan tason asiakkaan jos rakenne olisi päivätasolla. Jotta tämä voidaan välttää, tehdään segmentointi kuukausitasolla. Näin sosiaali- ja terveydenhuollon asiakkaan kehitystä voidaan seurata tehokkaasti Pärjääjä-mallin mukaisessa segmentoinnissa.

Uuden aputaulun tarkoitus on yhdistää jokainen asiakkaan palvelumerkintä vuoden samalle kuukaudelle. Koska faktataulu sisältää päivämäärän yleisenä muotona, voidaan hyödyntää D_kalenteri-dimensiotaulua, jonka avulla se yhdistetään jokaiselle asiakkaalle vuosi_kuukausi-sarakkeeseen. Tämän jälkeen samalle asiakkaalle kuuluvat ja saman vuosi_kuukausi arvon rivit ryhmitetään yhteen summaamalla dim_asiakkuusryhmakenttien arvot omille sarakkeilleen S1, S2, S3 ja S4. Näistä sarakkeista voidaan tehdä kuukausitason segmentointi. Asiakkaan kuukausi kuuluu segmenttiin 4 (Verkosto-asiakkuus), jos se sisältää yhtään palveluita jotka kuuluvat luokkaan 4, tai jos asiakkaalle on merkitty palveluja sekä luokasta 2 (Yhteistyö-asiakkuus) että 3 (Yhteisö-asiakkuus). Asiakkaan kuukausi segmentoidaan luokkaan 3 (Yhteisö-asiakkuus), jos hänelle on merkitty palveluja ryhmästä 3 (Yhteisö-asiakkuus). Asiakkaan kuukausi segmentoidaan luokkaan 2 tai 1 vastaavalla tavalla kuin segmentti 3. Lopuksi kuukausi segmentoidaan ryhmään 0, joka vastaa luokittelematonta kuukautta. Koska dynaaminen segmentointi ei ole täysin kattava kaikkien palveluiden osalta, sosiaali- ja terveydenhuollon laajuuden takia, ei osa kuukausista päädy dynaamisesti segmentoiduiksi. Tämä kuukausitason segmentointi-logiikka on esitetty kuvassa (4.2).

```
select

ASIAKASTUNNISTE,
VUOSI_KUUKAUSI,
sum(case DIM_ASIAKKUUSRYHMA when '1' then 1 else 0 end) as S1,
sum(case DIM_ASIAKKUUSRYHMA when '2' then 1 else 0 end) as S2,
sum(case DIM_ASIAKKUUSRYHMA when '3' then 1 else 0 end) as S3,
sum(case DIM_ASIAKKUUSRYHMA when '4' then 1 else 0 end) as S4,
sum(case DIM_ASIAKKUUSRYHMA when '-999996' then 1 else 0 end) as S0,
(case
    when S4 > 0 or (S2 > 0 and S3 > 0) then 4
    when S3 > 0 then 3
    when S2 > 0 then 2
    when S1 > 0 then 1
    else 0
end) as SEGMENTTI,

from
"F_PALVELUKAYTTO"

left join
"D_KALENTERI"
on "F_PALVELUKAYTTO".DIM_KALENTERI = "D_KALENTERI".DIM_KALENTERI

group by
ASIAKASTUNNISTE,
VUOSI_KUUKAUSI
```

Kuva 4.2: SQL-kysely, jonka avulla F_palvelukaytto-faktataulusta summataan yhteen kaikki asiakkaan kuukauden palvelutapahtumat, jotka kuuluvat Pärjääjä-mallin mukaiseen segmenttiin S1 (Omatoimi-asiakkuus), S2 (Yhteistyö-asiakkuus), S3 (Yhteisöasiakkuus) ja S4 (Verkosto-asiakkuus). S0 merkitsee palvelutapahtumia, joita ei ole voitu yhdistää mihinkään Pärjääjä-mallin segmenteistä. Näistä kuukausitason summista päätellään lopuksi asiakkaan kuukauden segmentti.

4.2 Aineiston keräys

Dynaamisen asiakassegmentoinnin jälkeen loput kerättävistä tiedoista kerätään Keski-Uudenmaan sote -kuntayhtymän tietovarastosta samalla kuukausitason rakenteella kuin dynaaminen segmentointi. Nämä tiedot muodostavat kuvassa (4.1) esitetyn rakenteen yhden kuukauden tietokentät. Keräykseen haetaan sosiaali- ja terveydenhuollon asiakkaan perustiedot iästä, kunnasta ja sukupuolesta. Lisäksi kirjauksista poimitaan THL:n mukaisia koodistoja. Keräykseen otetaan mukaan tautiluokitus ICD-10, perusterveydenhuollon luokitus ICPC-2 ja asiointitapakoodisto.

Sosiaali- ja terveydenhuollon asiakkaan kuukausitason segmenttien kaikista merkinnöistä muodostetaan summia, jotka ryhmitellään asiakastunnisteen ja yksilöllisen kuukauden avulla. Yksilöllinen kuukausi on tässä tapauksessa vuosi_kuukausi-muuttuja. Kuten dynaamisessa segmentoinnissa, käytetään F_palvelutapahtuma-faktataulua, johon yhdistetään kalenteri-dimensiotaulu. Tällöin voidaan yhdistää kaikki tapahtumat, jotka ovat merkitty samalla vuosi_kuukausi arvolla. Tämän jälkeen kerättävät arvot ryhmitetään summiksi. Näin kuukausitason tiedot saadaan omille sarakkeilleen. Sarakkeita muodostuu perustiedoista, dynaamisesta segmentoinnista ja koodistoista yhteensä 1606. Lisäksi keräyksessä tehdään suodatus, jonka avulla kerätään vain yli 70-vuotiaiden palvelutapahtumat. Suodatus tehdään, koska Pärjääjä-mallin mukainen segmentti on harvinaisempi nuoremmilla henkilöillä. Kuvassa (4.3) on esitetty lyhennetty SQL-kysely, jonka avulla kerätään ryhmitellyt palvelutapahtumamerkinnät. Datan keräyksessä pyritään mahdollisimman tasaiseen jakaumaan, koska epätasaisuus tuloksissa on todettu neuroverkkojen koulutusta vaikeuttavaksi ominaisuudeksi (Li ym., 2020). Lisäksi suodatetaan myös palvelutapahtumat ennen vuotta 2018, koska dynaamisen segmentoinnin merkintöjä ei ollut muina vuosina tarpeeksi keräystä varten.

Seuraavaksi muodostetaan aputauluja, jotka keräävät tietoja mitkä kuukausi-sarjat tulee kerätä asiakkaalta. Lisäksi yhdeltä asiakkaalta saadaan useita kuukausi-sarjoja liikuttamalla viimeisimmän palvelumerkinnän kuukautta taaksepäin. Tällä tavoin kerätään useita tapahtumia niistä henkilöistä, jotka ovat joutumassa verkosto-asiakkuuden segmenttimuutokseen. Tämä auttaa tasapainottamaan epätasaista segmenttimuutosten jakaumaa. Aputaulujen laskentaskripti on esitetty kuvassa (4.4).

Seuraavaksi luodaan toinen aputaulu, jonka avulla voidaan päätellä, onko ennustettavaa segmenttimuutosta tapahtunut. Taulussa (4.4) esitetyllä skriptillä luodaan taulu, johon yhdistetään alkuperäinen dynaamisen segmentoinnin taulu. Päättelemällä kymmenennen kuukauden, sekä ennustettavan kuukauden asiakassegmentit asiakaskohtaisesti, voidaan määritellä, tapahtuiko muutosta vai ei. Kaikki ne kuukausisarjat, jossa kymmenes

kuukausi on segmentoitu, eikä se ole ryhmässä 4, mutta ennustettavassa kuukaudessa segmentoitu luokka onkin ryhmässä 4.

```
select

ASIAKASTUNNISTE,
VUOSI_KUUKAUSI,
sum(PALVELU_LKM) as PALVELU_LKM,
avg(DIM_IKA) as IKA,
mode(DIM_SUKUPUOLI) as SUKUPUOLI,
mode(DIM_KUNTA) as KUNTA,

sum(case DIM_ASIOINTITAPA when 'R10' then 1 else 0 end) as R10,
...
sum(case DIM_ASIOINTITAPA when 'R90' then 1 else 0 end) as R90,

sum(case T1_ICD10_KOODI when 'A00-A09' then 1 else 0 end) as ICD10_A00_A09,
...
sum(case T1_ICD10_KOODI when 'ZA0-ZB9' then 1 else 0 end) AS ICD10_ZA0_ZB9,

sum(case T1_ICPC2_KOODI when 'A02' then 1 else 0 end) as ICPC2_A02,
...
sum(case T1_ICPC2_KOODI when 'W58' then 1 else 0 end) as ICPC2_W58

from "F_PALVELUKAYTTO"

left join
"D_KALENTERI"
on "F_PALVELUKAYTTO".DIM_KALENTERI = "D_KALENTERI".DIM_KALENTERI

left join
"D_ICD10"
on "F_PALVELUKAYTTO".DIM_ICD10 = "D_ICD10".DIM_ICD10

left join
"D_ICPC2"
on "F_PALVELUKAYTTO".DIM_ICPC2 = "D_ICPC2".DIM_ICPC2

where "D_KALENTERI".VUOSI_NUMERO > 2017 and DIM_IKA > 70

group by
ASIAKASTUNNISTE,
VUOSI_KUUKAUSI
```

Kuva 4.3: Lyhennetty SQL-kysely, joka kerää asiakkaan kuukauden ryhmittymäksi näkymän perustiedoista, tautiluokituksesta ICD-10, perusterveydenhuollon luokituksesta ICPC-2 sekä asiointitavan kirjauksista. Koodistojen kirjaukset summataan omille sarakkeilleen hierarkiatason mukaisesti jaoteltuna. Skripti suodattaa ennen vuotta 2018 tapahtuneet kirjaukset, sekä alle 70-vuotiaiden asiakkaiden kirjaukset.

Muut tapaukset ovat joko ei luokiteltuja muutoksia, muuttumattomia segmenttejä tai erilaisia segmenttimuutoksia, kuten ryhmästä 2 muutos ryhmään 3. Tämä päättely on esitetty kuvan (4.5) SQL-kyselyssä. Tämän SQL-kyselyn avulla voidaan myös suodattaa osa negatiivisista arvoista eli ne, jossa segmenttimuutosta ei tapahtunut. Tällöin datan tasapaino paranee entisestään.

```
select

ASIAKASTUNNISTE,
to_varchar(dateadd(month, -15, max(PAIVAMAARA))::date, 'YYYY-MM') as KK_1,
to_varchar(dateadd(month, -14, max(PAIVAMAARA))::date, 'YYYY-MM') as KK_2,
to_varchar(dateadd(month, -13, max(PAIVAMAARA))::date, 'YYYY-MM') as KK_3,
to_varchar(dateadd(month, -12, max(PAIVAMAARA))::date, 'YYYY-MM') as KK_4,
to_varchar(dateadd(month, -11, max(PAIVAMAARA))::date, 'YYYY-MM') as KK_5,
to_varchar(dateadd(month, -10, max(PAIVAMAARA))::date, 'YYYY-MM') as KK_6,
to_varchar(dateadd(month, -9, max(PAIVAMAARA))::date, 'YYYY-MM') as KK_7,
to_varchar(dateadd(month, -8, max(PAIVAMAARA))::date, 'YYYY-MM') as KK_8,
to_varchar(dateadd(month, -7, max(PAIVAMAARA))::date, 'YYYY-MM') as KK_9,
to_varchar(dateadd(month, -6, max(PAIVAMAARA))::date, 'YYYY-MM') as KK_10,
to_varchar(dateadd(month, 0, max(PAIVAMAARA))::date, 'YYYY-MM') as KK_ENNUSTETTAVA

from "UTIL_VUOSI_KUUKAUSI_ASIAKAS"

group by ASIAKASTUNNISTE
```

Kuva 4.4: SQL-kysely, jonka avulla luodaan aputauluja, joiden avulla pystytään hakemaan viimeisin kuukausi, jossa asiakkaalle on kirjattu palvelutapahtuma. Lisäksi palvelutapahtumasta lasketaan kymmenen vuosikuukauden arvot siten, että ne ovat kuusi kuukautta ennen viimeisintä tapahtumaa.

```
select
ASIAKASTUNNISTE,
ifnull(T1.SEGMENTTI,1) as KK_10_SEGMENTTI,
KK_ENNUSTETTAVA,
T2.SEGMENTTI as KK_ENNUSTETTAVA_SEGMENTTI,
case when T2.SEGMENTTI = 4 and T1.SEGMENTTI != T2.SEGMENTTI then 1 else 0 end as
    VAIHTUIKO_SEGMENTTI,
KK_1,
KK_2,
KK_3,
KK_4,
KK_5,
KK_6,
KK_7,
KK_8,
KK_9,
KK_10,
1 as BATCH

from "LOAD_ML_UTIL_ASIAKAS"

left join
LOAD_ML_UTIL_VUOSI_KUUKAUSI_ASIAKAS as T1
on LOAD_ML_UTIL_ASIAKAS.KK_10 = T1.VUOSI_KUUKAUSI
and LOAD_ML_UTIL_ASIAKAS.ASIAKASTUNNISTE = T1.ASIAKASTUNNISTE

left join
LOAD_ML_UTIL_VUOSI_KUUKAUSI_ASIAKAS as T2
on LOAD_ML_UTIL_ASIAKAS.KK_ENNUSTETTAVA = T2.VUOSI_KUUKAUSI
and LOAD_ML_UTIL_ASIAKAS.ASIAKASTUNNISTE = T2.ASIAKASTUNNISTE

where T2.SEGMENTTI > 0
and T1.SEGMENTTI > 0
```

Kuva 4.5: SQL-kysely, jonka avulla päätellään dynaamisen segmentoinnin muutos sosiaali- ja terveydenhuollon asiakkaan kymmenennen ja kuudennentoista kuukauden segmenteistä.

Kuvassa (4.5) esitetty SQL-kysely tuottaa koulutusdatan osalta oleellisen tuloksen, eli binäärisen jaottelun siitä, vaihtuuko sosiaali- ja terveydenhuollon asiakkaan potilas-

segmentti ryhmään 4 seuraavan kuuden kuukauden aikana viimeisestä merkinnästä. Tarvittavat tiedot kerätään ja niiden avulla luodaan keräyksen valmis taulu. Tätä taulua käytetään koulutusdatassa. Tämä on esitetty kuvan (4.6) SQL-kyselyssä.

```
select

BATCH,
ASIAKASTUNNISTE,
VAIHTUIKO_SEGMENTTI as TARGET_VALUE

from "PRE_DATASET"
```

Kuva 4.6: SQL-kysely, joka muodostaa lajitellun keräyksen tulokset, jotka ovat koulutusdatan y-arvoja.

Kuvassa (4.5) esitetyn SQL-kyselyn avulla tuotetaan uusi aputaulu, joka toimii pohjana siihen yhdistettäville tiedoille asiakkaan kuukausitason merkinnöissä. Aputaulun kymmenen kuukauden sarakkeet muunnetaan riveiksi. Tämä on esitetty kuvan (4.7) SQL-kyselyssä.

```
select BATCH, ASIAKASTUNNISTE, KK_1 AS KK from "PRE_DATASET"

union

...

union

select BATCH, ASIAKASTUNNISTE, KK_10 from "PRE_DATASET"
```

Kuva 4.7: Lyhennetty SQL-kysely, jonka avulla muodostetaan taulu kuukausitasolla. Tähän yhdistetään kaikki kymmenen kuukautta jokaisesta, kuukausikeräyksestä jotka muodostavat datapisteitä koulutusdatassa.

Lopuksi muodostetaan toinen aputaulu, johon asiakaskohtaisiin kuukausitason riveihin yhdistetään kuvan (4.8) SQL-kyselyn muodostama aputaulu. Lisäksi tässä vaiheessa varmistetaan, ettei null-arvoja kerätä datajoukkoon, koska kaikki asiakkaan kuukaudet eivät välttämättä sisällä palvelumerkintöjä tai diagnooseja. Tällöin kuukausidatan ryhmittelyriviä ei löydy myöskään aputaulussa, joka yhdistetään datapisteisiin. Lopuksi taulu lajitellaan erän, asiakastunnisteen ja kuukauden mukaan, jotta tieto yhdistyy oikein koulutusdatan y-arvojen kanssa. Kuvassa (4.10) koulutusdatan y-arvot kerätään ja suodatetaan.

```
select

BATCH,
ASIAKASTUNNISTE,
KK,
ifnull(SUM_SEG_1,0) as S1,
ifnull(SUM_SEG_2,0) as S2,
ifnull(SUM_SEG_3,0) as S3,
ifnull(SUM_SEG_4,0) as S4,
ifnull(SUM_SEG_0,0) as S0,
ifnull(IKA,0) as IKA,
ifnull(SUKUPUOLI,0) as SUKUPUOLI,
ifnull(KUNTA,0) as KUNTA,

ifnull(R10,0) as R10,
...
ifnull(R90,0) as R90,

ifnull( ICD10_A00_A09,0) as ICD10_A00_A09,
...
ifnull( ICD10_ZA0_ZB9,0) as ICD10_ZA0_ZB9,

ifnull( ICPC2_A02,0) as ICPC2_A02,
...
ifnull( ICPC2_W58,0) as ICPC2_W58

from "LOAD_ML_UTIL_DATAPISTEET"

left join
LOAD_ML_UTIL_VUOSI_KUUKAUSI_ASIAKAS
on LOAD_ML_UTIL_DATAPISTEET.KK = LOAD_ML_UTIL_VUOSI_KUUKAUSI_ASIAKAS.VUOSI_KUUKAUSI
and LOAD_ML_UTIL_DATAPISTEET.ASIAKASTUNNISTE = LOAD_ML_UTIL_VUOSI_KUUKAUSI_ASIAKAS.
    ASIAKASTUNNISTE

order by
ASIAKASTUNNISTE, KK
```

Kuva 4.8: SQL-kysely, joka muodostaa lajitellun koulutusdatan. Taulu yhdistää 10 kuukauden jaksot palvelutapahtumien merkintöihin. Lisäksi taulu lajitellaan henkilön ja kuukauden mukaan, jotta se vastaisi koulutusdatan tuloksien järjestystä. Koska kaikilla asiakkailla ei ole jokaisena kuukautena merkintöjä, tulee liitoksessa mukaan tyhjiä rivejä, jotka kaikki muunnetaan arvoksi 0.

4.3 Koulutusdatan muodostaminen

Seuraavaksi tietovarastolta kerätyt viimeistellyt taulut muutetaan koulutusdatalle sopivaan muotoon. Tietovarastolta data siirretään DataBricksin Python-työkirjaan, jossa se käsitellään. Koska klassiset neuroverkot, LSTM-arkkitehtuurin neuroverkot, sekä konvoluutioneuroverkot tarvitsevat syötteen hieman erilaisissa muodoissa, tulee tehdä erilaisissa formaateissa olevia koulutusdatajoukkoja. Koulutukseen käytettävässä datajoukossa on 111 280 riviä, jossa on 1607 saraketta. Ottaen huomioon, että rivit ovat kuukausitasolla olevia datapisteitä, joita käytetään kymmenen yhden syötteen rakentamiseen, tulee kokonaisuudeksi 11 128 datapistettä. Tulosjoukossa rivejä on vastaavasti 11 128. Datapisteistä 4533 on positiivisia, eli pisteitä joissa segmentti muuttuu ennustettavaan ryhmään 4. Negatiivisia arvoja datassa on 6595. Datajoukon jakauma on hieman vääristynyt, joka johtaa siihen että neuroverkoilla tuotetut mallit voivat painottua

liikaa negatiivisiin tuloksiin (Bishop, 2006).

Klassisen neuroverkon syötteet ovat yksi lista (Goodfellow ym., 2016), joka vastaa syötekerroksen neuronien määrää. Tästä syystä meidän tulee yhdistää kymmenen kuukauden datapisteet yhdeksi pitkäksi listaksi. Lisäksi sarakkeista poistetaan erän tunnus, vuosi_kuukausi, sekä rivinumerointi. Näin muodostetaan taulu, jossa on 11 128 riviä ja jokainen rivi koostuu 16 030 syötteen listasta. Lisäksi koulutuksen y-datasta siivotaan ylimääräiset sarakkeet pois, jättäen vain arvon, joka kertoo, muuttuiko potilaan segmentti vai ei. Tämä käsittely on esitetty kuvan (4.9) Python-koodissa.

```
import numpy as np

X = np.array(dataset_X)
X = X[:,4:X.shape[1]]
X = X.reshape((int((X.shape[0])/10), X.shape[1]*10))

Y = np.array(dataset_Y)
Y = Y[:,3]
```

Kuva 4.9: Python-koodi, jonka avulla koulutusdata muotoillaan klassisen neuroverkon syöteavaruuden mukaisesti.

LSTM- sekä konvoluutioneuroverkoissa x-taulun syötteistä tehdään moniulotteisempia. Aluksi muodostetaan lista, jossa on listoja, jotka sisältävät kuukauden datapisteet. Tällöin LSTM- ja konvoluutioneuroverkoissa muodostuu sama määrä syötteitä kuin klassista neuroverkkoa varten tehdyissä syötteissä, mutta luotu taulu on muodoltaan (11128, 10, 1603). Tämä vastaa kaksiulotteisia syötepisteitä, jotka ovat kronologisessa järjestyksessä kuukauden mukaan. Tämä on esitetty kuvan (4.10) koodissa.

```
X = np.array(dataset_X)
X = X[:,4:tmp_X.shape[1]]
X = X.reshape((int((X.shape[0])/10), 10, X.shape[1]))

Y = np.array(dataset_Y)
Y = Y[:,3]
```

Kuva 4.10: Python-koodi, jonka avulla koulutusdata muotoillaan LSTM- ja konvoluutioneuroverkkojen moniulotteisen syöteavaruuden mukaiseksi.

Seuraavaksi data tulee jakaa koulutusta, validointia ja testausta varten omiin joukkoihinsa. Data on jaettu siten, että koulutusta varten 11 128 rivistä otetaan 64%, validointia varten 16% ja testaukseen varten 20%. Käytetty kirjasto myös sekoittaa datan satunnaiseen järjestykseen, mikä varmistaa ettei vain tiettyjen potilaiden kuukausia joutuisi vain

yhteen datajoukkoon. Jako tapahtuu kuvassa (4.11) esitetyllä koodilla.

```
from sklearn.model_selection import train_test_split

x_train, x_test, y_train, y_test = train_test_split(tmp_X, tmp_Y, test_size=0.2)
x_train, x_valid, y_train, y_valid = train_test_split(x_train, y_train, test_size
    =0.2)
```

Kuva 4.11: Python-koodi, joka jakaa datan koulutusta, validointia ja testausta varten.

4.4 Neuroverkkojen koulutus

Neuroverkkojen kouluttaminen sisältää useita päätöksiä mallin arkkitehtuuriin, rakenteeseen, parametreihin ja koulutustyyleihin liittyen. Kaikkia mahdollisia parametriyhdistelmiä ja testejä ei ajan ja resurssien puutteessa pysty toteuttamaan, mutta muutamien parametriyhdistelmien ja arkkitehtuurien vertailu tuo hyvän näkökulman sille, soveltuvatko neuroverkot sosiaali- ja terveydenhuollon segmenttimuutoksien tehokkaaseen ennustamiseen.

Luvussa 4.4.1 kerrotaan kokeellisessa osuudessa neuroverkkomallien rakentamiseen ja kouluttamiseen käytetyistä kirjastoista Keras:ista ja TensorFlow:sta. Luvussa 4.4.2 käydään läpi käytettyjen neuroverkkomallien arkkitehtuurit ja esimerkkikoodeja. Luvussa 4.4.3 käydään läpi koulutettavien neuroverkkomallien tutkittavia parametreja.

4.4.1 Tensorflow ja Keras

Tutkimuksen kokeellisessa osuudessa neuroverkkomallit toteutetaan TensorFlow:n ja Keras:in avulla. TensorFlow on avoimen lähdekoodin alusta koneoppimisen tehtäviä varten. TensorFlow sisältää paljon erilaisia työkaluja, kirjastoja, sekä muita resursseja koneoppimisen sovelluksia varten (Google, 2021). Keras on pythonilla toteutettu korkean tason API, jolla voidaan luoda ja kouluttaa erilaisia neuroverkkomalleja, erityisesti syviä neuroverkkoja (”Keras Documentation”, 2021). Keras:in rajapinta pystyy ajamaan TensorFlow:n kirjastojen lisäksi myös muutamaa muuta koneoppimisen työkalua ja resurssikirjastoa. Keras tarjoaa käyttäjille helppokäyttöisen syntaksin, sekä monipuoliset ja laajennettavat ominaisuudet.

Keras soveltuu tutkielman tekemiseen erinomaisesti avoimen lähdekoodin ja helppokäyttöisen syntaksin takia. Keras:in avulla voidaan tuottaa kaikki kokeellisessa osuudessa tuotetut neuroverkkomallit sisältäen klassisen neuroverkkoarkkitehtuurin, LSTM-arkkitehtuurin ja konvoluutioneuroverkkoarkkitehtuurin.

4.4.2 Neuroverkkojen parametrit

Neuroverkoissa on monia parametreja, joiden määrittely on työlästä. Parametrien määrittäminen on usein myös riippuvainen käytössä olevasta datasta (Bishop, 2006). Tutkielman kokeellisessa osuudessa on ajan ja laskentatehon rajallisuuden takia käytetty kolmea erilaista neuroverkkoarkkitehtuuria, joista jokaista testataan sekä syvänä, että matalana neuroverkkona. Vaihdettavia parametreja kokeellisessa osuudessa ovat alkukerroksien aktivointifunktiot. Jokaisessa mallissa vastekerroksen aktivointifunktio on $Sigmoid$, koska binäärisessä luokitteluongelmassa arvoväli [0, 1] on tehokasta muuntaa pyöristämällä tulokset binäärisiksi tulokseksi. Muissa kerroksissa aktivointifunktiona käytetään $Sigmoid$:ia, $ReLU$:a ja $LReLU$:a. Virhefunktio on toinen vaihdettava parametri, jossa vertaillaan MSE:n ja risti-entropian vaikutusta koulutettavaan neuroverkkomalliin. Taulukossa (4.1) on esitetty kaikki tutkielman kokeellisessa osuudessa toteutetut mallit ja niiden parametriyhdistelmät. Lisäksi jokaisessa mallissa käytetään Adam-optimointialgoritmia vakioparametreilla. Mallien vertailtaviin parametreihin ei kuulu epoch-muuttuja, joka määrittelee kuinka monta kertaa koulutus toistetaan neuroverkkomallille. Epoch-parametri määritellään dynaamisesti käyttäen validointidataa ja aikaisen pysäytyksen -metodia, jota käsitellään tarkemmin luvussa 2.5 neuroverkkojen kouluttamisen yhteydessä.

4.4.3 Neuroverkkojen luominen ja koulutus

Tämän tutkielman kokeellisen osuuden tutkittaviksi neuroverkkoarkkitehtuureiksi valittiin klassinen neuroverkko, LSTM-neuroverkko, sekä konvoluutioneuroverkko. Konvoluutiomalli valittiin Mored et al. toteuttaman tutkimuksen perusteella (Morid ym., 2020), koska tutkielman kokeellisessa käytetty data on hyvin hajanaista. Konvoluutioverkot toimivat yleensä tehokkaasti hajanaisiin terveydenhuollon tietojoukkoihin (Pandey & Janghel, 2019). LSTM-neuroverkko valittiin tutkittavaksi, koska potilaista kerätty kuukausijaksoittainen data on ajanjaksollisesti sidottua, johon LSTM-mallit soveltuvat hyvin, niiden rakenteen vuoksi. Kolmanneksi malliksi valittiin klassinen neuroverkko, josta saadaan vertailupohja edellä mainittuihin neuroverkkoarkkitehtuureihin. Kaikista tutkittavista malleista tehdään yksinkertaiset versiot, joilla testataan neuroverkon

Taulukko 4.1: Koulutetuissa ennustemalleissa käytetyt parametriyhdistelmät

Malli	Arkkitehtuuri	Syvyys	Virhefunktio	Aktivointifunktio
M_1	Klassinen	Matala	MSE	$Sigmoid$
M_2	Klassinen	Matala	MSE	$ReLU$
M_3	Klassinen	Matala	MSE	$LReLU$
M_4	Klassinen	Matala	Risti-entropia	$Sigmoid$
M_5	Klassinen	Matala	Risti-entropia	$ReLU$
M_6	Klassinen	Matala	Risti-entropia	$LReLU$
M_7	Klassinen	Syvä	MSE	$Sigmoid$
M_8	Klassinen	Syvä	MSE	$ReLU$
M_9	Klassinen	Syvä	MSE	$LReLU$
M_{10}	Klassinen	Syvä	Risti-entropia	$Sigmoid$
M_{11}	Klassinen	Syvä	Risti-entropia	$ReLU$
M_{12}	Klassinen	Syvä	Risti-entropia	$LReLU$
M_{13}	LSTM	Matala	MSE	$Sigmoid$
M_{14}	LSTM	Matala	MSE	$ReLU$
M_{15}	LSTM	Matala	MSE	$LReLU$
M_{16}	LSTM	Matala	Risti-entropia	$Sigmoid$
M_{17}	LSTM	Matala	Risti-entropia	$ReLU$
M_{18}	LSTM	Matala	Risti-entropia	$LReLU$
M_{19}	LSTM	Syvä	MSE	$Sigmoid$
M_{20}	LSTM	Syvä	MSE	$ReLU$
M_{21}	LSTM	Syvä	MSE	$LReLU$
M_{22}	LSTM	Syvä	Risti-entropia	$Sigmoid$
M_{23}	LSTM	Syvä	Risti-entropia	$ReLU$
M_{24}	LSTM	Syvä	Risti-entropia	$LReLU$
M_{25}	Konvoluutio	Matala	MSE	$Sigmoid$
M_{26}	Konvoluutio	Matala	MSE	$ReLU$
M_{27}	Konvoluutio	Matala	MSE	$LReLU$
M_{28}	Konvoluutio	Matala	Risti-entropia	$Sigmoid$
M_{29}	Konvoluutio	Matala	Risti-entropia	$ReLU$
M_{30}	Konvoluutio	Matala	Risti-entropia	$LReLU$
M_{31}	Konvoluutio	Syvä	MSE	$Sigmoid$
M_{32}	Konvoluutio	Syvä	MSE	$ReLU$
M_{33}	Konvoluutio	Syvä	MSE	$LReLU$
M_{34}	Konvoluutio	Syvä	Risti-entropia	$Sigmoid$
M_{35}	Konvoluutio	Syvä	Risti-entropia	$ReLU$
M_{36}	Konvoluutio	Syvä	Risti-entropia	$LReLU$

syvyyden muutoksia sekä muutamaa parametrimuunnosta. Erilaisia ennustemalleja toteutetaan yhteensä 36 kappaletta.

Klassisella neuroverkolla tarkoitetaan rakennetta, jossa on syötekerroksen lisäksi jokin määrä piilotettuja neuronikerroksia. Kuvassa (4.12) oleva koodi esittää kuinka matala neuroverkko toteutetaan ja kuvassa (4.13) oleva koodi esittää syvän neuroverkon toteutuksen. Syvässä neuroverkossa piilokerroksia on enemmän, jolloin neuronien määrä verkossa kasvaa.

```
from tf.keras import Sequential
from tf.keras import Dense

model = Sequential()
model.add(Dense(64, activation='sigmoid', input_shape=X[0].shape))
model.add(Dense(1, activation='sigmoid'))
model.compile(loss='MSE', optimizer='adam')
```

Kuva 4.12: Kokeellisessa osuudessa käytetyn mallin M_1 koodiesimerkki. Neuroverkkomallin arkkitehtuuri on klassinen matala neuroverkko.

Kuvan (4.12) koodissa malli määritetään käyttäen tf.keras -kirjaston Sequential -mallia. Tämä toimii neuroverkon pohjakerroksena, jonka päälle muut kerrokset lisätään. Seuraavaksi malliin lisätään piilokerros, jossa on 64 neuronia ja jossa se yhdistyy datajoukon yhden datapisteen leveyteen. Lopuksi matalaan malliin lisätään yksi neuroni, joka tulee ennustamaan segmenttimuutoksen todennäköisyyttä. Mallin jokaiselle kerrokselle määritellään myös aktivointifunktio. Lisäksi malliin valitaan virhefunktio, joka on tässä mallissa MSE. Kuten kaikissa tulevissa malleissa, optimointialgoritmiksi on valittu Adam-algoritmi. Kuvassa (4.13) oleva koodi poikkeaa matalasta neuroverkosta siten, että siihen on lisättynä kaksi neuronikerrosta ennen vastekerrosta. Kaikki kokeellisen osuuden mallit tehdään syviksi neuroverkoiksi lisäämällä niihin nämä kaksi kerrosta, joiden molemmat kerrokset sisältävät viisi neuronia.

```
model = Sequential()
model.add(Dense(64, activation='sigmoid', input_shape=X[0].shape))
model.add(Dense(5, activation='sigmoid'))
model.add(Dense(5, activation='sigmoid'))
model.add(Dense(1, activation='sigmoid'))
model.compile(loss='MSE', optimizer='adam')
```

Kuva 4.13: Kokeellisessa osuudessa käytetyn mallin M_7 koodiesimerkki. Neuroverkkomallin arkkitehtuuri on klassinen syvä neuroverkko.

LSTM-arkkitehtuurilla rakennetun neuroverkkomallin määritellään kuvassa (4.14) esitetyssä koodissa. Kaikissa malleissa käytetään edellä mainittua Sequential -mallia. Neuroverkkomalliin lisätään tf.keras.layers -kirjastosta LSTM-kerros, jossa on 64 neuronia.

Vastekerros on yksi neuroni, joka antaa vasteeksi luvun, joka arvioi segmenttimuutoksen todennäköisyyttä. Kuvassa (4.15) on esitetty LSTM-arkkitehtuuri, johon on lisätty kaksi viiden neuronin kerrosta, tehden tästä rakenteesta syvän LSTM-neuroverkon.

```
from tf.keras.layers import LSTM

model = Sequential()
model.add(LSTM(64, activation='sigmoid', input_shape=X[0].shape))
model.add(Dense(1, activation='sigmoid'))
model.compile(loss='MSE', optimizer='adam')
```

Kuva 4.14: Kokeellisessa osuudessa käytetyn mallin M_{13} koodiesimerkki. Neuroverkkomallin arkkitehtuuri on matala LSTM-neuroverkko.

```
model = Sequential()
model.add(LSTM(64, activation='sigmoid', input_shape=X[0].shape))
model.add(Dense(5, activation='sigmoid'))
model.add(Dense(5, activation='sigmoid'))
model.add(Dense(1, activation='sigmoid'))
model.compile(loss='MSE', optimizer='adam')
```

Kuva 4.15: Kokeellisessa osuudessa käytetyn mallin M_{19} koodiesimerkki. Neuroverkkomallin arkkitehtuuri on syvä LSTM-neuroverkko.

Konvoluutioneuroverkkojen koodi on esitetty kuvan (4.16) koodissa. Malli on Sequential-pohjainen kuten muutkin Keras:in avulla tuotetut neuroverkot. Malliin lisätään kaksi Conv1D konvoluutiokerrosta, jotka ovat tf.keras.layers -kirjastosta. Conv1D konvoluutiokerrosta käytetään usein aikajaksoittaisen datan kanssa, joka ei ole yhtä moniulotteista kuin esimerkiksi kuvatiedostot. tf.keras.layers -kirjastosta saadaan myös MaxPooling1D -kerros, joka yksinkertaistaa ylempien kerroksien moninkertaistuneet syötteet. Tämän jälkeen moniulotteiset kerrokset muutetaan yksiulotteiseksi listaksi, koska konvoluution jälkeinen käsittely yhdistetään neuroverkkoon. tf.keras.layers- kirjastosta Flatten-kerros lisätään malliin toteuttamaan ulottuvuuksien muotoilu yksiulotteiseksi jonoksi. Lopuksi yksi neuroni lisätään viimeiseksi kerrokseksi. Kuvassa (4.17) on konvoluutioneuroverkon rakentava koodi, jossa vastekerrosta ennen lisätään kaksi viisineuronista kerrosta.

Kokeellisen osuuden kaikki neuroverkkomallit koulutetaan kuvan (4.18) koodin avulla. Kokeellisessa osuudessa käytetään tf.keras.callbacks -kirjastosta tuotua EarlyStopping funktiota, jonka avulla voidaan laskea validointidatalla jokaisen koulutuskierroksen jälkeen, kasvaako vai laskeeko validointidatan tulokset. Tämä estää mallin ylikouluttamisen koulutusdataan, jolloin ennustettavuus pysyy hyvällä tasolla uuteen dataan. Metodissa on patience-parametri, joka kertoo kuinka monta kierrosta validointidatalla

ajetun virhefunktion vaste voi kasvaa ennen kouluttamisen pysäyttämistä. Lopuksi suoritetaan neuroverkkomallin koulutusskripti, syöttämällä koulutusdata x_train ja y_train. Epoch-parametri määrittää koulutuskierroksien määrän, mutta koska tutkielmassa käytetään aikaisen pysäytyksen -metodia, ei todellisuudessa kaikkia kierroksia ajeta.

```
from tf.keras.layers import Conv1D
from tf.keras.layers import MaxPooling1D
from tf.keras.layers import Flatten

model = Sequential()
model.add(Conv1D(filters=64, kernel_size=3, activation='sigmoid', input_shape=X[0].
    shape))
model.add(Conv1D(filters=64, kernel_size=3, activation='sigmoid'))
model.add(MaxPooling1D(pool_size=2))
model.add(Flatten())
model.add(Dense(1, activation='sigmoid'))
model.compile(loss='MSE', optimizer='adam')
```

Kuva 4.16: Kokeellisessa osuudessa käytetyn mallin M_{25} koodiesimerkki. Neuroverkkomallin arkkitehtuuri on matala konvoluutioneuroverkko.

```
model = Sequential()
model.add(Conv1D(filters=64, kernel_size=3, activation='sigmoid', input_shape=X[0].
    shape))
model.add(Conv1D(filters=64, kernel_size=3, activation='sigmoid'))
model.add(MaxPooling1D(pool_size=2))
model.add(Flatten())
model.add(Dense(5, activation='sigmoid'))
model.add(Dense(5, activation='sigmoid'))
model.add(Dense(1, activation='sigmoid'))
model.compile(loss='MSE', optimizer='adam')
```

Kuva 4.17: Kokeellisessa osuudessa käytetyn mallin M_{30} koodiesimerkki. Neuroverkkomallin arkkitehtuuri on syvä konvoluutioneuroverkko.

```
from tf.keras.callbacks import EarlyStopping

stop = EarlyStopping(monitor='val_loss', mode='min', patience=5)
model.fit(x_train, y_train, validation_data=(x_valid, y_valid), epochs=4000,
    callbacks=[stop])
```

Kuva 4.18: Python-koodi, jonka avulla mallit koulutetaan hyödyntäen aikaisin pysäytys -metodia.

4.5 Kokeellisen osuuden tulokset

Neuroverkot laitettiin ennustamaan sosiaali- ja terveydenhuollon asiakkaan segmenttimuutoksen todennäköisyyttä. Tämä todennäköisyys muunnettiin binääriseksi vastaukseksi käyttäen pyöristämistä. Kokeellisen osuuden 36 neuroverkkoa ja niiden tuottamat tulokset ovat esitetty kolmessa taulukossa. Näihin tauluihin sisältyy tutkielman kokeellisessa osuudessa tuotettujen ennustemallien suorituskykymittarien arvot.

Taulukko (4.2) havainnollistaa, miten hyvin klassinen neuroverkko onnistui ennustamaan sosiaali- ja terveydenhuollon asiakkaan segmenttimuutosta. Klassisen neuroverkon parhaaksi malliksi tarkkuuden 0.838, sekä F-arvon 0.787 perusteella osoittautui malli M_{11}. Kyseinen malli on taulukosta (4.1) katsottuna syvä neuroverkko, jonka virhefunktiona on risti-entropia ja kerroksien aktivointifunktiona $ReLU$. Mallit olivat tarkkuuden, sekä f-arvon perusteella suhteellisen tasaisia, mutta huomionarvoista on malli M_2, joka ei oppinut tunnistamaan positiivisia tuloksia. Tämä voidaan nähdä taulukon (4.2) arvoista. Malli M_2 täsmällisyys ja herkkyys ovat 0, mutta johdonmukaisuus on 1. Kyseinen malli luokitteli siis kaikki tulokset negatiiviseen ryhmään binäärisistä tuloksista. Klassisilla neuroverkoilla toteutettujen ennustemallien saamat f-arvot vaihtelivat muiden mallien osalta 0.716 ja 0.787 välillä. Tämä osoittaa mallien ennustavat molempia luokkia suhteellisen tasaisesti.

Taulukko 4.2: Sosiaali- ja terveydenhuollon asiakkaan segmenttimuutoksen ennustamisen mittarit klassisiin neuroverkkoihin perustuvilla neuroverkkoarkkitehtuureilla

Malli	Tarkkuus	Täsmällisyys	Herkkyys	Johdonmukaisuus	F-arvo
M_1	0,816	0,845	0,674	0,914	0,750
M_2	0,590	0,000	0,000	1,000	0,000
M_3	0,789	0,799	0,649	0,887	0,716
M_4	0,827	0,834	0,720	0,900	0,773
M_5	0,805	0,745	0,795	0,811	0,769
M_6	0,826	0,813	0,747	0,881	0,778
M_7	0,795	0,751	0,746	0,829	0,748
M_8	0,822	0,799	0,755	0,868	0,777
M_9	0,818	0,877	0,647	0,937	0,744
M_{10}	0,801	0,752	0,768	0,824	0,760
M_{11}	0,838	0,855	0,729	0,914	0,787
M_{12}	0,785	0,692	0,855	0,736	0,765

Taulukko (4.3) näyttää kuinka LSTM-neuroverkot onnistuivat ennustamaan sosiaali- ja

terveydenhuollon asiakkaan segmenttimuutosta. LSTM-neuroverkkojen parhaaksi malliksi tarkkuuden 0.854, sekä F-arvon 0.825 perusteella osoittautui malli M_{13}. Kyseinen malli on taulukosta (4.1) katsottuna matala neuroverkko, jonka virhefunktiona on MSE ja kerroksien aktivointifunktiona $Sigmoid$. Tämän arkkitehtuurin mallit olivat myös tarkkuuden, sekä f-arvon perusteella suhteellisen tasaisia. Myöskään mikään malli ei päätynyt ennustamaan vain toista arvoista. F-arvot vaihtelivat mallien osalta 0.751 ja 0.825 välillä.

Taulukko 4.3: Sosiaali- ja terveydenhuollon asiakkaan segmenttimuutoksen ennustamisen mittarit LSTM-neuroverkkoihin perustuvilla neuroverkkoarkkitehtuureilla

Malli	Tarkkuus	Täsmällisyys	Herkkyys	Johdonmukaisuus	F-arvo
M_{13}	0,854	0,840	0,810	0,886	0,825
M_{14}	0,840	0,822	0,794	0,874	0,808
M_{15}	0,848	0,842	0,789	0,891	0,814
M_{16}	0,853	0,850	0,793	0,897	0,821
M_{17}	0,818	0,892	0,649	0,942	0,751
M_{18}	0,838	0,889	0,706	0,935	0,787
M_{19}	0,851	0,849	0,789	0,897	0,818
M_{20}	0,850	0,826	0,817	0,874	0,821
M_{21}	0,832	0,920	0,661	0,958	0,769
M_{22}	0,849	0,906	0,717	0,945	0,800
M_{23}	0,836	0,803	0,813	0,853	0,808
M_{24}	0,793	0,937	0,548	0,973	0,692

Taulukko (4.4) näyttää kuinka konvoluutioneuroverkot oppivat ennustamaan sosiaali- ja terveydenhuollon asiakkaan segmenttimuutosta. Konvoluutioon pohjautuvan arkkitehtuurin parhaaksi malliksi tarkkuuden 0.843, sekä F-arvon 0.807 perusteella osoittautui malli M_{29}. Kyseinen malli on taulukosta (4.1) katsottuna matala neuroverkko, jonka virhefunktiona on risti-entropia ja kerroksien aktivointifunktiona $ReLU$. Huomioitavaa on myös kahden mallin M_{32} ja M_{35} epäonnistumiset, jossa mallit oppivat myös ennustamaan vain negatiivisia tuloksia. F-arvot vaihtelivat muiden mallien osalta 0.763 ja 0.807 välillä.

Taulukoiden (4.2 - 4.4) perusteella voidaan havaita, että mallien tarkkuus ja f-arvo vaihtelevat jossain määrin. Vaihtelut pysyivät tarkkuuden osalta välillä 0.716 - 0.825, poislukien ne mallit, jotka eivät oppineet ollenkaan ennustemaan segmenttimuutoksia. Tuloksia vertaillen voidaan todeta, että LSTM-arkkitehtuuri oli selvästi soveltuvin arkkitehtuuri datan oppimiseen. LSTM-arkkitehtuurisssa kaikki mallit oppivat erottelemaan positiivisia ja negatiivisia syötteitä. Lisäksi tarkkuuden ja f-arvon perusteella tehokkain malli oli myös LSTM-arkkitehtuuriin perustuva.

Taulukko 4.4: Sosiaali- ja terveydenhuollon asiakkaan segmenttimuutoksen ennustamisen mittarit konvoluutioneuroverkkoihin perustuvilla neuroverkkoarkkitehtuureilla

Malli	Tarkkuus	Täsmällisyys	Herkkyys	Johdonmukaisuus	F-arvo
M_{25}	0,823	0,757	0,813	0,830	0,784
M_{26}	0,832	0,801	0,763	0,876	0,782
M_{27}	0,792	0,695	0,846	0,757	0,763
M_{28}	0,843	0,783	0,833	0,849	0,807
M_{29}	0,842	0,852	0,726	0,918	0,784
M_{30}	0,812	0,744	0,799	0,821	0,771
M_{31}	0,835	0,802	0,772	0,875	0,787
M_{32}	0,605	0,000	0,000	1,000	0,000
M_{33}	0,834	0,874	0,678	0,936	0,764
M_{34}	0,843	0,815	0,778	0,885	0,796
M_{35}	0,605	0,000	0,000	1,000	0,000
M_{36}	0,817	0,736	0,837	0,804	0,783

Mikään toteutetuista ennustemalleista ole optimoitu ennustamaan Pärjääjä-mallin mukaista segmenttimuutosta. Testatut yhdistelmät olivat yksinkertaisia ja perustuivat yleisiin käytäntöihin sekä muihin tutkimuksiin. On todennäköistä, että on olemassa parempia neuroverkkoihin perustuvia ennustemalleja, joilla voidaan saavuttaa suurempi ennustetarkkuus tälle datajoukolle. Tutkielman kokeellisen osuuden tulokset ovat kuitenkin lupaavia, ja niitä muokkaamalla voitaisiin mahdollisesti saavuttaa ennustevoimaltaan vieläkin tehokkaampia neuroverkkoja.

5. Johtopäätökset ja yhteenveto

Tämä tutkielma rakentui kokeellisen osuuden ympärille. Kokeellisessa osuudessa tutkittiin sosiaali- ja terveydenhuollon asiakkaan Pärjääjä-mallin mukaisen segmenttimuutoksen ennustamista puolen vuoden päähän. Tarkoituksena oli tuottaa mahdollisimman tarkka segmenttimuutoksen ennuste neuroverkkojen avulla. Tämän lisäksi tarkastelun kohteena oli erilaisten neuroverkkoarkkitehtuurien vaikutus ennustemallien tarkkuuteen. Tutkielman kokeellinen osuus pohjautui toteutettuun kirjallisuuskatsaukseen. Kirjallisuuskatsaus koostuu aiemmista tutkimuksista, jotka liittyvät sosiaali- ja terveydenhuollon asiakkaan segmenttimuutoksen ennustamiseen. Tutkielmassa syvennyttiin myös sosiaali- ja terveydenhuollon operatiivisten järjestelmien tuottaman datan hyödyntämiseen.

Kokeellisessa osuudessa ennustemallien koulutusta varten kerätty datajoukko haettiin Keski-Uudenmaan sote-kuntayhtymän tietovarastosta. Rakenteelliseksi tiedoksi kerättiin yli 70-vuotiaiden sosiaali- ja terveydenhuollon asiakkaiden palvelumerkintöjä, jotka sisälsivät ICD-10, ICPC-2 ja asiointitapa-koodeja. Lisäksi datajoukkoon kerättiin asiakkaan perustiedot kuten ikä, sukupuoli ja kunta. Kerätty datajoukko muunnettiin kuukausitason ryhmittelyiksi, ja asiakkaan jokaiselle kuukaudelle tehtiin Pärjääjä-mallin mukainen dynaaminen segmentointi. Koulutusdata koostettiin kymmenen kuukauden jaksoiksi, ja koulutuksen kohteeksi otettiin kuuden kuukauden jälkeinen kuukausi, viimeisimmästä asiakkaan palvelutapahtumamerkinnästä. Dynaamisen segmentoinnin perusteella pääteltiin, tapahtuiko segmenttimuutos verkosto-asiakkuusryhmään vai ei. Kokeellisessa osuudessa valittiin käytettäväksi klassinen neuroverkkoarkkitehtuuri, LSTM-arkkitehtuuri ja konvoluutioneuroverkkoarkkitehtuuri. Jokaista arkkitehtuuria varten tehtiin useita parametriyhdistelmiä, joissa neuroverkon syvyyttä lisättiin, aktivointifunktioita muutettiin ja virhefunktioita vaihdettiin. Ajan ja resurssien rajallisuuden takia muita parametreja ei vertailtu kokeellisessa osuudessa.

Luvussa 5.1 kootaan vastauksia ensimmäiseen tutkimuskysymykseen. Luvussa 5.2 syvennytään tutkielman löydöksiin toisen tutkimuskysymyksen osalta. Luku 5.3 käsittelee viimeisen tutkimuskysymykseen löytyneitä vastauksia. Luvussa 5.4 käsitellään

tutkielman kokeellisessa osuudessa tuotettujen ennustemallien vaikutuksia. Viimeisessä luvussa 5.5 perehdytään tutkielman mahdollisiin jatkotutkimusaiheisiin.

5.1 Ensimmäinen tutkimuskysymys

Tässä tutkielmassa tutkittiin, kuinka sosiaali- ja terveydenhuollon dataa voidaan kerätä, jotta sitä voidaan hyödyntää koneoppimisen ennustemallien kouluttamisessa. Luvussa 3.2 tarkasteltiin sosiaali- ja terveydenhuollon tuottaman tiedon ominaisuuksia, joista keskeiseksi tekijäksi muodostui datan siiloutuminen ja poikkeavat tietorakenteet. Luvussa 3.2.3 perehdyttiin tietovaraston toteutukseen Suomessa kehitetyn käsitemallinnuksen avulla. Käsitemallinnus ja oikein toteutettu tietovarastointi pystyvät yhdistämään siiloutuneita operatiivisia tietojärjestelmiä yhtenäisiksi kokonaisuuksiksi. Lisäksi luvussa 3.2.1 käsiteltiin erilaisia sosiaali- ja terveydenhuollon koodistoja. Näiden koodistojen avulla pystytään muodostamaan dataa, joka on samanlaista kansallisesti tai jopa kansainvälisesti. Koodistot ja standardit ovat siis merkittävässä asemassa, sillä sosiaali- ja terveydenhuollon operatiiviset järjestelmät käyttävät näitä koodistoja.

Toisaalta sosiaali- ja terveydenhuollon operatiiviset tietojärjestelmät voivat vaikeuttaa datan yhdistämisen prosessia, koska ne tallentavat dataa erimuodossa. Voidaan kuitenkin todeta, että sosiaali- ja terveydenhuollon dataa voidaan kerätä operatiivisista tietojärjestelmistä hyödyntäen tietovarastoinnin tekniikoita ja Suomessa kehitettyä sosiaali- ja terveydenhuollon käsitemallinnusta. Lisäksi luvussa 3.2.4 käsiteltiin, kuinka tietovarastossa olevaa dataa voidaan hyödyntää koneoppimisen koulutuksessa. Esimerkiksi dimensio- ja faktataulut vähentävät datan käsittelyssä vaadittavaa prosessointitehoa.

5.2 Toinen tutkimuskysymys

Tutkielmassa tutkittiin sitä, kuinka tarkasti neuroverkot pystyvät ennustamaan sosiaali- ja terveydenhuollon asiakkaiden segmenttimuutoksia. Paras ennustetarkkuus kokeellisessa osuudessa tuotetuilla ennustemalleilla oli 0.853. Tämä tulos saatiin hyödyntämällä kaikkia palvelumerkinnöistä saatavia koodistoja sekä asiakkaan perustietoja. Parhaan tarkkuuden ennustemalli M_{13} perustui LSTM-arkkitehtuuriin, jonka ensimmäisessä kerroksessa oli 64 neuronia ja vastekerroksessa 1 neuroni. Kaikissa kerroksissa käytettiin $Sigmoid$-aktivointifunktiota ja virhefunktiona oli MSE-funktio. Parhaan tarkkuuden ennustemalli sai myös parhaan f-arvon 0.825, joka kertoo ennustemallin tasapainosta sekaannusmatriisissa. Saavutettu tarkkuus oli hyvä, kun vertaillaan muihin sosiaali- ja

terveydenhuollossa oleviin neuroverkoilla toteutettuihin ennustemalleihin (Morid ym., 2020; Ng ym., 2020). Toisaalta suora ennustemallien vertailu tutkimusten välillä on haasteellista, koska käytetty data sekä ennustettava luokka eivät vastaa täysin muiden tutkimuksien lähtökohtia. Lisäksi viitatuissa tutkimuksissa datamäärä, ennusteen pituus ja ikäryhmät vaihtelivat paljon. Tutkielman kokeellisen osuuden datajoukko oli huomattavasti pienempi, mikä voi omalta osaltaan selittää, miksi tarkkuus jäi muita tutkimuksia alhaisemmaksi.

Tutkielmassa perehdyttiin kirjallisuuskatsauksen avulla neuroverkkojen ennustetarkkuuteen sosiaali- ja terveydenhuollon asiakkaan segmenttimuutoksien suhteen. Luvussa 3.3 käytiin läpi kirjallisuuskatsauksena muutamia tutkimuksia. Morid ym. (2020) toteuttaman tutkimuksen kokeellisen osuuden tarkkuus oli 0.9453. Tässä tutkimuksessa ennustettiin potilaan segmenttimuutosta kalleimpaan kustannusluokkaan. Tämä ennustemalli toteutettiin konvoluutioarkkitehtuurin avulla.

Kokeellisen osuuden ja kirjallisuuskatsauksen perusteella voidaan todeta, että neuroverkkojen kyky ennustaa sosiaali- ja terveydenhuollon asiakkaan segmenttimuutosta on erittäin hyvä. Kokeellisessa osuudessa huomattiin myös, ettei kaikki neuroverkkoyhdistelmät ole kykeneviä ennustamaan sosiaali- ja terveydenhuollon asiakkaan segmenttimuutoksia. Neuroverkkojen parametriyhdistelmät vaikuttavat siis vahvasti ennustemallien tarkkuuteen.

5.3 Kolmas tutkimuskysymys

Tutkielmassa tutkittiin neuroverkkoarkkitehtuurien vaikutusta ennustemallien suorituskykyyn. Vertailtavat arkkitehtuurit olivat klassinen neuroverkkoarkkitehtuuri, LSTM-arkkitehtuuri ja konvoluutioneuroverkkoarkkitehtuuri. LSTM-arkkitehtuurilla tuotetut ennustemallit olivat keskimääräisesti tarkempia sekä f-arvoiltaan suurempia kuin muiden arkkitehtuurien ennustemallit. Lisäksi LSTM-arkkitehtuurilla tuotettu ennustemalli oli kaikista malleista paras tarkkuuden ja f-arvon mittareilla. Kokeellisessa osuudessa huomattiin myös, että osa klassiseen neuroverkkoarkkitehtuuriin ja konvoluutioarkkitehtuuriin perustuvista malleista ei oppinut ennustamaan segmenttimuutosta. Nämä mallit päätyivät ennustamaan, että segmentti ei muutu, mikä johti huonoon lopputulokseen tarkkuuden ja f-arvon kannalta.

Lisäksi luvussa 2.4 perehdyttiin kokeellisessa osuudessa käytettyihin neuroverkkoarkkitehtuureihin. Tutkielmassa huomattiin, että LSTM-neuroverkot ja konvoluutioneu-

roverkot toimivat erilaisen datajoukon kanssa. LSTM-neuroverkot ovat tehokkaita peräkkäisen aikasarjadatan kanssa, kun taas konvoluutioneuroverkot ovat tehokkaita harvemman datajoukon kanssa. Myös opettamiseen liittyviä eroja huomattiin. Voidaan siis todeta, että arkkitehtuurilla on suuri vaikutus sosiaali- ja terveydenhuollon asiakkaan segmenttimuutoksen ennusteen tarkkuuteen.

5.4 Tulosten vaikutusten arviointi

Tutkielman kokeellisessa osuudessa tuotettuja ennustemalleja tullaan hyödyntämään Keski-Uudenmaan sote -kuntayhtymän tietovarastointihankkeessa. Valitun ennustemallin avulla voidaan tuottaa tietovarastolla oleville sosiaali- ja terveydenhuollon asiakkaille ennuste seuraavan kuukauden päästä segmenttimuutoksen riskistä. Suuri määrä ennusteita mahdollistaa myös tietojohtamisen tasolla hyötyjä. Esimerkiksi organisaatiotasolla voidaan mallin tuottamien ennusteiden avulla arvioida, kuinka palvelutarpeen määrä tulee muuttumaan. Tällä tavoin suurempiin hoitotarpeisiin pystyttäisiin mahdollisesti varautumaan, jolloin sosiaali- ja terveydenhuollon palvelut eivät ruuhkautuisi yhtä paljon. Tutkielman kokeellisessa osuudessa saavutettu tarkkuus 0.853 on hyödynnettävässä mallissa mielestäni riittävä, sillä tarkoitus on löytää asiakkaita, joiden riskit ovat suuremmat. Sosiaali- ja terveydenhuollon asiakkaita on todella paljon, ja resurssien ohjaus on tässä kontekstissa erittäin tärkeää. Tutkielmassa tuotettu malli antaa lukuarvon 0 ja 1 väliltä. Ennustemallin mukaan lukuarvo 1 viittaa hyvin suureen segmenttimuutoksen riskiin. Tästä syystä sosiaali- ja terveydenhuollon asiakkaille voitaisiin esittää riskilukua, joka voisi auttaa resurssien kohdentamisessa.

Kokeellisessa osuudessa tuotetun mallin ei ole tarkoitus toimia lääkinnälliseen laitteeseen verrattavana välineenä, eikä sen perusteella ole tarkoitus toteuttaa mitään sosiaali- ja terveydenhuollon päätöksiä. Sitä voidaan kuitenkin mahdollisuuksien mukaan käyttää asiakkaiden riskien tunnistamiseen, sekä organisaation tasolla tapahtuvaan tiedolla johtamiseen. Tästä syystä väärät luokittelut, eivät aiheuta sosiaali- ja terveydenhuollon asiakkaiden kannalta haittapuolet ovat hyvin vähäiset. Tarkoituksena on toimia työkaluna, jonka avulla riskejä voidaan tunnistaa.

5.5 Jatkotutkimuskohteet

Kokeellinen osuus oli tarkasti rajattu ajan ja resurssien rajallisuudesta johtuen. Jatkotutkimusta voisi tehdä useista aihealueista. Käyttöön otetuista 1603:sta parametrista, jotka

sisälsivät perustiedot, ICD-10, ICPC-2 ja asiointitavan koodit, voisi tutkia tarkemmin esimerkiksi hierarkiatason valinnan vaikutuksia tuloksiin. Lisäksi tietovaraston muita tietoja voisi mahdollisesti myös hyödyntää, ja tutkia sitä, miten nämä tiedot vaikuttaisivat ennustemallien suorituskykyyn. Neuroverkkojen arkkitehtuuriin ja parametrisoitiin voisi myös keskittää paljon jatkotutkimusta, sillä tutkielmassa tuotetut 36 ennustemallia olivat hyvin yksinkertaisia rakenteeltaan ja yleisiltä parametreiltaan. Kaikkia näitä osia ja niiden vaikutuksia voitaisiin jatkotutkia sekä kehittää lisää tulevaisuudessa. Aktivointifunktioiden, virhefunktioiden ja optimointialgoritmien vaikutusta ennustemalleihin voitaisiin tutkia laajemmin. Neuroverkkojen kerroksien määrä, neuronien määrä sekä muut neuroverkoissa usein käytetyt toteutustavat ovat hyvin potentiaalisia tutkimuskohteita ennustemallin suorituskyvyn kannalta. Yksi potentiaalinen tutkimuskohde olisi myös datan esikäsittely ja sen vaikutukset ennustemalliin. Esimerkiksi autoenkoodaajan (Autoencoder) käytön vaikutuksista voisi tehdä jatkotutkimusta.

Kokeellisessa osuudessa tutkittiin ainoastaan Pärjääjä-mallin mukaisen segmentaation muutosta verkosto-asiakkuudeksi kuuden kuukauden päähän. Jatkotutkimusta voisi tehdä esimerkiksi ennustepituuden kasvattamisesta 12, 24 tai jopa 36 kuukauteen. Lisäksi ennustemallista voisi tehdä binäärisen luokitteluennustajan sijasta moniluokkaisen ennustemallin, jossa sosiaali- ja terveydenhuollon asiakkaan tuleva segmentti ennustetaan datan perusteella. Esimerkiksi Pärjääjä-mallin mukainen moniluokkainen ennustaminen olisi tehokas tapa organisaation tasolla hallinnoida kaikkia sosiaali- ja terveydenhuollon asiakkaita. Tällöin ennustetta voitaisiin hyödyntää resurssitarpeen tai kustannusten arviointiin. Myös erilaiset segmentointimallit olisivat yksi jatkotutkimuksen aihe. Esimerkiksi luvussa 3.1 mainitun Bridges to Health -mallin mukainen segmenttimuutoksen ennustaminen olisi todella mielenkiintoinen vaihtoehto segmenttimuutosten ennustamiseen Suomen sosiaali- ja terveydenhuollossa.

Viitteet

Albu, A. & Stanciu, L. (2015). Benefits of using artificial intelligence in medical predictions. *2015 E-Health and Bioengineering Conference (EHB)*, 1–4.

Bishop, C. M. (2006). Pattern recognition. *Machine learning*, *128*(9).

Chechulin, Y., Nazerian, A., Rais, S. & Malikov, K. (2014). Predicting patients with high risk of becoming high-cost healthcare users in Ontario (Canada). *Healthcare Policy*, *9*(3), 68.

Chen, T. & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 785–794.

Chong, J. L., Lim, K. K. & Matchar, D. B. (2019). Population segmentation based on healthcare needs: a systematic review. *Systematic reviews*, *8*(1), 1–11.

Chong, J. L. & Matchar, D. B. (2017). Benefits of population segmentation analysis for developing health policy to promote patient-centred care. *Ann Acad Med Singapore*, *46*(7), 287–289.

De, S., Maity, A., Goel, V., Shitole, S. & Bhattacharya, A. (2017). Predicting the popularity of instagram posts for a lifestyle magazine using deep learning. *2017 2nd International Conference on Communication Systems, Computing and IT Applications (CSCITA)*, 174–177.

Dumoulin, V. & Visin, F. (2016). A guide to convolution arithmetic for deep learning. *arXiv preprint arXiv:1603.07285*.

Franz, L., Shrestha, Y. R. & Paudel, B. (2020). A deep learning pipeline for patient diagnosis prediction using electronic health records. *arXiv preprint arXiv:2006.16926*.

Fukushima, K., Miyake, S. & Ito, T. (1983). Neocognitron: A neural network model for a mechanism of visual pattern recognition. *IEEE transactions on systems, man, and cybernetics*, (5), 826–834.

Goodfellow, I., Bengio, Y. & Courville, A. (2016). *Deep Learning*. MIT Press.

Google. (2021). Tensorflow [Haettu 13.8.2021 osoitteesta tensorflow.org].

He, K., Zhang, X., Ren, S. & Sun, J. (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *Proceedings of the IEEE international conference on computer vision*, 1026–1034.

Hutson, M. (2017). Self-taught artificial intelligence beats doctors at predicting heart attacks. *Science*, *14*(04).

Häyrinen, K., Saranto, K. & Nykänen, P. (2008). Definition, structure, content, use and impacts of electronic health records: a review of the research literature. *International journal of medical informatics*, *77*(5), 291–304.

Jean-Baptiste, D., O'MALLEY, A. & Shah, T. (2017). Population Segmentation and Targeting of Health Care Resources: Findings from a Literature Review December 2017. *Mathematica Policy Research*.

Joynt, K. E., Figueroa, J. F., Beaulieu, N., Wild, R. C., Orav, E. J. & Jha, A. K. (2017). Segmenting high-cost Medicare patients into potentially actionable cohorts. *Healthcare*, *5*(1-2), 62–67.

Kai, Y., Lei, J., Yuqiang, C. & Wei, X. (2013). Deep learning: yesterday, today, and tomorrow. *Journal of computer Research and Development*, *50*(9), 1799.

Kattan, A., Abdullah, R. & Geem, Z. W. (2011). *Artificial neural network training and software implementation techniques*. Nova Science Publishers, Inc.

Keras Documentation [Haettu 13.8.2021 osoitteesta keras.io]. (2021).

Kingma, D. P. & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Krizhevsky, A., Sutskever, I. & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, *25*, 1097–1105.

Le, X.-H., Ho, H. V., Lee, G. & Jung, S. (2019). Application of long short-term memory (LSTM) neural network for flood forecasting. *Water*, *11*(7), 1387.

Li, S., Ben-Nun, T., Girolamo, S. D., Alistarh, D. & Hoefler, T. (2020). Taming unbalanced training workloads in deep learning with partial collective operations. *Proceedings of the 25th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*, 45–61.

Linstedt, D. & Olschimke, M. (2015). *Building a scalable data warehouse with data vault 2.0*. Morgan Kaufmann.

Lynn, J., Straube, B. M., Bell, K. M., Jencks, S. F. & Kambic, R. T. (2007). Using population segmentation to provide better health care for all: the "Bridges to Health" model. *The Milbank Quarterly*, *85*(2), 185–208.

Maas, A. L., Hannun, A. Y., Ng, A. Y. ym. (2013). Rectifier nonlinearities improve neural network acoustic models. *Proc. icml*, *30*(1), 3.

Mattila, K. (2016). Suuntima-työkalun kokeilu ja käytön ohjeet asiakaslähtöisyyden vahvistumiseksi osana monisairaan potilaan hoitoketjua.

Miotto, R., Li, L. & Dudley, J. T. (2016). Deep learning to predict patient future diseases from the electronic health records. *European Conference on Information Retrieval*, 768–774.

Morid, M. A., Sheng, O. R. L., Kawamoto, K. & Abdelrahman, S. (2020). Learning hidden patterns from patient multivariate time series data using convolutional neural networks: A case study of healthcare cost prediction. *Journal of Biomedical Informatics*, *111*, 103565.

Mubarak, A. A., Cao, H. & Ahmed, S. A. (2021). Predictive learning analytics using deep learning model in MOOCs' courses videos. *Education and Information Technologies*, *26*(1), 371–392.

Muniasamy, A., Tabassam, S., Hussain, M. A., Sultana, H., Muniasamy, V. & Bhatnagar, R. (2019). Deep learning for predictive analytics in healthcare. *International Conference on Advanced Machine Learning Technologies and Applications*, 32–42.

Mäkinen. (2018). Alueelliset asiakaslähtöiset hoitoketjut sote-integraation välineeksi [Haettu 13.8.2021 osoitteesta https://asiakas.kotisivukone.com/files/gpfinland.kotisivukone.com/tiedostot/YLpvt_2018/Makinen_Alueelliset_hoitoketjut.pdf].

Neittaanmäki, P. & Lehto, M. (2018). Suomen kansalliset SOTE-tiedonlähteet ja tietojen hyödyntäminen. *Informaatioteknologian tiedekunnan julkaisuja/Jyväskylän yliopisto*, (2018, 49).

Neittaanmäki, P., Lehto, M., Ruohonen, T., Kaasalainen, K. & Karla, T. (2019). Suomen terveysdata ja sen hyödyntäminen.

Neittaanmäki, P., Tuominen, H., Äyrämö, S. & Vähäkainu, P. (2019). Tekoäly ja terveydenhuolto Suomessa.

Ng, S. H. X., Rahman, N., Ang, I. Y. H., Sridharan, S., Ramachandran, S., Wang, D. D., Khoo, A., Tan, C. S., Feng, M., Toh, S.-A. E. S. ym. (2020). Characterising and predicting persistent high-cost utilisers in healthcare: a retrospective cohort study in Singapore. *BMJ open*, *10*(1), e031622.

Niemelä, J. & Kivipelto, M. (2019). Asiakaslähtöinen palvelupolkumalli tulevaisuuden sote-keskusten lähtökohdaksi.

Nwankpa, C., Ijomah, W., Gachagan, A. & Marshall, S. (2018). Activation functions: Comparison of trends in practice and research for deep learning. *arXiv preprint arXiv:1811.03378*.

Pak, M. & Kim, S. (2017). A review of deep learning in image recognition. *2017 4th international conference on computer applications and information processing technology (CAIPT)*, 1–3.

Pandey, S. K. & Janghel, R. R. (2019). Recent deep learning techniques, challenges and its applications for medical healthcare system: A review. *Neural Processing Letters*, *50*(2), 1907–1935.

Pirkanmaan sairaanhoitopiiri. (2021). Suunta Suuntimasta ja kohti arjessa pärjäämistä [Haettu 13.8.2021 osoitteesta https://www.tays.fi/fi-fi/ohjeet/Hoitoketjut/Suuntima].

Prechelt, L. (1998). Early stopping-but when? *Neural Networks: Tricks of the trade* (s. 55–69). Springer.

Shamshirband, S., Fathi, M., Dehzangi, A., Chronopoulos, A. T. & Alinejad-Rokny, H. (2020). A review on deep learning approaches in healthcare systems: Taxonomies, challenges, and open issues. *Journal of Biomedical Informatics*, 103627.

Shanker, M. S. (1996). Using neural networks to predict the onset of diabetes mellitus. *Journal of chemical information and computer sciences*, *36*(1), 35–41.

Sokolova, M. & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information processing & management*, *45*(4), 427–437.

Sosiaali- ja terveysministeriö. (2021). Toisiolaki mahdollistaa sosiaali- ja terveystietojen tietoturvallisen käytön [Haettu 13.8.2021 osoitteesta https://stm.fi/sote-tiedon-hyodyntaminen].

Stein, B. & Morrison, A. (2014). The enterprise data lake: Better integration and deeper analytics. *PwC Technology Forecast: Rethinking integration*, *1*(1-9), 18.

Terveyden ja hyvinvoinnin laitos. (2021a). Kuntaliitto - ICPC Perusterveydenhuollon luokitus [Haettu 13.8.2021 osoitteesta https://koodistopalvelu.kanta.fi/codeserver/pages/classification-view-page.xhtml?classificationKey=210versionKey=282].

Terveyden ja hyvinvoinnin laitos. (2021b). THL - Asiointitapa [Haettu 13.8.2021 osoitteesta https://koodistopalvelu.kanta.fi/codeserver/pages/classification-view-page.xhtml?classificationKey=2003versionKey=2263].

Terveyden ja hyvinvoinnin laitos. (2021c). THL - avoHILMO-koulutus [Haettu 13.8.2021 osoitteesta https://thl.fi/documents/10531/123923/AvoHILMO-koulutus%20ICPC-luokitus%202010.pdf].

Terveyden ja hyvinvoinnin laitos. (2021d). THL - Tautiluokitus ICD-10 [Haettu 13.8.2021 osoitteesta https://koodistopalvelu.kanta.fi/codeserver/pages/classification-view-page.xhtml?classificationKey=23versionKey=58].

Virta-hanke/ DigiFinland Oy. (2021). Ylläpidettävät käsitemallit [Haettu 13.8.2021 osoitteesta https://digifinland.fi/toimintamme/virta-hanke/kasitemallit/].

Wood, R., Murch, B. & Betteridge, R. (2019). A comparison of population segmentation methods. *Operations Research for Health Care*, *22*, 100192.

Zhang, Z., Geiger, J., Pohjalainen, J., Mousa, A. E.-D., Jin, W. & Schuller, B. (2018). Deep learning for environmentally robust speech recognition: An overview of recent developments. *ACM Transactions on Intelligent Systems and Technology (TIST)*, *9*(5), 1–28.